



11071 LCD Lab Handout

LAB 1 – Initializing the LCD Driver Module

In this first exercise, we will modify the registers needed to initialize the LCD Driver module. The values that are loaded into these registers are dependent upon the LCD specifications (this is provided in the handouts) and other human factors. After initializing the registers, the project will be compiled and uploaded onto the PIC18F85J90 and a message should appear on the LCD. The exercise will also be an introduction to the MPLAB IDE, and programming the PICDEM LCD2 with the MPLAB ICD 2.

- Open Project “C:\Masters\11071\Labs\Lab1\LAB1.mcp” using MPLAB
- Add the needed code to the LCD_CONFIG_R() routine in LAB1.c (Ln 297):

The LCDPS, LCDSEn, LCDCON, and LCDDATAN registers need to be initialized.

Note: Use the attached PIC18F85J90 register maps and LCD data sheet to help determine how to configure each register.

- Ensure that the R<->C switch points to the R.
- Compile the project, program the target device, and release the device from reset.

HINTS:

- Each line in C programming should end with a colon.
- The example below shows how individual bits of a register are accessed in C18.

```
LCDPSbits.WFT = 1;
```

In this case, the LCDPS bit WFT would be set to '1.'



11071 LCD Lab Handout

LAB2 Part1 – Creating LCD Display Routines

Part I – Mapping LCDDATA registers to Segment Names

The creation of LCD display routines can be simplified in a number of ways. We will first look at how mapping LCDDATA register names as segment names can achieve this. For this exercise you will need to use the attached LCD Segment versus COM table, LCD to microcontroller schematic, and LCDDATA register map to fix the missing section of code in LCD.h. We will complete one example together.

- Open Project “C:\Masters\11071\Labs\Lab2\Lab2a\LAB2a.mcp” using MPLAB
- Modify the code that is commented out in LCD.h so that the LCDDATAx registers can be referred to by the segment names.
 - o An example is given the section.
- Compile the project, program the target device, and release the device from reset.



11071 LCD Lab Handout

LAB2 Part 2 – 7-Segment Digit Routines

A very common LCD display routine is one which turns on the corresponding segments of a 7-segment digit based on the number it is given. A useful C structure that can assist in this is the switch case statement. You will modify the code in LAB2b.c to build a routine that accomplishes this task. For this exercise you will need the LCD Segment diagram attached to this page. You will use the segments that you defined in LCD.h in the previous lab in the routine that you will write.

C Programming Hints:

```
// Switch case example
switch ( <variable> ) {
case this-value:
    Code to execute if <variable> == this-value
    break;
case that-value:
    Code to execute if <variable> == that-value
    break;
...
default:
    Code to execute if <variable> does not equal the value
    following any of the cases
    break;
}
// End Switch case example
```

- Open Project “C:\Masters\11071\Labs\Lab2\Lab2b\LAB2b.mcp” using MPLAB
- Modify the code that is commented out in LAB2b.c in the S1Num() routine (Ln 522). The routine should take in an integer. The segments corresponding with that number should be displayed.
- Compile the project, program the target device, and release the device from reset.



11071 LCD Lab Handout

LAB2 Part 3 – Bar Graph Display

The LCD glass used in this lab has a bar graph display on the top of the screen. This can be useful in displaying voltage levels. In this exercise, we will use it to display the voltage across the potentiometer on the bottom left of the PICDEM LCD 2 board. When the potentiometer is turned counter clockwise, the A/D conversion should be 0. When turned clockwise, the A/D conversion should increase linearly to 1023. Design a routine similar to the Part 2 of this lab that displays a properly scaled number of segments on the bar graph display as the potentiometer is twisted. For this exercise you will need the LCD Segment diagram attached to this page. The A/D routines have already been written and will return the A/D value.

- Open Project “C:\Masters\11071\Labs\Lab2\Lab2c\LAB2c.mcp” using MPLAB
- Modify the code that is commented out in LAB2c.c BarGraph() routine (Ln 325). The routine should take in an integer 0-1023 and display a properly scaled number of segments. For example, a value of 512 from the A/D should enable half of the bar graph, while a value of 1023 should enable all of the segments.
- Compile the project, program the target device, and release the device from reset.
- Twist the potentiometer to see how the display responds to a changing voltage across the potentiometer.

Hints:

- There are 20 bar graph segments
 - o $1023/20 = 51$
- If-statements may be an effective C structure in this routine.



11071 LCD Lab Handout

LAB2 Part 4 – Advanced Lab

For those of you that have finished early, congratulations. Here is an additional lab to show how easy it is to design LCD applications with a PIC microcontroller that has an LCD Driver Module. Available on the PICDEM LCD2 are various external components like a 32.768 kHz crystal on Timer1 and pushbuttons on AN0.

You have already completed all the necessary routines to create a basic clock. The only thing that is missing is the code and that initializes Timer 1 and code that increments the clock data values (seconds, minutes, hours). Included to assist you in developing the clock is the Timer 1 section of the PIC18F85J90 data sheet.

Your task is to design a clock with any features you see fit, such as the ability to change hours, minutes, and seconds, stop watch functionality, or a visual alarm. There are also many other icons and segments on the LCD that can be used for a clock application.

- Open Project C:\Masters\11071\Labs\Lab2\Lab2d\LAB2d.mcp” using MPLAB
- Create the code and routines needed to display time in LAB2d.c. Add other features as time permits.
- Compile the project, program the target device, and release the device from reset.

Hints:

- Preloading Timer 1 with the value 32768 (0x8000) will cause a roll over interrupt once per second.
- The interrupt service routine for Timer 1 interrupts has been created (Ln322), but the clock code is missing. You need to fill this in.



11071 LCD Lab Handout

LAB3 – Part 1

Goal: -

Understand how to enable and use the Software Contrast Control feature using PIC18F85J90.

Hardware Connections: -

- Make sure the S6 switch is flipped towards C since we are going to enable the charge pump in this lab.

Software:-

- Open 11071_LAB3.mcp
- Make sure LAB_3A.c is selected in the Source Files.
- Make sure LCD_85J90.h is selected in the Header Files.
- Make sure 18F85J90.lkr is selected in the Linker Scripts.
- Click Build All icon.
- You must get an Error message. Double click the error message.
- Fill in the missing section to configure the LCD module to enable the charge pump.

Hints:-

- Enable Charge Pump.
- Configure for Maximum contrast.
- Select Bias appropriate for the glass being used.
- Select INTRC source for the Regulator clock.
- Build, download and Run. Select ICD 2 as programmer.

Hint:-

- Remove the RJ11 cable after programming.

Test your code:-

- After briefly turning on all the segments, now only MCHP and Varitronix logos will be on.
- Press and hold S1 to confirm it is Lab3A.
- Release S1 to display bias.
 - Note:-
 - The three bits displayed in the lower line are the bias bits in the LCDREG.
 - These bits are updated in real time from the LCDREG register.
- Pressing S3 will increase the bias bits values.
- Pressing S4 must decrease the bias bits values.
- Observe the Change in contrast.



11071 LCD Lab Handout

LAB3 – Part 2

Goal: -

Understand how the various human factors affect the display quality.

Hardware Connections: -

- Make sure the jumper is on S6 switch is flipped towards C since we are going to enable the charge pump in this lab.
- Make sure on J11 jumper AN0 and POT is connected.

Software:-

- Remove LAB_3A.c from the source file and add LAB_3B.c
- Build, and program the device. Select ICD 2 as programmer.

Hint:-

- Remove the RJ11 cable after programming.

Analyze:-

- After briefly turning on all the segments, now only MCHP and Varitronix logos will be on.
- Press and hold S1 to confirm it is Lab3B.
- Release S1 to execute the program.

Note:-

- The data displayed in the lower line is the frame frequency.
- The data displayed in right hand corner of the upper line is the value of LP bits updated in real time form the LCDPS register.
- Use POT R44 to change the frame frequency and observe the how it affects the display quality.

Analyze: -

- Press S3 to increase contrast
- Press S4 to decrease contrast
 - Note: -
 - The data displayed in right hand corner of the upper line is the value of BIAS bits updated in real time form the LCDREG register.
- Observe how the change in contrast affects the display quality along with the refresh rate.
 - Example: -
 - Set the contrast to least and vary the frame frequency for the entire range.



11071 LCD Lab Handout

Quiz

Question 1:-

- What is the minimum frequency required for an optimum display quality?

Ans:-

Question 2:-

- For your minimum frequency setting, is the quality of display good in all viewing angles? If the answer is 'no' then redo the question 1.

Ans:-

Question 3:-

- Does highest refresh rate give you the best display quality?

Ans:-

Question 4:-

- Fill out the table. Set contrast to maximum value for this test.

Table 1:

LP3:LP0	Frame Frequency in Hz	Your Display Quality Rating
0	250	
1	125	
2	83	
3	62	
4	50	
5	41	
6	35	
7	31	
8	27	
9	25	
10	22	
11	20	
12	19	
13	17	
14	16	
15	15	

Note: - Rate using 0 to 5 (5 mean the best quality).

Question 4:-

- What happens to the power consumption when the refresh rate is high?

Ans:-