



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





**LCD0821**  
**Technical Manual**

**Revision: 2.1**

# Contents

<b>Contents</b>	<b>ii</b>
<b>1 Getting Started</b>	<b>1</b>
1.1 Display Options Available . . . . .	1
1.2 Accessories . . . . .	2
1.3 Features . . . . .	3
1.4 Connecting to a PC . . . . .	4
1.5 Installing the Software . . . . .	5
1.5.1 uProject . . . . .	5
<b>2 Hardware Information</b>	<b>6</b>
2.1 Power/Data Connector . . . . .	6
2.2 RS-232 Header . . . . .	7
2.3 Manual Override . . . . .	7
2.4 General Purpose Output . . . . .	8
2.5 Protocol Select Jumpers . . . . .	9
<b>3 Troubleshooting</b>	<b>10</b>
3.1 The display does not turn on when power is applied. . . . .	10
3.2 The display module is not communicating. . . . .	10
3.3 The display module is communicating, however text cannot be displayed. . . . .	11
<b>4 Communications</b>	<b>11</b>
4.1 Introduction . . . . .	11
4.1.1 I <sup>2</sup> C Communication Summary . . . . .	11
4.1.2 I <sup>2</sup> C Transaction Example . . . . .	12
4.1.3 Serial Communication . . . . .	13
4.2 Changing the I <sup>2</sup> C Slave Address . . . . .	13
4.3 Changing the Baud Rate . . . . .	13
4.4 Setting a Non-Standard Baud Rate . . . . .	14
<b>5 Text</b>	<b>15</b>
5.1 Introduction . . . . .	15
5.1.1 Character Set . . . . .	16
5.1.2 Control Characters . . . . .	17
5.2 Auto Scroll On . . . . .	18
5.3 Auto Scroll Off . . . . .	18
5.4 Clear Screen . . . . .	18
5.5 Changing the Startup Screen . . . . .	18
5.6 Set Auto Line Wrap On . . . . .	19
5.7 Set Auto Line Wrap Off . . . . .	19
5.8 Set Cursor Position . . . . .	19
5.9 Go Home . . . . .	20
5.10 Move Cursor Back . . . . .	20

5.11	Move Cursor Forward . . . . .	20
5.12	Underline Cursor On . . . . .	21
5.13	Underline Cursor Off . . . . .	21
5.14	Blinking Block Cursor On . . . . .	21
5.15	Blinking Block Cursor Off . . . . .	21
<b>6</b>	<b>Special Characters</b>	<b>22</b>
6.1	Introduction . . . . .	22
6.2	Creating a Custom Character . . . . .	22
6.3	Saving Custom Characters . . . . .	23
6.4	Loading Custom Characters . . . . .	24
6.5	Save Startup Screen Custom Characters . . . . .	24
6.6	Initialize Medium Number . . . . .	25
6.7	Place Medium Numbers . . . . .	25
6.8	Initialize Horizontal Bar . . . . .	26
6.9	Place Horizontal Bar Graph . . . . .	26
6.10	Initialize Narrow Vertical Bar . . . . .	26
6.11	Initialize Wide Vertical Bar . . . . .	27
6.12	Place Vertical Bar . . . . .	27
<b>7</b>	<b>General Purpose Output</b>	<b>28</b>
7.1	Introduction . . . . .	28
7.2	General Purpose Output Off . . . . .	28
7.3	General Purpose Output On . . . . .	28
7.4	Set Startup GPO state . . . . .	29
<b>8</b>	<b>Display Functions</b>	<b>29</b>
8.1	Introduction . . . . .	29
8.2	Display On . . . . .	29
8.3	Display Off . . . . .	30
8.4	Set Brightness . . . . .	30
8.5	Set and Save Brightness . . . . .	30
8.6	Set Contrast . . . . .	31
8.7	Set and Save Contrast . . . . .	31
<b>9</b>	<b>Data Security</b>	<b>32</b>
9.1	Introduction . . . . .	32
9.2	Set Remember . . . . .	32
9.3	Data Lock . . . . .	33
9.4	Set and Save Data Lock . . . . .	34
9.5	Write Customer Data . . . . .	35
9.6	Read Customer Data . . . . .	35
<b>10</b>	<b>Miscellaneous</b>	<b>35</b>
10.1	Introduction . . . . .	35
10.2	Read Version Number . . . . .	35
10.3	Read Module Type . . . . .	36

<b>11 Command Summary</b>	<b>38</b>
11.1 Communications . . . . .	38
11.2 Text . . . . .	38
11.3 Special Characters . . . . .	39
11.4 General Purpose Output . . . . .	40
11.5 Display Functions . . . . .	40
11.6 Data Security . . . . .	41
11.7 Miscellaneous . . . . .	41
11.8 Command By Number . . . . .	41
<b>12 Appendix</b>	<b>43</b>
12.1 Specifications . . . . .	43
12.1.1 Environmental . . . . .	43
12.1.2 Electrical . . . . .	43
12.1.3 Optical . . . . .	43
12.2 Physical Layout . . . . .	44
12.3 Ordering Information . . . . .	46
12.4 Definitions . . . . .	46
12.5 Contacting Matrix Orbital . . . . .	47
12.6 Revision History . . . . .	47



# 1 Getting Started

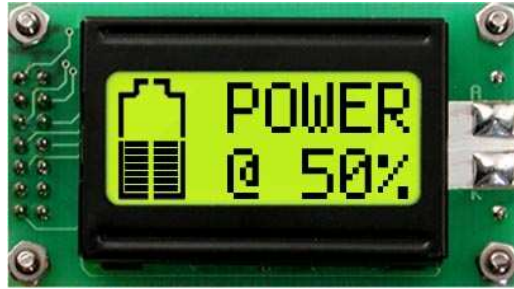


Figure 1: LCD0821

The LCD0821 is an intelligent LCD display designed to decrease development time by providing an instant solution to any project. With the ability to communicate via serial RS-232/TTL and I<sup>2</sup>C protocols, the versatile LCD0821 can be used with virtually any controller. The ease of use is further enhanced by an intuitive command structure to allow display settings such as backlight brightness, contrast and baud rate to be software controlled. Additionally, up to forty custom characters such as character sets for bar graphs, medium and large numbers may be stored in the non-volatile memory to be easily recalled and displayed at any time.

## 1.1 Display Options Available

The LCD0821 comes in a wide variety of colors including the standard yellow/green or inverse yellow, the popular blue/white and the crisp white/grey as well as inverse red which is excellent for viewing at night. Extended voltage, and temperature options are also available, to allow you to select the display which will best fit your project needs.

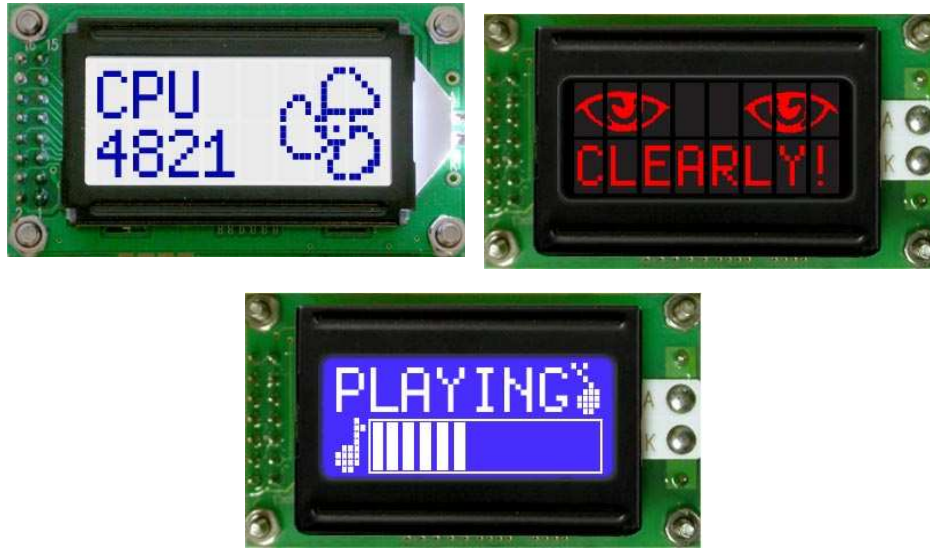


Figure 2: LCD0821 Options

## 1.2 Accessories

---

**NOTE** Matrix Orbital provides all the interface accessories needed to get your display up and running. You will find these accessories and others on our e-commerce website at <http://www.matrixorbital.com>. To contact a sales associate see Section 12.5 on page 47 for contact information.

---



Figure 3: 5V Power Cable Adapter



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



Figure 5: Breadboard Cable

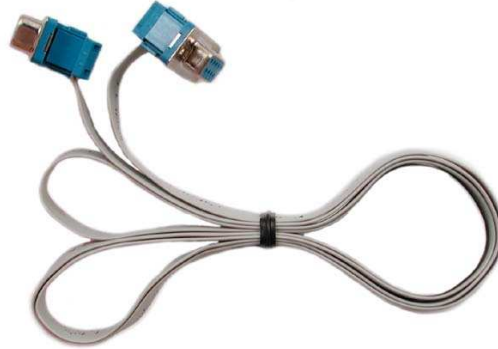


Figure 6: Serial Cable 4FT

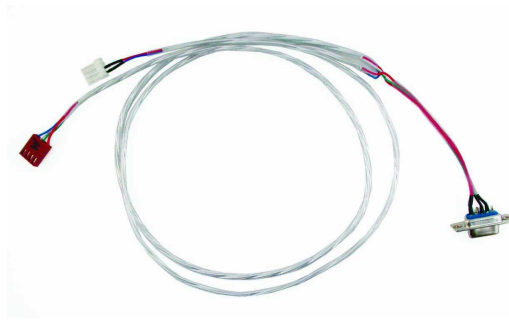


Figure 7: Communication and 5V Power Cable

### 1.3 Features

- 8 column by 2 line alphanumeric liquid crystal display
- Selectable communication protocol, serial at RS-232 or TTL levels or I<sup>2</sup>C
- One, 5V -20mA, general purpose output for a variety of applications
- Lightning fast communication speeds, up to 57.6 kbps for RS-232 and 100 kbps for I<sup>2</sup>C
- Extended temperature available for extreme environments of -20C to 70C
- Extended voltage power supply available
- Built in font with provision for up to 8 user defined characters
- Up to 127 modules addressable on the same 2 wire I<sup>2</sup>C interface
- Ability to add a customized splash / startup screen
- Software controlled contrast and brightness with configurable time-out setting up to 90 minutes
- Horizontal or vertical bar graphs

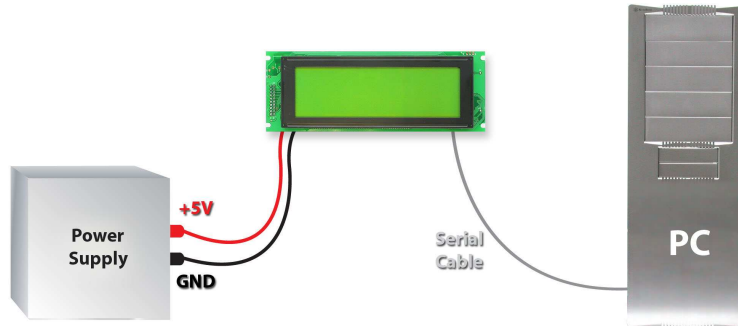


## 1.4 Connecting to a PC

The LCD0821 connects seamlessly to a PC and it is an excellent means of testing the functionality. To connect your display to a PC, you will require a standard RS-232 9-pin serial cable such as the one pictured in *figure 6 on the previous page*, as well as a modified 5V power adapter such as the one pictured in *figure 3 on page 2*.

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

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



---

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

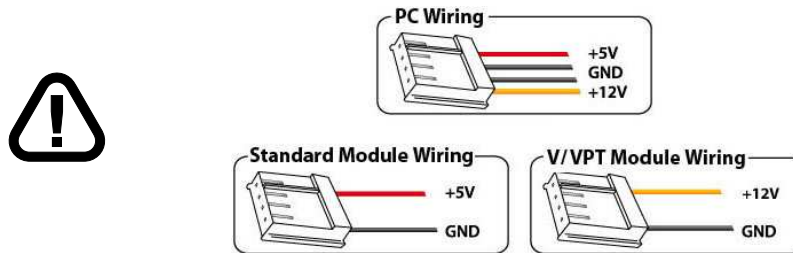


Figure 8: PC vs Matrix Orbital Display Module Wiring

## 1.5 Installing the Software

### 1.5.1 uProject

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

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

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

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



<b>Comport</b>	The serial port the display is plugged in to.
<b>Baudrate</b>	The communication speed the display module is set to. (Default 19,200)

Figure 9: uProject Settings

---

#### NOTES

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

## 2 Hardware Information

Refer to the following diagram for this chapter:

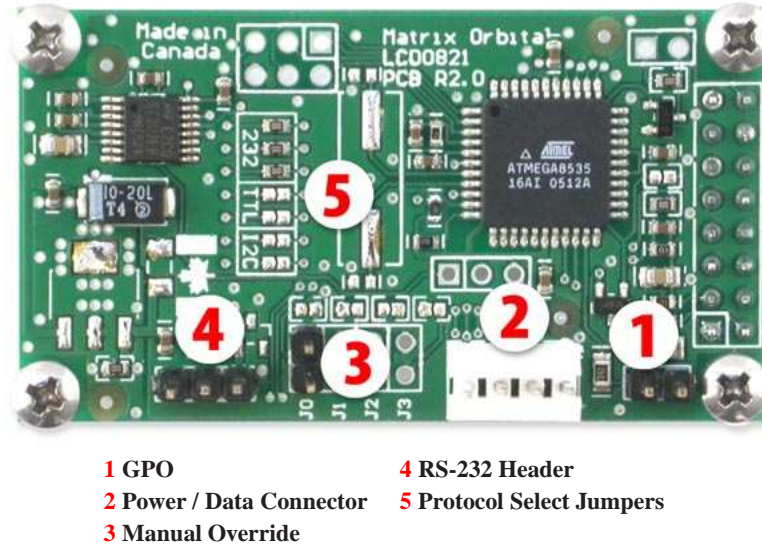
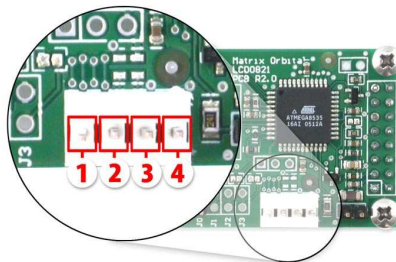


Figure 10: LCD0821

### 2.1 Power/Data Connector

The *Power/Data Connector* provides a standard connector for powering the display module. The LCD0821 requires five volts for the standard display module and between nine and fifteen for the wide voltage (V) model. The voltage is applied through pins one and four of the four pin *Power/Data connector*. Pins two and three are reserved for serial transmission, using either the RS-232/TTL or the I<sup>2</sup>C protocol, depending on what has been selected by the *Protocol Select Jumpers*. Pins two and three may be reversed by changing the *Legacy Connector Jumpers* in order to be compatible with previous PCB revisions.



- Pin 1** PWR (See table 1 on the next page)
- Pin 2** Rx \ SCL (I<sup>2</sup>C clock)
- Pin 3** Tx \ SDA (I<sup>2</sup>C data)
- Pin 4** GND

Figure 11: Power Connector and Pin out

Table 1: Power Requirements

	Standard	-V
Supply Voltage	+5Vdc $\pm$ 0.25V	+9V to +15V
Backlight On Supply	105 mA typical	
Backlight Off Supply	30 mA	

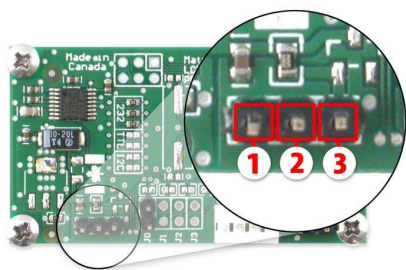


#### WARNINGS

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

## 2.2 RS-232 Header

The *RS-232 Header* provides an alternate serial transmission connector.



- Pin 1** GND
- Pin 2** Rx
- Pin 3** PWR (See table 1)

Figure 12: Power Connector and Pin out

**NOTE** This header is for RS-232 transmission only.

## 2.3 Manual Override

The *Manual Override* is provided to allow the LCD0821 to be reset to factory defaults. This can be particularly helpful if the display module has been set to an unknown baud rate or I<sup>2</sup>C Slave Address and you are no longer able to communicate with it. If you wish to return the module to its default settings you must:

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

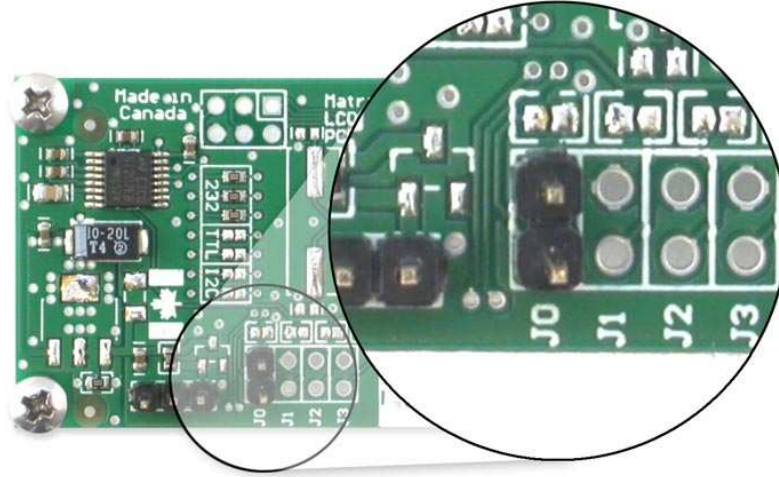


Figure 13: Manual Override Jumper

Table 2: Default Values

<b>Brightness</b>	255
<b>Baud Rate</b>	19.2 kbps
<b>I<sup>2</sup>C Slave Address</b>	0x50
<b>Data Lock</b>	False
<b>RS232AutoTransmitData</b>	True

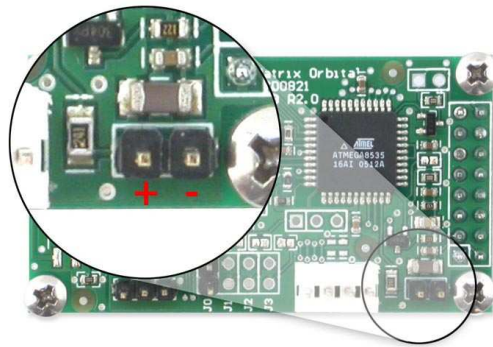
---

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

---

## 2.4 General Purpose Output

A unique feature of the LCD0821 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 as pictured below in *figure 14*. 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.



- GND
- + MAX: 20 mA, +5Vdc

Figure 14: General Purpose Output



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

---

## 2.5 Protocol Select Jumpers

The *Protocol Select Jumpers*, pictured below in *figure 15*, provide the means necessary to toggle the display module between RS-232, TTL, and I<sup>2</sup>C protocols. As a default, the jumpers are set to RS-232 mode with solder jumps on the 232 jumpers. In order to place the display module in I<sup>2</sup>C mode you must first remove the solder jumps from the 232 jumpers and then place them on the I2C jumpers. The display will now be in I<sup>2</sup>C 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<sup>2</sup>C jumpers and solder them to the TTL jumpers.



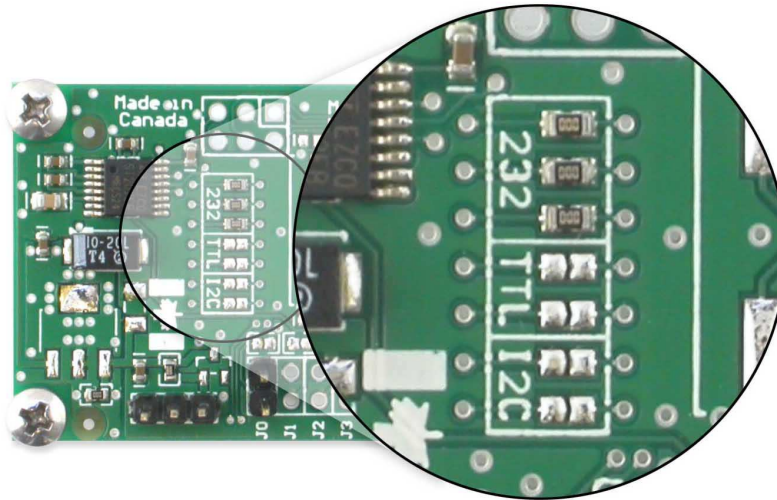


Figure 15: Protocol Select Jumpers

## 3 Troubleshooting

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

- First, you will want to make sure that you are using the correct power connector. Standard floppy drive power cables from your PC power supply may fit on the Power/Data Connector however they do not have the correct pin out as can be seen in *figure 8 on page 4*. 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 LCD0821. If the *Power / Data Connector* has become loose, or you are unable to resolve the issue, please contact Matrix Orbital, see *12.5 on page 47* 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 2.5 on the preceding page*.
- 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<sup>2</sup>C please ensure that the data is being sent to the correct address. The default slave address for the display module is 0x50.

---

**NOTE** I<sup>2</sup>C 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 2.3 on page 7*.

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

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

---

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

If you are unable to resolve any issue please contact Matrix Orbital. See *12.5 on page 47* for contact information.

---

## 4 Communications

### 4.1 Introduction

The commands listed in this chapter describe how to configure data flow on the LCD0821.

#### 4.1.1 I<sup>2</sup>C Communication Summary

The LCD0821 is capable of communicating at 100 KHz in I<sup>2</sup>C mode, with 127 units addressable on a single I<sup>2</sup>C communication line. However, in order to communicate via I<sup>2</sup>C 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 I<sup>2</sup>C. 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 I<sup>2</sup>C data lines operate at 5V normally or 3.3V for -1U style units. The LCD0821 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 LCD0821 should only be sent addresses that are even (LSB is 0). When the I<sup>2</sup>C master wishes to write to the display, the effective address is \$50 (0101 0000) , since the LSB has to be

0 for an I<sup>2</sup>C master write. When the I<sup>2</sup>C master wishes to read the LCD0821, the effective address is \$51 (0101 0001), since the LSB has to be 1 for an I<sup>2</sup>C master read.

If we take a standard Phillips 7 bit address of \$45 (100 0101), Matrix Orbital's LCD0821 would describe this Phillips I<sup>2</sup>C 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 I<sup>2</sup>C the LCD0821 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 I<sup>2</sup>C 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 LCD0821 has some speed limitations, especially when run in I<sup>2</sup>C mode. Here are some considerations when writing I<sup>2</sup>C 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 I<sup>2</sup>C output buffer instead of the RS232 output buffer. Please note that due to a 16 byte output buffer, query commands that reply with more than 16 bytes cannot be read (eg cmd Get FileSystem Directory)

- \* 3ms delay between the read commands

- \* 625us delay in between data bytes within a transaction is necessary

- \* 375us between transactions is necessary

---

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

---

## 4.1.2 I<sup>2</sup>C Transaction Example

The typical I<sup>2</sup>C 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<sup>2</sup>C Transaction Algorithm

<b>START</b>	Toggle SDA high to low
<b>Address</b>	0x50
<b>Information</b>	0x48 0x45 0x4C 0x4C 0x4F
<b>STOP</b>	Toggle SDA low to high

### 4.1.3 Serial Communication

In addition to being able to communicate via I<sup>2</sup>C the LCD0821 communicates natively through the RS-232 protocol at a default baud rate of 19,200 bps and is capable of standard baud rates from 9600 to 115,200 bps. Furthermore the LCD0821 is also capable of reproducing any non-standard baud rate in between using values entered into our baud rate generation algorithm and set through command 164 (0xA4). The display module communicates at standard voltage levels of -30V to +30V or at TTL levels of 0 to +5V by setting the *Protocol Select Jumpers* to TTL.

## 4.2 Changing the I<sup>2</sup>C Slave Address

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

## 4.3 Changing the Baud Rate

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

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

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

---

**NOTE** This command is not available in I<sup>2</sup>C mode.

---

Remembered Always  
 Default 19,200 bps

## 4.4 Setting a Non-Standard Baud Rate

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

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

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

Remembered Always

## Examples

**Crystal Speed** 16 Mhz

**Desired BAUD** 13,500

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

$$speed = 148.15 - 1$$

$$speed = 147.15$$

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

---

### NOTES

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

---

### NOTES

- This command is not available in I<sup>2</sup>C mode.
- 

## 5 Text

### 5.1 Introduction

The LCD0821 is an intelligent display module, designed to reduce the amount of code necessary to begin displaying data. This means that it is able to display all ASCII formatted characters and strings that are sent to it, which are defined in the current character set. The display module will begin displaying text at the top left corner of the display area, known as home, and continue to print to the display as if it was a page on a typewriter. When the text reaches the bottom right row, it is able to automatically scroll all of the lines up and continue to display text, with the auto scroll option set to on.



## 5.1.1 Character Set

		Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)																
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)	0	CG RAM (1)	⌘		0	E	P	'	P	G	é	á		ñ	ñ	ö	t	
	1	CG RAM (2)	⌘	.	1	A	Q	a	q	u	a	é	í		ü	t	y	o
	2	CG RAM (3)	⌘	"	2	R	b	r	e	é	é	ó		°	ø	ö	ý	
	3	CG RAM (4)	⌘	#	3	O	S	c	s	á	á	ó		'	P	l	e	q
	4	CG RAM (5)	⌘	\$	4	D	T	d	t	á	á	ó		'	e	r	z	o
	5	CG RAM (6)	⌘	%	5	E	U	e	u	á	á	ó		'	ñ	ñ	ñ	ñ
	6	CG RAM (7)	⌘	&	6	F	V	v	á	á	ó		'	ñ	ñ	ñ	ñ	
	7	CG RAM (8)	⌘	'	7	G	W	w	á	á	ó		'	ñ	ñ	ñ	ñ	
	8	CG RAM (1)	⌘	(	8	X	H	x	á	á	ó		'	ñ	ñ	ñ	ñ	
	9	CG RAM (2)	⌘	)	9	I	Y	y	á	á	ó		'	ñ	ñ	ñ	ñ	
	A	CG RAM (3)	⌘	*	A	Z	J	z	á	á	ó		'	ñ	ñ	ñ	ñ	
	B	CG RAM (4)	⌘	+	B	K	K	C	i	á	á	ó		'	ñ	ñ	ñ	
	C	CG RAM (5)	⌘	,	C	L	L	I	i	á	á	ó		'	ñ	ñ	ñ	
	D	CG RAM (6)	⌘	-	D	=	M	O	n	á	á	ó		'	ñ	ñ	ñ	
	E	CG RAM (7)	⌘	.	E	>	N	n	á	á	ó		'	ñ	ñ	ñ	ñ	
	F	CG RAM (8)	⌘	/	F	?	O	_	á	á	ó		'	ñ	ñ	ñ	ñ	

Figure 16: European Character Set

		Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)																		
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F			
Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)	0	CG RAM (1)			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	1	CG RAM (2)			1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
	2	CG RAM (3)			2	3	4	5	6	7	8	9	A	B	C	D	E	F		
	3	CG RAM (4)			3	4	5	6	7	8	9	A	B	C	D	E	F			
	4	CG RAM (5)			4	5	6	7	8	9	A	B	C	D	E	F				
	5	CG RAM (6)			5	6	7	8	9	A	B	C	D	E	F					
	6	CG RAM (7)			6	7	8	9	A	B	C	D	E	F						
	7	CG RAM (8)			7	8	9	A	B	C	D	E	F							
	8	CG RAM (1)			8	9	A	B	C	D	E	F								
	9	CG RAM (2)			9	A	B	C	D	E	F									
	A	CG RAM (3)			A	B	C	D	E	F										
	B	CG RAM (4)			B	C	D	E	F											
	C	CG RAM (5)			C	D	E	F												
	D	CG RAM (6)			D	E	F													
	E	CG RAM (7)			E	F														
	F	CG RAM (8)			F															

Figure 17: Japanese Character Set

---

**NOTE** The default font set for these units is Japanese, save for the -WB color option. Please consult the PCN pages at [www.matrixorbital.ca](http://www.matrixorbital.ca) or a sales representative to confirm font set availability.

---

### 5.1.2 Control Characters

In addition to a full text set, the LCD0821 display supports the following ASCII Control characters:

**0x08** Backspace

**0x0C** Clear screen / New page

**0x0D** Carriage return

**0x0A** Line feed / New line

## 5.2 Auto Scroll On

Syntax	Hexadecimal	0xFE 0x51
	Decimal	254 81
	ASCII	254 “Q”
Description	When auto scrolling is on, it causes the display to shift the entire display’s contents up to make room for a new line of text when the text reaches the end of the last row.	
Remembered	Yes	
Default	On	

## 5.3 Auto Scroll Off

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

## 5.4 Clear Screen

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

## 5.5 Changing the Startup Screen

Syntax	Hexadecimal	0xFE 0x40
	Decimal	254 64
	ASCII	254 “@”
Description	In order to change the text that is displayed by the LCD0821 when it starts up simply send the command bytes 254 64 followed by the characters that you wish to display, starting from the top left. This command will automatically line wrap the characters that are sent to it.	

Remembered Yes

## 5.6 Set Auto Line Wrap On

Syntax           Hexadecimal   0xFE 0x43  
                  Decimal       254 67  
                  ASCII        254 "C"

Description      Enabling Auto Line Wrap will allow the cursor to automatically wrap over to the next line when the current line is full.

---

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

---

Remembered Yes

## 5.7 Set Auto Line Wrap Off

Syntax           Hexadecimal   0xFE 0x44  
                  Decimal       254 68  
                  ASCII        254 "D"

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

Remembered Yes

## 5.8 Set Cursor Position

Syntax           Hexadecimal   0xFE 0x47 [col] [row]  
                  Decimal       254 71 [col] [row]  
                  ASCII        254 "G" [col] [row]

Parameters	Parameter	Length	Description
	col	1	Column
	row	1	Row

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

---

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

---

Remembered No

## 5.9 Go Home

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

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

Remembered No

## 5.10 Move Cursor Back

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

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

Remembered No

## 5.11 Move Cursor Forward

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

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

## 5.12 Underline Cursor On

Syntax	Hexadecimal 0xFE 0x4A Decimal 254 74 ASCII 254 “J”
Description	This command will cause the LCD0821 to display an underline cursor at the current text insertion point.
Remembered	Yes

## 5.13 Underline Cursor Off

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

## 5.14 Blinking Block Cursor On

Syntax	Hexadecimal 0xFE 0x53 Decimal 254 83 ASCII 254 “S”
Description	This command will cause the LCD0821 to display a block cursor at the current text insertion point.
Remembered	Yes

## 5.15 Blinking Block Cursor Off