Liquid Crystal Display – LCD

Learning Goals:

- Learn how to use more complicated forms of output.
- Get practice setting up circuits using a schematic.

Materials

1x Arduino
1x Breadboard
1x 220 Ohm resistor
1x 16x2 LCD
1x Potentiometer
Wires

Prelab:

Follow this tutorial to get your LCD screen scrolling hello world:

- https://www.arduino.cc/en/Tutorial/LiquidCrystalScroll

Note that the written instructions on hooking up your LCD screen are incomplete - make sure to follow the schematic! The potentiometer will control the screen brightness. Other LCD info:

- https://www.arduino.cc/en/Tutorial/LibraryExamples/HelloWorld
- https://www.arduino.cc/reference/en/libraries/liquidcrystal/

Lab 3 - Due: Week 5 Monday 2/5 – depending on your scheduled time

In order to become familiar with more functions of the Arduino, you are required to use the sprintf() function to create a format string for your name/nickname you will display on the bottom row of the LCD. Below is a tutorial showing how to use the function for Arduino.
https://www.programmingelectronics.com/sprintf-arduino/

Display a favorite quote and your name on the 16x2 LCD display:

- Your quote on the first line
  - Your quote must be longer than 16 characters long (so is does not all fit on the LCD screen)
  - The quote is to scroll across the screen so it can be read from beginning to end
  - The quote is to scroll again when done. This should continue indefinitely.
- Your name on second line
  - Your name is to be 8 characters or less (this can be a nickname if your name is longer than 8 characters – perhaps your UIC NetID)
  - The name is to be centered on the LCD (as close as possible)
  - The name must NOT scroll/move.

To be considered completed "on time", this Lab needs to be demonstrated by end of Lab on Monday 2/5/2024.

Your code must be submitted to Gradescope BEFORE you demo your lab!

Late Policy

- Lateness is determined by the time the lab is demonstrated, not when the .ino file is submitted.
- Labs that are not demonstrated get a score of 0.
    - -50% for no demonstration
    - -50% for being late
- Any lab demonstrated after 4:50pm on the day on the day when the lab is due is considered late.
- Late Submission 1: by Thursday after Lab day
    - Demonstrated after the end of Lab Time (4:50PM) on 2/5, but before 11:59pm on Thursday 2/8
    - NOTE: This cut-off for on-time is NOT at Midnight!
    - 25% Penalty
- Late submission 2: up to 1 week late
    - Demonstrated on Friday 2/9 through 11:59pm Monday 2/12/2024
    - 50% Penalty
- Labs Demonstrated after 1 week late receive a score of 0 (as if they were never demonstrated).

What should I include with my .ino Code File?

As with any code file, it should be written in Good Coding Style: in a manner that will help other people read and understand the intent, purpose, operation of the code. So your code must include:

- Name the .ino file with your NetId and Lab Number
    - I.E. something like: ptroy4Lab2.ino
- Header Comments (including the following)
    - // FirstName LastName, UIN and NetID
    - // Lab x - Title
    - // Description - what is this code supposed to do?
    - // Include any assumptions you may have made, what do you expect from the hardware, pinouts, particular arduino versions, etc.
    - // References - where did you find code snippets, ideas, inspirations? If no references used state: "no references used". NOTE: It is hard to justify that you didn't use the links given in the prelab. (To be on the safe side, include those links in this section!)
- Code is well documented/formatted with comments, indentations, and descriptive variable names
- Actual code - the functions in the cpp/ino file

Academic Integrity Guidelines:

You may use any resources linked from this lab, or posted by the professor or TAs on piazza/class web page/etc. You should not look at any other internet resources for this. This is an individual assignment, and should be completed on your own. You should not show anyone your code, or look at anyone else's code. You are responsible for writing the code on your own that converts the input to binary which lights up the proper LEDs.