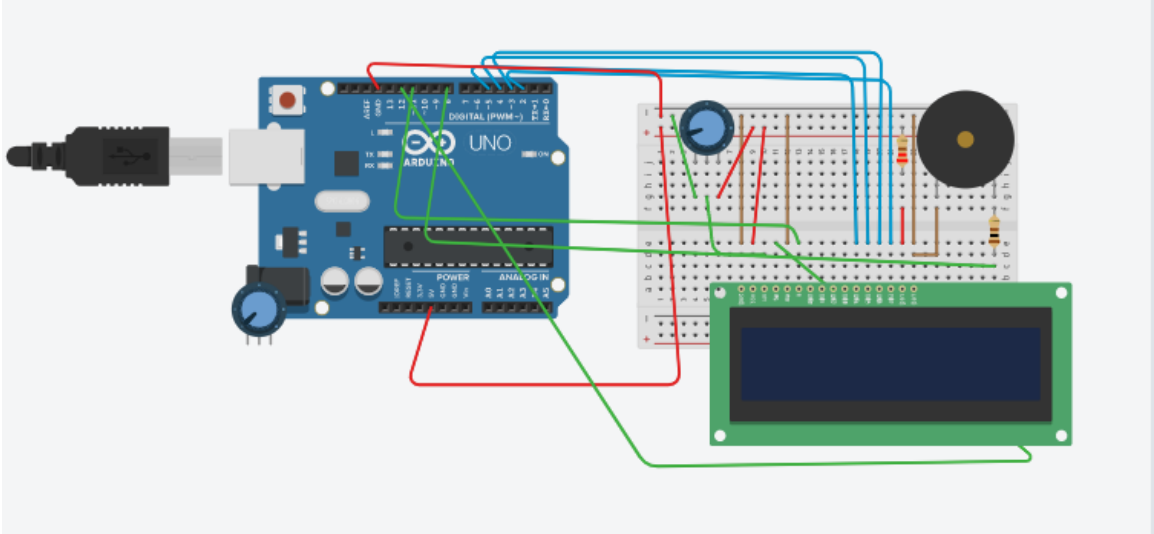
LAB REPORT #5

Name: Kenilkumar Patel
Date: 4/11/21

Screenshot + components:



The components were:

- An Arduino GPU
- A small Breadboard
- A Potentiometer
- A Piezo
- 2 Resistors
- A 16x2 LCD
- wires

Summary:

The lab includes two parts. The first was just setting up the environment, and the second was to modify the code to display the output message “Welcome cs341 students! ” 16 bytes at a time. The first part seemed complicated at first since there were many wires and other parts involved. After wiring the Arduino and the breadboard, properly, “Welcome cs341 St” displayed on the motherboard. The second art was to modify the code so that the LCD would display the whole message in a loop. To implement this, we created a function that would move the characters in the data array to the left by 1 slot. And then would get add 1 character from p. We modified the variable position to go from 0 to the length of the string (not 16 but 24). The function is only called after the first 16 characters of p are stored in data. The value of p[position] will always be the character after the last character of data.

Results:

- The LCD correctly displayed the message by continuously looping.

Conclusions:

We learned how to better use modulo to ensure that the value stays within a specific range. We also learned how to use the LCD specific methods like print and setCursor.

Code:

```
// Program to simulate a ticker tape display
#include <LiquidCrystal.h>
#define DISPLAY_LEN 16  // LCD display width
#define LINE_1 0       // LCD LINE 1
#define LINE_2 1       // LCD LINE 2

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
int buzzer = 8;       // I/O pin for controlling the buzzer
char p[] = "Welcome CS341 Students! ";
int string_length = 24;
char data[DISPLAY_LEN+1];
int position=0;
int counter =0;

void setup() {
 // set up the LCD's number of columns and rows:
 Serial.begin(9600);
 lcd.begin(DISPLAY_LEN, 2);
}
void change(){
 for(int i =1; i<DISPLAY_LEN;i++){
  //move all the characters down 1 slot
        data[i-1] = data[i];
 }
 //in the last slot, add 1 character from p
 data[DISPLAY_LEN-1]=p[position];
}

void loop() {
 // print the number of seconds on line 2
 // starting at the middle:
 lcd.setCursor(8, LINE_2);
 lcd.print(millis() / 1000);

 // turn on the buzzer for a tone =1500 hz
 tone(buzzer,1500);
```

```
 delay(10);

 // turn the buzzer off
 noTone(buzzer);
 //if there is nothing in data, add the first 16 characters from p to data
 if(data[0] == NULL){
  for (int j=0; j < DISPLAY_LEN; j++)
  {
   data[j]=p[j];
   //make position go from 0 to 24
   position =(position+1)% string_length ;
   //Serial.println("pass");
  }
  //since i update position b4 i check if j<display_len, position will end up
being 17, i need it to be 16 for the char 'u'
  //so at the end of the loop, icrement position by 1
  position--;
 }
 else{
        change();
 }
 //data[position] = p[counter];
 //counter = (counter+1)%string_length;
 //position =(position +1)% DISPLAY_LEN ;

 // set the cursor to column 0, line 1
 // (note: line 1 is the second row, since counting begins with 0):
 lcd.setCursor(0, LINE_1);
 lcd.print(data);
 delay(400);

 //still need to update position to add a character from p to data
 position =(position +1)% string_length ;
 //Serial.print(position);
 //Serial.println(data);
}
```