EE 337: Microprocessors Laboratory (Spring 2023)

Indian Institute of Technology Bombay

Lab 6: 20 points Date: February 13, 2023

1. [5 points] In this experiment, you will learn to display content on the LCD connected to the Pt-51 kit. Download `lcd.h`, `lcd.c` files and `lcd-control-made-easy.pdf` from the following links.

   - `https://ee337.github.io/2023/downloads/lcd.h`
   - `https://ee337.github.io/2023/downloads/lcd.c`
   - `https://ee337.github.io/2023/downloads/lcd-control-made-easy.pdf`

   The `lcd-control-made-easy.pdf` has general information about LCD operation which is helpful in understanding the code in `lcd.h`. Also, `lcd.h` has comments for each line, try to understand the comments by going through the code line by line.

   - Compile `lcd.c` with header file `lcd.h` and load the hex file on to the kit. Make sure the output on the LCD screen is as shown below:

     ```
       Pt-51
     IIT Bombay
     ```

   - Study the functions used in the `lcd.h` code and their usage. Modify `lcd.c` to display "EE337-2023" on the first line and your **first name** on the second line (truncate to 16 characters if you have a longer name). **Pad the display lines with spaces such that these are centered on the LCD when displayed.** You should load and run this program on the Pt-51 kit.

2. [15 points] Write a program in Embedded C to read five numbers of 4-bit each, sort these numbers in ascending order and display the numbers in the sorted order. Then follow the below steps. For generating the given delays improvise and use `msdelay` function given in `lcd.h` file which generates 1 ms delay.

   - At the start, display **"START PROGRAM"** on the first row of LCD screen for 5 seconds and keep all LEDs OFF.
   - While taking the first input, display **"FIRST INPUT"** on the first row of LCD screen for 5 seconds.
   - Read the first input number from port 1 (P1.3-P1.0) DIP switches and display it on the LEDs connected to port 1 (P1.4 to P1.7) for 5 seconds. Also, store the first input number in an array.
   - While the first input is being displayed on LED, give the second input number using DIP switches and display **"NEXT INPUT"** on the LCD.
   - After the the first input is displayed for 5 seconds, pause for 1 second keeping all LEDs off.
   - Display the second input number on the LEDs for 5 seconds and continue to display **"NEXT INPUT"** on the LCD. Also, store the second input number in the same array. Give the next input through DIP switches during this time.
   - Repeat the above process for the next three numbers as well and place each of them in the array.
   - While the last input is being displayed on the LEDs for 5 seconds, display **"SORTING..."** on the LCD.

- After the last input is displayed, pause for 1 second keeping all LEDs off and display **”SORTING”** in the first row, **”COMPLETED”** in the second row.
- Sort the five numbers in ascending order and display it on LEDs, each for 5 seconds with a pause of 1 second with all LEDs off.
- After displaying the sorted array, give a pause of 1 second with all LEDs off. Then turn on all the LEDs for 5 seconds where you have to take the input to be searched and display **”NUMBER TO BE”** in the first row and **”SEARCHED”** in the second row of the LCD.
- Give a pause of 1 second by clearing both LEDs and LCD.
- Perform any search algorithm on the sorted array of five numbers.
- If the number is present in the array, display the index at which the number is present (assume array index starts from 1) on the LEDs and display **”THE INDEX IS”** on LCD.
- If the number is absent, ON all the LEDs and display **”NUMBER”** in the first row and **”NOT FOUND”** in the second row of the LCD.

TA Checkpoints

- Check the understanding of LCD operation and the `lcd.c` code.
- For question 1, check that the text is properly centered.
- For question 2, check the working of the experiment using the following test cases.
  1. Input array = $\{8, 2, 15, 12, 4\}$
     Item to be searched = `2`
  2. Input array = $\{7, 6, 13, 9, 11\}$
     Item to be searched = `14`