

Interfacing Graphic LCDs to the BL1000 and the BL1100

Summary

The Seiko SED1330 LCD controller works with most LCD graphics modules that use a clocked 4-bit interface.

The SED1330 controller board connects directly to the BL1000 and BL1100 LCD interface.

The SED1330 has a number of complex features. While these might be useful for specialized applications, they are not suitable for a generic driver. Thus, the Z-World LCD driver is organized as a two-page display. Each page can be displayed or hidden, independently of the other. All driver commands write to the page that is currently selected. This is a common practice that allows the user to write one page and display another.

Interface Variables

The following variables are used in the LCD driver:

- **int color**
The **color** variable determines whether pixels are turned on (**color** modulo 1 = 1) or turned off (**color** modulo 1 = 0). This variable is used by the functions **putpixel**, **line**, **lcdputc**, and **lcdputs**. The value of **color** is destroyed by calls to **line**. (See “**void line**” below.)
- **int style**
An 8-bit map of the pixel pattern used by the **line** function. The value of **style** is copied serially to the color variable prior to line plotting. As each bit is plotted, the **style** bit controls the color of that pixel. After the pixel is drawn, the **style** variable is “rotated.” Table 1 shows samples of bit patterns.

Table 1. Sample Pixel Patterns

Value	Pattern	Description
0xFF	••••••••	Solid Line
0x00		No Line
0xAA	• • • •	Dotted Line
0xCC	•• ••	Dashed Line
0xF6	•••• ••	Alternate Dashes

Interface Functions

The following are the functions in the driver for the SED1330, with the most basic commands listed first. Most drawing commands specify some pixel coordinate. Since these library functions do not check screen bounds, be sure to specify values within the pixel range of the screen being used.

- **void lcdinit(void)**
Initializes the LCD driver. This call must be made for any of the other calls to work, and must be customized for the display being used. The LCD driver is initially set up for the Seiko G191C display. (See the section on customization.)
- **void lcddisp(int pages)**
Sets the display flags for pages 0 and 1:

Value	Result
PAGE0	Display page 0 only
PAGE1	Display page 1 only
PAGE0 PAGE1	Display logical OR of pages 0 and 1
- **void lcdpage(int page)**
Specify the current page for graphics commands.

Value	Result
PAGE0	Graphic commands to page 0
PAGE1	Graphic commands to page 1
- **void lcdcls(void)**
Clears the current page.
- **void lcdclb(int x0, int y0, int x1, int y1)**
Clears a rectangle on current page with (**x0**, **y0**) at the upper left hand corner and (**x1**, **y1**) at the lower right hand corner.

- **void lcdputc(int x, int y, int ch)**
Draws the ASCII character equivalent of **ch** at (**x**, **y**) on the current display. Characters are up to 8 × 8 pixels.
- **void lcdputs(int x, int y, char* s)**
Outputs a string **s** of ASCII characters starting at (**x**, **y**) on the current display. Characters are up to 8 × 8 pixels.
- **void lcdscl(int pixels)**
Scrolls the current page *left* a given number of pixels. Pixels must in the range [0, Display Width]. Lines newly shifted into the display are blank.
- **void lcdscrr(int pixels)**
Scrolls the current page *right* a given number of pixels. Pixels must in the range [0, Display Width]. Lines newly shifted into the display are blank.
- **void lcdscrd(int pixels)**
Scrolls the current page *down* a given number of pixels. Pixels must in the range [0, Display Length]. Lines newly shifted into the display are blank.
- **void lcdscru(int pixels)**
Scrolls the current page *up* a given number of pixels. Pixels must in the range [0, Display Length]. Lines newly shifted into the display are blank.
- **void lcdscrup(int x0,int y0, int x1,int y1, int pixels)**
Scrolls the specified rectangle up by a given number of pixels. The rectangle scrolled has (**x0**, **y0**) as the upper left hand pixel and (**x1**, **y1**) as the lower right hand pixel.
- **void putpixel(int x, int y)**
If color modulo 1 = 1, the pixel at (**x**, **y**) is set (turned black). Otherwise, the specified pixel is reset (turned white).

- **void line(int x0,int y0, int x1,int y1)**
Plots a line from pixel (**x0**, **y0**) to (**x1**, **y1**). The line pattern drawn is specified by the **style** variable. (A call to **line** destroys the color variable.)

Customization (New Displays)

The SED1330 controller and Z-World's graphic driver are compatible with a variety of displays, such as those listed in Table 2.

Table 2. LCD Displays

Manufacturer	Display	Size
Seiko	G2436	240 x 64
	G191C	92 x 128
Optrex	DMF660N	240 x 128
	DMF682AN	256 x 128
	DMF6102N	480 x 128
	DMF690N	640 x 64
	DMF651AN	640 x 200
AND	AND1241ST	480 x 128
	AND932ST	640 x 200
	AND561ST	640 x 200
Stanley	GMF640201S	640 x 200

The SED1330 driver (as supplied in the library) is set up for the G191C display from Seiko. A few modifications are required for use with other displays compatible with the SED1330. Two sections of code require changes.

The first section requiring change is the definition of macros for the machine language routines. The block for the SED1330 is shown in Figure 1.

LCD:	equ 062h	;Address of LCD Port
DIMX:	equ 019h	;Display Width (Bytes)
SCREEN:	equ 0C80h	;Screen Size (Bytes)
WIDTH:	equ 192	;Display Width (Pixels)
HEIGHT:	equ 128	;Display Length
CHWID:	equ 6	;Character Width + Spaces

Figure 1. SED1330 Macros

Table 3 gives a detailed description of each macro.

Table 3. Machine Language Routine Macros

Macro	Description
LCD	The I/O address of the lowest of the two LCD ports. the BL1000's LCD interface is located at 0A000 _{HEX} ; the BL1100's LCD interface is at 062 _{HEX} .
WIDTH	Screen width in pixels.
HEIGHT	Screen height in pixels.
DIMX	Screen width in bytes, plus 1. $DIMX = WIDTH/8 + 1$.
SCREEN	Memory required for one full display. $SCREEN = DIMX \times HEIGHT$.
CHWID	The number of pixels occupied by each alphanumeric character, including intercharacter spacing. The default font is 5 pixels wide plus 1 space between characters.

The second section that will need modification is the actual initialization sequence sent to the controller. The initialization sequences for the G191C are shown in Figure 2.

```
int G191C_init[] = {0x140,0x30,0x87,0x07,0x18,0x7C,0x7F,0x19,0x00,
                  0x144,0x00,0x00,0x7F,0x80,0x0C,0x7F,
                  0x15D,0x07,0x87,
                  0x15A,0x00,
                  0x15B,0x1C,
                  0x14C};
```

Figure 2. G191C Initialization Sequences

Following are detailed descriptions of each sequence. Only sections which need modification are explained.

- **0x140, 0x30, 0x87, 0x07, <C/R>, <TC/R>, <L/F>, <LSB AP>, <MSB AP>**

This is the system command. If it is set improperly, the display will not work.. Each of the modifiable fields is described in Table 4.

Table 4. System Command Modifications

Field	Description
<C/R>	Screen width in bytes ($DIMX$).
<L/F>	Screen length in lines, minus 1 ($HEIGHT - 1$).
<A/P>	16 bits version of screen length, plus 1 ($DIMX + 1$).
<TC/R>	A timing value which relates screen refresh rates, controller clock speed, and display dimensions.

For simplicity,

$$\langle \mathbf{TC/R} \rangle = \langle \mathbf{Fosc} \rangle / (9 \times \langle \mathbf{L/F} \rangle \times \langle \mathbf{Frame Freq} \rangle)$$

where $\langle \mathbf{Frame Freq} \rangle$ is the framing frequency (in Hz) for the display being used, and $\langle \mathbf{Fosc} \rangle$ is the crystal frequency of the SED1330. For example, the G191C has 128 lines ($\langle \mathbf{L/F} \rangle = 128$) and a framing frequency of 70 Hz; the SED1330 controller has a 10 MHz crystal. For this configuration, then,

$$0x7C = 10,000,000 / (9 \times 128 \times 70)$$

- **0x144, 0x00, 0x00, $\langle \mathbf{L/F} \rangle$, $\langle \mathbf{LSB SCREEN} \rangle$, $\langle \mathbf{MSB SCREEN} \rangle$, $\langle \mathbf{L/F} \rangle$**

This sequence sets the memory locations for the start of each of the two display pages. The first screen starts at the base of memory, which is shown by (0x00, 0x00). The other immediately follows it. Thus, the start address for page 1 is SCREEN bytes into memory.

- **0x15D, 0x07, 0x87**

Hides cursor.

- **0x15A, 0x00**

Sets horizontal scroll of screen to 0.

- **0x15B, 0x1C**

Sets 2 page graphics format.

- **0x14C**

Sets cursor advance direction to right.

After these modifications, the SED1330 graphics driver will work with any compatible display.

Custom Fonts

The driver as distributed comes with a 5×7 character font, which is the smallest size that allows for the display of a complete set of unique characters. The software, however, is set up to allow fonts of up to 8×8 , such as the standard PC graphics font. Two modifications are required to use such a font:

1. The **CHWID** macro must be set to the appropriate value.
2. The font table must be modified. Each character is specified by eight bytes. The first byte represents pixels on the top row, the next byte, pixels on the second row, and so on. Bit 7 is the leftmost pixel of the row and bit 0 is the rightmost pixel. Characters less than eight bits in width must be left justified, with unused bits zero.

Design Notes

Larger LCDs are more sensitive to deviations from specified parameters. There are two important problems to consider. First, if $\langle \mathbf{TC/R} \rangle$ is set incorrectly, the display will become unstable or may not display at all. Second, the specified LCD voltage (usually negative) is very sensitive, and should be supplied within .5 volts of its specified value. Otherwise, even a properly configured display will appear blank because there is insufficient bias voltage to align the liquid crystals.

Unless a custom SED1330-based design is planned, obtain an evaluation board from Seiko for development. The G191C easily connects to the BL1000 or BL1100. Two cables are required, one from the controller to the SED1330, and one from the SED1330 to the G191C. These cables are laid out illustrated in Figures 3 and 4 on the following page.

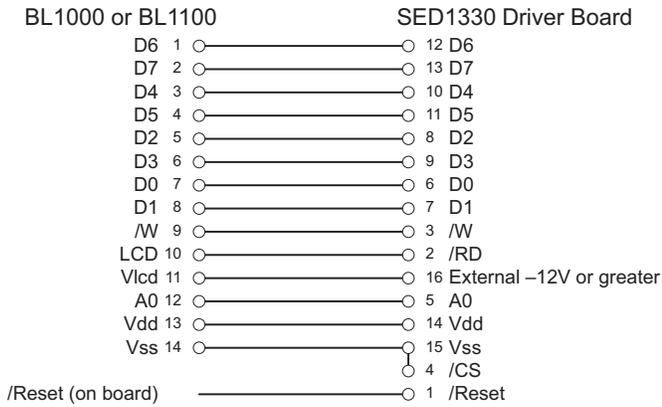


Figure 3. Controller to Driver Board Connections

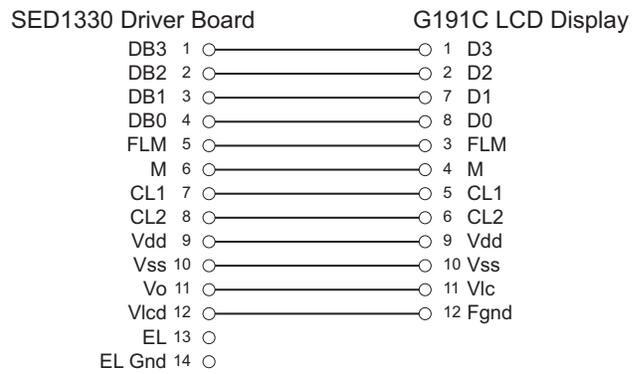


Figure 4. Driver Board to LCD Display Connections

Note:

Pins 15 and 4 on the first cable (SED1330 side) are tied together.

The reset line (not available on the LCD interface) must be taken from somewhere else on the board.

An external power supply with a large enough negative voltage to bias the display must also be used.

The board must be set for 6800 interface mode.

To better interface with the LCD module, J1 on the BL1000 and J6 on the BL1100 are numbered in mirror image compared to a standard 14-pin connector.



Z-World Corporate Headquarters
 2900 Spafford Street
 Davis, California 95616 U.S.A.
 530.757.3737 • Fax: 530.753.5141

<http://www.zworld.com>