Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

GLK12232-25-SM Technical Manual

Revision: 2.1

Contents

1 Getting Started

Figure 1: GLK12232-25-SM

The GLK12232-25-SM is an intelligent graphic LCD display designed to decrease development time by providing an instant solution to any project. With the ability to communicate via serial RS-232/TTL and I^2C protocols, the versatile GLK12232-25-SM can be used with virtually any controller. The ease of use is further enhanced by an intuitive command structure to allow display settings such as backlight brightness, contrast and baud rate to be software controlled. Additionally, text and fonts may be uploaded to the display and stored in the on board memory.

1.1 Display Options Available

The GLK12232-25-SM comes in two colour options, white text with blue background and blue text with white background. Extended voltage and temperature options are also available to allow you to select the display which will best fit your project needs.

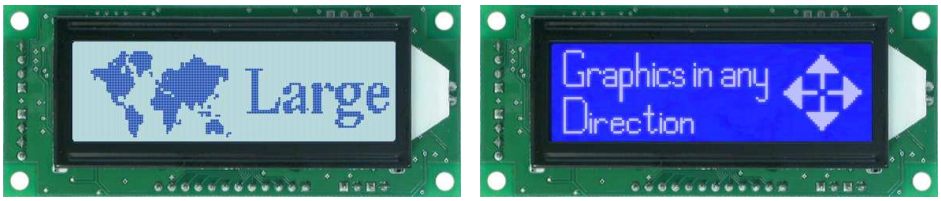


Figure 2: GLK12232-25-SM Options

1.2 Accessories

NOTE Matrix Orbital provides all the interface accessories needed to get your display up and running. You will find these accessories and others on our e-commerce website at http://www.matrixorbital.com. To contact a sales associate see Section 16.6 for contact information.

Figure 3: 5V Power Cable Adapter

Figure 4: 12V Power Cable Adaptor (V Models)

Figure 5: Breadboard Cable

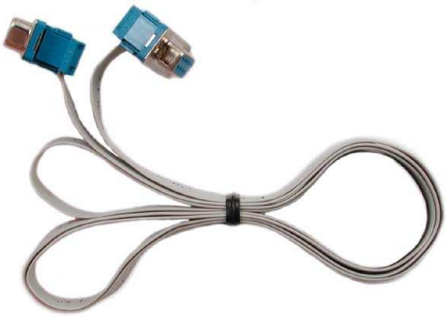


Figure 6: Serial Cable 4FT

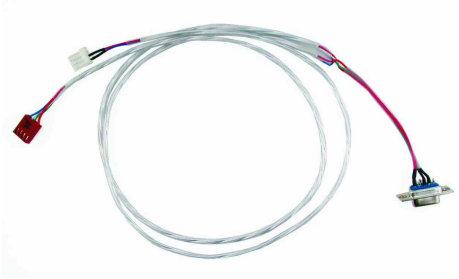


Figure 7: Communication and 5V Power Cable

Figure 8: 4x4 Keypad

1.3 Features

- 122 x 32 pixel graphics display
- Selectable communication protocol, Serial at RS-232 or TTL Levels or I^2C
- Two 5V - 20mA General Purpose Outputs
- 16 KB memory for fonts and bitmaps
- Lightning fast communication speeds, up to 115 kbps for RS-232 and 100 kbps for I^2C
- Adjustable contrast and backlight brightness

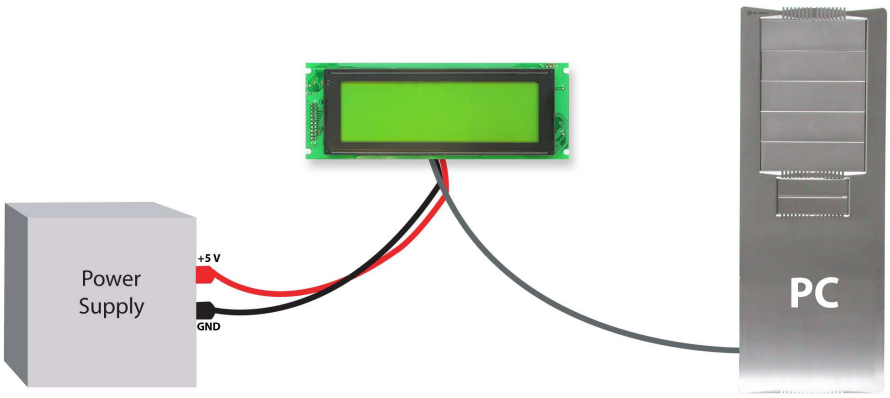
- Extended temperature available for extreme environments of -20C to 70C
- Extended voltage and efficient power supply available
- Support for up to a twenty five key matrix style keypad

1.4 Connecting to a PC

The GLK12232-25-SM connects seamlessly to a PC and it is an excellent means of testing the functionality and uploading new fonts and bitmaps. You will require a Communication and 5V Power Cable such as the one shown in Figure 7.

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

1. Plug the DB9 end of the Communication and 5V Power cable cable into the com port you wish to use.
2. Connect the power connector end of the Communication and 5V Power cable into the PC power supply (you will have to open your computer case if you do not have a separate power supply).
3. Connect the power and data connector of the Communication and 5V Power cable into the back of the display, see Section 2.1 for details.



1.5 Installing the Software

1.5.1 MOGD#

MOGD# is the latest updated version of MOGD and can be used to manage font and graphics downloads as well as exercise all of the features of our graphical displays. MOGD# provides a new user friendly interface as well as many feature enhancements.

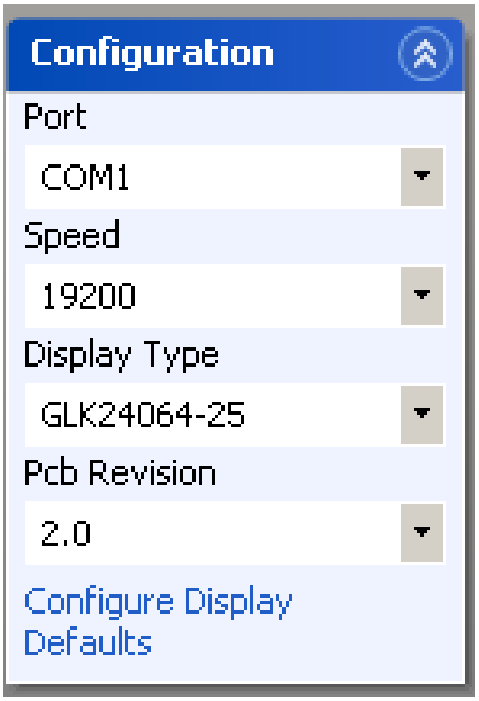
To install MOGD# from the Matrix Orbital website, follow the following steps:

1. Go to the website location: http://www.matrixorbital.ca/software/software_graphic/MOGDsharp/
2. Click on “Download Here”

3. Locate the file MogdSharp.zip on your desktop
4. Unzip MogdSharp.zip to a temporary directory using a program such as Winzip, Pkzip, etc.
5. Double click on "setup.exe"
6. Follow the instructions on the screen to complete the installation
7. MOGD# requires the .NET framework 2.0 and will download and install it automatically

After the installation is complete there will be a Matrix Orbital entry under "Start->Programs->Matrix Orbital" in the start menu. Click on the 'Mogd Sharp' entry to run the program.

Be sure to check the information selected in the configuration panel the first time MOGD# is run. Once this information is entered correctly the program can be used to control all functions of the graphic display.



Port The serial port the display is plugged in to.

Speed The communication speed the display module is set to. (Default 19,200)

Display Type The type of display (GLK12232-25-SM)

PCB Revision The revision of the display you are using. (Found on the back of the PCB).

Figure 9: Mogd Sharp Settings

- Winzip is available as a free download from http://www.winzip.com

2 Hardware Information

Refer to the following diagram for this chapter:

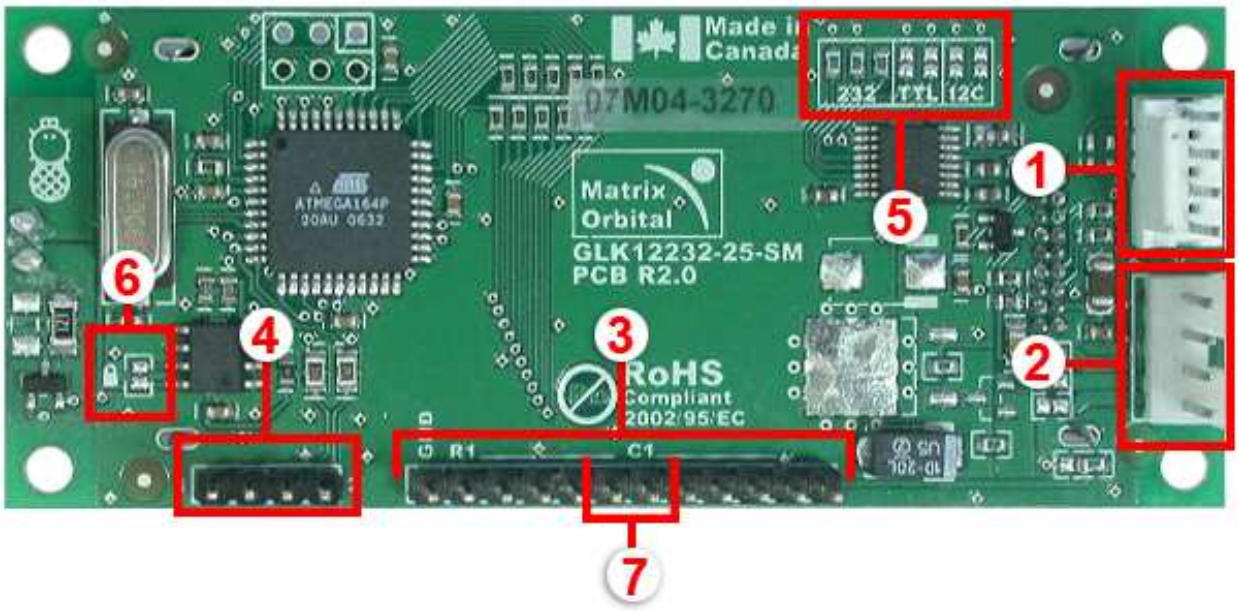
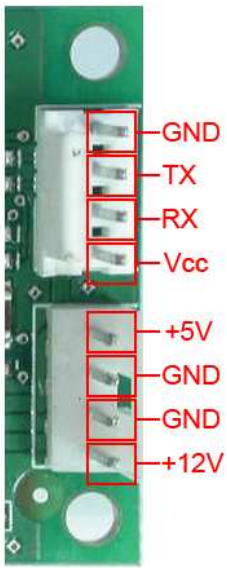


Figure 10: GLK12232-25-SM

Table 1: Hardware information

1 Power/Data Connector	5 Protocol Select Jumpers
2 Floppy Power Connector	6 Filesystem Lock Jumper
3 Keypad Interface Connector	7 Manual Override
4 GPO	

2.1 Power/Data Connector



Power/Connector	Regular GLK12232-25-SM input	-V option input
Vcc	+5V only	9-15V
+5V	+5V only	N/C
+12V	N/C	+12V

NOTE +12V is nominal for the -V but supply of 9-15V is acceptable

Figure 11: Power and Data Configuration

To power up and communicate with a GLK12232-25-SM (standard or wide voltage options), the users have two options for connections:

- power and data connections using the Power/Data Connector of the GLK12232-25-SM (supply Vcc with the proper voltage)
- power connections using the Floppy Power connector and data (RX/TX) and ground using the Power/Data connector

NOTES

- A common ground should be used at all times
- Each module ordered has specific voltages and cannot be interchanged

WARNINGS

- Do not apply and power with reversed polarization
- Do not apply any voltage other than the specified voltage
- Do not use any cables other than the cables supplied by Matrix Orbital, unless aware of the modifications required
- Do not under any circumstances use an unmodified floppy drive power cable in the wrong connector

2.2 Protocol Select Jumpers

The *Protocol Select Jumpers*, pictured below in Figure 12, provide the means necessary to toggle the display module between RS-232, TTL and I^2C protocols. As a default, the jumpers are set to RS-232 mode with zero ohm resistors on the 232 jumpers. In order to place the display module in I^2C mode you must first remove the zero ohm resistors from the 232 jumpers and then solder the resistors on to the I2C jumpers, or bridge solder across the pads. The display will now be in I^2C mode and have a default slave address of 0x50 unless it has been changed. Similarly, in order to change the display to TTL mode, simply remove the zero ohm resistors from the 232 or I^2C jumpers and solder them to the TTL jumpers.

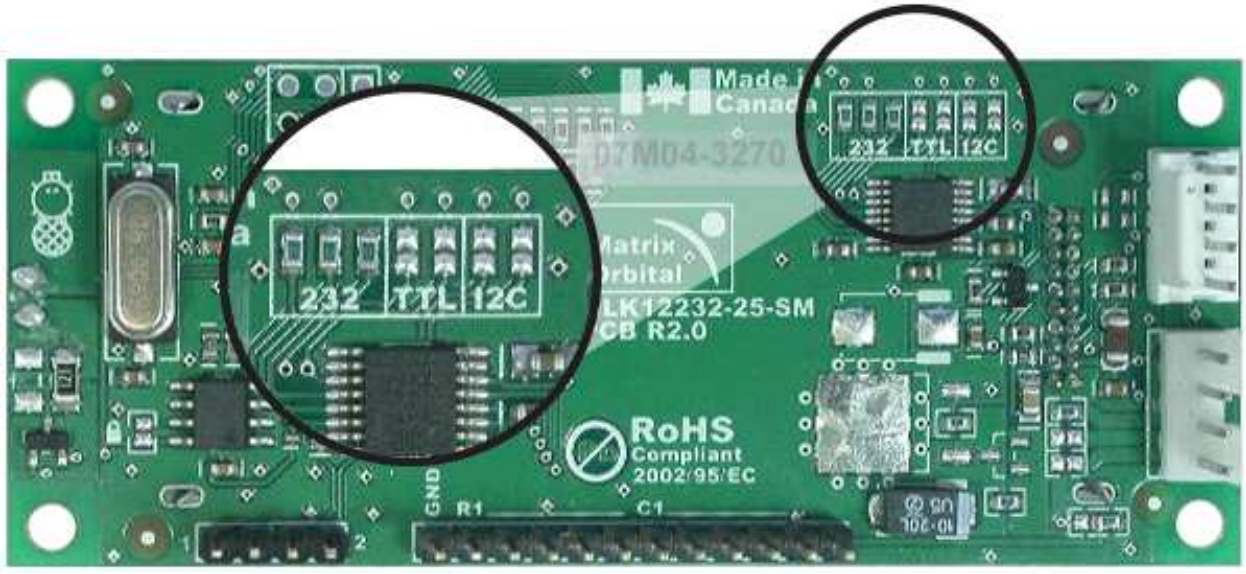


Figure 12: Protocol-Select-Jumpers

2.3 Keypad Interface Connector

The GLK12232-25-SM provides a *Keypad Interface Connector* which allows for up to a five by five matrix style keypad 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, which is associated with the particular key press, is automatically sent on the Tx communication line. If the display module is running in I^2C 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, for more detailed information see the Keypad Section.

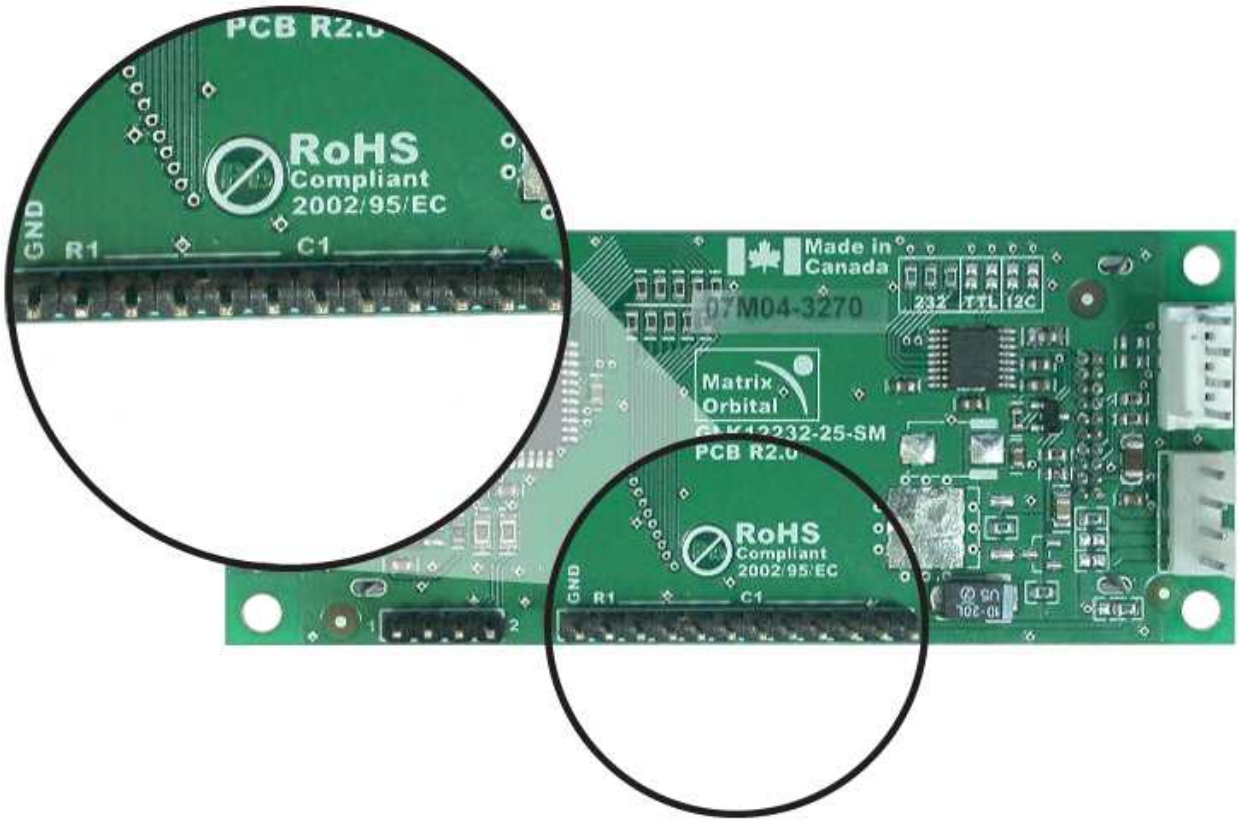


Figure 13: Keypad Interface Connector

2.4 GPO

A unique feature of the GLK12232-25-SM is the ability to control relays and other external devices using a *General Purpose Output* (3), 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 directly above the positive pin as pictured below in Figure 14 on the following page. 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.

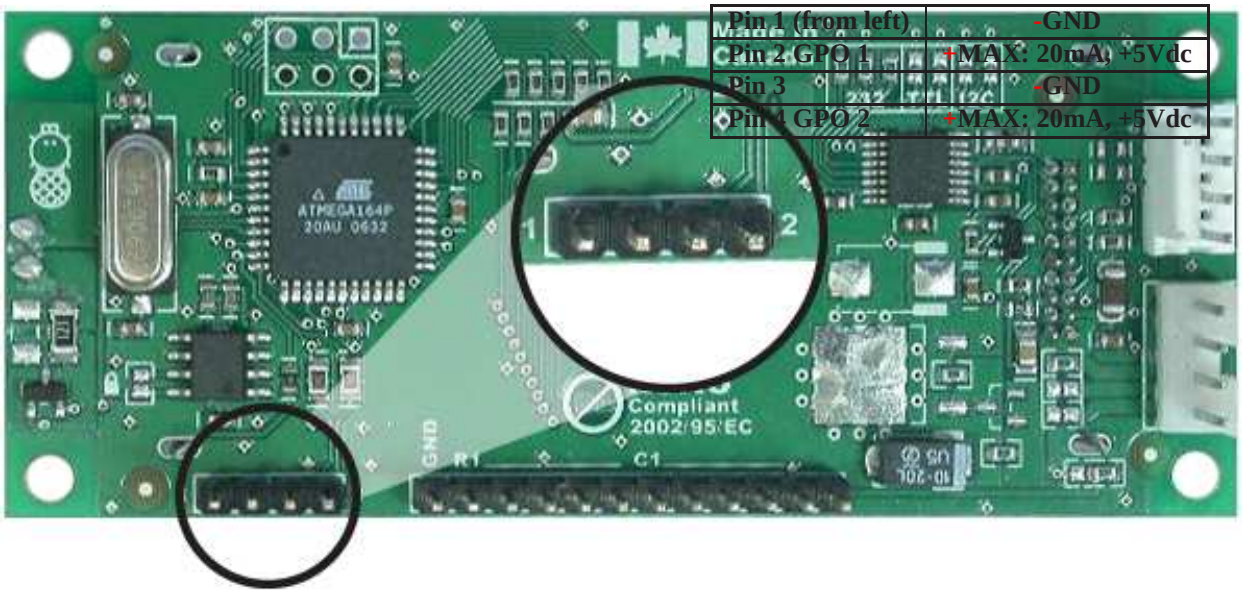


Figure 14: General Purpose Output

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

2.5 Manual Override

The *Manual Override* is provided to allow the GLK12232-25-SM to be temporarily reset to some of the factory defaults. This can be particularly helpful if the display module has been set to an unknown baud rate or I^2C Slave Address and you are no longer able to communicate with it. If you wish to return the module to its default settings you must:

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

Figure 15: Manual Override Jumper

Table 2: Default Values

Contrast	128
Backlight	255
Baud Rate	19.2 kbps
I^2C Slave Address	0x50
Data Lock	0x00
RS232AutoTransmitData	True

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

2.6 Filesystem Lock Jumper

The Filesystem Lock Jumper allows you to lock the filesystem on the GLK12232-25-SM so that no fonts or bitmaps can be either written or deleted from the on board memory. This feature is useful in order to protect data integrity of production units, if protection of other settings is required see Section 13

To lock the filesystem, solder a zero ohm resistor or use a solder jumper pictured in Figure 16 below.

Figure 16: Filesystem Lock Jumper

3 Troubleshooting

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

- First, you will want to make sure that you are using the correct power connector. Standard floppy drive power cables from your PC power supply may fit on the Power/Data Connector however they do not have the correct pinout as can be seen in Figure **??**. Matrix Orbital supplies power cable adapters for connecting to a PC, which can be found in the Accessories Section on page 2.
- The next step is to check the power cable which you are using for continuity. If you don't have an ohm meter, try using a different power cable, if this does not help try using a different power supply.
- The last step will be to check the *Power / Data Connector* on the GLK12232-25-SM. If the *Power / Data Connector* has become loose, or you are unable to resolve the issue, please contact Matrix Orbital see 16.6 on page 64 for contact information.

3.2 The display module is not communicating.

- First, check the communication cable for continuity. If you don't have an ohm meter, try using a different communication cable. If you are using a PC try using a different Com port.
- Second, please ensure that the display module is set to communicate on the protocol that you are using, by checking the *Protocol Select Jumpers.* To change the protocol used by the display module see Section 2.2 on page 8.
- Third, ensure that the host system and display module are both communicating on the same baud rate. The default baud rate for the display module is 19200 bps.

- If you are communicating to the display via I^2C please ensure that the data is being sent to the correct address. The default slave address for the display module is 0x50.

NOTE I^2C communication will always require pull up resistors.

- Finally, you may reset the display to it's default settings using the Manual Override Jumper, see Section 2.5 on page 10.

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

- The cause of this is often that no font has been loaded onto the display. To load a font onto the display see Section 4.2.1 on page 16.
- Another common cause may be that the contrast settings have been set to low. The solution to this problem is to adjust the contrast settings, the default setting that will work in most environments is 128

NOTE Optimal contrast settings may vary according to factors such as temperature, viewing angle and lighting conditions.

3.4 There is a problem uploading fonts or bitmaps.

- First, ensure that you can communicate to the display. A good test is to use a PC, with MOGD# installed, to connect to the display. See Section 1.4 on page 4for setting up a PC to test the GLK12232-25-SM.
- Second, ensure that the Filesystem Lock Jumper has not been set. See Section 2.6 on page 11.
- Third, please ensure that the display module's memory is not full. The GLK12232-25-SM has 16 Kb of memory for fonts and bitmaps.

NOTE If you are unable to resolve any issue please contact Matrix Orbital. See 16.6 on page 64 for contact information.

---

4 Communications

4.1 Introduction

The commands listed in this chapter describe how to configure data flow on the GLK12232-25-SM.

4.1.1 I²C Communication Summary

The GLK12232-25-SM is capable of communicating at 100 KHz in I²C mode, with 127 units addressable on a single I²C communication line. However, in order to communicate via I²C you must first ensure that pull up resistors, with a nominal value of 1K to 10K, are placed on the SCL and SDA communication lines coming from pins two and three of the Data / Power Connector respectively. Data responses by the module are automatically output via RS232, in case the host will be querying the module, it is necessary for the host to inform the module that its responses are to be output via I²C. This can be done by sending command 254 /160 / 0 to turn off auto transmission of data in RS232. This will keep the data in the buffer until the master clocks a read of the slave. The I²C data lines operate at 5V normally or 3.3V for -1U style units. The GLK12232-25-SM uses 8-bit addressing, with the 8th or Least Significant Bit (LSB) bit designated as the read/write bit, a 0 designates a write address and a 1 designates a read address. The default read address of the display module will be 0x51, whereas the write address is 0x50 by default. This address may be changed by using cmd 254 / 51 / <address>. The GLK12232-25-SM should only be sent addresses that are even (LSB is 0). When the I²C master wishes to write to the display, the effective address is $50 (0101 0000) , since the LSB has to be 0 for an I²C master write. When the I²C master wishes to read the GLK12232-25-SM, the effective address is $51 (0101 0001), since the LSB has to be 1 for an I²C master read.

If we take a standard Phillips 7 bit address of $45 (100 0101), Matrix Orbital's GLK12232-25-SM would describe this Phillips I²C address as $8A (1000 1010). The read address would be $8B (1000 1011).

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

When communicating in I²C the GLK12232-25-SM will send an ACK on the 9th clock cycle when addressed. When writing to the display module, the display will respond with a ACK when the write has successfully been completed. However if the buffer has been filled, or the module is too busy processing data it will respond with a NAK. When performing a multiple byte read within one I²C transaction, each byte read from the slave should be followed by an ACK to indicate that the master still needs data, and a NAK to indicate that the transmission is over.

The GLK12232-25-SM has some speed limitations, especially when run in I²C mode. Here are some considerations when writing I²C code:

* to be able to read the replies of query commands (eg. cmds 54, 55) the following command must be sent (only needs to be sent once, so this can be done somewhere in init): 254 / 160 / 0 this command puts the reply data in the I²C output buffer instead of the RS232 output buffer. Please note that due to a 16 byte output buffer, query commands that reply with more than 16 bytes cannot be read (eg cmd Get FileSystem Directory)
* 3ms delay between the read commands
* 625us delay in between data bytes within a transaction is necessary
* 375us between transactions is necessary

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

4.1.2 I²C Transaction Example

The typical I²C transaction contains four parts: the start sequence, addressing, information, and stop sequence. To begin a transaction the data line, SDA, must toggle from high to low while the clock line, SCL,

is high. Next, the display must be addressed using a one byte hexadecimal value, the default to write to the unit is 0x50, while read is 0x51. Then information can be sent to the unit; even when reading, a command must first be sent to let the unit know what type of information it is required to return. After each bit is sent, the display will issue an ACK or NACK as described above. Finally, when communication is complete, the transaction is ended by toggling the data line from low to high while the clock line is high. An example of the use of this algorithm to write a simple "HELLO" message can be seen in 3.

Table 3: I^2C Transaction Algorithm

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

4.1.3 Serial Communication

In addition to being able to communicate via I^2C the GLK12232-25-SM communicates natively through the RS-232 protocol at at a default baud rate of 19,200 bps and is capable of standard baud rates from 9600 to 115,200 bps. Furthermore the GLK12232-25-SM is also capable of reproducing any non-standard baud rate in between using values entered into our baud rate generation algorithm and set through command 164 (0xA4). The display module communicates at standard voltage levels of -30V to +30V or at TTL levels of 0 to +5V by setting the *Protocol Select Jumpers* to TTL.

4.2 Turn Flow Control On

Syntax

Hexadecimal	0xFE 0x3A [full] [empty]
Decimal	254 58 [full] [empty]
ASCII	254 ":" [full] [empty]

Parameters

Parameter	Length	Description
full	1	Bytes remaining before issuing a almost full message. (Full is 0)
empty	1	Bytes available before issuing a almost empty message. (Empty is 128)

Description

This command enables flow control. When the buffer fills so that only [full] bytes are available, the display will return an “almost full” message (0xFE) to the host controller. When the buffer empties so that only [empty] bytes remain, the display will return an “almost empty” message (0xFF) to the host controller.
The display will return the “almost full” message for every byte sent to the display until the used buffer space once more drops below the [full] level. Whether the user is in ‘flow control mode’ or not, the module will ignore display or command bytes which would overrun the buffer.
While in ‘flow control mode’ the unit will return 0xFE when buffer is almost full even though it may have already thrown rejected data away.
The buffer size for the display is 128 bytes.
When using this command in an application, selection of the value for the buffer [full] should be considered very carefully. This is a critical aspect to be able to use this feature to it’s full potential. When using a host system or PC which contains a FIFO, the user should set the value of [full] equal to or greater than the size of the FIFO. The reason for this is that the FIFO may be full when the host system receives 0xFE. In the case of 16550 UART the size at its maximum is 16, therefore the value of [full] should be set to 16 or greater. It is suggested that the “almost full” parameter be equal to the largest chunk of data the host will be sending the display (should be less than 127).

NOTE This command is not available in I^2C mode.

Remembered Yes
Default Off

4.3 Turn Flow Control Off

Syntax

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

Description

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

NOTE This command is not available in I^2C mode.

Remembered Yes

4.4 Changing the I^2C Slave Address

Syntax

Hexadecimal	0xFE 0x33 [adr]
Decimal	254 51 [adr]
ASCII	254 “3” [adr]

Parameters

Parameter	Length	Description
adr	1	The new I^2C write address (0x00 - 0xFF).

Description

This command sets the I^2C write address of the module between 0x00 and 0xFF. The I^2C write address must be an even number and the read address is automatically set to one higher. For example if the I^2C write address is set to 0x50, then the read address is 0x51.

NOTE The change in address is immediate.

Remembered Always
Default 0x50

4.5 Changing the Baud Rate

Syntax

Hexadecimal	0xFE 0x39 [speed]
Decimal	254 57 [speed]
ASCII	254 “9” [speed]

Parameters

Parameter	Length	Description
speed	1	Hex value corresponding to a baud rate.

Description

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

Hex Value	Baud Rate
0xCF	9600
0x8A	14400
0x67	19200
0x44	28800
0x33	38400
0x22	57600
0x19	76800
0x10	115200

NOTE This command is not available in I^2C mode.

Remembered Always
Default 19,200 bps

4.6 Setting a Non-Standard Baud Rate

Syntax

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

Parameters

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

Description

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

Remembered Always

Examples

Crystal Speed 16 Mhz

Desired BAUD 13,500

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

$$speed = 148.15 - 1 \qquad speed = 147.15$$

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

NOTES

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

NOTES

- This command is not available in I^2C mode.

5 Fonts

5.1 Introduction

The GLK12232-25-SM comes loaded with the 'Small Filled' and 'Futura Bk BT 16' fonts by default. However, it is capable of displaying any font that is uploaded to it in the correct format. MOGD# provides a simple method of generating font files from your installed fonts. For instructions on how to install MOGD# see ***Section 1.5.1 on page 4***.

5.1.1 Font File Format

A font file consists of three parts, a header, a character table and bitmap data.

1. Header (4 bytes)

   (a) Nominal Width (1 byte)
   (b) Height (1 byte)
   (c) ASCII Start Value (1 byte)
   (d) ASCII End Value (1 byte)

2. Character Table (3 bytes for every character between the ASCII Start and End values inclusive)

   (a) High Offset MSB (1 byte)
   (b) Low Offset LSB(1 byte)
   (c) Character Width (1 byte)

3. Bitmap Data

5.1.2 Creating a Font

The following is an example of how to create a font file for the letters *h*, *i* and *j*.

First you must create the bitmaps containing the character data in bitmap form. ***Figure 17*** below illustrates the bit pattern for the *h*, *i* and *j* bitmap data.

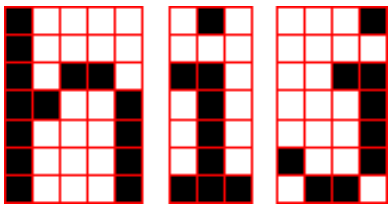


Figure 17: Bitmaps for h, i, and j

Second you may begin to create the font file starting with the header. The header will contain the nominal width, the height and the ASCII start and end values inclusive that you wish to create characters for.

Table 9: Font File Header

Nominal Width	Height	ASCII Start Val	ASCII End Val
0x05	0x07	0x68	0x6A

Next we will have to find out how many bytes each character will use up, in order to create the character table. The bitmaps are encoded horizontally and may have variable widths, *h* has a width of five, *i* a width of three and *j* a width of four, see the figure below for an example of encoding the first letter *h:*