SLCD6 Controller Manual V2.21

August 14, 2009

For SLCD6 Hardware Revision A, RoHS Compliant

Table of Contents

Figures

Hardware Limited Warranty and Software License Agreement

0.1. *Hardware Limited Warranty*

REACH TECHNOLOGY, Inc. warrants its hardware products to be free from manufacturing defects in materials and workmanship under normal use for a period of one (1) year from the date of purchase from REACH. This warranty extends to products purchased directly from REACH or an authorized REACH distributor. Purchasers should inquire of the distributor regarding the nature and extent of the distributor's warranty, if any. REACH shall not be liable to honor the terms of this warranty if the product has been used in any application other than that for which it was intended, or if it has been subjected to misuse, accidental damage, modification, or improper installation procedures. Furthermore, this warranty does not cover any product that has had the serial number altered, defaced, or removed. This warranty shall be the sole and exclusive remedy to the original purchaser. In no event shall REACH be liable for incidental or consequential damages of any kind (property or economic damages inclusive) arising from the sale or use of this equipment. REACH is not liable for any claim made by a third party or made by the purchaser for a third party. REACH shall, at its option, repair or replace any product found defective, without charge for parts or labor. Repaired or replaced equipment and parts supplied under this warranty shall be covered only by the unexpired portion of the warranty. Except as expressly set forth in this warranty, REACH makes no other warranties, expressed or implied, nor authorizes any other party to offer any warranty, including any implied warranties of merchantability or fitness for a particular purpose. Any implied warranties that may be imposed by law are limited to the terms of this limited warranty. This warranty statement supercedes all previous warranties, and covers only the Reach hardware. The unit's software is covered by a separate license agreement.

0.2. *Returns and Repair Policy*

No merchandise may be returned for credit, exchange, or service without prior authorization from REACH. To obtain warranty service, contact the factory and request an RMA (Return Merchandise Authorization) number. Enclose a note specifying the nature of the problem, name and phone number of contact person, RMA number, and return address.

Authorized returns must be shipped freight prepaid to Reach Technology Inc. 4575 Cushing Parkway, Fremont, California 94538 with the RMA number clearly marked on the outside of all cartons. Shipments arriving freight collect or without an RMA number shall be subject to refusal. REACH reserves the right in its sole and absolute discretion to charge a 15% restocking fee, plus shipping costs, on any products returned with an RMA.

Return freight charges following repair of items under warranty shall be paid by REACH, shipping by standard ground carrier. In the event repairs are found to be non-warranty, return freight costs shall be paid by the purchaser.

0.3. *Software License Agreement*

PLEASE READ THIS SOFTWARE LICENSE AGREEMENT CAREFULLY BEFORE DOWNLOADING OR USING THE SOFTWARE .

This License Agreement ("Agreement") is a legal contract between you (either an individual or a single business entity) and Reach Technology Inc. ("Reach") for software referenced in this guide, which includes computer software and, as applicable, associated media, printed materials, and "online" or electronic documentation (the "Software").

BY INSTALLING, COPYING, OR OTHERWISE USING THE SOFTWARE, YOU AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, DO NOT INSTALL OR USE THE SOFTWARE. IF YOU HAVE PAID A FEE FOR THIS LICENSE AND DO NOT ACCEPT THE TERMS OF THIS AGREEMENT, REACH WILL REFUND THE FEE TO YOU PROVIDED YOU (1) DO NOT INSTALL THE SOFTWARE AND (2) RETURN ALL SOFTWARE, MEDIA AND OTHER DOCUMENTATION AND MATERIALS PROVIDED WITH THE SOFTWARE TO REACH TECHNOLOGY INC AT: REACH TECHNOLOGY INC., 4575 CUSHING PARKWAY, FREMONT, CALIFORNIA 94538.

Reach Technology Inc. ("Reach") and its suppliers grant to Customer ("Customer") a nonexclusive and nontransferable license to use the Reach software ("Software") in object code form on one or more central processing units owned or leased by Customer or otherwise embedded in equipment provided by Reach.

EXCEPT AS EXPRESSLY AUTHORIZED ABOVE, CUSTOMER SHALL NOT: COPY, IN WHOLE OR IN PART, SOFTWARE OR DOCUMENTATION; MODIFY THE SOFTWARE; REVERSE COMPILE OR REVERSE ASSEMBLE ALL OR ANY PORTION OF THE SOFTWARE; OR RENT, LEASE, DISTRIBUTE, SELL, OR CREATE DERIVATIVE WORKS OF THE SOFTWARE.

Customer agrees that aspects of the licensed materials, including the specific design and structure of individual programs, constitute trade secrets and/or copyrighted material of Reach. Customer agrees not to disclose, provide, or otherwise make available such trade secrets or copyrighted material in any form to any third party without the prior written consent of Reach. Customer agrees to implement reasonable security measures to protect such trade secrets and copyrighted material. Title to Software and documentation shall remain solely with Reach.

SOFTWARE LIMITED WARRANTY. Reach warrants that for a period of ninety (90) days from the date of shipment from Reach: (i) the media on which the Software is furnished will be free of defects in materials and workmanship under normal use; and (ii) the Software substantially conforms to its published specifications. Except for the foregoing, the Software is provided AS IS. This limited warranty extends only to Customer as the original licensee. Customer's exclusive remedy and the entire liability of Reach and its suppliers under this limited warranty will be, at Reach's option, repair, replacement, or refund of the Software. In no event does Reach warrant that the Software is error free or that Customer will be able to operate the Software without problems or interruptions.

This warranty does not apply if the software (a) has been altered, except by Reach, (b) has not been installed, operated, repaired, or maintained in accordance with instructions supplied by Reach, (c) has been subjected to abnormal physical or electrical stress, misuse, negligence, or accident, or (d) is used in ultrahazardous activities.

DISCLAIMER. EXCEPT AS SPECIFIED IN THIS WARRANTY, ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS, AND WARRANTIES INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE, ARE HEREBY EXCLUDED TO THE EXTENT ALLOWED BY APPLICABLE LAW.

IN NO EVENT WILL REACH OR ITS SUPPLIERS BE LIABLE FOR ANY LOST REVENUE, PROFIT, OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR PUNITIVE DAMAGES HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE EVEN IF REACH OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

 In no event shall Reach's or its suppliers' liability to Customer, whether in contract, tort (including negligence), or otherwise, exceed the price paid by Customer. The foregoing limitations shall apply even if the above-stated warranty fails of its essential purpose. SOME STATES DO NOT ALLOW LIMITATION OR EXCLUSION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES.

The above warranty DOES NOT apply to any beta software, any software made available for testing or demonstration purposes, any temporary software modules or any software for which Reach does not receive a license fee. All such software products are provided AS IS without any warranty whatsoever.

This License is effective until terminated. Customer may terminate this License at any time by destroying all copies of Software including any documentation. This License will terminate immediately without notice from Reach if Customer fails to comply with any provision of this License. Upon termination, Customer must destroy all copies of Software.

Software, including technical data, is subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Customer agrees to comply strictly with all such regulations and acknowledges that it has the responsibility to obtain licenses to export, re-export, or import Software.

This License shall be governed by and construed in accordance with the laws of the State of California, United States of America, as if performed wholly within the state and without giving effect to the principles of conflict of law. If any portion hereof is found to be void or unenforceable, the remaining provisions of this License shall remain in full force and effect. This License constitutes the entire License between the parties with respect to the use of the Software.

1. Introduction

1.1. *Overview*

The SLCD6 controller provides complete Graphical User Interface for embedded systems. It was designed to work with specific LCD TFT panels. Using the SLCD family of controllers is simply the quickest way to generate a user interface without a lot of graphical programming. It has a small size to fit in space-constrained applications.
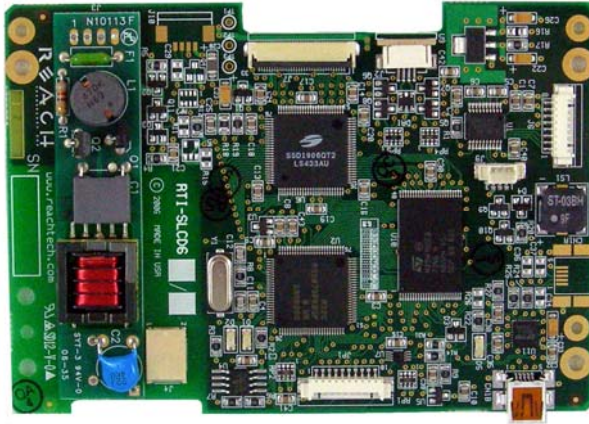
1.2. *Features*

♦ Fits exact footprint of LB040Q02-TD01 4" LCD Module

♦ Touch controller (4 wire resistive) on board

♦ Beeper for audible touch feedback and alarms

♦ On-board CCFL inverter

♦ Low power (40mA typical at 5V)

♦ RS232 / 3.3V CMOS up to 230,400 baud or USB level interface up to 460,800 baud

♦ User downloadable bitmaps with RLE compression (4Mb of flash memory)

♦ Backlight enable with wide range brightness control

♦ Low cost

♦ Supports either landscape or portrait mode display (firmware option)

♦ Can be modified for specific OEM requirements

♦ RoHS Compliant as of Jan 1, 2007

1.3. *Board dimensional drawing*
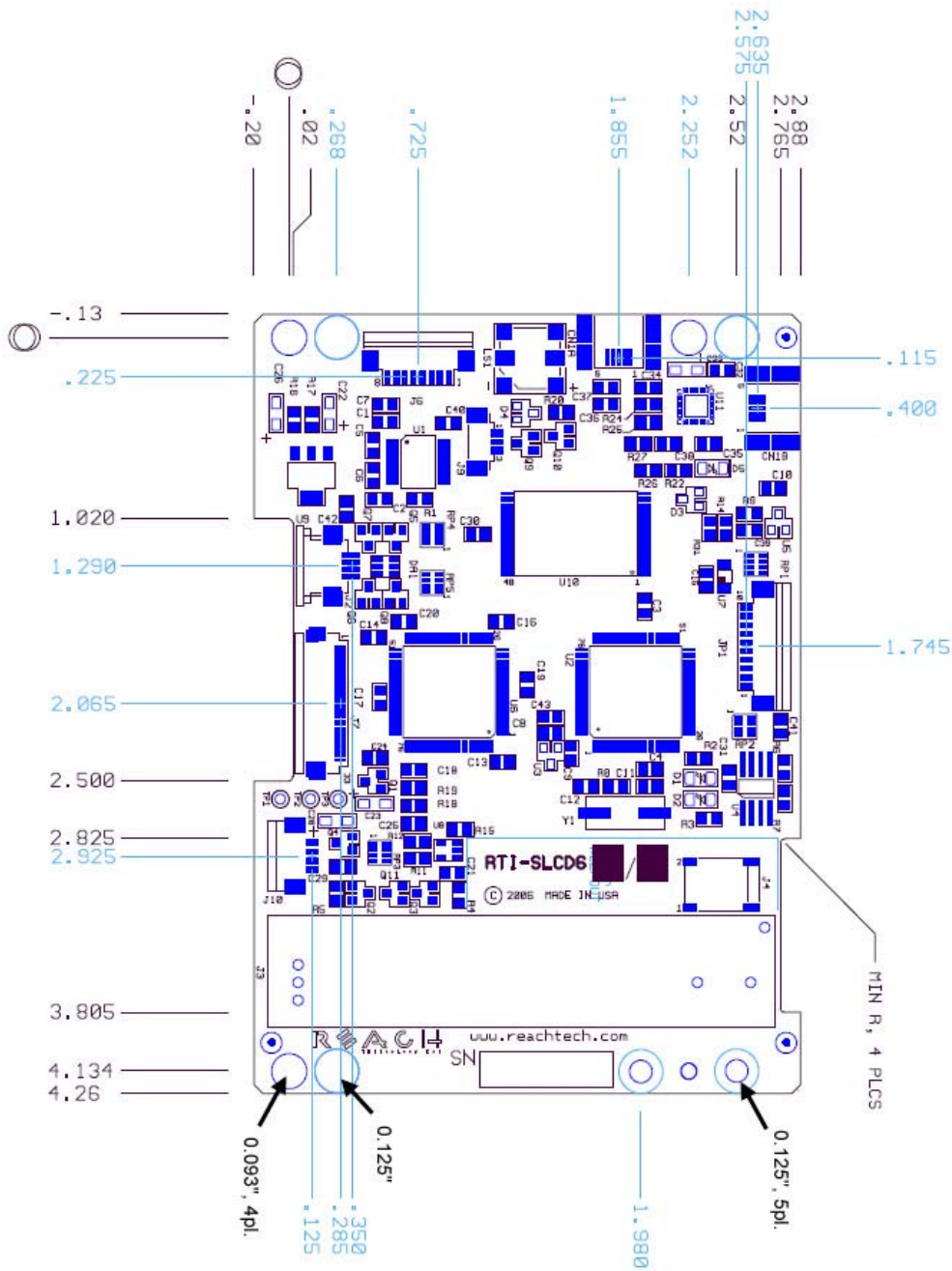
Figure 1: Rev G Board Physical Dimensions (inches)

1.4. *Dimensions.*

.

Width	4.39" / 114.4mm
Length	3.08" / 78mm
Total Height (back of board to top of on-board inverter)	0.46" / 11.7mm

1.5. *Electrical Characteristics*

The SLCD6 board requires 5V DC.

Current draw is as follows:

Configuration	Typical A	Max. A
SLCD6 with LB040Q02 (4") panel	0.8	1.0
SLCD6 with Data Image (5.6") panel	0.95	1.25

1.6. *Environmental*

The standard SLCD6 controller is rated for commercial temperature operation of 0 to 70ºC. The Industrial Temperature -40 to 85ºC version is available as a special order.

2. System Overview

2.1. *General - SLCD6 controller*

The SCLD6 plus attached LCD touch panel acts as a "smart terminal" and is generally connected to a "host" processor that implements the desired Graphical User Interface (GUI). The host can be any kind of processor from an eight bit microcontroller to a PC. The host issues commands to the SLCD6 and receives button press responses from the SLCD6. In this manual, the term "host" is used to describe the device connected to the SLCD6.

Note that it is possible to use the SLCD6 as a host in a limited way by using macros and the OUTPUT command. If true "host" capabilities are needed, contact Reach Technology and ask for the beta firmware with embedded interpreter capability.

The SLCD6 board contains flash memory that is used for bitmap and macro storage. (This is sometimes referred to as "external" flash to distinguish it from the processor's internal flash memory that stores the SLCD6 processor firmware.) A bitmap is equivalent to a Windows ™ bitmap file – it is a rectangular image. Appendix D describes bitmaps and the BMPload program used to store these into the SLCD6. Macros are a sequence of SLCD6 commands and are described in Appendix E.

The SLCD6 is connected to the host processor via a serial port. The SLCD6 has 3 separate serial ports plus a USB/Serial device port. There are several reasons for having multiple ports:

a) Host program development and debugging. One port is connected to the host and another to a PC. The PC is used to download images and macros that the host uses. The two ports allow both the host and PC to be connected without having to switch cables. The PC can also be used for interactive command execution / testing.

b) The SLCD6 supports serial pass-through via the "aout" and "ain" commands. This allows serial peripherals to be attached to the SLCD6 and accessed by the host.

2.2. *Overview - SLCD6 Evaluation Kits*

The SCLD6 is available in an evaluation kit form, and also as a complete enclosed unit. In the kit or unit, it comes pre-loaded with bitmaps and macros that implement a demo. The demo runs if the unit is powered on with the main communications port looped back transmit to receive. This loopback is via external plug in the case of the enclosed unit, or via the "DEMO" jumper on the "PowerCom" board in the case of the kit. The demo macro

is #1. Appendix Section E.6 contains a listing of the pre-loaded macro file. This can be used as an example of how to use macros.

The SLCD6 board in the evaluation kit has panel-specific hardware and firmware options loaded. In general, it is not advised to use an evaluation kit board to drive a different panel. Though it seems like this should not be an issue if the connectors look like they match, there are many details such as the inverter connection and timing that can cause a panel not to work.

The evaluation kit includes a two-port DB9 interface board that makes it easier to set up a development environment. One port can be used to download bitmaps from a PC while the other connects to the "host" computer. The SLCD6 also has a USB device port that can be used to connect it to a PC.

2.2.1. Getting Started

The SLCD6 kit as shipped contains a demo that allows you to verify its functionality. Plug the supplied power supply into the barrel connector on the triangular shaped PowerCom 4 board. The display should light up and lead you through various touch-activated screens.

Note that the demo is pre-loaded into the flash memory, and includes both bitmap files and a macro file. To best learn how the SLCD6 board and this kit works, start with simple commands using the serial interface and leave the creation and use of macros for later. Appendix G of the SLCD6 manual provides a short tutorial.

2.2.2. Connecting the kit to a PC - Serial or USB

The kit should be connected to a PC so that the serial command interface can be experimented with. This is a preliminary step before the unit is connected to the embedded system that will control the display in the embedded system.. ***PLEASE NOTE: In order to communicate over the main serial port, the Demo jumper JP1 on the PowerCom4 board must be removed.*** This jumper loops back transmit to receive on the serial port and this is what tells the display to run the demo.

As shipped, both DB9 serial ports is set to 115,200 baud, 8 bit, 1 start, 1 stop, no parity. Connect the PC using a straight through cable to the DB9 marked "MAIN" (P1). If the PC does not have a serial port, use the SLCD6 USB port. The USB port is implemented with the FTDI FT232R chip. To use this port, download and install the VCOM (Virtual COM port) drivers from the manufacturer here. The driver should be installed before the USB port is plugged into the PC.

A USB-to-serial adapter cable can also be used. *Note: the Belkin USB-serial adapter has software compatibility issues and is not recommended; Reach recommends either Prolific or FTDI chip based adapters.*

Once connected, use RealTerm (preferred), HyperTerminal, or similar terminal emulator to send and receive commands from the kit. HyperTerminal has limitations that can cause problems; specifically, some versions cannot send the "escape" character. We recommend RealTerm (shareware – Google "realterm" to download).

Comment [wpj1]: I have added a Task (OnTime) to add configuration files to the Download Center for both Hyperterminal and RealTerm.

2.2.3. Developing the application GUI with an embedded controller as host

To develop the host-driven GUI, connect the SLCD6 main port to the host and any of the other ports to a PC. This way either the host or the PC can drive the SLCD. Typically the PC is used to experiment and determine what command is needed for a graphical element, and to download new bitmaps and macros without having to disconnect the embedded system.

Comment [JM2]: Include a diagram to show this would work from a user perspective. This would make it a little easier for the less technical users.

How this dual port connection works is as follows. Only one SLCD6 port is the active or "main" port at any time. However, when any other port receives three <return> characters in a row, it auto switches to become the main port. This way, the PC can take control of the SLCD6 communications channel and send commands or download new bitmaps. The reset button, *RESET command, or a power cycle restores the previously active main port.

[Note: the *prevCons command will also restore the previously active main port. Also, the control port autoswitch character is <return> by default but can be set to a different value with the *auxEsc command.)

The SLCD6 has non-volatile memory which remembers which port is the "main" control port. This allows any of the SLCD6 serial/USB ports to be the main port on power-on. See the SET CONTROL PORT command.

2.3. PowerCom4 Board
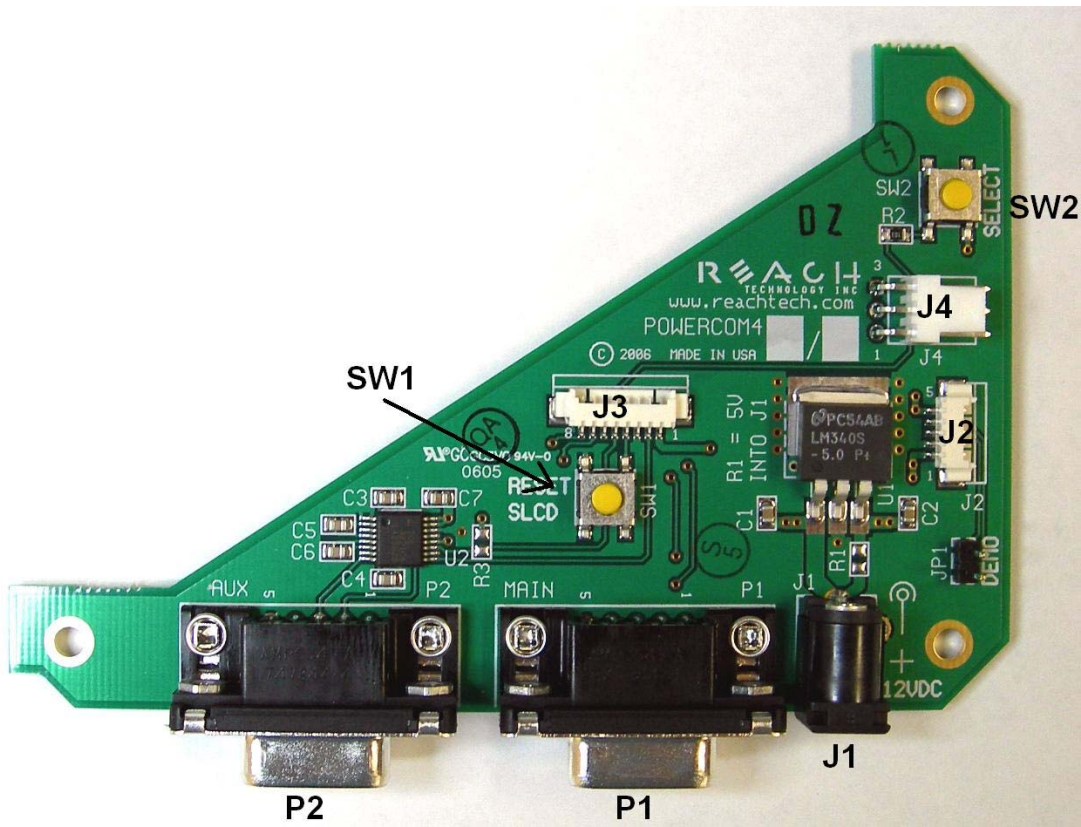
PowerCom4 Board

2.4. *PowerCom4 Schematic*

2.5. *Kit / PowerCom4 Operational Notes*

1. The unit default baud rate is 115200. The unit does not echo characters (for communications efficiency), so you must select "echo characters locally" or "half duplex" in your PC communications program. Also, all return strings are terminated by a <return> only, so you need to specify "add line feed to line return" as well.

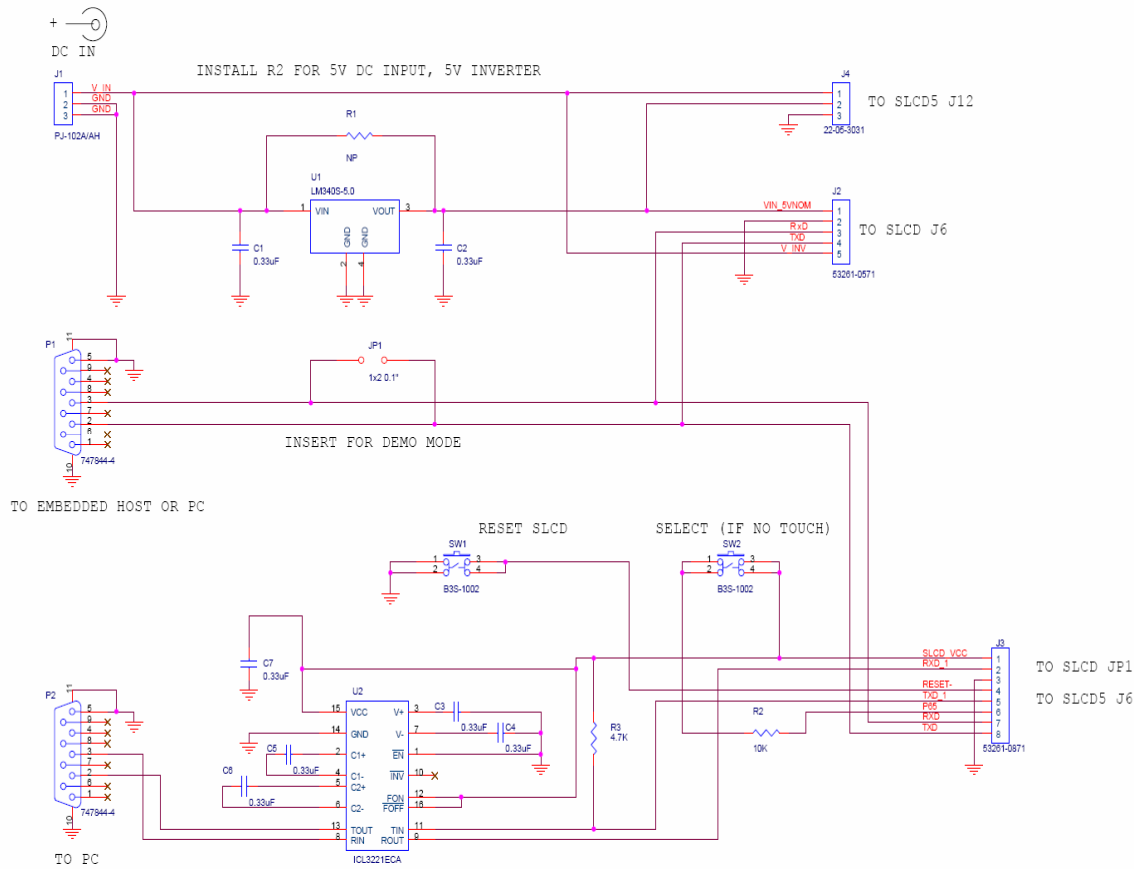2. The internal demo starts with an optional touch calibration. In order for the touch screen to work reliably ensure the LCD frame is grounded to the SLCD6 mounting holes.

3. The demo requires a certain set of bitmaps to be loaded. These are loaded as part of the kit. If these are not present, it will not run correctly. Copies of these are provided in the "BMPs and Macros" directory on the CD provided. Use the BMPload program and load the demo.lst and macros.mac files to restore the demo.

4. The SW1 "RESET" button on the PowerCom4 board resets the SLCD6 processor and performs the equivalent of a power-on reset.

5. The SW2 "SELECT" button on the PowerCom 4 board is intended for use with kits that don't have a touch screen, and is not implemented on a standard kit.

6. Jumper JP1 is the "DEMO" serial loopback jumper that is installed at the factory in order to automatically run the demo at power up. Remove the jumper prior to attempting serial communications with SLCD6 controller.

7. The J1 barrel connector is the external power supply connector for the development kit. With the SLCD6, the power supply is 5VDC input and the barrel connector is 2.5mm center pin positive.

8. Connector J2 provides the communications path for the P1 "MAIN" RS232 serial port. It connects to J6 of the SLCD6 controller. Connector J2 also provides 5VDC power to the SLCD6 controller.

9. Connector J3 of the PowerCom4 board is the communications path for the P2 "AUX" rs232 serial port. It connects to JP1 of the SLCD6 controller. This provides the path for the "RESET", and "SELECT" signal buttons. As well as the communications path for downloading of bitmaps and macros to the SLCD6 controller.

10. Connector J4 is reserved for future use.

2.6. *Communications Interface*

General

♦ Default communication is at a baud rate of 115200 with no parity, software (XON/XOFF) flow control, 8 bits of data, and 1 stop bit. The baud rate can be set to a different initial value on power-on by using the POWER-ON MACRO feature.

♦ Characters are not echoed and all responses end only with a <return> character (0x0d). This is done to maximize communications line efficiency.

♦ ASCII commands consist of a command (one or more ASCII characters) followed by the data associated with that command, followed by a carriage return. In this manual, the return character (value 0x0D, decimal 13) is signified by <return>.

♦ Binary commands consist of series of hex bytes. The general format is as follows, where each <…> descriptor is a single byte. Note that the first byte indicates the command length and there is no trailing <return>.

<0x80+number of bytes to follow><command byte><data0><data1>…<data n>

♦ Screen pixel values start at the upper left-hand corner. This is point x=0, y=0. The lower right corner is point x=319, y=239 (landscape mode).

♦ The maximum length of any command including the termination character is 127 characters.

Compressed Command Syntax

♦ All ASCII commands are shown with a space after the command mnemonic, for example:

```
p <pixels>
```

This command sets the line drawing width. This space is optional in all commands where the first argument is numeric (e.g. not text display) and can be removed to reduce code space and transmission overhead. Here is an example:

```
p2<return>
```

The above command sets the line width to 2.

2.7. *SLCD6 Input Buffer Processing*

Input Buffer

The SLCD6 has a 512 byte input circular buffer. As commands are received, they are queued in the buffer and executed first come first served. After a command has been processed, the SLCD6 issues a "prompt" character followed by a <return> indicating the success or failure of the command. The '>' prompt indicates success and the '!' prompt indicates failure. Failure can be due to either a syntax error or an out-of-bounds parameter. Depending on how long a command takes to execute, one or more commands may be stacked in the input buffer. The SLCD6 will issue a prompt for each command after it executes. These prompts may be issued while the host is sending a command to the SLCD6 (full duplex operation).

The purpose of the circular buffer is to provide overlapped command issue and execution with full duplex communication. If this is not needed, the host can wait for the prompt before sending another command.

The SLCD6 controller issues a prompt when it has finished processing a command. This includes the null command which is just a <return>.

There is no special "power-on" prompt supplied when the unit first powers on. To detect that the board is available for commands, the host should send a null command (single <return> character) and wait at least 10ms for a success prompt back. Alternatively the POWER-ON MACRO command / feature can be used together with the OUTPUT command to send a unique message indicating that the unit is up and running.

Flow Control

The SLCD6 implements software flow control using the XON (decimal 17) and XOFF (decimal 19) characters. When the circular buffer is approximately ¾ full, an XOFF is issued to the host. An XON is then issued when the buffer is approximately ¼ full. If the host cannot or does not want to accommodate software flow control, the host can make sure that no more than 2 commands are outstanding at any time. Given that the maximum length of any command is 127 bytes, this guarantees that the host will not be sent an XOFF character.

Buffer Limit Discussion

The input buffer can become full and unable to accept more data in two scenarios, both of which should never happen in normal operation. This discussion is presented because buffer overflow issues have presented security and reliability problems in PC and internet

devices. The two scenarios are as follows. In both cases, the buffer limit event happens when the buffer is full and one more character is received and has to be thrown away.

Scenario #1: The host sends data that a) does not conform to the command specification, and b) keeps doing so until the buffer size limit is reached, and c) ignores the XOFF request from the SLCD. ASCII commands are limited to a total of 127 characters including the <return>. Input buffer limit will occur when enough data is sent without a <return> to fill the buffer. This indicates a flaw in the host protocol or a hardware failure (for example, the communication line is chattering).

Scenario #2: The host sends valid commands that take a long time to execute and ignores the XOFF request from the SLCD. The limit event can occur when the buffer is full of unexecuted commands.

In both of the previous cases, when the SLCD6 detects a buffer limit it does the following:

- Discards the received character that caused the limit event, and resets (flushes) the entire input buffer. This is done in an attempt to make the error obvious to the GUI user. If a buffer overflow occurs it is a serious system error.

- Sends an overflow prompt to the host. The overflow prompt is '^'<return>. That is, shift-6 or caret followed by a return.

- Sends an XON character to the host (matches the XOFF that was previously sent)

Prompt Summary

The SLCD6 can issue the following prompts. These are in addition to any result of a command or button press event.

- '>'<return>   Indicates the a command has been executed successfully

- '!'<return>   Indicates that the command had a syntax or parameter error

- '^'<return>   Indicates that an input buffer full event occurred.

- '?'<return>   Indicates that a transmission line error occurred. This includes parity, framing, and receive overrun errors

2.8. *Touch interface*

The SLCD6 contains a touch controller that interfaces to a four wire resistive touch screen. Touch sensitive areas of the display are defined as either "hotspots" or "buttons". When either of these is pressed or released, the SLCD6 can either notify the host directly or execute a "macro", or both. A macro is a predefined sequence of SLCD6 commands.

Hotspot

The term "hotspot" refers to an area of the display that is touching sensitive. There are two types of "hotspots": visible and invisible. A visible hotspot is the standard type and when touched, the display area of the hotspot is color inverted (technically XOR'd with the foreground color) to provide a visual indication that a hotspot has been activated. An invisible hotspot does not provide any visual indication when touched.

The invisible hotspot is useful where a touch control is used to switch display screens. If a visible hotspot is used, and the host redraws the screen when the hotspot is pressed, the hotspot area can become inverted when the user removes their finger from the screen.

Button

A button is a touch sensitive area that has two bitmaps associated with it. These bitmaps correspond to the two states of the button – 1) normal /not pressed and 2) active / pressed. This allows a button to look like any GUI object including pushbuttons, toggle switches, radio buttons, check boxes, and so forth.

There are two major types of buttons: normal (momentary) and latching. A momentary button changes visual state only when pressed. This is like a momentary pushbutton or a keyboard key. A latching button is like a checkbox – press and release it once and the checkbox is filled, press and release again to clear it.

Host Notification

When a touch sensitive area is pressed or released, the SLCD6 can either notify the host, execute a macro or both. See the BUTTON DEFINE and TOUCH MACRO ASSIGN commands for details.

2.9. *Host input processing*

When integrated into a host environment, the SLCD6 sends prompts, touch activity notifications, and user-defined text to the host it is connected to. In general, all SLCD6 messages are terminated with a <return>.

There can be no guarantee as to the order of arrival for prompts, touch notifications, etc. It is guaranteed that the messages arrive complete and do not overwrite each other. The debounce timer for touch processing ensures that the host is not overwhelmed by touch notifications.

2.10. *LED indicators*

The SLCD6 board has two LEDs to indicate operational status.

D2 is closest to the board edge. It turns on when power is applied and the microcontroller is operating correctly. It can also be turned off and on via the SET LED command.

D1 is lit while the SLCD6 is executing a command or running a looping macro. It is useful to determine the activity level of the SLCD

2.11. *Control port autoswitch*

For added flexibility, the main control port of the SLCD6 can be switched on the fly to any of the other serial ports. This is called "control port autoswitch". It is affected by sending four auxEscape characters on the port to be switched to. These characters must not be interrupted by any other auxiliary IO. The first three characters cause the control port to switch and the last generates a prompt on the new control port. The auxEscape character is the <return> character by default and is programmable.

3.   Configuration Guide

3.1. *Power*

The SLCD6 requires 5VDC power. See Section 3 for connector pinouts. Power is applied to connector J6 which also contains the COM0 serial interface.

3.2. *Serial*

The SLCD6 has four serial ports. Any of them can be used to send commands to the board.

The physical interface options are as follows:

COM0 -   RS232 (factory option for 3.3V CMOS levels)

COM1 -   3.3V CMOS level signaling with RTS handshaking for an external transceiver such as RS485

COM2 -   RS232 only

COM3 -   USB (looks like a serial port to the host PC)

By default, all serial communications is 115200 baud, 8 data bits, no parity, with 1 stop bit, and receive flow control via XON / XOFF. This can be changed by using a "power-on macro" which is described here.

3.3. *TFT Panel Orientation*

The SLCD6 provides a programmable panel signal for flipping the image 180 degrees via the "*orient" command (See Section 5). This command is only supported for panels that have a hardware option for flipping the image.

3.4. *Grounding*

In order for the touch screen to work properly with a panel that has a CCFL backlight, the panel metal frame must be grounded to the SLCD6 ground. All SLCD6 mounting holes are connected to SLCD6 ground. Note that this must be connected to Chassis ground either directly or through a transient protection diode for ESD.

4. Connectors and Jumpers
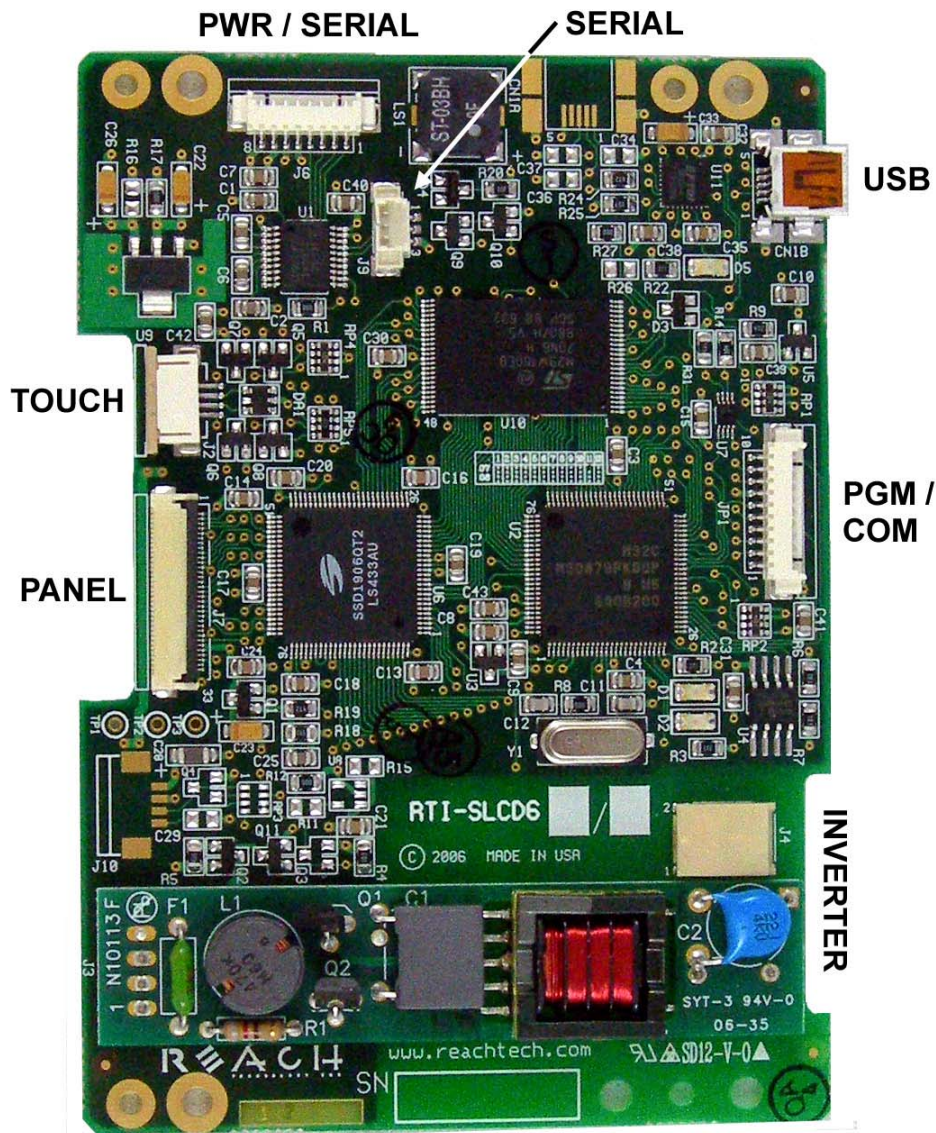
Figure 2: Connectors and Jumpers

4.1. *J6 - Power and COM0 RS232 communications*

J6 8 Pin Molex 53261-0871 for Power and Communications

Pin	RS232 Mode – R1 installed on board
1	do not connect
2	do not connect
3	RS232 input * (ESD protected)
4	RS232 output (ESD protected)
5	5V DC input
6	GND (power and communications)
7	5V DC input
8	GND (power and communications)

*   RS232 input valid only if R1 installed as zero ohms.
    This is the factory default.

For ESD protection details, refer to the specifications of the ICL3222E
transceiver chip.

Typical connection from a PC is as follows:
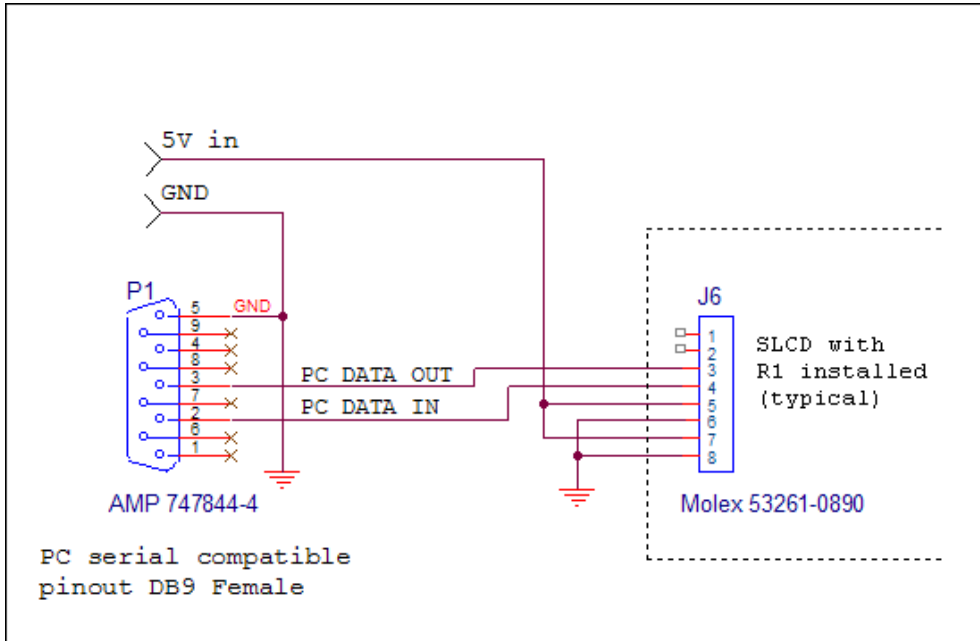
4.2. *J2 - 4 Wire Touch*

J2 4 Pin Molex 52271-0469 or equivalent 1mm pitch bottom
contact Zero-Insertion-Force Connector

Pin	Signal
1	X Right
2	Y Down
3	X Left
4	Y Up

*Note: touch calibration can accommodate swap between right and
left, and / or up and down.*

4.3. *J7 - 33 pin 0.5mm Flat Flex LCD Connector*

J9 33 Pin Omron XF2H-3315-1LW or equivalent 0.5mm pitch Zero-Insertion-Force Connector

Pin	Signal	Pin	Signal
1	GND	18	Green 5
2	LCD Clock	19	GND
3	LCD Line Pulse	20	Blue 0 (= Blue 4)
4	LCD Frame Pulse	21	Blue 1 (= Blue 5)
5	GND	22	Blue 2
6	Red 0 (= Red 4)	23	Blue 3
7	Red 1 (= Red 5)	24	Blue 4
8	Red 2	25	Blue 5
9	Red 3	26	GND
10	Red 4	27	LCD DE
11	Red 5	28	LCD VCC
12	GND	29	LCD VCC
13	Green 0 (= Green 4)	30	R/L *
14	Green 1 (= Green 5)	31	U/D *
15	Green 2	32	HVR *
16	Green 3	33	GND
17	Green 4		

** These signals' value are set via the *orient command but may also
be set to static high or low values depending on which panel the
firmware supports*

4.4. *J9 – COM2*

This connector is used for the COM2 serial port. It is RS232 only.

J9 10 Pin Molex 53398-0371

Pin	Signal
1	RS232 output (ESD protected)
2	RS232 input (ESD protected)
3	Ground

4.5. *CN1 – USB / COM3*

This connector is a mini-USB type which connects to an FTDI FT232R USB-to-Serial bridge chip. To use this port, download the VCP (Virtual Com Port) drivers at: www.ftdichip.com/Drivers/VCP.htm

CN1 Molex 54819-0572

Pin	Signal
1	VBUS
2	D-
3	D+
4	n/c
5	GND

4.6. *JP1 – COM1 (Programming Header)*

This connector is used *either* for updating the firmware of the SLCD, or for communicating using the COM1 serial port. It also has a hardware reset signal for the SLCD6 processor. (For firmware update, use the Reach P/N 42-0056 USB programmer.)

JP1 10 Pin Molex 53261-1071

Pin	Signal
1	VCC (typical 3.3V) – can be used to power external transceiver
2	DE (active high) – use if needed to enable external 422 / 485 driver
3	Do not use
4	RxD1 – Com1 input
5	Do not use
6	Do not use
7	GND
8	RESET- input; drive low to hardware reset the SLCD6 processor
9	Do not use
10	TxD1 – Com1 output

TxD1, RxD1, DE, RESET- electrical specifications:

Input voltage	-0.3 / VCC+.03	Absolute Min / Max
Input voltage	0.8xVCC	High min
Input voltage	0.2xVCC	Low max
Output current High	1mA	Min, $V_{out} = 0.9xVCC$
Output current Low	-1mA	Min, $V_{out} = 0.1xVCC$

4.7. *J10 - Backlight / Inverter Control*

J10 4 Pin Molex 53261-0471

Pin	Signal
1	Backlight power (connected to J6 pin 5)
2	Ground
3	Backlight on/off control
4	Backlight brightness control

This connector is typically not stuffed if the SLCD6 has an on-board inverter for the panel. It can be used to power and control an external backlight inverter. The sense of the on/off control (active high or low) is set in the firmware. The sense and range of the brightness voltage output is also set in the board firmware.

5.   Software Command Reference

The software commands and utilities are described in a separate document, the SLCD, SLCD6 Software Reference Manual.

Appendix A - Panels compatible with the SLCD6 controller

The SLCD6 controller has been tested with a variety of TFT panels. Please contact Reach Sales at 510-770-1417 sales@reachtech.com for specific panel compatibility and support.

Appendix B - Parts and suppliers for SLCD6 controller connections

B.1   *Connectors and cables for J6, J10*

The board connector for J6, JP1, etc is Molex type 53261-0890 (J6) and 53261-0490 (J10). The mating connector is made of two parts: receptacle housing and crimp pins. A special tool is needed to make the crimps. Alternatively, custom cables can be purchased. See B.3 for cable vendors.

J6 Receptacle housing Molex P/N 51021-0800

J10 Receptacle housing Molex P/N 51021-0400

Crimp pins Molex 50079 or 50058

Prototype (small qty) crimp tool Molex 63811-0200

Production crimp tool Molex 63811-0000

All of the above are available from www.digikey.com

B.2   *Cables for J9*

These connectors attach to a flat flex cable that then attaches to the display unit. These cables can be ordered from either

www.axoncable.com or www.quadrangleproducts.com

B.3   *Discrete wire cable vendors*

The cables needed for J6, etc. can be specified and supplied as assembled cables by contacting:

www.intcomptech.com

Appendix C - Ordering information

C.1   Contact Reach directly for ordering information.

Reach Technology Inc
4575 Cushing Parkway
Fremont, California 94538
www.reachtech.com
(510) 770-1417

Appendix D - Troubleshooting

D.1   *Touch unreliable or non-operative*

If the touch screen is unreliable or non-operative, do the following:

1.   For displays with CCFL backlights, make sure the metal shell of the display is connected to one of the SLCD6 mounting holes. This is the same as saying that the display case should be grounded to SLCD6 ground. Note that this only applies to displays with CCFL backlights, not to EL or LED backlights.

2.   Run the TOUCH CALIBRATE command, "tc". This will reset the calibration values and allow you to recalibrate the touch screen.

If after completing the step(s) above the touch is still non-operative, check the touch connection into the SLCD6 board. Some touch panels use conductive ink that can be easily scraped off by too many or incorrect connector insertion cycles. If there are holes you can see through on the touch connector end where it plugs into the SLCD6 connector this is the problem.

To determine the accuracy and sensitivity of the touch, you can use the "debug" command as follows:

*debug 1<return>

This puts an "X" on the screen whenever a valid touch is recognized. To turn off, use:

*debug 0<return>

Appendix E – Using COM0 / J6 for 3.3V CMOS Interface

The SLCD6 serial port COM0 is typically used for communication using RS232 levels. However, by removing resistor R1, it can use 3.3V CMOS logic levels. This is typically a factory option, but the user may make this change as needed. R1 is located near U1 which is near connector J6.

COM0 Logic Level Mode:

1. Resistor R1 (next to J6) <u>must be removed</u>.

With R1 removed, connector J6 must be wired for Logic Level Mode:

J6 8 Pin Molex 53261-0890 for Power and Communications

Pin	Logic level mode – R24 not installed
1	TxD - output from microcontroller.
2	RxD - input to microcontroller *
3	do not connect
4	do not connect
5	5V Input
6	GND (power and communications)
7	5V Input
8	GND (power and communications)

* R1 must be removed to avoid conflict with RS232 receiver.

Backlight Voltage

5V in

GND

3.3V CMOS
LEVELS *

$\overline{\text{TxD}}$　out

$\overline{\text{RxD}}$　in

J6

1
2
3
4
5
6
7
8

SLCD WITH
R24 REMOVED

Molex 53261-0890

* 5V CMOS LEVELS is panel power is 5V