MOS-AL162F

Technical Manual

Revision 2.0

PCB Revision: 1.0 or Higher

Firmware Revision: 5.0 or Higher

1 Revision History

Revision	Description	Author
2.0	Initial Release	Martino

2 Contents

3 Introduction

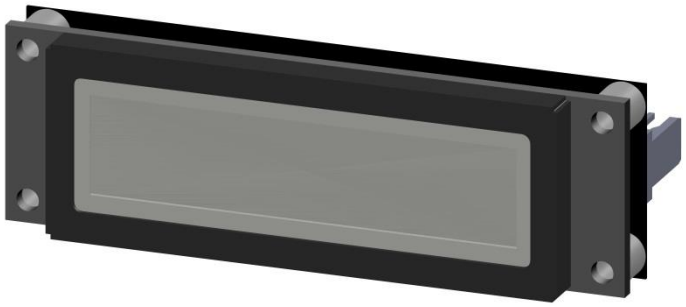


Figure 1: MOS-AL162F

The MOS-AL162F is an intelligent LCD display designed to decrease development time by providing an instant solution to any project. With the ability to communicate via serial RS232/TTL protocol, the versatile MOS-AL162F can be used with virtually any controller. The ease of use is further enhanced by an intuitive command structure to allow display settings such as backlight brightness and contrast to be software controlled. A general purpose output allows the controller to switch a single electronic or electro-mechanical device by issuing commands to the display unit. This can be used for controlling LEDs, relays, etc. Additionally, up to thirty-two custom characters such as character sets for bar graphs, and medium numbers may be stored in the non-volatile memory to be easily recalled and displayed at any time.

4 Quick Connect Guide

4.1 Standard Module

Figure 2: Standard Connections

Table 1: Standard Headers

#	Header	Mate
1	Communication/Power	SCCPC5V/BBC
2	Alternate Communication/Power	SCCPC5V/BBC
3	GPO/9600Bps	JMPR

The MOS-AL162F allows for user configuration of two common serial protocol logic levels. Either RS232 or TTL levels can be selected using the Protocol Select Jumpers and accessed through the four pin Communication/Power Header as outlined in the Serial Connections section.

Recommended Parts

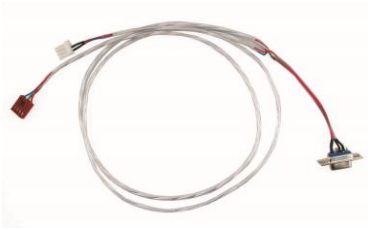


*Figure 3: *Communication/Power Cable (SCCPC5V)*

The most common cable choice for any standard 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 3: Breadboard Cable (BBC)*

For a more flexible interface to the MOS-AL162F, 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.

***Note:** Physical modification of cable may be required to ensure correct fit..

4.2 Serial Connections

Serial protocol provides a classic connection to the MOS-AL162F. 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 Baud Rate jumper.
   - 9600bps: Place a Jumper on the GPO Header before the unit is powered up.
   - 19200bps: Pulled up as default.
2. 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.

3. Make the connections.
   a. Connect the four pin female header of the Communication/Power Cable to the Communication/Power Header of your MOS-AL162F.
   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.
4. Create.
   - uProject or hyperterminal will serve to get you started, then move on with your own development. Instructions for the former can be found below and a variety of application notes are available for the latter at www.matrixorbital.ca/appnotes.

5 Software

The multiple communication protocols available and simple command structure of the *MOS-AL162F* 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. While many software programs are available to communicate with the *MOS-AL162F*, a number of more common samples are detailed in depth below.

Table 2: Reserved Control Characters

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

5.1 Hyperterminal

Installed on most Windows computers, hyperterminal can be run by selecting run and typing 'hypertrm' in the command line. This basic program will allow communication between a PC and your display.

When starting up, a name must be given to your connection, and an icon may be chosen, neither is consequential. Next, it's important to select the appropriate communication port to which your display is connected. Finally, the settings below must be entered to complete the port setup.

Table 3: Hyperterminal Settings

BPS	Data Bits	Parity	Stop Bits	Flow Control
19200*	8	None	1	None

***Note:** 9600bps Baud may also be selected on the MOS-AL162F.

Once a port is successfully set up, data can be sent to an attached display by typing on the keyboard. At this point, it may be helpful to echo keys to the monitor by selecting properties from the file menu and opening the ASCII settings from settings tab.

Commands can be sent to an attached display by issuing decimal commands using the number pad. While the ALT key is held down, four digit decimal values can be sent as a single ASCII character. For example, to clear the screen, try the following sequence.

ALT +0254 ALT +0088

Figure 4: Hyperterminal Command

Any commands or text desired can be sent to the communication port using this method to provide total control of any Matrix Orbital display.

5.2 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 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 basic GUI functions in the main window.

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

5.3 Application Notes

Full demonstration programs and code are available for Matrix Orbital displays in the C# language from Simple C# AppNote Pack in the Matrix Orbital 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.

6 Hardware

6.1 Standard Model

Communication/Power Header

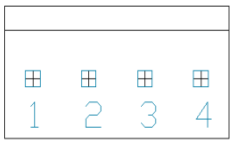


Figure 6: Communication/Power Header

Table 4: Communication/Power Pinout

Pin	Function
1	Vcc
2	Rx
3	Tx
4	Gnd

The Communication/Power Header provides a standard connector for interfacing to the serial series *MOS-AL162F*. Voltage is applied through pins one and four of the four pin Power/Data connector. Please ensure the correct voltage input for your display by referencing the electrical specifications in Table 13 before connecting power. Pins two and three are reserved for serial transmission, using either RS-232 or TTL logic levels, depending on what has been selected by the Protocol Select Jumpers.

Alternate Communication/Power Connector

Figure 7: Alternate Power Connector

Table 5: Alternate Power/Data Pinout

Pin	Function
3	VCC
2	Rx
1	Gnd

The MOS-AL162F also provides a three pin version of the Power/Data Connector which permits the device to receive commands only, no response is possible. The Molex 22-27-2031 style header is particularly useful for connecting to an unmodified floppy power cable, a Molex 22-01-2031 for example, from a PC power supply for a simple bench power solution.

Protocol Select Jumpers

The Protocol Select Jumpers, provide the means necessary to toggle the display module between RS-232 and TTL protocols. As a default, the jumpers are set to RS-232 mode with jumps on the 232 jumpers. In order to change the display to TTL mode, simply remove the zero ohm resistors from the 232 jumpers and solder them to the TTL jumpers.

6.2 Common Features

General Purpose Outputs

Figure 8: GPO Header

Table 6: GPO Pinout

Pin	Function	Pin	Function
1	GPO 1	6	Gnd

A unique feature of the MOS-AL162F is the ability to control relays and other external devices using a General Purpose Output, which can provide up to 20 mA of current and +5Vdc from the positive side of the GPO. This is limited by a 240 ohm resistor which is located to the above right of the GPOs as pictured below in figure 10. If the device, which is being driven by a GPO, requires a relatively high current (such as a relay) and has an internal resistance of its own greater than 250 ohms, then the 240 ohm resistor may be removed and replaced with a Jumper.

***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.

7 Troubleshooting

7.1 Power

In order for your Matrix Orbital display to function correctly, it must be supplied with the appropriate power. If the screen does not illuminate, power may 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.
- The next step is to check the power cable which you are using for continuity. If you don't have an ohm meter, try using a different power cable, if this does not help try using a different power supply.
- The last step will be to check the 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.

7.2 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.
- 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.

7.3 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 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 protocol, ensure that the host system and display module are both communicating on the same baud rate. The display will communicate at either 9600bps or 19200bps only.
- Match Rx from the *MOS-AL162F* to the transmitting pin from your host and the Tx pin to the receiving pin.
- Unlock the display. See the Set and Save Data Lock command for more info.

8 Commands

Text

1.1. Auto Scroll On		
	Dec	254 81
	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 auto scroll on.

1.2. Auto Scroll Off		
	Dec	254 82
	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.

1.3. Clear Screen		
	Dec	254 88
	Hex	FE 58
	ASCII	■X

Clears the contents of the screen.

1.4. Changing the Start Up Screen			
	Dec	254 64	Characters
	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	80 bytes, space characters can be added as needed

1.5. Set Auto Line Wrap On	Dec	254 67
	Hex	FE 43
	ASCII	■ C

Text will wrap to the next consecutive line once a row becomes full.

1.6. Set Auto Line Wrap Off	Dec	254 68
	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.

1.7. Set Cursor Position	Dec	254 71	Column Row
	Hex	FE 47	Column Row
	ASCII	■ G	Column Row

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

Column	1 byte, value between 1 and display width
Row	1 byte, value between 1 and display height

1.8. Go Home	Dec	254 72
	Hex	FE 48
	ASCII	■ H

Returns the cursor to the top left of the screen.

1.9. Move Cursor Back	Dec	254 76
	Hex	FE 4C
	ASCII	■ L

Moves cursor one position to the left. Cursor will obey wrap settings.

1.10. Move Cursor Forward	Dec	254 77
	Hex	FE 4D
	ASCII	■ M

Moves cursor one position to the right. Cursor will obey wrap settings.

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

1.12. Underline Cursor Off	Dec 254 75 Hex FE 4B ASCII ■ K
Removes line under current cursor position.	

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

1.14. Blinking Block Cursor Off	Dec 254 84 Hex FE 54 ASCII ■ T
Removes blinking block over current cursor position.	

Special Characters

1.15. Creating a Custom Character	Dec	254 78	ID Data
	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	1 byte, character ID 0-7
Data	8 bytes, character pixel data as shown below

Table 7: Custom Degree Character

Data1	000	p1	p2	p3	p4	p5	00001000	8
Data2	000	p1	p2	p3	p4	p5	00010100	20
Data3	000	p1	p2	p3	p4	p5	00001000	8
Data4	000	p1	p2	p3	p4	p5	00000011	3
Data5	000	p1	p2	p3	p4	p5	00000100	4
Data6	000	p1	p2	p3	p4	p5	00000100	4
Data7	000	p1	p2	p3	p4	p5	00000011	3
Data8	000	p1	p2	p3	p4	p5	00000000	0

1.16. Saving Custom Characters	Dec	254 193	Bank ID Data
	Hex	FE C1	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	1 byte, memory bank ID (0-4)
ID	1 byte, character ID (0-7)
Data	8 bytes, character pixel data as above

Table 8: Custom Character Banks

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

1.17. Loading Custom Characters	Dec	254 192	Bank
	Hex	FE C0	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	1 byte, memory bank ID (0-4)

1.18. Save Start Up Screen Custom Characters	Dec 254 194 ID Data Hex FE C2 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	1 byte, character ID (0-7)
Data	8 bytes, character pixel data, see custom character example

1.19. Initialize Medium Number	Dec 254 109 Hex FE 6D ASCII ■ m
Loads the medium number custom character bank into memory. Medium numbers must be initialized before they are used.	

1.20. Place Medium Numbers	Dec 254 111 Row Column Digit 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	1 byte, value between 1 and 20
Column	1 byte, value between 1 and 4
Digit	1 byte, single decimal digit to display

1.21. Initialize Horizontal Bar	Dec 254 104 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.	

1.22. Place Horizontal Bar Graph	Dec 254 124 Column Row Direction Length Hex FE 7C 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	1 byte, value between 1 and 4
Row	1 byte, value between 1 and 20
Direction	1 byte, 0 for right and 1 for left
Length	1 byte, length in pixels of the graph (0-100)

1.23. Initialize Vertical Bar	Dec 254 118 Hex FE 76

	ASCII	■ v	

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

1.24. Place Vertical Bar			
	Dec	254 61	Column Length
	Hex	FE 3D	Column Length
	ASCII	■ =	Column Length

Places a vertical bar graph on the screen beginning at the column and row specified. The bar extends upwards to the length indicated. A new bar will over write the old.

Column	1 byte, value between 1 and 4
Length	1 byte, length in pixels of the graph between 0 and 32

General Purpose Output

1.25. General Purpose Output Off			
	Dec	254 86	Number
	Hex	FE 56	Number
	ASCII	■ V	Number

Turns the specified GPO off by supplying a zero volt output.

Number	1 byte, GPO (1-6) to be turned off

1.26. General Purpose Output On			
	Dec	254 87	Number
	Hex	FE 57	Number
	ASCII	■ W	Number

Turns the specified GPO on by supplying a five volt output.

Number	1 byte, GPO (1-6) to be turned off

1.27. Set Start Up GPO State			
	Dec	254 195	Number State
	Hex	FE C3	Number State

Sets the start up state of the specified GPO. Changes will only be seen on start up.

Number	1 byte, GPO (1-6) to be turned off
State	1 byte, 1 for on or 0 for off

Display Functions

1.28. Display On	Dec	254 66	Minutes
	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	1 byte, number of minutes to leave backlight on (0 for infinite)

1.29. Display Off	Dec	254 70
	Hex	FE 46
	ASCII	■ F

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

1.30. Set Brightness	Dec	254 153	Brightness
	Hex	FE 99	Brightness

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

Brightness	1 byte, brightness level from 0(Dim)-255(Bright)

1.31. Set and Save Brightness	Dec	254 152	Brightness
	Hex	FE 98	Brightness

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

Brightness	1 byte, brightness level from 0(Dim)-255(Bright)

1.32. Set Contrast	Dec	254 80	Contrast
	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	1 byte, contrast level from 0(Light)-255(Dark)

1.33. Set and Save Contrast	Dec	254 145	Contrast
	Hex	FE 91	Contrast

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

Contrast	1 byte, contrast level from 0(Light)-255(Dark)

Data Security

1.34. Set Remember	Dec	254 147	Switch
	Hex	FE 93	Switch
Allows changes to specific settings to be saved to the display memory. Writing to memory can be slow and each change consumes 1 write of approximately 100,000 available. Set and save functions are always remembered. Remember is off by default.			
Switch	1 byte, 1 for on or 0 for off		

1.35. Set Data Lock	Dec	254 202 245 160	Level
	Hex	FE CA F5 A0	Level
Temporarily locks certain aspects of the display to ensure no inadvertent changes are made. A new level overrides the old, and levels can be combined. Default is 0.			
Level	1 byte, each bit representing a level, see table below		

Table 9: Data Lock Bits

Unlock	Reserved	Reserved	Reserved	Setting	Reserved	Command	Display
0	1	2	3	4	5	6	7

Table 10: Lock Parameters

Unlock	Unlocks all setting locks
Reserved	Place holders only, should be 0
Setting	Locks settings such as backlight and contrast
Command	Locks all commands
Display	Locks entire display, no new information can be displayed

1.36. Set and Save Data Lock	Dec	254 203 245 160	Level
	Hex	FE CB F5 A0	Level
Permanently locks certain aspects of the display to ensure no inadvertent changes are made. A new level overrides the old, and levels can be combined. Default is 0.			
Level	1 byte, see data lock table		

Miscellaneous

1.37. Read Version Number	Dec	254 54
	Hex	FE 36
	ASCII	■ 6
Causes display to respond with its firmware version number.		
Response	1 byte, convert to hexadecimal to view major and minor revision numbers	

1.38. Read Module Type	Dec	254 55
	Hex	FE 37
	ASCII	■ 7
Causes display to respond with its module number.		
Response	1 byte, module number, see partial list below	

Table 11: Sample Module Type Responses

4F	MOS Displays

9 Appendix

The versatility of the MOS-AL162F allows it to be interfaced to a variety of different displays. As a result, Character Sets, Environmental tolerances, Electrical specifications, and Optical characteristics will be display dependent. Applicable MOS-AL162F values are shown below for reference.

9.1 Environmental

Table 12: Environmental Specifications

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

9.2 Electrical

Table 13: Electrical Specifications

Parameter	Min	Typical	Max	Units
Current Draw (Board Only)	-	37.5	-	mA
Current Draw (Backlight)	-	18.9	-	mA
Maximum Single GPO Current	-	-	20	mA
Input Voltage (Vcc)	4.75	5.00	5.25	V

9.1 Optical Characteristics

Table 14: Optical Characteristics

Character x Lines	16 columns x 2 rows
Module Size	69.0 mm x 21.5 mm x 23.0 mm
Character Size	2.56 mm x 4.90 mm
Viewing Area	51.0 mm x 14.8 mm
LED Backlight Life	100,000 hours typical

9.2 Dimensional Drawings

Please see display datasheets for screen dependent dimensions and drawings.

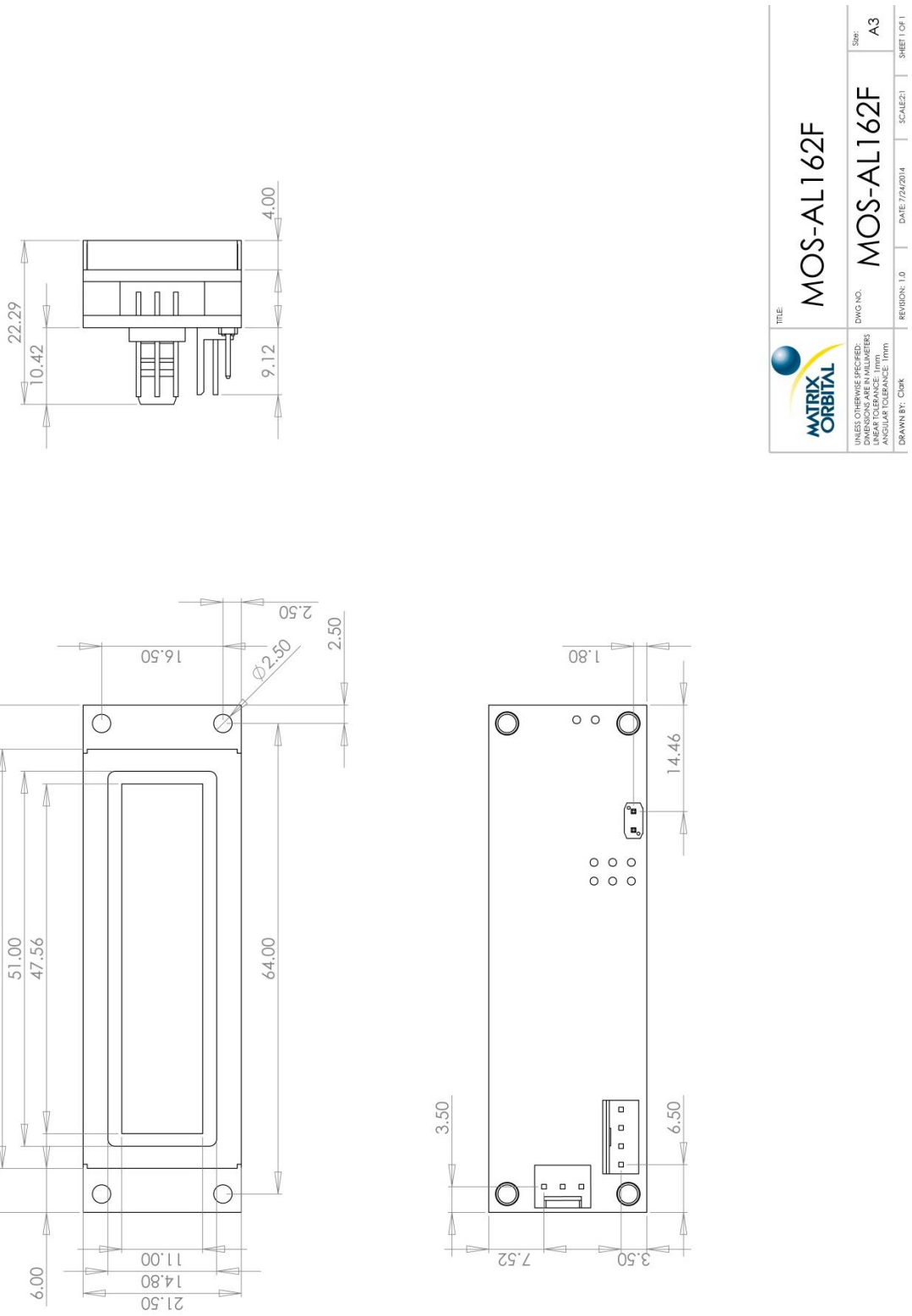


Figure 9: MOS-AL162F Model Dimensional Drawing

10 Ordering

10.1 Part Numbering Scheme

Table 15: Part Numbering Scheme

MOS	-AL	-162	F
1	2	3	4

10.2 Options

Table 16: Display Options

#	Designator	Options
1	Interface Type	MOS: Serial protocol using RS232 or TTL levels
2	Display Type	AL: Alphanumeric Liquid Crystal Display
3	Display Size	162: Sixteen columns by two rows
4	Display Model	F: Smaller Screen and Text

10.3 Accessories

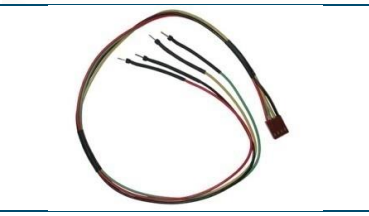
Power

Table 17: Power Accessories

PCS	Standard Power Cable	

Communication

Table 18: Communication Accessories

SCCPC5V	Serial Communication/5V Power Cable	
BBC	Breadboard Cable	

11 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.

FFSTN: Double film super-twisted nematic in reference to an LCD. The addition of two layers of film between the STN display and polarizer improves contrast.

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

GUI: Graphical user interface.

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

I^2C: Inter-integrated circuit protocol employing a clock and data line to communicate a short distance at slow speeds between a master and up to 128 addressable slave devices. A Matrix Orbital 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.

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 between phases to providing contrast than TN models.

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

12 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