

NetMedia RoHS Serial LCD Module V1.0

Table of Contents:

Pinout	2
Connections.....	2
Jumper Settings.....	3
Contrast Adjustment	3
Interfacing.....	4
LCD Control Codes	5
Creating Custom Characters	7
Complete BasicX-24 Example Program.....	8
2x16 Specifications.....	9
4x20 Specifications.....	9
2x16 Current Specs.....	9
4x20 Current Specs.....	9
Optional Bezel Dimensions	10
Dimensions	11

Warranty

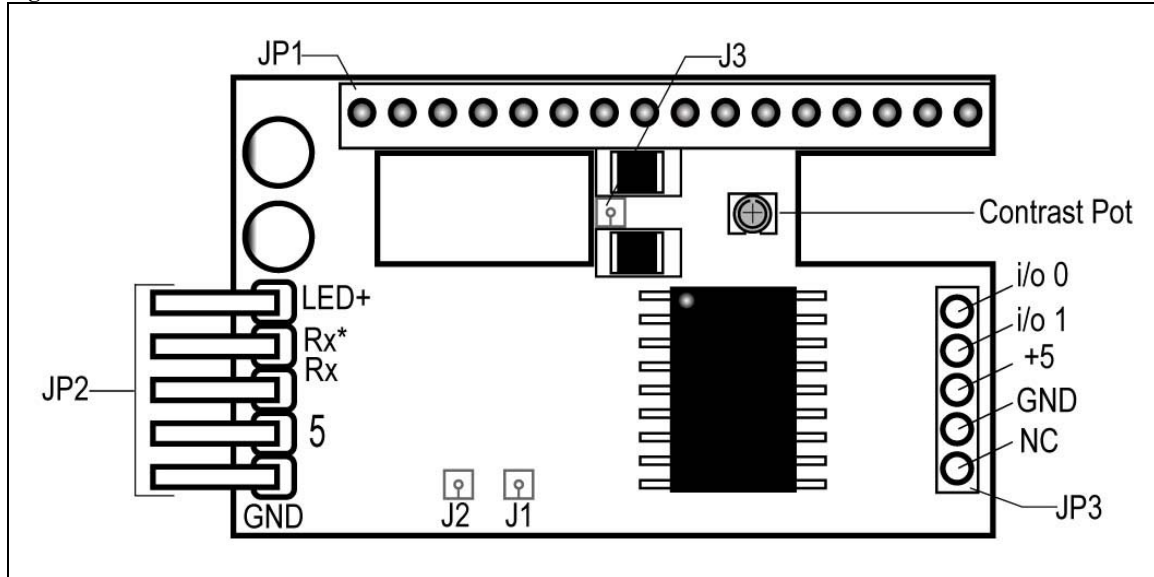
NetMedia, Inc. warrants this product against defects in materials and workmanship for a period of 90 days from purchase date. Repaired/replaced products will be returned via standard shipping. Expedited return shipping is available at customer's expense. Any product that has been abused, modified or had its polarity reversed is not covered under warranty. Our technicians check all returns. Items damaged by customer abuse/misuse will not be warranted and will only be returned at the customer's request and expense.

Extended warranties are available for large volume OEM customers. Please contact a NetMedia sales representative for more information.

Pinout

All power and communications connections are made via header JP2. Connection JP1 connects the serial module to the LCD header. JP3 contains additional power connectors and user TTL outputs 0 & 1.

Figure1



Connections

JP1 Pins 1 - 8	Description	JP1 Pins 9 -16	Description
Pin1	Ground	Pin9	D2 (Not Used)
Pin2	VCC (+5)	Pin10	D3 (Not Used)
Pin3	Contrast	Pin11	D4
Pin4	Data/Command (R/S)	Pin12	D5
Pin5	Read/Write (W)	Pin13	D6
Pin6	Enable (E1)	Pin14	D7
Pin7	D0 (Not Used)	Pin15	VCC (LEDSV+)
Pin8	D1 (Not Used)	Pin16	Ground

JP2 Pins	Description	Notes
GND	Power Ground	Reversing Polarity voids warranty!
+5	Power + 5 supply	Tie to +5vdc 15mA+ supply
RX	Serial Input	RS232 or logic level (if used leave RX* disconnected)
RX*	Serial Input (Inverted)	RS232 or logic level (if used leave RX disconnected)
LED+	Backlight Power	Tie to 5-14vdc 100mA+ supply

JP3 Pins	Description	Notes
1	User Output Pin 0	
2	User Output Pin 1	
3	Power +5 Supply	Ties to +5 pin on JP2
4	Power Ground	Ties to GND pin on JP2
5	Factory Program Pin	DO NOT USE!

Jumper Settings

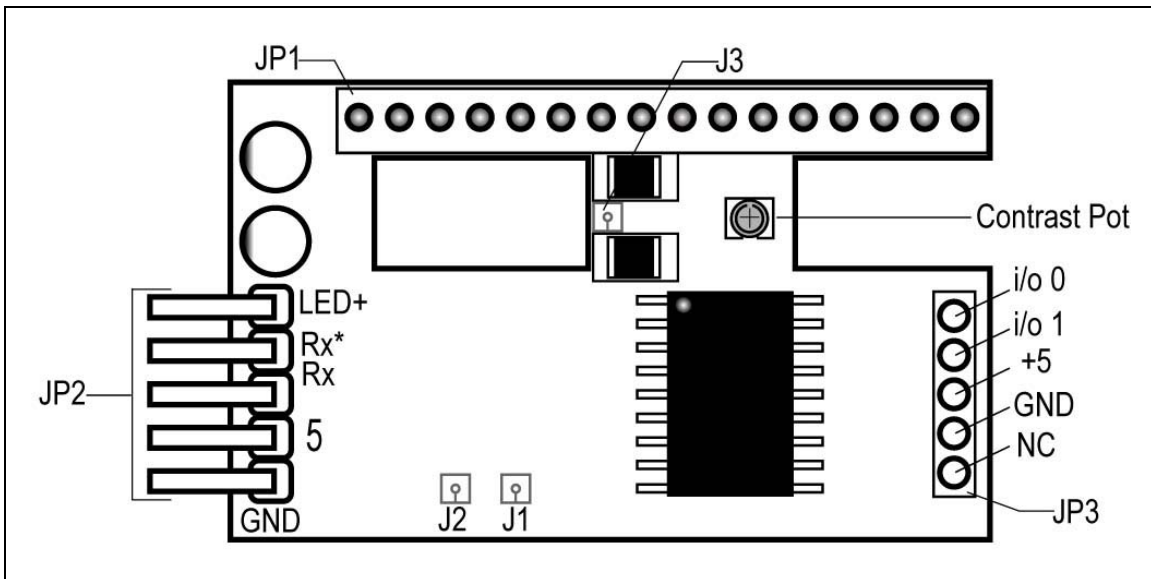
Jumper connections J1 and J2 are used to set the modules baud rate and display type, J3 sets the backlight current.

Note: Always disconnect power before changing jumper positions!

Jumpers J1, J2 & J3	Description	
J1	J1 Not Soldered = 9600 Baud, Soldered = 2400 Baud	
J2	J2 Not Soldered = 2x16, Soldered = 4x20	
J3	Backlight current, Not soldered = 2x16, Soldered = 4x20	

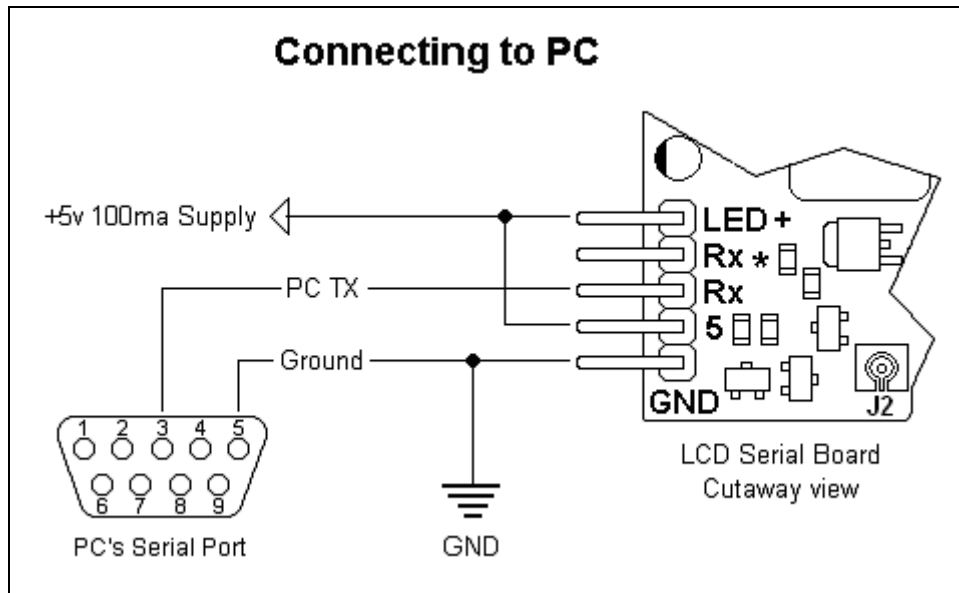
Contrast Adjustment

When purchased with an attached LCD the contrast is preset at the factory and should not need any major adjustment. The LCD contrast is adjusted using the potentiometer shown in Figure1 (Marked Contrast Pot). All adjustments must be made using a plastic potentiometer adjustment tool! Anything else may damage the potentiometer and void your warranty.



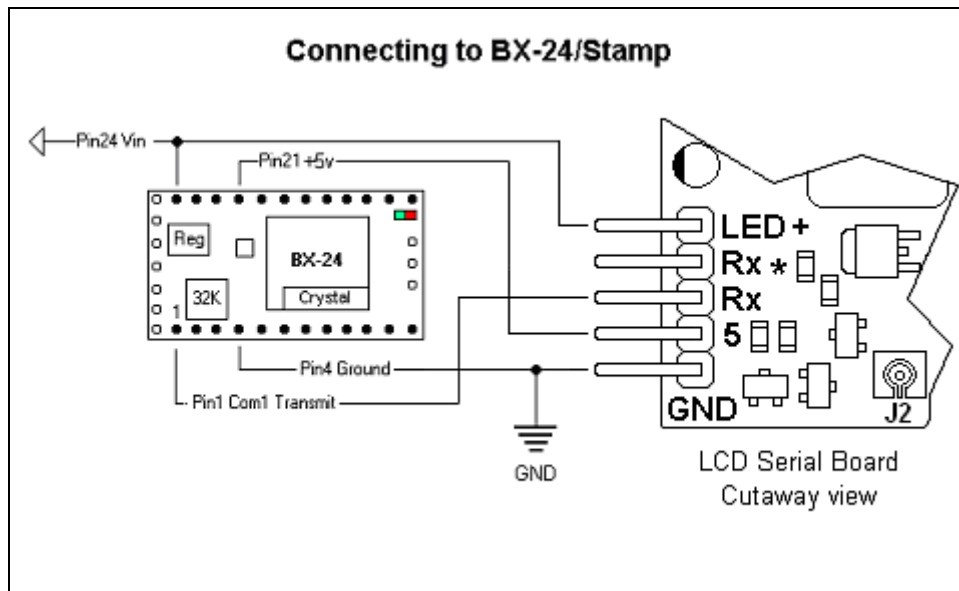
Interfacing

The following diagrams show two common methods for interfacing the 2x16 LCD.



Connection LEDSV+ is 5-14vdc.

Note: Never connect LED+ to +5 on the BX-24! The BX-24 +5 regulator cannot supply enough current to power the LCD backlight and will overheat.



LCD Control Codes

Description	Keyboard Code	ASCII or Decimal value
Display custom character 0-7	Ctrl-@ -Through- Ctrl-G	0 - 7
Back Space	Ctrl-H	8
Horizontal Tab	Ctrl-I	9
New Line	Ctrl-J	10
Vertical Tab	Ctrl-K	11
Form Feed (Clear Screen)	Ctrl-L	12
Carriage Return	Ctrl-M	13
Reset Controller	Ctrl-N	14
Set Geometry	Ctrl-O	15
Set Tab Size	Ctrl-P	16
Set Cursor Position	Ctrl-Q	17
Set Custom Character *New Command*	Ctrl-R	18
Set Contrast	Ctrl-S	19
Set Backlight	Ctrl-T	20
Command Escape	Ctrl-U	21
Data Escape	Ctrl-V	22
Raw Data Escape	Ctrl-W	23
Toggle TTL Outputs	Ctrl-X	24
Display an ASCII Character	None	25 - 255

Back Space Ctrl-H

Causes the cursor to move back once space. The cursor will wrap from the first column of a line to the last column of a previous line. Sending backspace when at the home position causes the cursor to wrap to the last character position of the last line.

Horizontal Tab Ctrl-I

Causes the cursor to move forward to the next tab position. If the cursor is near the end of the line and no more tab positions are on the line, then the cursor will advance to the next line. The LCD Controller is initial set up with tab positions at every 4th column. To set the tab position at a different column use the “Set Tab Size” command.

New Line Ctrl-J

Causes the cursor to advance to column 1 of the next line. If the cursor is on the last line, it will wrap to the home position.

Vertical Tab Ctrl-K

Causes the cursor to advance to the next line, but stay on the same column. If the cursor is on the last line of the screen, it will wrap to the first line of the screen.

Form Feed Ctrl-L

Causes the screen to be cleared and the cursor positioned to the home position. The form feed command takes some time to complete. It may take up to 2ms to complete. Since the LCD Controller has a finite amount of internal buffer space (16 bytes) for storing commands and data before sending to the LCD, you may overrun the internal buffer when sending multiple form feed commands in succession followed by other data.

Carriage Return Ctrl-M
Causes the cursor to go to column 1 of the current line

Reset Controller Ctrl-N

Resets the LCD controller as if it had been just power on. This command will also cause the hardware jumpers to be reread, so you can use it if you want to change baud rate or display size. This command takes about 1 second to complete.

Set Geometry Ctrl-O

Defines the layout (or geometry of the LCD). This command needs to be issued before any other commands if the LCD is not a 20 character by 4 line LCD. You need to send 5 additional bytes after sending a "Set Geometry" command.

Example:

15,20,0x80, 0xC0,0x94,0xD4 Standard 20 column by 4 line display

15,16,0x80, 0xC0,0x94,0xD4 Standard 16 column by 4 line display

15,16,0x80, 0xC0,0x80,0x80 Standard 16 column by 2 line display

If the display is less than 4 lines then you still need to send the line 1 starting address for the unused starting addresses. See the appendix for the typical layout of various LCD modules.

Set Tab Size Ctrl-P

Sets the size of a tab. You need to send a byte that is the negative of the tab size. The default tab size is 4. The tab size should be a power of 2 (i.e. 0, 1, 2, 4, 8, 16)

Set Cursor Position Ctrl-Q

Sets the cursor position. The following 2 bytes specify the zero based row and column of the cursor position. The bytes need to be within the display range. Sending bytes outside the display range will position the cursor to unpredictable locations.

Set Custom Character Ctrl-S ** New Command ******

Sets custom character within the LCD module. To define a custom character you send a Dec 18 followed by the new characters address and the eight byte that make up that character. Explained later in this text.

Set Contrast Ctrl-S ** No longer Used ******

Sets the display contrast. The byte following the "Set Contrast" command will set the display contrast. A contrast of 0 is no contrast and a contrast of 255 is full contrast. The contrast is automatically set to 50% at power up.

Set Backlight Ctrl-T

Sets the display backlight brightness. The byte following the "Set Brightness" command will set the display brightness. A brightness level of 0 will turn off the backlight completely. A brightness level of 255 is full brightness. The brightness is set to 80% after power up.

Command Escape Ctrl-U

The following byte is sent to the LCD controller as a raw LCD controller command. See the appendix for a list of commands that the LCD controller supports. You will mostly use this command to define custom characters and to set the cursor shape and visibility.

Data Escape Ctrl-V

The following byte is treated as data. This command is used to send bytes that would normally be interpreted as commands. Some LCD displays (in particular the European font LCDs) have characters in the same range as the commands of the LCD controller. This command allows these characters to be sent. After data is output to the LCD controller, the cursor is updated properly.

Raw Data Escape Ctrl-W

The following byte is treated as raw data. This command is used to send bytes that are used for the creation of custom characters. No attempt is made to advance the cursor since this would interfere with custom character creation.

Toggle TTL Outputs Ctrl-X ** New Command ******

This command sets the state of on-board TTL user outputs. Outputs can sink/source 25mA and can be used for operating piezo buzzers, relays, LEDs or whatever other device you can come up with. After sending the Ctrl_X byte the next byte selects the TTL output number (0 or 1) and the last byte selects the state (0 for low and 1 for high). The Serial LCD module sets both user outputs to low at power up.

Creating Custom Characters

Creating custom character is an easy way to add missing letters, symbols or characters to your display. Up to 8 characters can be created. A character is added by sending Chr(18) + Chr(RAM Storage Address) + Character (row byte) x 8. The RAM Storage address is the location where the character will be stored.

Character RAM StorageAddresses Table

Character Number	Character RAM storage address
0	64 (Dec)
1	72 (Dec)
2	80 (Dec)
3	88 (Dec)
4	96 (Dec)
5	104 (Dec)
6	112 (Dec)
7	120 (Dec)

Calculating Character Row Byte Value

Pixel Position Value		Pixel Position Value									
16	8	4	2	1		16	8	4	2	1	
1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Row 1 Value = 4	1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Row 1 Value = 0
2	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Row 2 Value = 4	2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Row 2 Value = 0
3	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Row 3 Value = 4	3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Row 3 Value = 10
4	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Row 4 Value = 4	4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Row 4 Value = 0
5	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Row 5 Value = 4	5	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Row 5 Value = 17
6	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Row 6 Value = 31	6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Row 6 Value = 14
7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Row 7 Value = 14	7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Row 7 Value = 0
8	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Row 8 Value = 4	8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Row 8 Value = 0

Custom Character BasicX Code Snippet:

```
' Load Down Arrow as char 0  
Call PutQueueStr(Com3Out,Chr(18) & Chr(64) & Chr(4) & Chr(4) & Chr(4) & Chr(4) & Chr(4) & Chr(31) & Chr(14) & Chr(4))
```

```
' Print char 0 (new down arrow)  
Call PutQueueStr(Com3Out,Chr(0))
```

Complete BasicX-24 Example Program

' This program turns on the LCD 2X16 backlight and displays a "Hello World" message.

' Connections:

```
' LCD Gnd to BasicX-24 Pin23  
' LCD +5 to BasicX-24  
' LCD LEDSV+ to BasicX-24 Pin24 or 5-12Vdc 150mA + supply  
' LCD RX to BasicX-24 Pin 5
```

' Define Com3 buffer sizes

```
Dim Com3In(1 to 15) As Byte  
Dim Com3Out(1 to 40) As Byte
```

' Define the LCD control constants we will use

```
Const BackLite As Byte = 20  
Const Clear_LCD As Byte = 12  
Const Set_Cursor As Byte = 17
```

Sub Main()

' Wait 1/2 second after power up for LCD to stabilize

```
Call Sleep(256)  
' Open Com3 Buffers  
Call OpenQueue(Com3In, 15)  
Call OpenQueue(Com3Out, 40)  
' Set Com3 to Inverted Logic, 8 Data Bits, No Parity, Pin12 TX, 0 = NO RX pin  
Call DefineCom3(0, 5, bx1000_1000)  
' Open Com3  
Call OpenCom(3, 9600, Com3In, Com3Out)  
' Run greeting subroutine  
Call Greeting
```

```
Do ' We are done. Sit here and do nothing  
Loop
```

End Sub

Sub Greeting()

```
' Set backlight to full brightness  
Call PutQueueStr(Com3Out,Chr(BackLite) & Chr(255))  
' Send Clear LCD command and first 1/2 of message "Hello World!"  
Call PutQueueStr(Com3Out,Chr(Clear_LCD) & " Hello World!")  
' Move cursor to Row 2 column 4  
Call PutQueueStr(Com3Out,Chr(Set_cursor) & Chr(1) & Chr(3))  
' Display the rest of the message "I'm Alive!"  
Call PutQueueStr(Com3Out,"I'm Alive!")  
End Sub ' Return
```


2x16 Specifications

Power4.9-5.2 Vdc @5mA (No Backlight), 135mA (Full Backlight)
Serial Input.....8N1, 9600 or 2400 Baud, RS232 or TTL/CMOS level
Maximum Operating Temperature.....0° - 50° C

4x20 Specifications

Power4.9-5.2 Vdc @5mA (No Backlight), 225mA (Full Backlight)
Serial Input.....8N1, 9600 or 2400 Baud, RS232 or TTL/CMOS level
Maximum Operating Temperature.....0° - 50° C

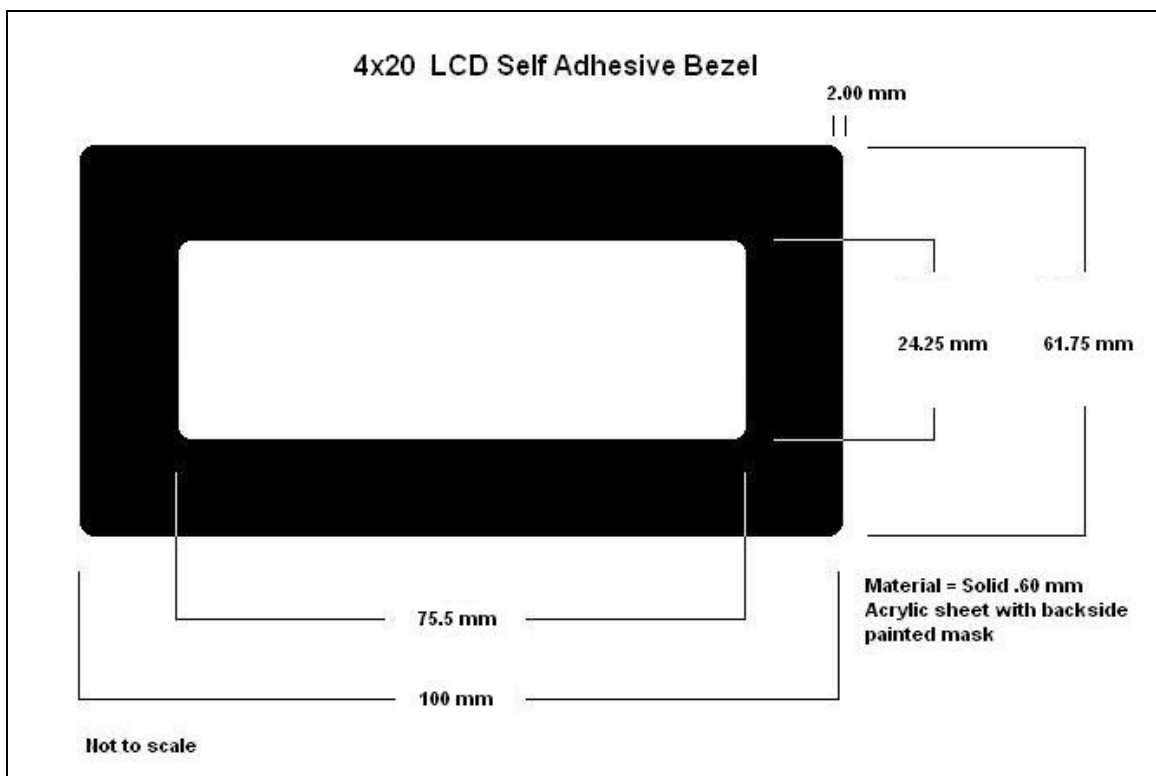
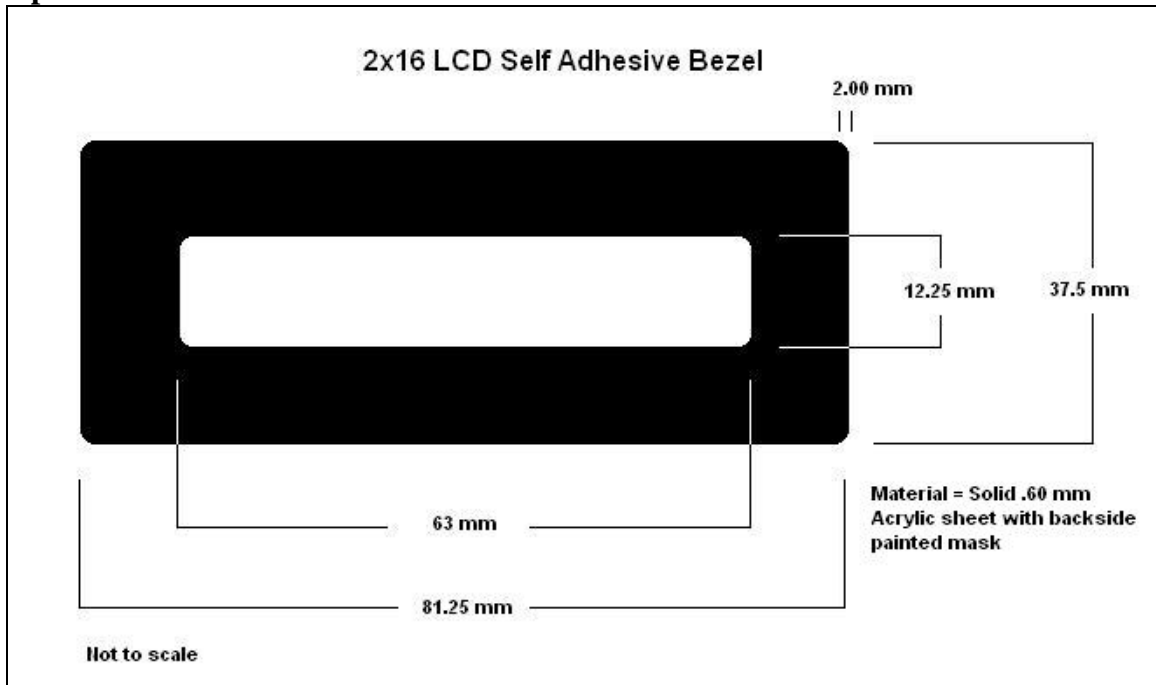
2x16 Current Specs

Current Usage		Notes
Backlight Off	4.75mA	All Colors
Backlight Full On	130mA	LCD with Blue Backlight
Backlight Full On	66mA	LCD with Red Backlight
Backlight Full On	66mA	LCD with Green Backlight

4x20 Current Specs

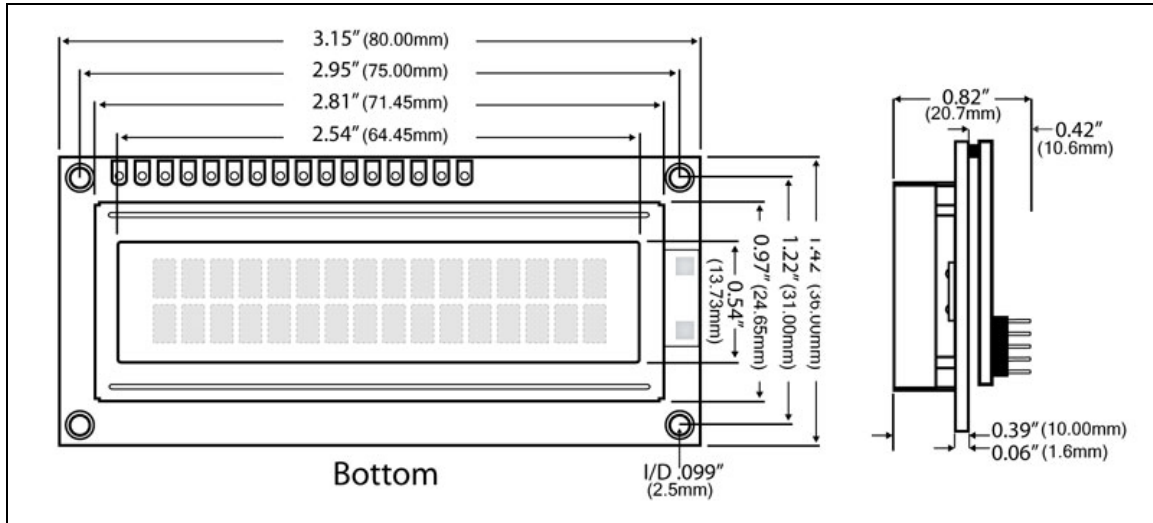
Current Usage		Notes
Backlight Off	4.75mA	All Colors
Backlight Full On	88mA	LCD with Blue Backlight
Backlight Full On	86mA	LCD with Red Backlight
Backlight Full On	225mA	LCD with Green Backlight

Optional Bezel Dimensions

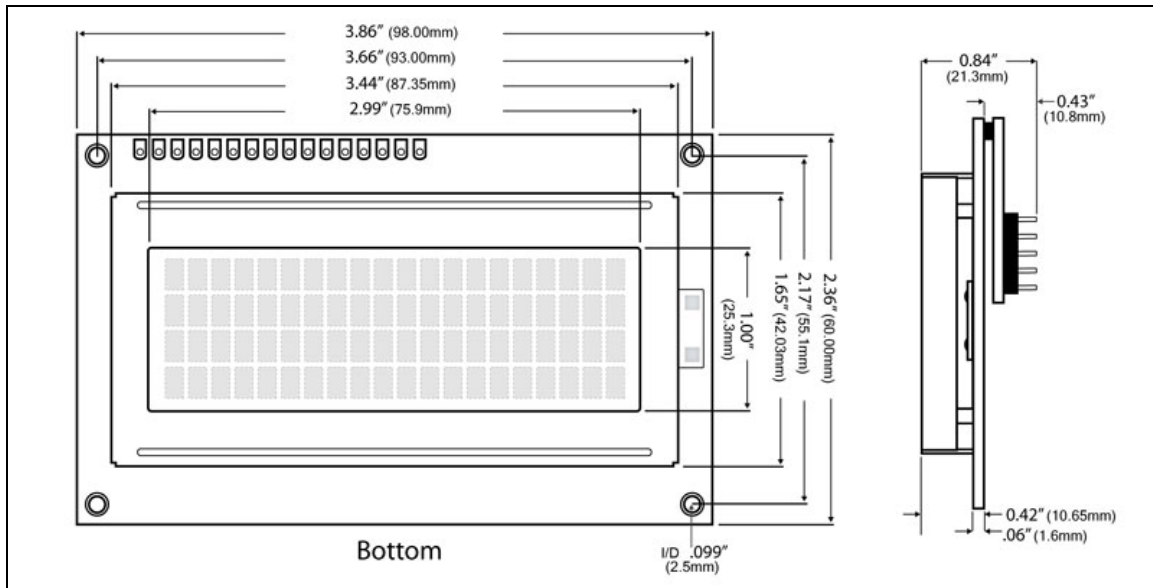


Dimensions

Serial 2x16



Serial 4x20



Copyright 2006 by NetMedia, Inc.
 10940 N. Stallard Place
 Tucson, AZ 85737
 (520) 544-4567