ECE383: Microcomputers – Lab 7
LCD Interfacing with a PIC24

Goals: The goals of this lab are to introduce students to LCD interfacing with the PIC24.

1. Introduction

Liquid Crystal Displays (LCD) have tremendous popularity among developers and consumers. These devices have been used in a wide variety of electronic products. Versatility and low power consumption are among the most attractive features of LCDs. Industry standard LCDs are controlled by Hitachi's HD44780 chips. These chips allow output and input of character information to and from the LCD displays. In this lab you will:

- Interface a PIC24 microcontroller to a 16x2 (16 characters by 2 lines) LCD via a 4-bit interface
- Output text to LCD

Read through the entire lab and scan any supplied files before starting work. The reporting requirements have you verify operations performed by the PIC24 system and corresponding C programs.

3. TASK 1: Connecting the LCD to the PIC24 system

For this task you will construct an LCD interface (4-bit mode) to the basic PIC24 system constructed in previous labs. The LCD module used in this lab is HDM16216L-5-L30S from Hantronix. It allows character output in 2 lines with 16 characters per line. This LCD supports both 4-bit and 8-bit interfaces. The datasheet for this LCD module can be found here: http://www.hantronix.com/down/16216l5.pdf.

The LCD interface is divided into control lines (E, R/W#, RS), data lines (D7:D0) and power (VDD, VSS, VL, K, A). The 4-bit interface mode used in this lab is used to reduce the number of connections between the PIC24 and LCD module, allowing 4 transfer lines to be used (D7:D4). D3:D0 are unused. Figure 1 shows a PIC24 LCD interface for a HDM16216L-5-L30S standardized around the HD44780 LCD controller. **Additional information on the LCD interface and programming can be found in Chapter 8 of the textbook (pages 302-310)**.

The LCD requires a 5V power supply and ground. During semesters in which students have access to the laboratory spaces on campus this 5V is provided using power supplies available in the lab (Option 1). During semesters in which hybrid or remote lab instruction is required, this 5V supply will be provided from a USB port on a computer, a series of connected AAA batteries, or similar alternative arrangements that are part of the course materials list (Option 2). Video details of the USB power, batteries, and alternative arrangements will be posted on Blackboard.

Also, recall that logic voltage level translation can be accomplished by using the open drain configuration and pull-up resistors. In addition, all “digital only” pins on PIC24 are 5V tolerant. To use this functionality for this lab, wire the LCD as follows:

- Wire all data connections as shown in Figure 1
- Pins RB[4:7] on the PIC should be connected to D[4:7] on the LCD module (U2 on the schematic) leaving D[0:3] on the LCD module unconnected. Connect Pins RB[8, 9, and 13] with E, R/W#, RS.
- Lines D[4:7] and E, R/W#, RS should also be pulled up to +5V using resistors.
- V_{BB} on the LCD module should be connected to ground (GND). **Note: It is important that the GND for both LCD and Microstick are common, that is connected together. If these parts do not share a "common ground" you can get unexpected behavior. Please double and triple check that the GNDs are connected.**
- V_{DD} on the LCD module should be connected to 5V (Power Supply).
- The $V_{DD} – V_L$ voltage difference determines the intensity of the characters displayed on the LCD. V_L can be connected to the wiper terminal of a thumbwheel potentiometer, with another lead on the potentiometer connected to the V_{DD} node and the other connected to ground. In option 1, we will connect V_L to the ground for maximum intensity.

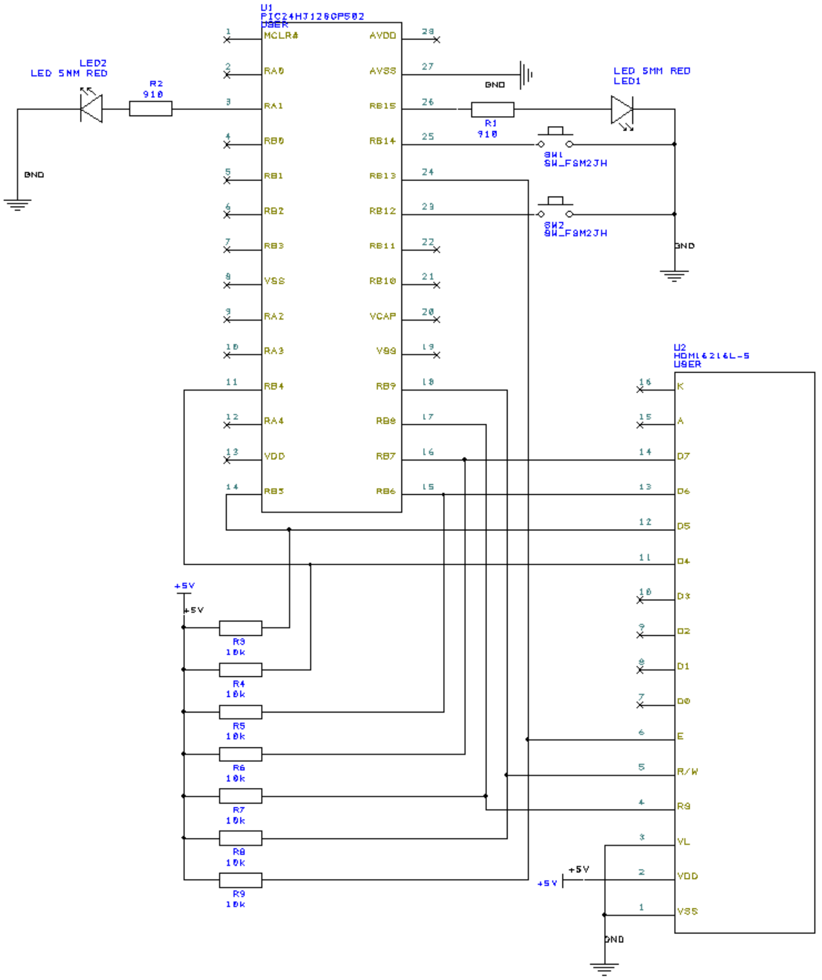

Figure 1. LCD Interface to Microstick II with 5V supply from external in-lab power supply (Option 1)

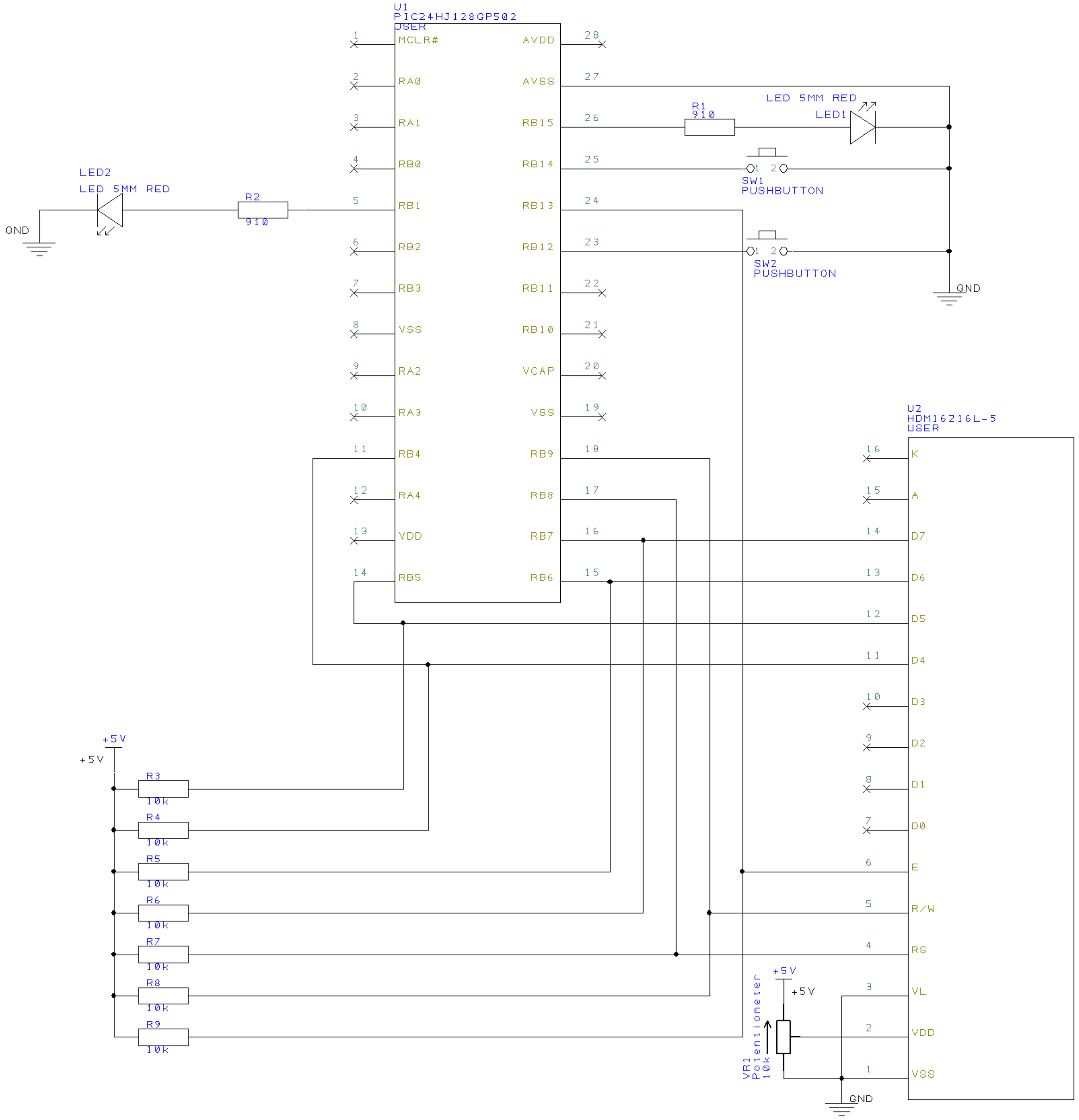

Figure 2. LCD Interface to Microstick II with 5V supply (expected from modified MicroStickII, USB cable, batteries) with potentiometer (variable resistor) for dimming control.

<u>Deliverable 1:</u> Capture/upload a photo of your breadboarded circuit that clearly shows all components (MicrostickII, LCD, pull-up resistors, push-buttons, etc). Full marks will only be awarded for cleanly wired diagrams that can be easily traced when viewing.

As an example of a poorly wired system, see Figure 3. Note: do not literally follow this example, as the connections shown are not the most efficient and easy to replicate).

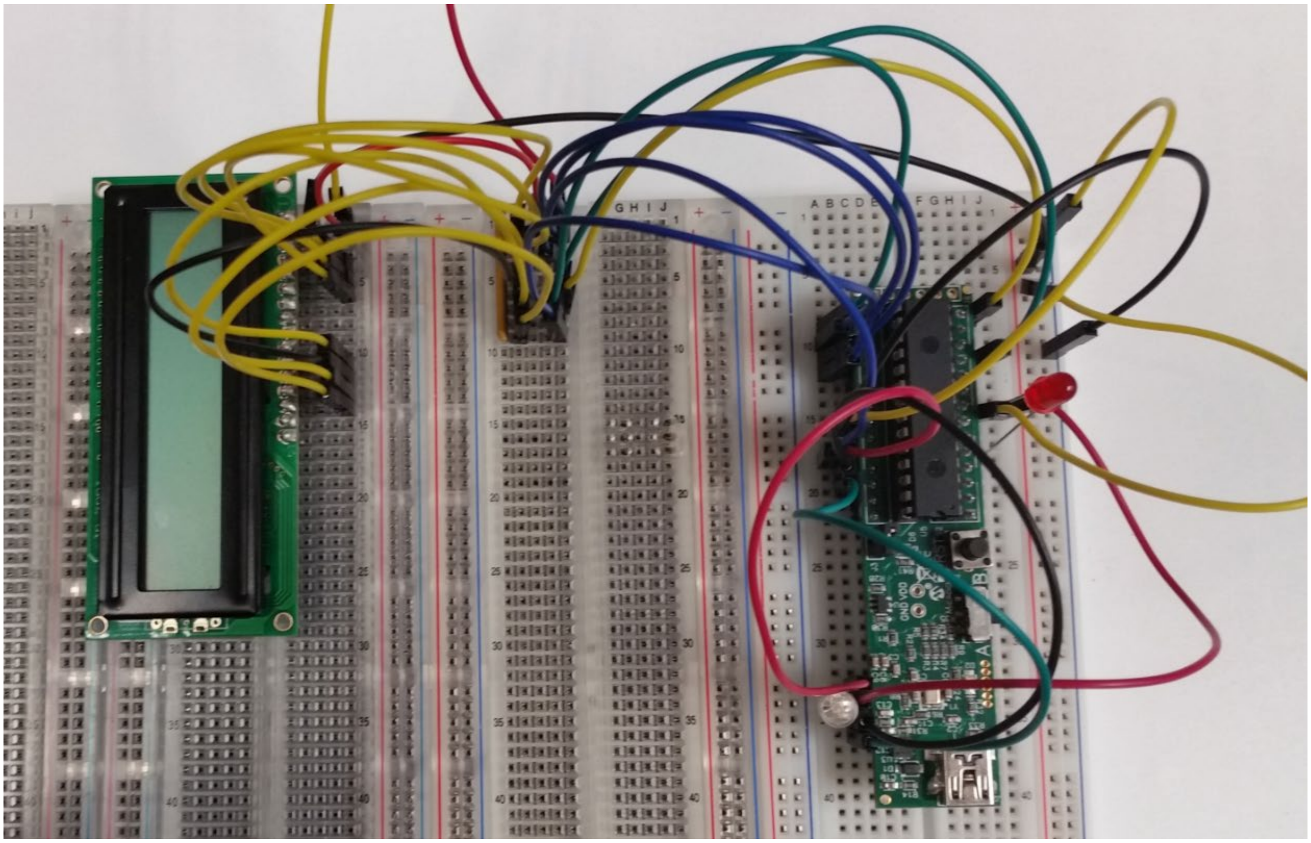

Figure 3. A (bad) example of a wired LCD.

4. TASK 2: Character Output to LCD

- Copy the *chap8* files to a suitable directory on your PC hard disk.
- Start MPLAB. Open the project lcd4bit.mcp (note: you must start MPLAB first, then open the project file). Create a new C source file named “output_name.c”. Add this file to the project, and remove the source file “lcd4bit.c” from the project.
- Modify the pin values in the code according to Figure 1.
- Modify the code to use open drain configuration for all pins connected to the LCD and configured as
- outputs (data lines, R/W, etc).
- With “lcd4bit.c” as a reference, create a program that is capable of displaying your first and last name on the LCD screen as well as your email address.
  - Write a function that displays your first and last name on the top line of the LCD, moves the cursors to the bottom line (1st position) of the LCD, then writes your email address on the bottom line of the LCD.
  - To be fair to those with long names, the characters on the screen should scroll across the display (moving right to left) at a rate of 200 ms. This can be done by taking advantage of the writeLCD function provided in “lcd4bit.c”.
  - The HD44780 LCD controller datasheet is a good reference.

Deliverable 2: Upload a video that captures your TASK 2 program running on the PIC24 hardware (the Microstick II). This video needs to show the successful build, download, and execution of the code. During execution, this video needs to capture the complete details of your scrolling name on the LCD display to confirm all functions were achieved. At the end of your video also make sure to include a final shot that shows your ACT card or similar photo ID. Videos without this final detail will be given a grade of zero.

Deliverable 3: Upload the C-code source file (.c file) you wrote to meet the TASK 2 requirements. Remember that comments are required for all source code to detail the intent of the code. Source code without comments will be given a zero.

5. TASK 3: Counter output to LCD

For this task, you will implement a counter with a visual output to the LCD.

- Using the project lcd4bit.mcp as a basis, create a new project named counter_LCD.mcp.
- For this task, you will create a program (counter_LCD.c) that will output an incremental counter to the LCD screen as follows:
  - Upon initialization of the program, output a character string "000" to the LCD screen.
  - When SW1 is pressed and released, the number on the LCD screen should increment by 1.
  - Once program counts to "015", the number of the LCD screen should change back to "000" and begin incrementing again on the press and release of SW1.
  - To display the integer output, use the following:

```
char str[3];
sprintf(str,"%.3d",count);
outStringLCD(str);
```

  - Pressing SW2 should clear the count and change the displayed value back to "000".
  - Turn on LED1 when the count reaches "015". Turn LED1 off when the count resets back to "000".

An interrupt service routine should be programmed to handle the state changes of the pushbutton. Use change notification interrupts to detect button press and release. Programs that do not use interrupts will be given a grade of zero for both demo and code.

Deliverable 4: Upload a video that captures your TASK 3 program running on the PIC24 hardware (the Microstick II). This video needs to show the successful build, download, and execution of the code. During execution, this video needs to capture all functionality (SW1 incrementing LCD count, rollover from "015" to "000", SW2 resetting count, correct LED1 operation). At the end of your video also make sure to include a final shot that shows your ACT card or similar photo ID. Videos without this final detail will be given a grade of zero.

Deliverable 5: Upload the C-code source file (.c file) you wrote to meet the TASK 3 requirements. Remember that comments are required for all source code to detail the intent of the code. Source code without comments will be given a zero.