Portfolio of work

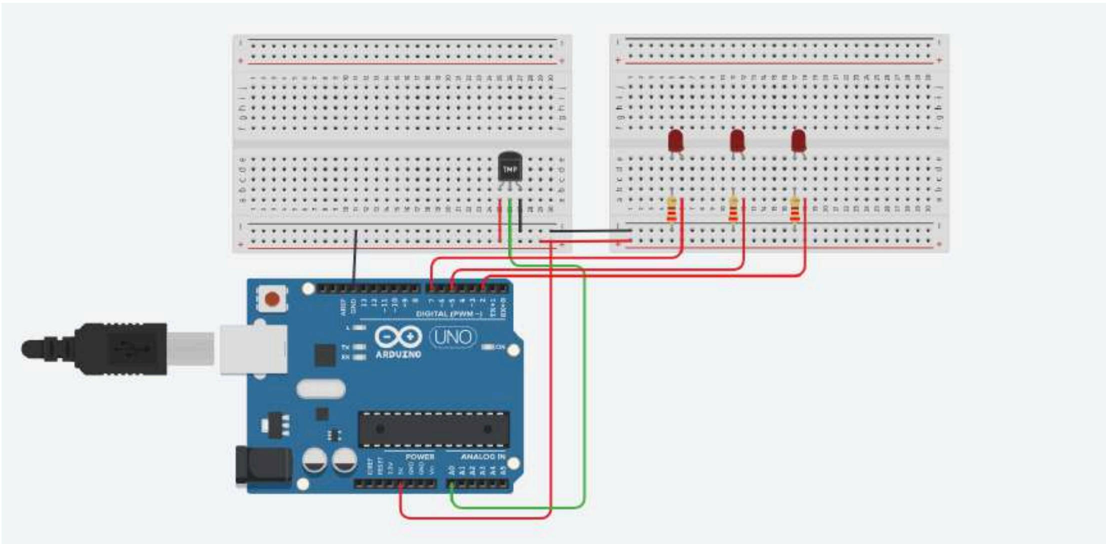
My Github link : https://github.com/msc-creative-computing/p-comp-week-1-labs-orionmel

Weekly Project

Week1

Code Link:

https://github.com/msc-creative-computing/p-comp-week-1-labs-orionmel/blob/main/week1/week1.ino



The survive equipment

The background:

In the future, because of the constant demand for human materials to the earth, the earth has become barren, the ambient temperature has become unpredictable, it is difficult to survive, people as the only living things on the earth need to have special equipment with the temperature of the protective clothing (named "survive equipment") to carry out external activities.

The function of this survive equipment:

This device can maintain a suitable temperature for human survival for a certain period of time, but over time, the temperature inside the device will decrease, the device will initially display three lit cue lights, but when the temperature drops to 25 degrees (this is the most suitable temperature for human survival), it will turn off a lamp, until the light is lowered to the last lamp (finally, when the temperature is below 5 degrees), the user must immediately find a survival base, the device need to be charged, but the last lamp will turn on the device's the thermostatic mechanism, The time of this thermostatic mechanism lasts only 30 seconds until all lights are finally turned off and the people inside the device die because the equipment is too cold.

The “affordance” of this survive equipment:

Three warning lights are a way of connecting the user to the environment, the temperature inside the device is reduced because of the continuous low temperature of the external environment, and the temperature mechanism unique to the device is constantly approaching the external environment with time changes, three lights are the only way for the user to remind, people need to sense the reaction of the light through visual sense to give cognition feedback

Of course, this process need a certain learning process, like Gibson and Pick (2003,p7) said, Affordances do not automatically present themselves to the actor. Instead, they typically must be discovered through perceptual learning, and actors must learn to use the affordances, which in some cases “…may require much exploration, patience, and time”

Finally, the use of visual reminders is not enough, after that, I will also consider using sound and feeling information for further thinking of the device.

reference:

Gibson, E. J. and Pick, A. D. An Ecological Approach to Perceptual Learning and Development. Cary, NC: Oxford University Press (2003).

Week2

My code in each lab :

https://github.com/msc-creative-computing/p-comp-week-1-labs-orionmel/tree/main/week2

Lab0: serial and parallel LED

Q1:How many leds can you connect to a 220K resistor in series before the voltage runs out?

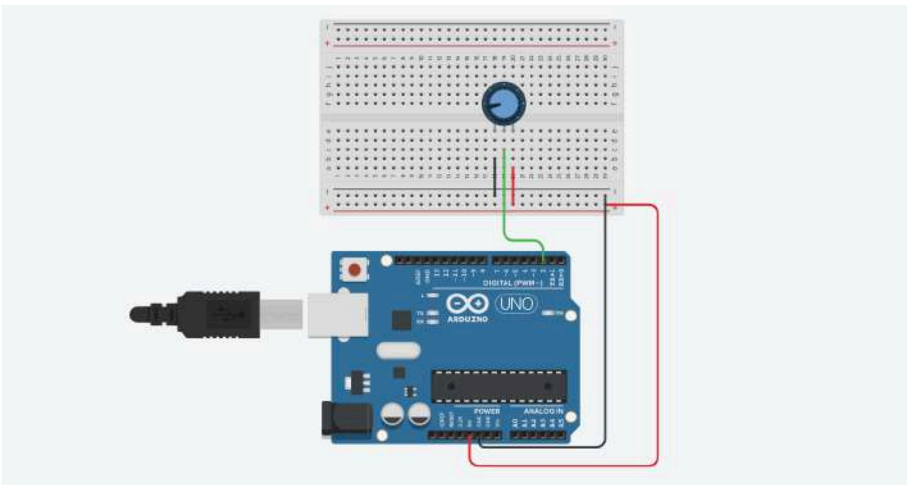
A1:3 Leds

Q2:How many leds you connect in parallel with the 220K resistor before the voltage runs out?

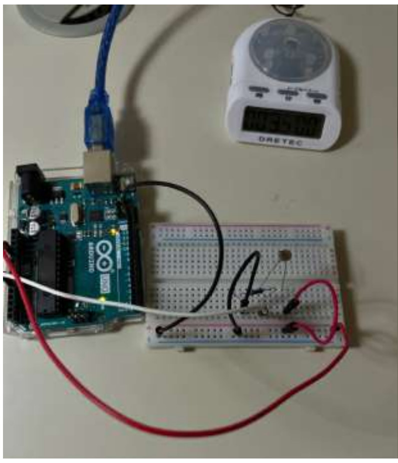
A2: Unlimited

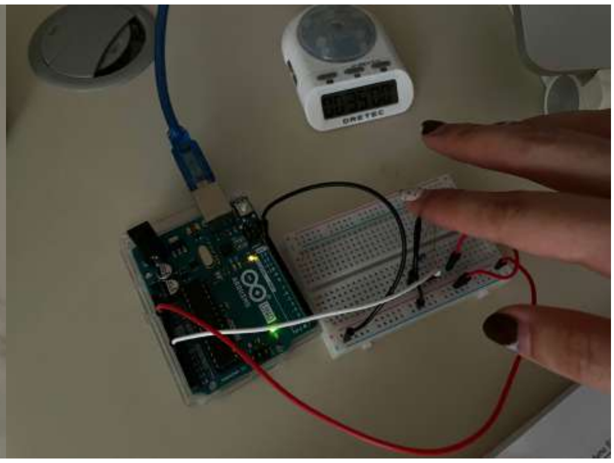
Lab1: Experiment 01-serial data

Lab2: potentiometer



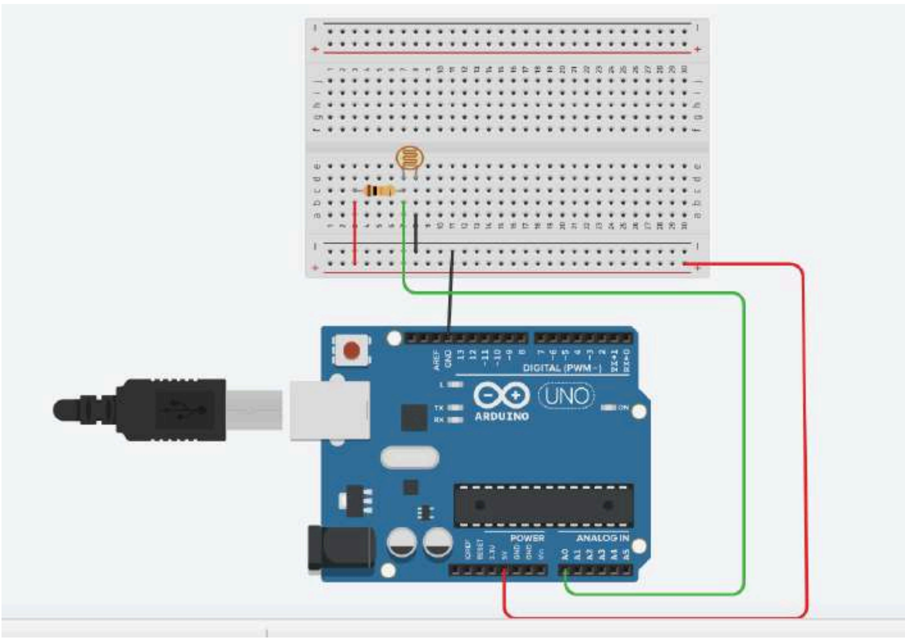
Lab3: photosensitive





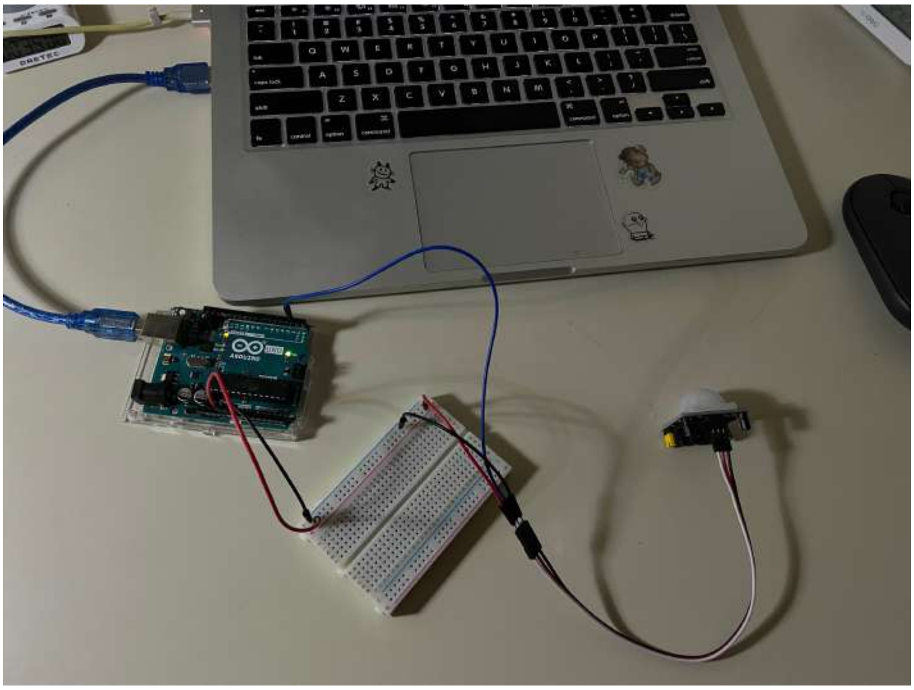
/dev/cu.usbmodem14101

```
Voltage: 1.03 V
Resistance: 17933.27 ohms

Voltage: 1.02 V
Resistance: 18040.86 ohms

Voltage: 1.02 V
Resistance: 18040.86 ohms

Voltage: 1.03 V
Resistance: 17933.27 ohms

Voltage: 1.02 V
Resistance: 18040.86 ohms
```

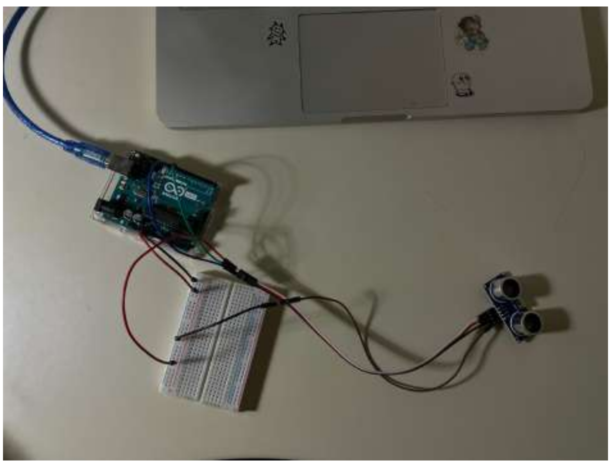


Lab4: play with common sensors

1. **PIR**

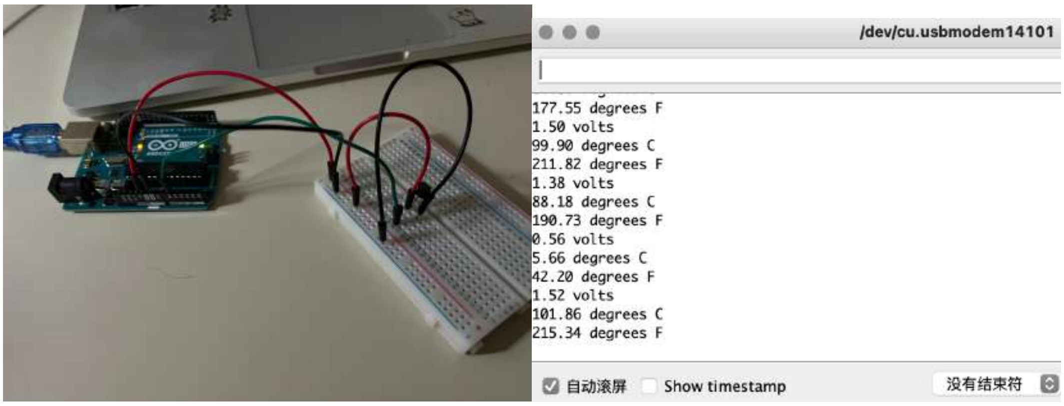


2. Ultrasonic transducer

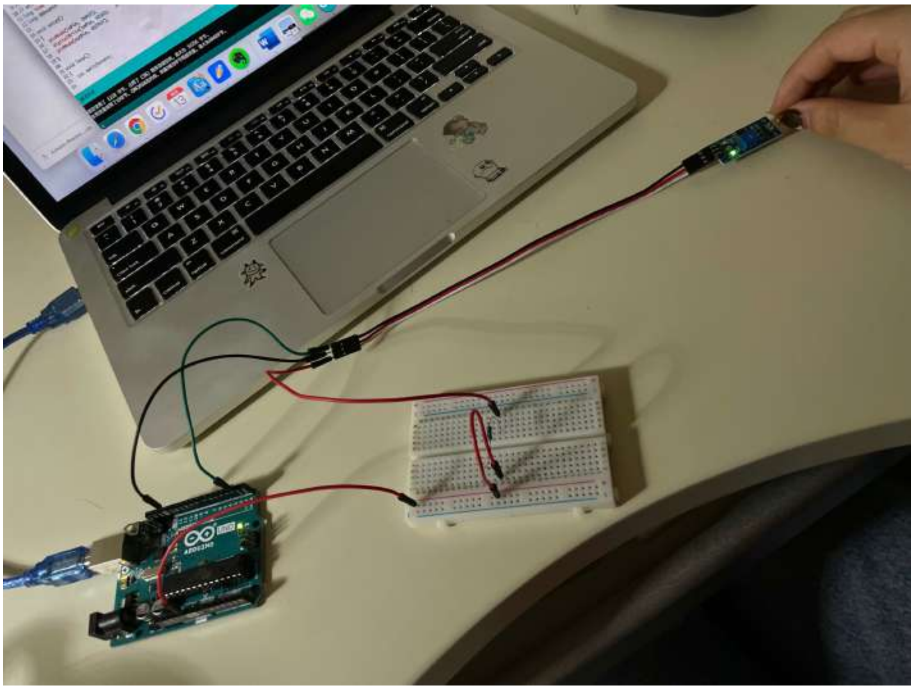


3. LED_receiver

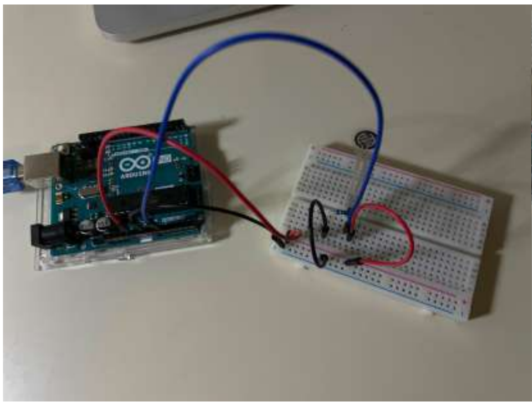
4. TMP36 temperature

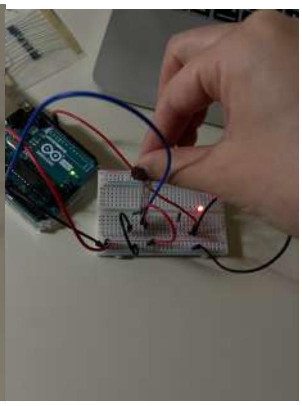


5. Tilt sensor

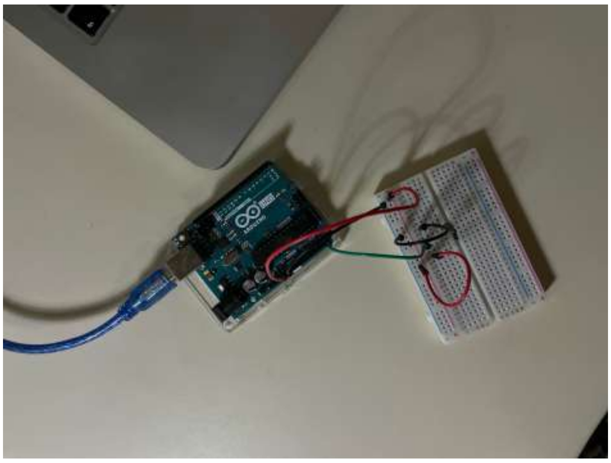


6. Force-sensitive resistor

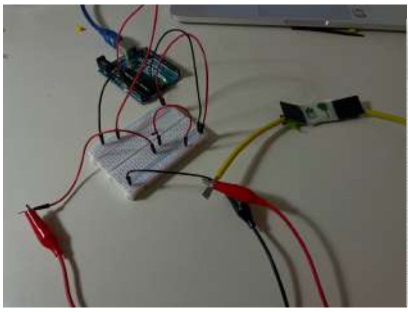




7. LDR



Lab5: Home-made work [advanced]



Week3

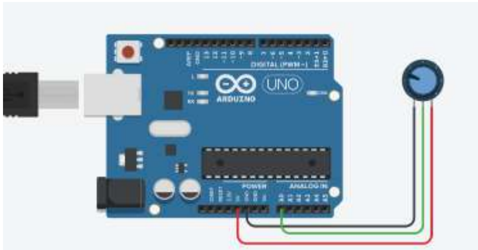
My code link:

https://github.com/msc-creative-computing/p-comp-week-1-labs-orionmel/tree/main/week3

Lab1: PWM

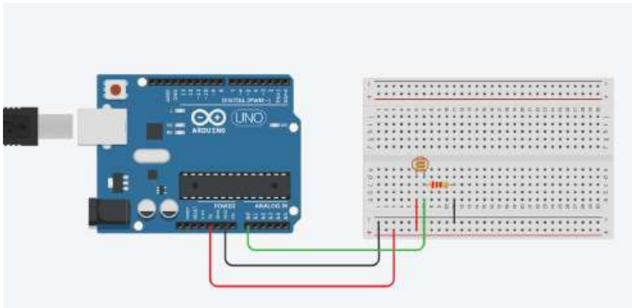
Video: https://vimeo.com/636991759

Lab2: Voltage Dividers



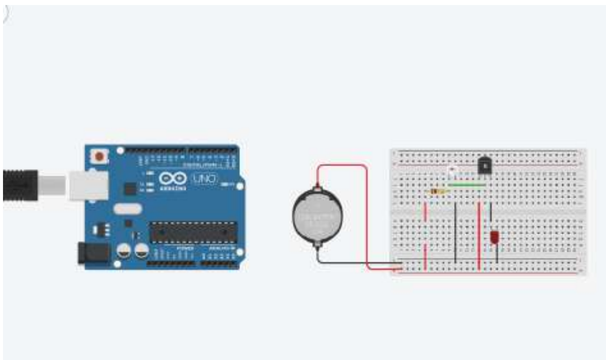

```
int sensorValue = 0;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  // read the input on analog pin 0:
  sensorValue = analogRead(A0);

  // print out the value you read:
  Serial.println(sensorValue);

  delay(10); // Delay a little bit to improve simulation performance
}
```

Lab2: using LDR




```
// C++ code
//
int sensorPin =A0;
int ledPin = 13;
int sensorValue = 100;

void setup()
{
  pinMode(ledPin, OUTPUT);
}

void loop()
{
  sensorValue = analogRead(sensorPin);
  digitalWrite(ledPin,HIGH);
  delay(sensorValue);
  digitalWrite(ledPin,LOW);
  delay(sensorValue);
}
```

Lab3: phototransistor



Lab4: Feedback systems

Description:

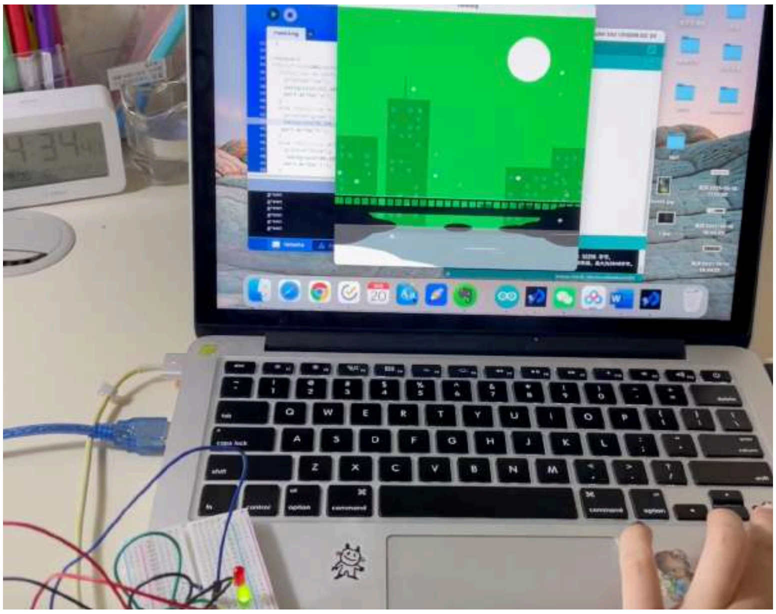
Systems which feedback on themselves often feel more alive to us than static systems which only output data. Design a system, based on what we have learned so far, which feedbacks on itself. How could interaction work within a system which is continually feeding back on itself?

Outcome:

Video: https://vimeo.com/636998279

According to Don Norman's affordance theory, I imagine a certain feedback mode in the future, and the entire city's work system is in the hands of the highest class people.

The day and night in the city are no longer provided by the ecosystem, but by the buttons which in the hands of the controller: [left] [right] [down]. People use the urban environment to give feedback on their work day and night, but the controller controls the urban environment, creating a strange vicious circle

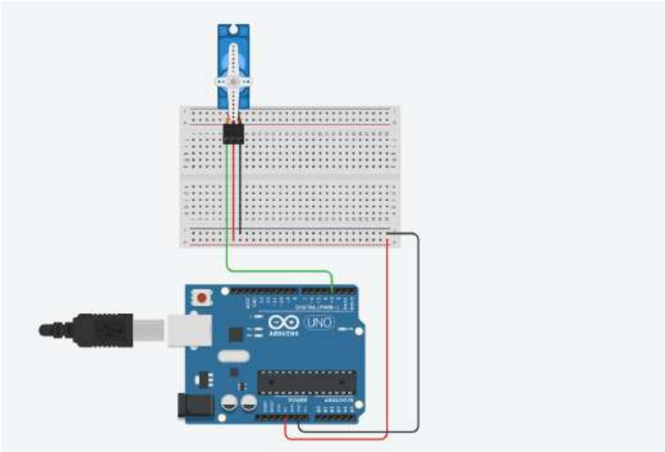


Week4

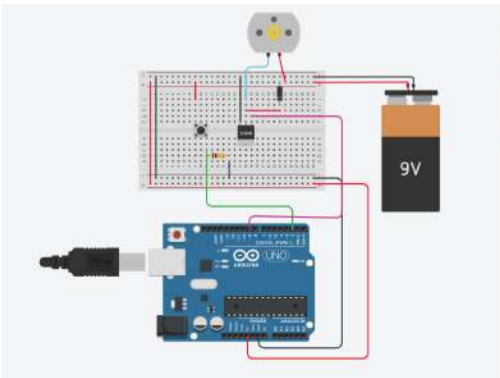
My code link:

https://github.com/msc-creative-computing/p-comp-week-1-labs-orionmel/tree/main/week4

Lab2: Hook up a servo

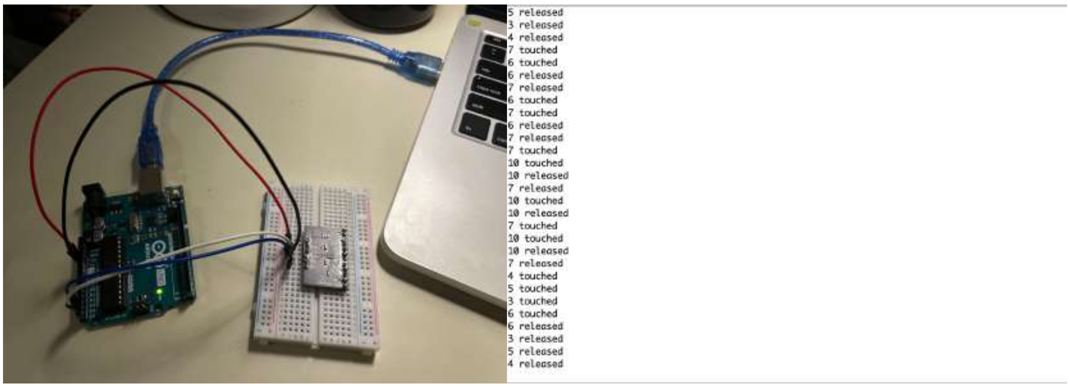


Lab3: Hook up a motor




```
const int switchPin =2;
const int motorPin =9;
int switchState =0;

void setup()
{
  pinMode(motorPin,OUTPUT);
  pinMode(switchState,INPUT);
}

void loop()
{
  switchState = digitalRead(switchPin);

  if(switchState == HIGH){
    digitalWrite(motorPin,HIGH);
  }
  else{
    digitalWrite(motorPin,LOW);
  }
}
```

Lab4: Hook up an MPR121



Lab5: The Capsense library

Video: https://vimeo.com/639425157