

Design and Implementation of a Lighting System for SHU Engineering Electric Vehicle

Kaitlyn Mangano

Tolga Kaya

ENGR-314-A - Directed Research in Engineering

Abstract

In this project, I utilized a coding software and computer-aided design to create a lighting system for the engineering department electric vehicle. For instance, I design a cone shaped compartment for the front and rear lights of the vehicle. I coded two LEDs for the back of the vehicle so that when it is put in reverse, the LEDs blink. The LEDs in the front are wired to turn on when the vehicle turns. I attached an LED strip inside the seating compartment. As the driver presses on the throttle the LEDs will turn on based on the different voltage caused by the pressure on the throttle when powered. The harder the throttle is pushed; the more LEDs turn on.



Methods and Materials

The materials used for this project were 4 big LEDs, breadboard, Arduino, jumper wires, (2) 1-ohm resistors, LED strip, Prusa i3 MK3, power source, and printer cable. The software I used was Arduino IDE.

First, I designed a cone shape in Fusion 360 by creating a 5-inch circle in the sketch tab. Then, I created another circle with a diameter of .20 inches on the same plane just 3.905-inch higher up. Next, I used the loft tool to create the cone. By using the shell tool, I was able to make the shape hollow with a thickness of .114 inches. Afterwards, I extruded the .20-inch circle inwards towards the cone to make a cut out at the top for the LED to rest in. After I 3D printed it, I soon realized that it was too big and decreased the diameter by half and the length by 1.833 inches.

I put two big LEDs side by side on a breadboard with both positive and negative legs in the same row facing the same direction. Then, I connected a 1-ohm resistor in the same row as the anode side of both LEDs. The other side of the resistor I connected to power on the breadboard (5 volts). I grounded the cathode side of both LEDs to the breadboard. I connected the ground from the Arduino to the breadboard. Next, I connected 5 volts from the Arduino to the breadboard. Afterwards, I set up another small similar circuit which consisted of two big LEDs with both positive and negative legs in the same row facing the same direction. Then, I connected a 1-ohm resistor in the same row as the anode side of both LEDs. The other side of the resistor I connected to pin 6 on the Arduino. I grounded the cathode side of both LEDs to the breadboard.

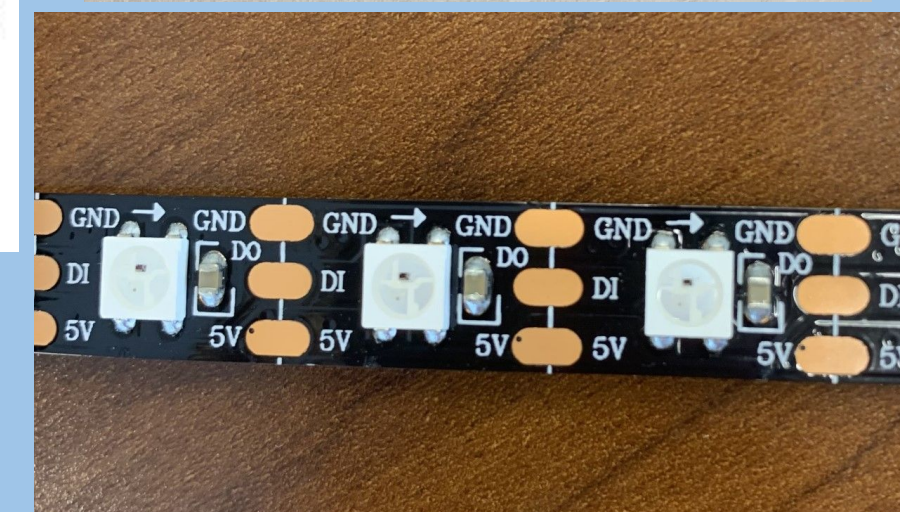
For the LED strip, I connected the red wire on the strip to power, 5 volts, on the breadboard. The green wire I connected to A0 on the Arduino and the black wire to ground on the breadboard.

```
4_LEDS | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help

4_LEDS

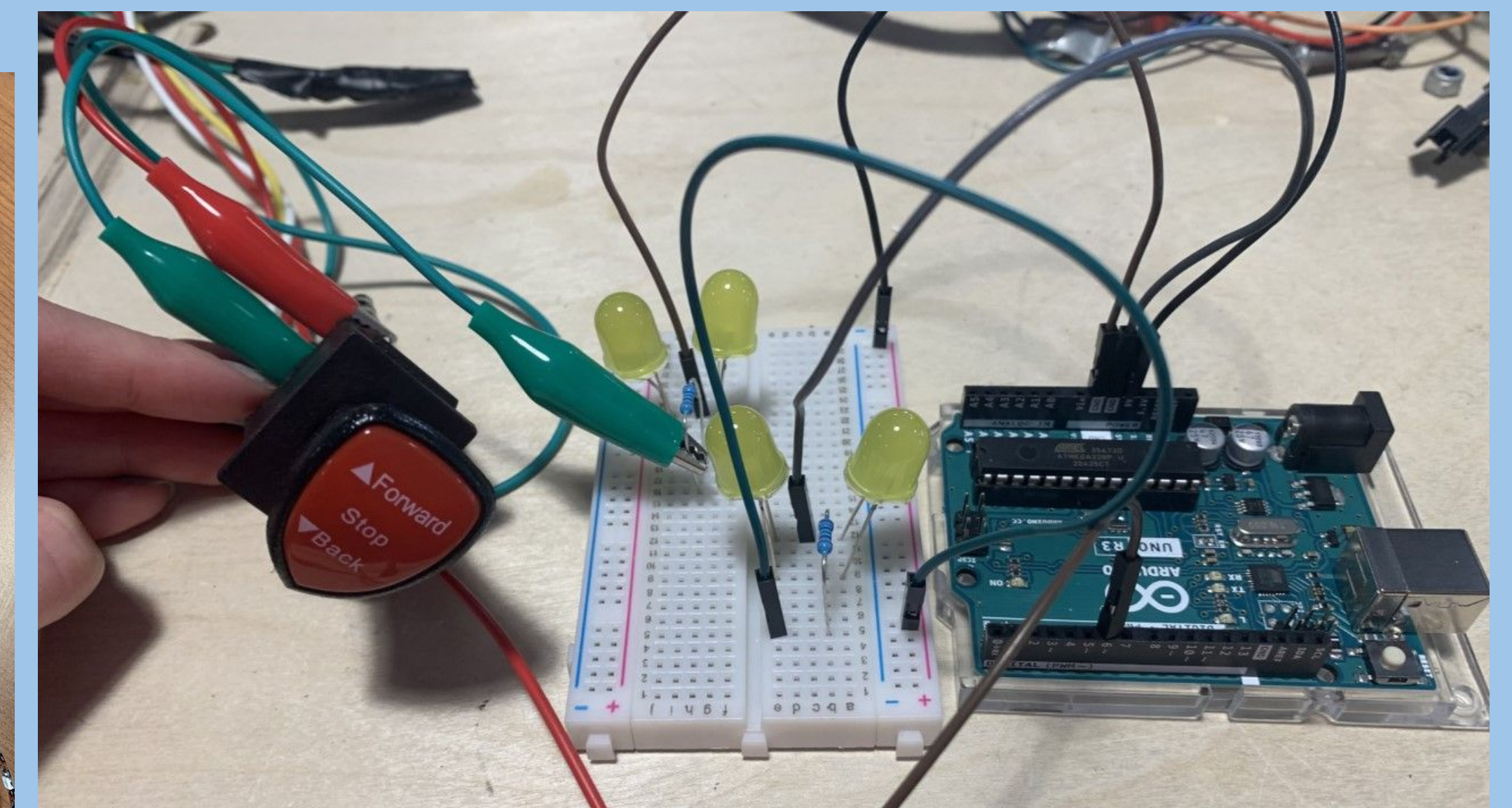
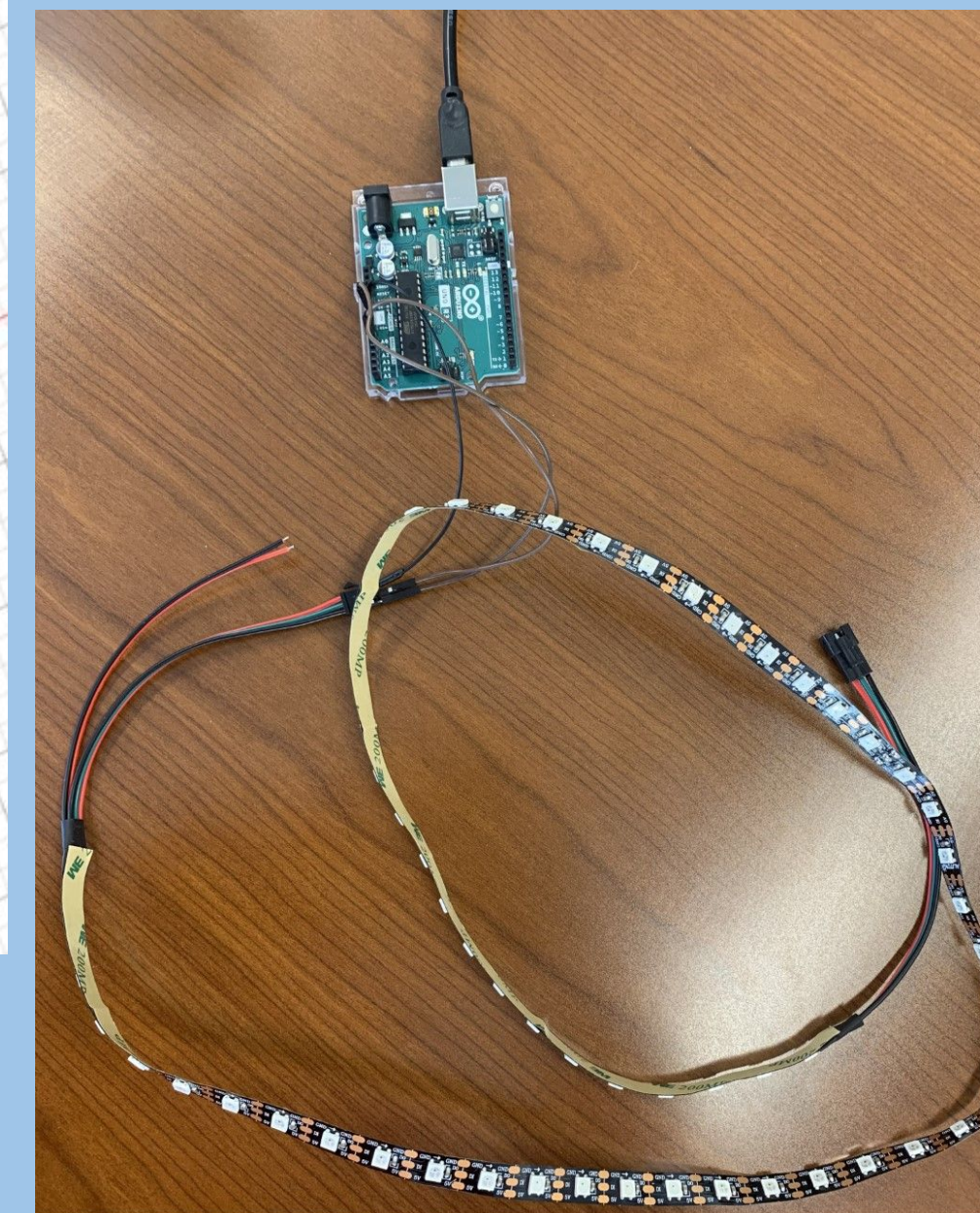
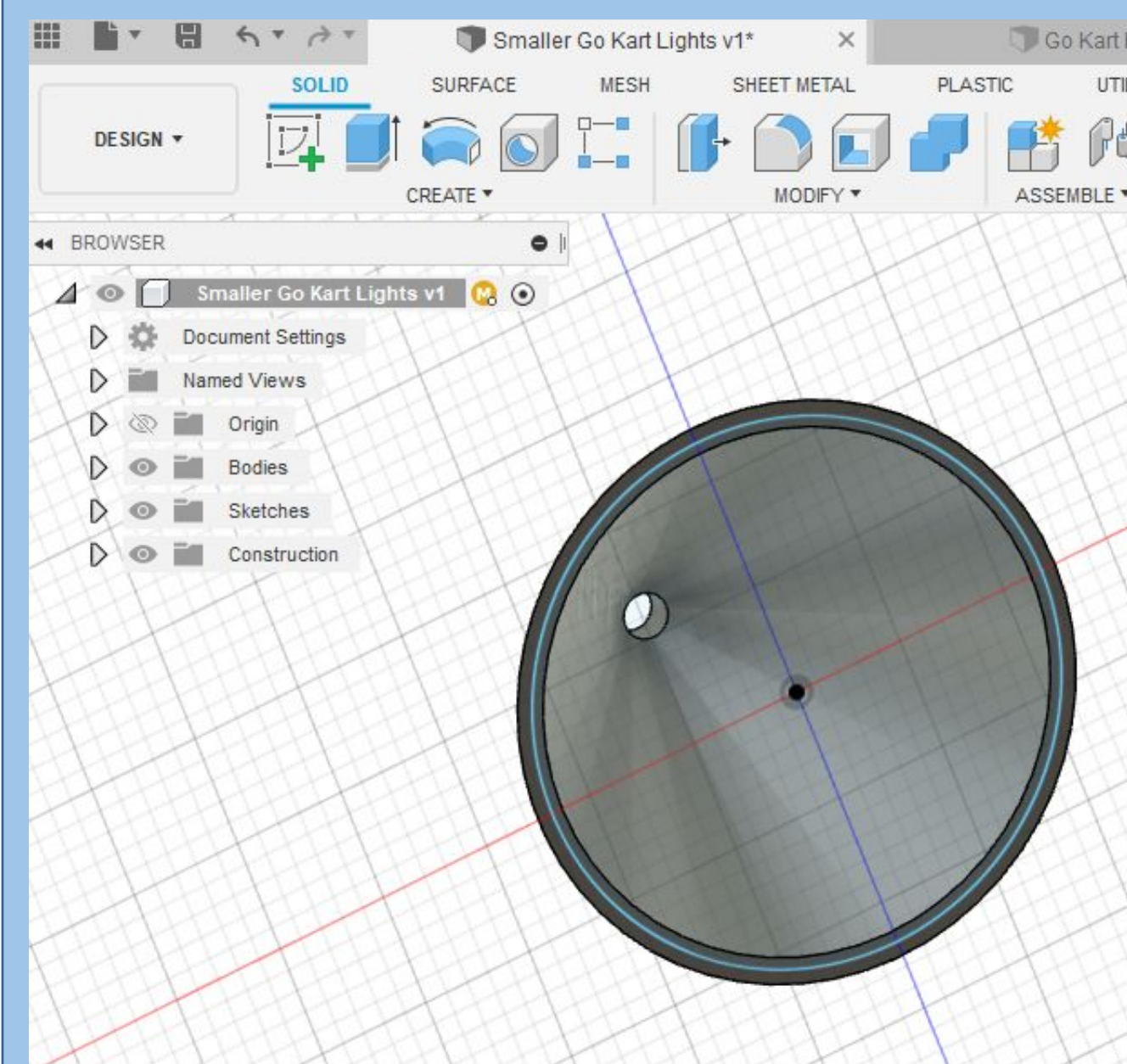
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(6, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(6, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);           // wait for a second
  digitalWrite(6, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);           // wait for a second
}
```



Code

I wrote my code for the project in Arduino IDE. In the beginning, I setup the function. Then, I initialized the digital pin as an output. Next, I created a loop function that runs repeatedly. Within the loop, I turn on the LED making the voltage level high. I pause the code for 1 second and then turn the LED off by making the voltage low. At the end, I pause the code for 1 second. This allows the LEDs to blink on and off.



Plan

I plan to code the LED strip according to the different voltage produced by the pressure on the throttle. The range is between 3.8 and .8 volts. I was able to test this by opening up the throttle and using a multimeter. To write this code, I will be using Arduino IDE and create an if else statement. For example, if voltage is between 3.8 and 3.6 slowly turn on 4 LEDs one by one, elseif if the voltage is between 3.6 and 3.4 slowly turn on 10 LEDs one by one...

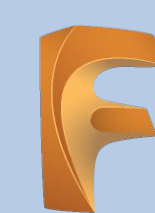
To assemble my design, I will need to use mini breadboards, one for each LED so that I could position them properly on the vehicle. As a result, each LED will most likely need its own resistor. It would need to be 50 ohms as I currently using 100 ohms for two LEDs. I will insert each LED into the cone before attaching it to the breadboard. I will work with my team to find the best way to secure the headlights on. I will do the same process for the rear lights. I would continue to collaborate with my team to find the proper placement for the Arduino inside the vehicle. If I want the front LEDs to turn on when the vehicle does, I need to solder two wires to the ignition key and connect the other side to the LEDs. For the rear lights, I could either solder or use alligator clips to connect the rear two lights to the forward/reverse switch on the vehicle. I would need to connect the side of the resistor that isn't connected to the LEDs to the middle prong on the forward/reverse switch and connect pin 6 on the Arduino to the top prong on the forward/reverse switch. If each LED is on its own breadboard, then I would need to connect the two resistors on each with a wire and the other end of that wire would be connected to the middle prong.

Since the Arduino only needs a small amount of voltage and the batteries on the vehicle are 48 volts, I would buy a regulator. I would connect it to the battery and Arduino so I could control the amount of voltage I am providing the Arduino with.

Contact

Kaitlyn Mangano

manganok@mail.sacredheart.edu



AUTODESK®
FUSION 360™



Sacred Heart
UNIVERSITY

ENGINEERING