Jonathan Hess

Section 3.2

## Week 8 Lab Report: Circuits (II)

# Lab Report Rubric

|  |  |  |
| --- | --- | --- |
| **Category** | **Student Score** | **Grader Score** |
| **Organization** | | |
| **Appropriate sections** | **1/1** | **/1** |
| **Appearance and formatting** | **2/2** | **/2** |
| **Spelling, grammar, sentence structure** | **1/1** | **/1** |
| **Work** | | |
| **Experimental procedure** | **2/2** | **/2** |
| **Results (data, code, figure, graph, tables, etc.)** | **1.5/2** | **/2** |
| **Conclusion** | **1.5/2** | **/2** |
|  |  |  |
| **Total** | **9/10** | **/10** |

# Introduction

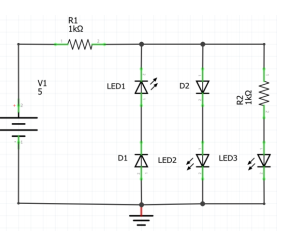
This week we worked on circuits again making use of our arduino kits. We used the parts to learn about diodes and pushbuttons. We programmed a bit with arduino making use of the inputs and outputs provided.

# Procedure

## Procedure

E**xplain in detail** what you did to find the answer to your question. The reader should be able to follow your procedure and repeat exactly what you did.

*First I set up this circuit bellow:*

**

*Before connecting it to power I predicted that Leds 2 and 3 should turn on because diodes point in the direction of conventional current (using the triangle as an arrow towards negative). Also side note, the diodes would have a voltage drop of .7 volts, the leds around 1.7.*

*led2 voltage = (if ignoring led 3)*

*r voltage = 5-.7 -1.7 = 2.6 volts*

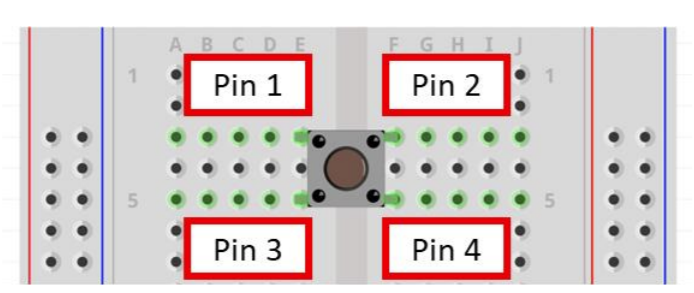
*r amperage = 26 mA*

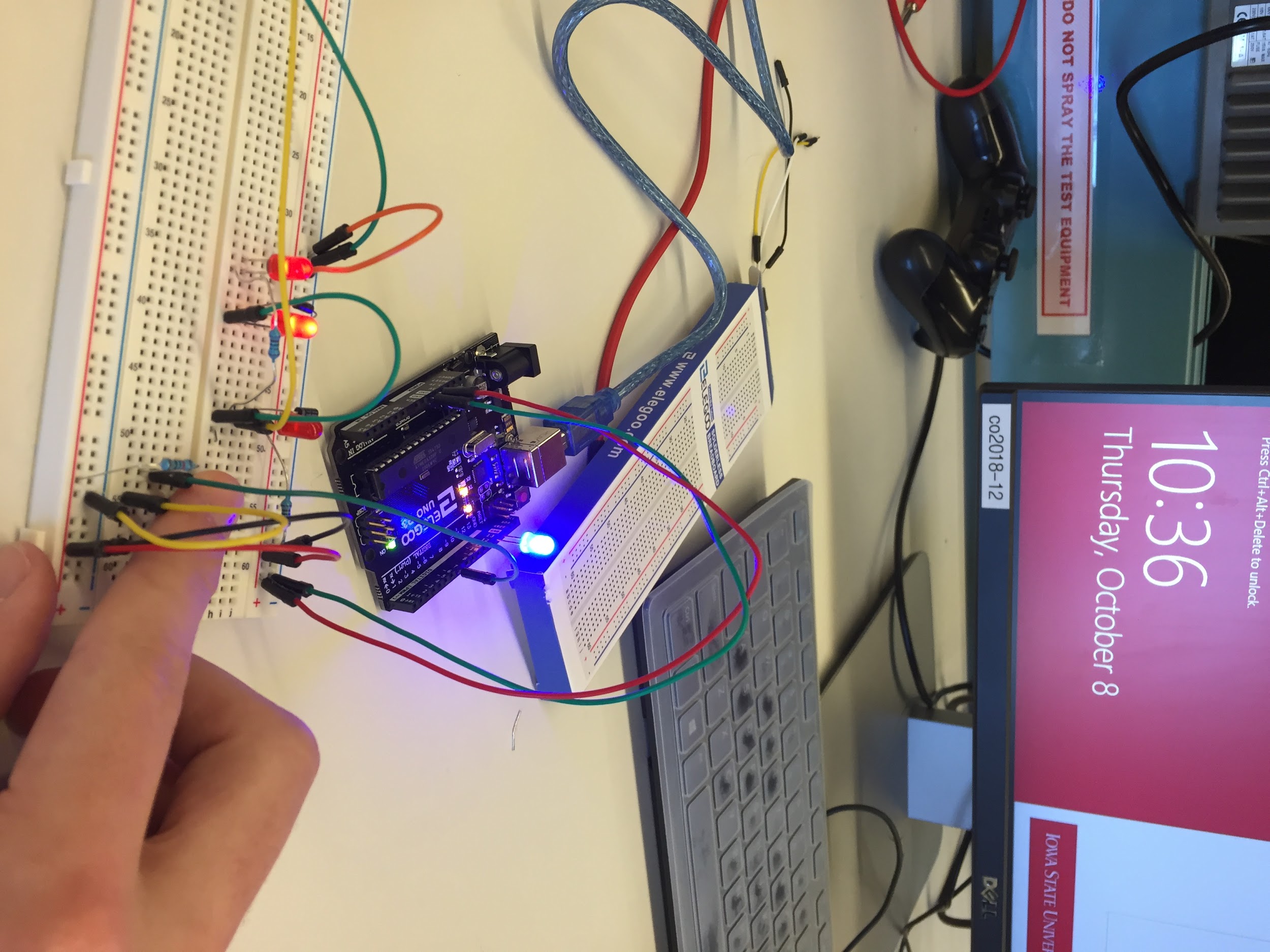
*Next we were told to turn the voltage around to get -5 volts instead. I swapped the power leads.*

*I suspected that this would make LED1 turn on while 2 and 3 would turn off. This was for the same reasons as above.*

*Lastly before playing with arduino we worked with push buttons*

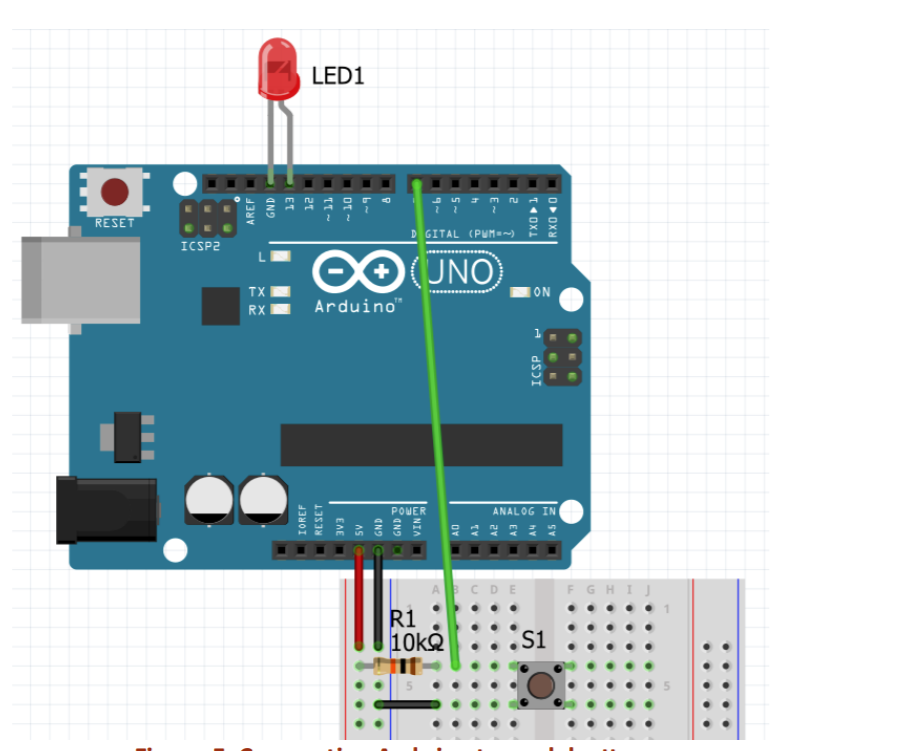
*The push button has 4 pins the lab labeled them as follows:*

**

*Using the multimeter continuity test and Leds I was able to find that pins 1 and 3 are connected with the press of the button as well as pins 2 and 4. I did this by connecting a basic led circuit to the button and trying the different pins, later I used the multimeter to check what I did.*

*Last we were allowed to mess around with arduino*

*The task was to wire the arduino like so:*

**

*and upload the code given to get a working led switch via arduino.*

*Questions:*

*1. Will the Arduino take in the first three lines considering they are not in the setup() or loop()*

*loops? What is their purpose?*

*Yes, this space is for declaring and setting initial values for variables.*

*2. What is ‘boolean’ in line 3? How does that affect the variable button?*

*Boolean means either true or false so it will either be 1 (True) or 0 (False)*

*3. In line 7, what is ‘INPUT\_PULLUP’? Use the Arduino language reference website provided on Canvas to research this. Try to explain it in your own words for the report.*

*This means that a HIGH voltage means a false and LOW voltage equals true. This is because the arduino is using its pull up resistor which brings a 20k resistor between the 5 volts and the pin so a drop in voltage (like a button press) drops the voltage to 0.*

*4. (Related to Lab 2) What do line 8 and subsequently line 18 do?*

*The Serial.begin(9600); sets up a serial connection (with the computer) at 9600 as the baud rate. Line 18 then tells the value of the button into the serial connection.*

*5. Can you figure out what is the purpose of this code? How about the circuit design?*

*The button when pressed brings the pin to 0 volts and turns the led on because of the code. The code checks the state of the button and changes the state of the led accordingly.*

*6. Are you able to draw a systems level diagram based on this code?*

**

*1. Play around with your circuit. What happens when you activate the push button?*

*The LED will turn on when you press the push button, this is because the input pin will go to 0 triggering it to turn on the LED.*

*2. Open the Serial Monitor. What do you see? Activate the push button, what do you see? What does this tell you about inputs to the Arduino?*

*The Serial Monitor is outputting the value of the pushbutton. This is because of the Serial.println(button); It is a pulldown*

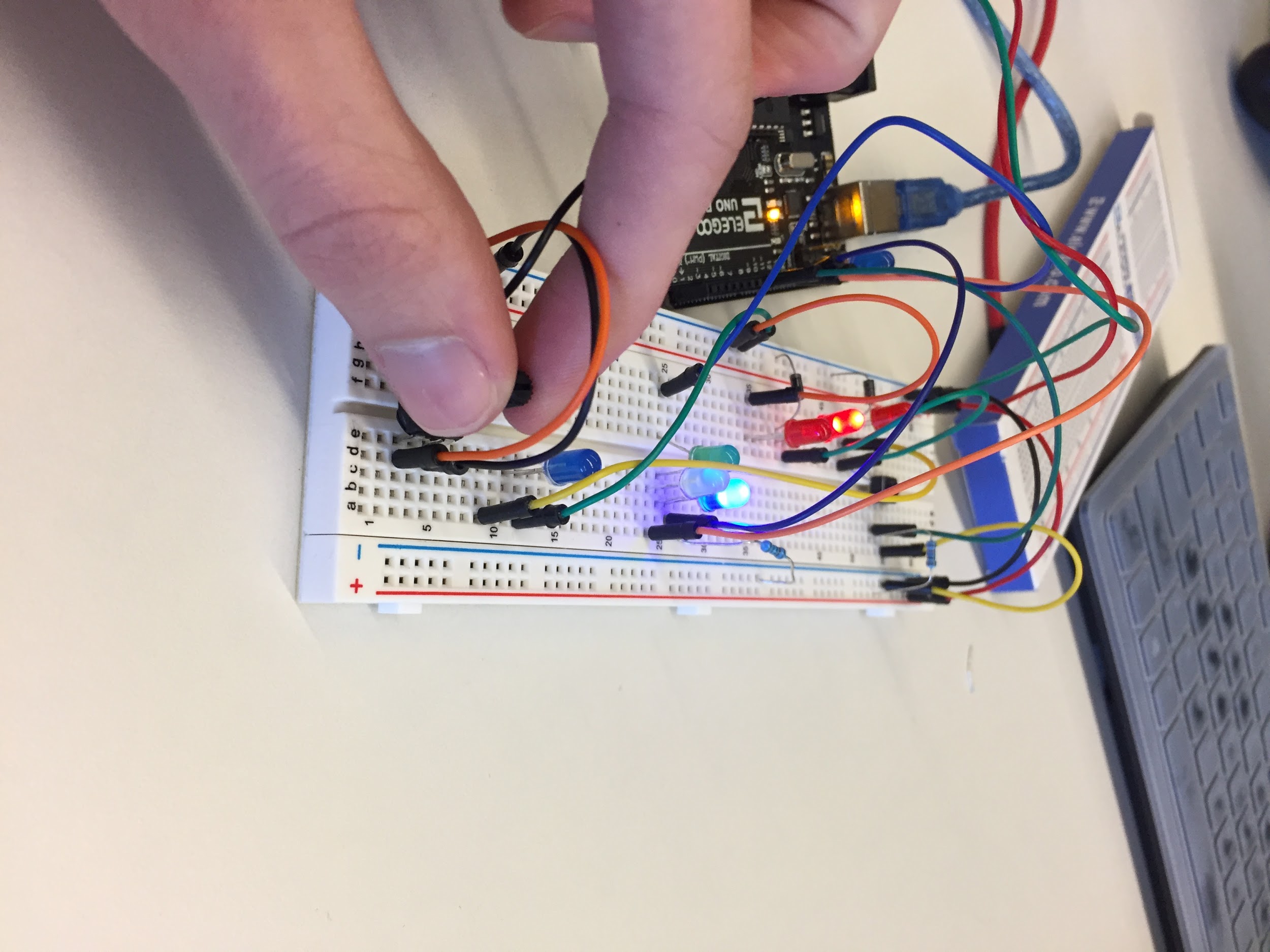
*3. What is the purpose of the 10 kΩ resistor?*

*Looking at it, I have no idea why there is a 10k resistor. The pullup resistor is already 20k so it doesn’t seem to help with current and it doesn’t really do anything, I took it out of the circuit and nothing changed.*

*4. What else can you explore?*

*For this last question I took it upon myself to use an analog input rather than digital so I could switch between the 3 colored LEDS.*

*The code: (I edited the previous code with a few if statements and with analog input)*



int inputPin = 7;

boolean button = 0;

int blue = 11;

int green = 12;

int yellow = 10;

int anolog = 0;

void setup() {

pinMode(ledPin, OUTPUT);

pinMode(inputPin, INPUT\_PULLUP);

//mycode

pinMode(blue, OUTPUT);

pinMode(green, OUTPUT);

pinMode(yellow, OUTPUT);

Serial.begin(9600);

}

void loop() {

anolog = analogRead(A0);

digitalWrite(green, LOW);

digitalWrite(yellow, LOW);

digitalWrite(blue, LOW);

if(anolog <=50){

digitalWrite(green, HIGH);

}

else if(anolog <=800){

digitalWrite(blue, HIGH);

}

else{

digitalWrite(yellow, HIGH);

}

Serial.println(anolog);

}

*Extra Challenge: Make a Traffic Light*

[*Video Link for Demo*](https://www.youtube.com/watch?v=fjhzPZUS1WU)

*Most of the challenge was with setting up the wires. Each LED was connected to its own pin on the arduino and then the anode was connected to ground via a 1k resistor to limit current.*

*The code I used was:*

int r1 = 10;

int y1 = 11;

int g1 = 12;

int r2 = 4;

int y2 = 5;

int g2 = 6;

int inputPin = 7;

int button = 1;

void setup() {

// put your setup code here, to run once:

pinMode(r1, OUTPUT);

pinMode(y1, OUTPUT);

pinMode(g1, OUTPUT);

pinMode(r2, OUTPUT);

pinMode(y2, OUTPUT);

pinMode(g2, OUTPUT);

}

void loop() {

// put your main code here, to loop

button = digitalRead(inputPin);

if(!button){

digitalWrite(y1, HIGH);

digitalWrite(g1, LOW);

delay(5000);

digitalWrite(r2, LOW);

digitalWrite(g2, HIGH);

digitalWrite(r1, HIGH);

digitalWrite(y1, LOW);

delay(5000);

digitalWrite(y2, HIGH);

digitalWrite(g2, LOW);

delay(5000);

digitalWrite(r2, HIGH);

digitalWrite(y2, LOW);

}

else{

digitalWrite(g1, HIGH);

digitalWrite(r2, HIGH);

digitalWrite(y1, LOW);

digitalWrite(r1, LOW);

}

}

*This is very simple code that probably could have been shrunk by making a function that sets all to low. Before writing this code, I didn’t think the code would look as messing as it does.*

# Results

For the first part I found it interesting how the diodes worked and seeing the voltage drop of them and the LEDS. I relearned which way a diode would have to face in order to allow the current to flow. With the push buttons I found it interesting that you could have 2 different inputs. Also while working with the pushbutton, I learned more about pullup and pulldown resistors, I learned a bit about them when it came to game system controllers.

# Conclusions and Reflection

In this lab I got to work more with diodes and push buttons. I have had some experience before with diodes and wished I could have used some of the fancy types. I have only ever read about zener diodes or photodiodes, maybe in a future lab we will use more variety. One thing that was new was the multimeter. I had never used a continuity tester on a multimeter, it seems very helpful for finding accidental shorts in a circuit or finding gaps. Other than that I found the push buttons interesting because they are wired so that pins 1 and 2 aren’t connected so you could run two different circuits off of one push button. I found the use of the push button with the arduino will be more useful when the arduino is programmed to do something other than turn on an LED. In one of the free labs I plan on making a controller that interfaces with some sort of system so I can’t wait until I get to use it there. Also it reminded me of a button matrix that might be useful for that future project.