

GUIDE TO CONNECTING THE OPTREX 20434 LCD TO YOUR BOARD

Introduction

Unfortunately, the old Optrex 20434 data sheet is a very cryptic read. The data sheet includes very little useful text and contains a terrible error. This guide is meant to add some useful information where Optrex fell short in its data sheet as well as correct the mentioned error. In addition, adding the Optrex 20434 to your board is deceptively simple. Some of the hints provided will help to guide you to a working LCD with minimal frustration. For better information, refer to the Hitachi HD44780U LCD Controller data sheet.

Hardware

This section of the guide is meant to lead you through the portion of the LCD hardware design that is a bit tricky because it is tempting to connect certain LCD signals without putting in the thought that is necessary to meet the LCD timing requirements. The design solution will not be given to you, but the following hints will help you to perform your design with all of the information necessary to recognize a potential design pitfall. The bottom line here is that **you must meet the LCD timing requirements to get the LCD to work!**

- ❑ Look at the DMC20434 timing chart given in the Optrex data sheet. Notice that the parameter, t_{AS} , specified in the timing chart and shown in the write and read timing diagrams (Figures 1 and 2 respectively) is the setup time for AS **and** R/W. Also notice that this setup time is relative to the rising edge of the LCD Enable pulse. **Pay close attention to the LCD timing parameter, t_{AS} , as you perform your design.**
- ❑ Now look at the timing diagram for the 8051 MOVX instruction. Since reads and writes to the LCD will be performed with this instruction, its timing is relevant to the LCD interface.
- ❑ Think about your board decoder logic. You may want to have another 74LS138 decoder that divides the upper 8 KB of your memory into eight 1 KB blocks, although for the purposes of interfacing to the LCD, this is not necessary. Assuming you will want to generate your enable pulse using this decoder logic, when will the enable pulse reach the LCD after a MOVX instruction has been issued to the LCD by the 8051? One way to see and understand this timing is to sketch a to-scale timing diagram of the following signals on a piece of graph paper:
 - De-multiplexed 8051 address bus
 - De-multiplexed 8051 data bus
 - 8051 WR/ or RD/ signal -- depending on the type of transfer you would prefer to look at. Write is probably easier to analyze at this stage because the 8051 supplies the data.
 - LCD Enable signal
- ❑ Based on your timing diagram from above, would it be a good idea to directly connect the 8051 WR/ pin to the R/W pin of the LCD? This is the aforementioned pitfall. Again, think about the LCD timing parameter, t_{AS} . Don't fall victim to the R/W pitfall and use all of the timing diagrams shown above to perform your interface design.
- ❑ After you have completed your design and before you begin writing any firmware, you may want to test your design by doing a write to the LCD. You can do this by issuing a MOVX instruction using MON51. You don't even need to have the actual LCD connected to your

board to do this. Observe the signals to the LCD using a logic analyzer. If you have not met the timing requirements of the LCD, modify your design before continuing on to the LCD firmware design.

- Note that the LCD contrast must be set correctly, or you will not be able to view the display.

Firmware

LCD Instructions

LCD instructions are used to perform various activities with the LCD panel. The instruction table in the HD44780 or Optrex data sheet provides a summary of the LCD instructions. An LCD instruction is simply a read or a write to the LCD with the pins asserted as shown in the instruction table.

LCD Initialization Sequence

Before writing any other firmware, you must write the firmware that initializes the HD44780 LCD controller chip. This firmware is called the initialization sequence and is just a sequence of LCD commands. Without performing this initialization your LCD will not work. **There is an error in the initialization sequence flow chart given in some old Optrex data sheets.** If you compare the Display ON/OFF instruction, given in the instructions table, to the stage that has "Display ON" written next to it, you will notice that they do not agree. The comment in the Optrex data sheet should have read "Display Clear". This error has been corrected in the flow chart shown on the next page, as well as in the HD44780U data sheet. In addition, the parameters missing in the data sheet LCD initialization commands have been filled in for you.

Getting the LCD to Display Text

After successfully initializing the LCD and turning the display on, you can begin to display messages on your LCD by sending the correct instructions to it. First of all, it is helpful to understand the LCD layout and the Display Data RAM (DDRAM).

Each LCD location where a character can be written has a unique address. The addresses for the DMC20434 are mapped to the LCD panel as follows:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53
14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67

Figure 1: LCD Addresses (given in hexadecimal)

In order to display a character in the bottom right hand corner of the LCD panel, a character has to be written to display address 67H. In actuality, the LCD addresses correspond to specific locations in the HD44780 DDRAM. Therefore, in displaying a

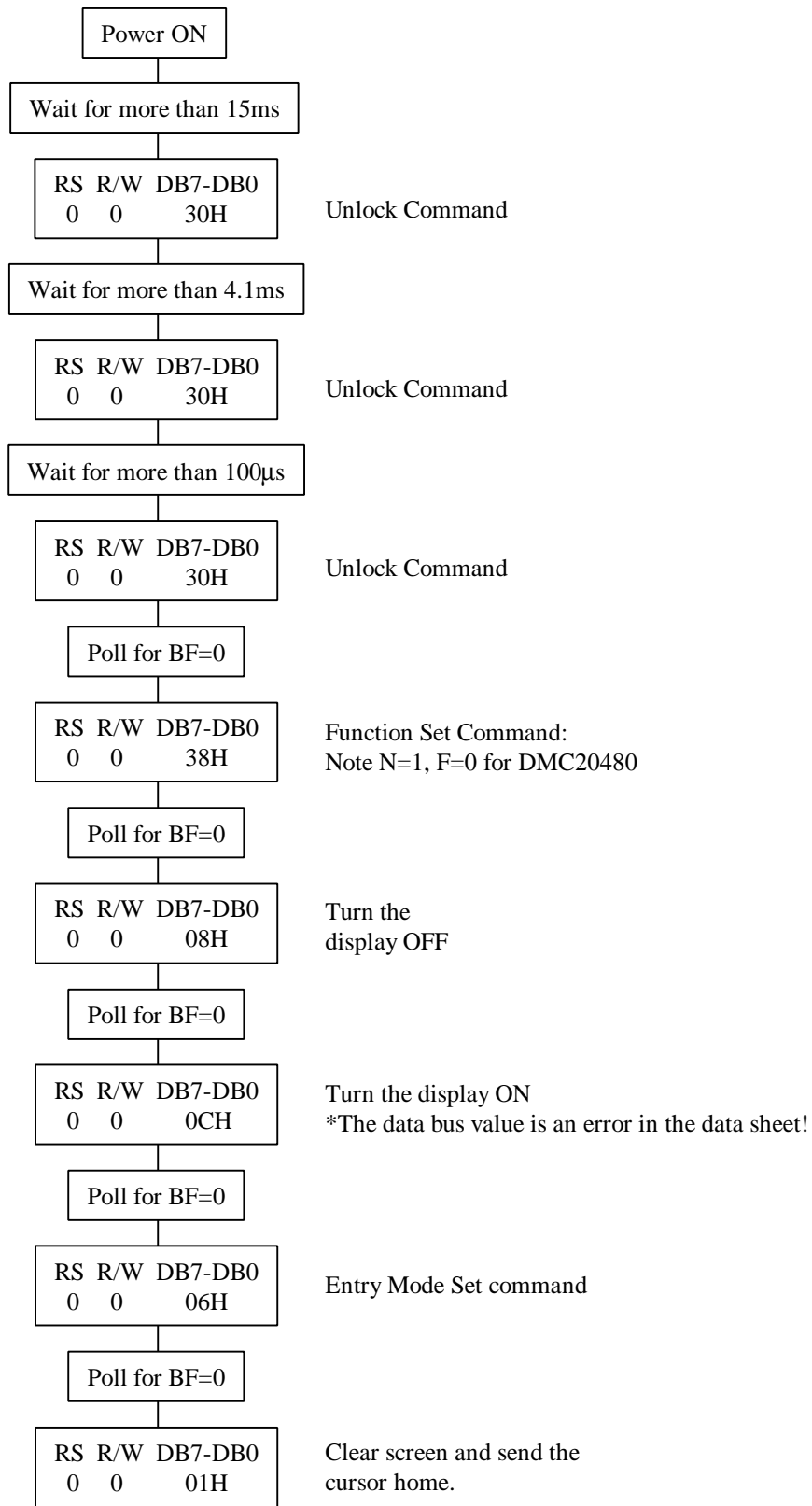


Figure 2: LCD Initialization Sequence (Corrected over old Optrex 20434 data sheet)

Getting the LCD to Display Text (cont.)

text message, the HD44780 reads all of its DDRAM locations and writes the character stored in each location to its corresponding location on the screen.

Getting the LCD to display text is a two step process. First, the LCD's cursor must be moved to the LCD address where the character is to be displayed. This is done with the "DDRAM Address Set" command. Second, the actual character must be written to the cursor in order to store it in the DDRAM at the cursor's location. This is performed with the "CGRAM/DDRAM Data Write" command. Refer to the instructions table of the Optrex data sheet to get the specific pin settings for both of these commands. Refer to the Font Table in the Optrex data sheet to see the characters that can be displayed on the LCD panel and their corresponding 8-bit values. Note that the Optrex 20434 supports the standard ASCII character set.

One last point to note about displaying text is that if the initialization sequence given in the flow chart is followed, the LCD is set to automatically increment the cursor after each character has been written to DDRAM. This is nice if you want to write text messages to your LCD panel because you only need to set the DDRAM address once. Then you can continue sending the "CGRAM/DDRAM Data Write" until you reach the end of a line. However, note that the cursor address is simply incremented by one. Also note that the LCD addresses are not sequential from line to line. So, using this feature without being careful can yield some weird results. Finally, keep in mind that you can turn the automatic increment off or change it to automatic cursor decrement. Refer to the instructions table for details.

LCD Busy Flag Polling

The HD44780 will not accept new commands while it is busy with some internal operation. This condition must be tested before sending a new command to the LCD. The "Busy Flag/Address Read" instruction should be used for this purpose. Please refer to the LCD instructions table. The BF bit in this instruction is the busy flag. When this bit is a 1, the LCD controller is busy. When BF becomes 0, the LCD is ready for the next command. In addition to busy flag polling, this instruction can be used to determine where the address of the LCD cursor. The AC field of the "Busy Flag/Address Read" instruction will contain this address.

Defining LCD Custom Characters

One of the nice features of the HD44780 is that it allows for the creation and use of up to eight unique, user-defined characters. Character Generator RAM (CGRAM) has been added to the HD44780 for this purpose.

Before a custom character can be used it must be created. Each character that can be displayed by the LCD is composed of a 5x8 grid of pixels, or dots. Each of these dots can either be turned ON or OFF when a character is being displayed. Therefore, in order for the LCD controller to display a character, it must have a definition of which of the character dots need to be turned ON. To that end, the LCD Character Definition Table can be used.

Binary Row #	Row Bit Number					Hex. Row Value
	4	3	2	1	0	
000	ON	ON		ON	ON	0x1B
001	ON	ON		ON	ON	0x1B
010						0x00
011			ON			0x04
100			ON			0x04
101	ON				ON	0x11
110		ON	ON	ON		0x0E
111						0x00

Figure 3: LCD Character Definition Table – Demonstrating the definition of a smiley face.

Each square of the table represents one dot of the LCD character. As an example, the character dots for a smiley face have been given in the character definition table. The hexadecimal row values are a byte of data that is generated by replacing all of the ONs in a given row with binary ones. Three zeros must be inserted at beginning of the row byte value to create the full 8 bits. For example, the row 000 value, 0x1B, has the binary representation, 00011011. Notice that the binary ones in this binary representation match the ONs of the LCD Character Definition Table.

After defining a character and calculating each of its 8 row values, that character definition can be sent to the LCD. Specifically, each of the character's 8 row values must be written to the correct HD44780 CGRAM addresses. A 6-bit CGRAM address is generated as follows:

Custom Character Number			Character Row Number		
C ₂	C ₁	C ₀	R ₂	R ₁	R ₀

Figure 4: CGRAM Address Format

- C₂C₁C₀ is a 3-bit field in the CGRAM address that contains the binary value of the custom character number to be defined (000₂ to 111₂). Eight different custom 5x8 characters may be defined for the LCD.
- R₂R₁R₀ is a 3-bit field specifying the row number for this character. This field corresponds to the row number value of the LCD Character Definition Table.
- **Example:** If the smiley face shown in the LCD Character Definition Table is to be defined as custom character number 1 (001₂), row number 3 (011₂) of the smiley face would have the address: 001,011₂.

Perform the following steps to define your custom character:

- 1) Set the character's row 000 address to the CGRAM using the CGRAM Address Set instruction defined in the instructions table of the LCD data sheet.
- 2) Write the character's row 000 value to the row 000 CGRAM address using the CGRAM/DDRAM Data Write instruction.
- 3) Repeat steps 1 and 2 for all your character's remaining rows.

Printing a defined custom character to the LCD screen

Once you have defined your custom character, you may use it just as you would any other character that is listed in the data sheet Font Table. Therefore, just write the character's value from the Font Table to the DDRAM address corresponding to the LCD location at which you want the character to be displayed. Notice, as shown in the first column of the Font Table, that the custom characters can be referenced using their 8-bit character numbers.

Example:

Suppose you have defined custom character number 3 (011_2). To print that character to the upper right hand corner of the LCD screen, perform the following steps:

- 1) Move the cursor to the upper right hand corner of the LCD panel by issuing a DDRAM Address Set instruction. Set the DDRAM Address to $0x13$ (see the LCD Addresses table above).
- 2) Write your custom character to the cursor with the CGRAM/DDRAM Data Write instruction. The WRITE DATA field of this instruction should contain the custom character's number or, in this case, the value $0000,0011_2$.