

Physical Computing

Designing Physical Interactions for a Digital World

ARTS 370

Fall 2019

Wednesday 1:40PM – 5:30PM

Klapper 107

Professor Danne Woo

pcomp.dannewoo.com

dwoo@qc.cuny.edu

Week 1-9

Week 1: What is Physical Computing?

Week 2: Introduction to Electronics

Week 3: Arduino, Hello World

Week 4: Analog In and Output

Week 5: Digital and Analog Review

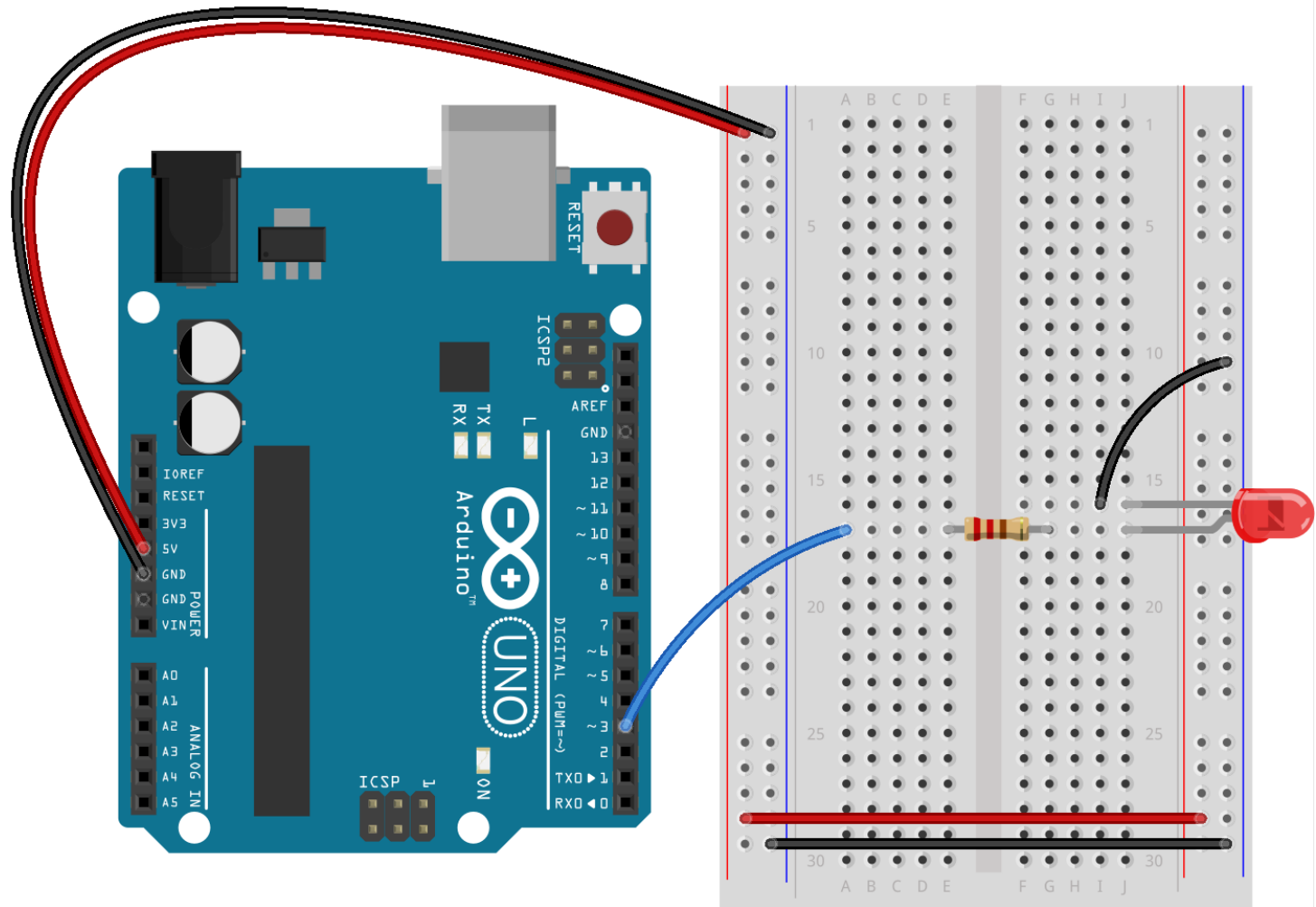
Week 6: Enclosures

Week 7: Serial Communication, Processing and p5.js

Week 8: Soldering Workshop

Week 9: Midterm Presentation

Digital Write

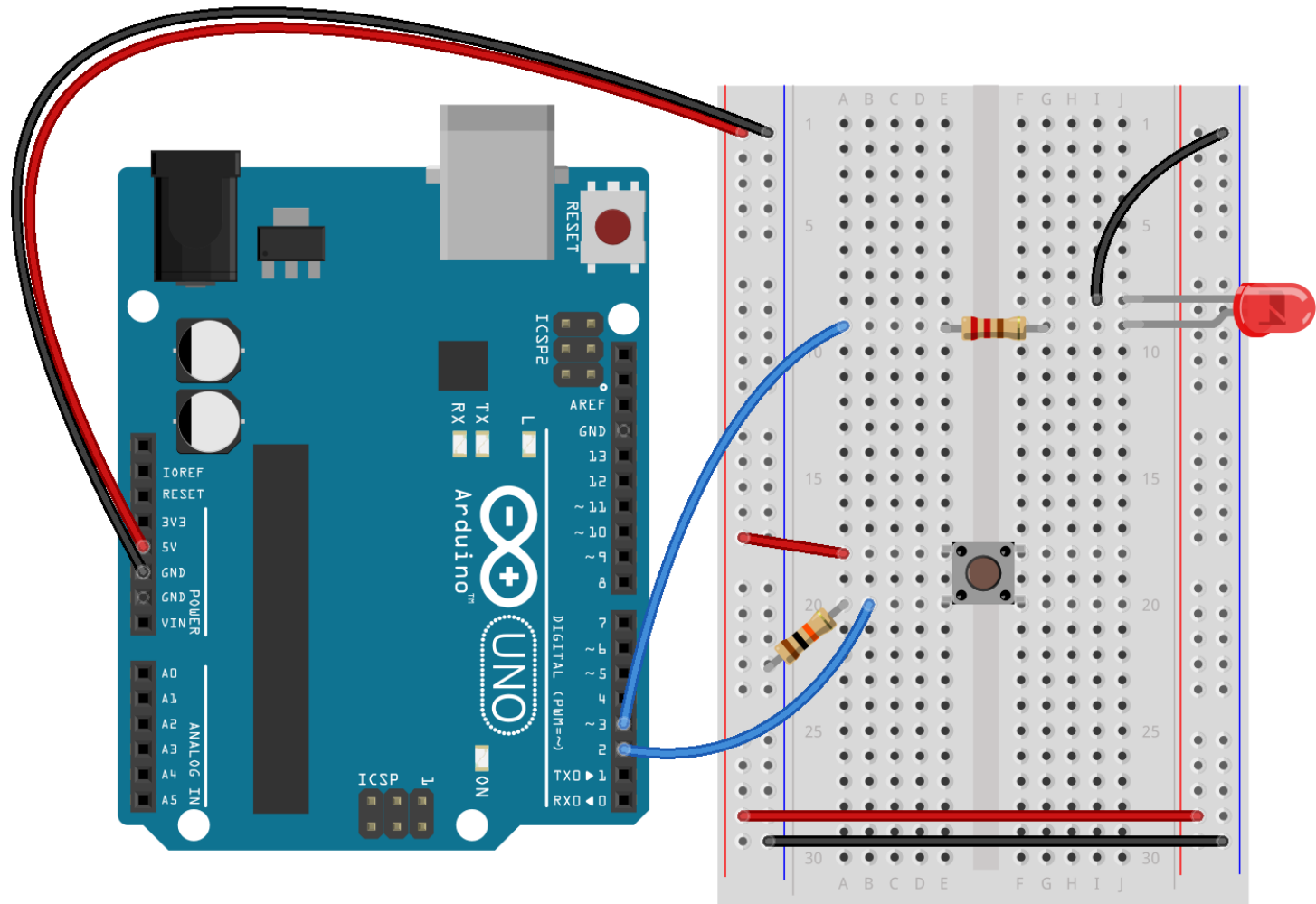


Digital Write

A screenshot of the Arduino IDE interface. The window title is "digi_out | Arduino 1.0.4". The toolbar shows icons for check, run, upload, and download. The file name is "digi_out". The code is as follows:

```
void setup() {  
  // Define which pin you are using and if it is input or output.  
  pinMode(3, OUTPUT);  
}  
  
void loop() {  
  // Send power to pin 3  
  digitalWrite(3, HIGH);  
  // Delay for 1000 milliseconds or 1 second  
  delay(1000);  
  // Turn off power to pin 3  
  digitalWrite(3, LOW);  
  // Delay for 200 milliseconds  
  delay(200);  
}
```


Digital In and Digital Out

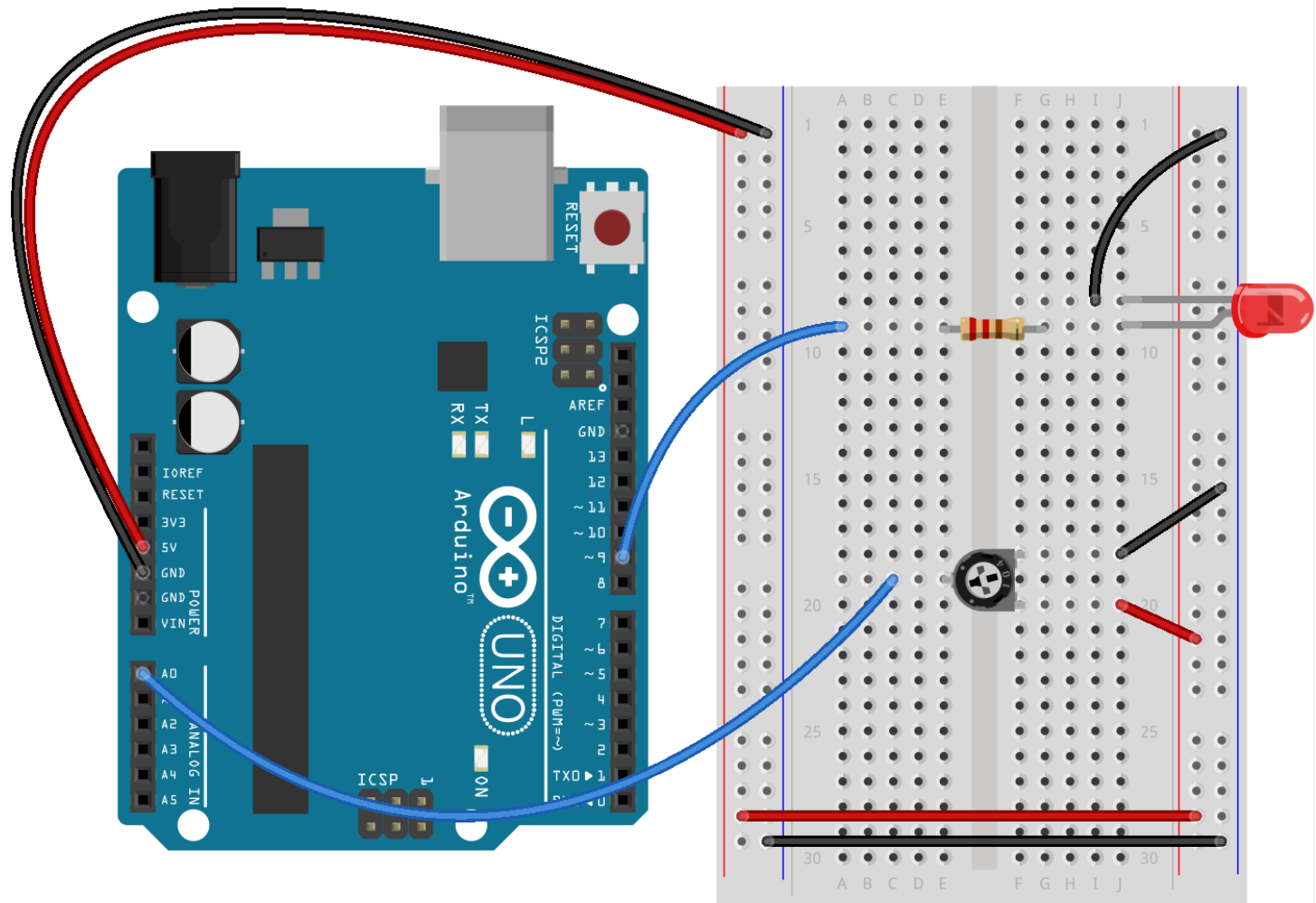


Digital In and Digital Out



```
digi_in_digi_out_1 | Arduino 1.0.4  
digi_in_digi_out_1  
void setup() {  
  pinMode(2, INPUT);  
  pinMode(3, OUTPUT);  
}  
  
void loop() {  
  if (digitalRead(2) == HIGH) {  
    digitalWrite(3, HIGH);  
  } else {  
    digitalWrite(3, LOW);  
  }  
}
```

Analog In and Out (Potentiometer)



Analog In and Out (Potentiometer)

The image shows a screenshot of the Arduino IDE interface. The window title is 'analog_in_pot | Arduino 1.0.4'. The main editor area contains the following C++ code:

```
int ledPin = 9;
int analogValue = 0;
int brightness = 0;

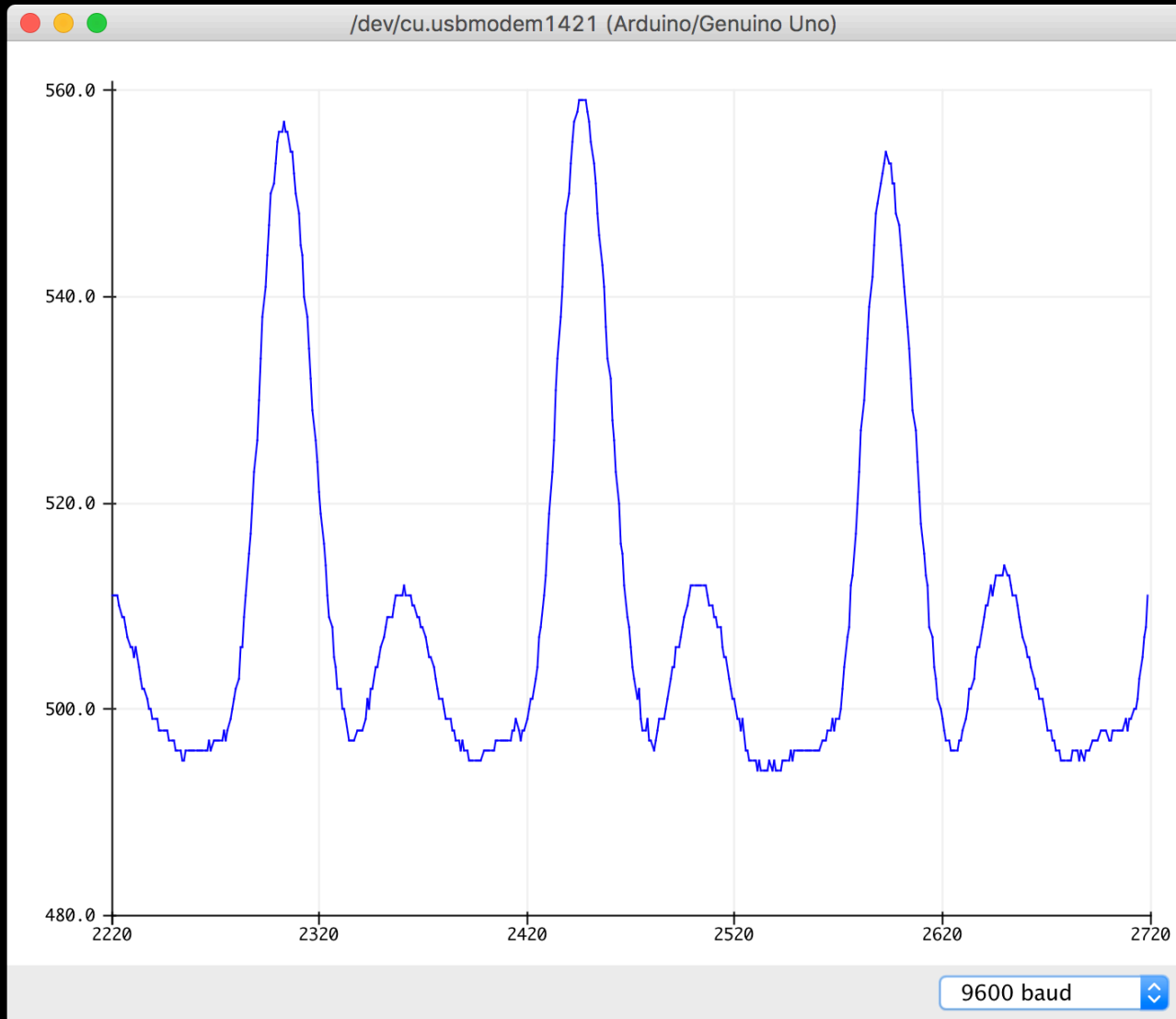
void setup() {
  Serial.begin(9600);
  pinMode(ledPin, OUTPUT);
}

void loop() {
  // Value between 0 and 1023
  analogValue = analogRead(A0);
  brightness = analogValue / 4;
  analogWrite(ledPin, brightness);
  Serial.println(analogValue);
}
```

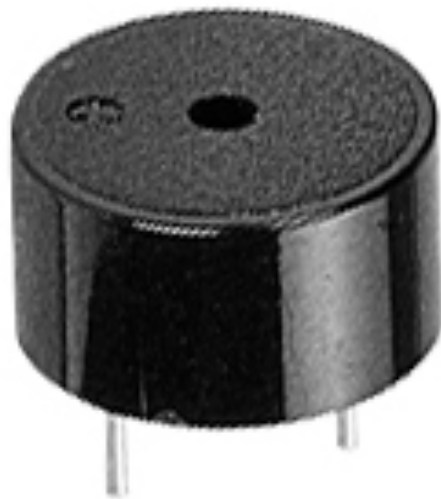
Pulse Sensor



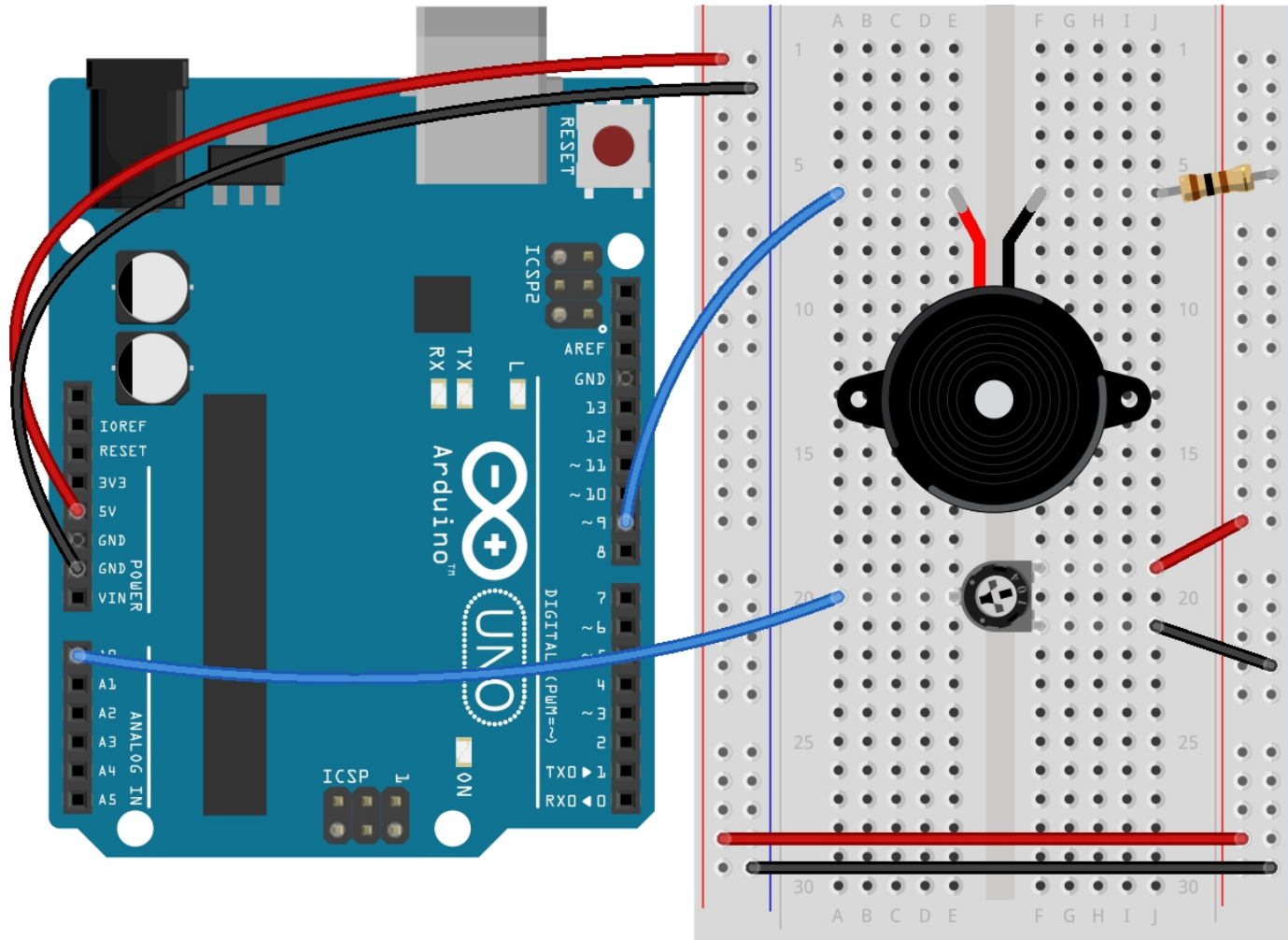
Serial Plotter



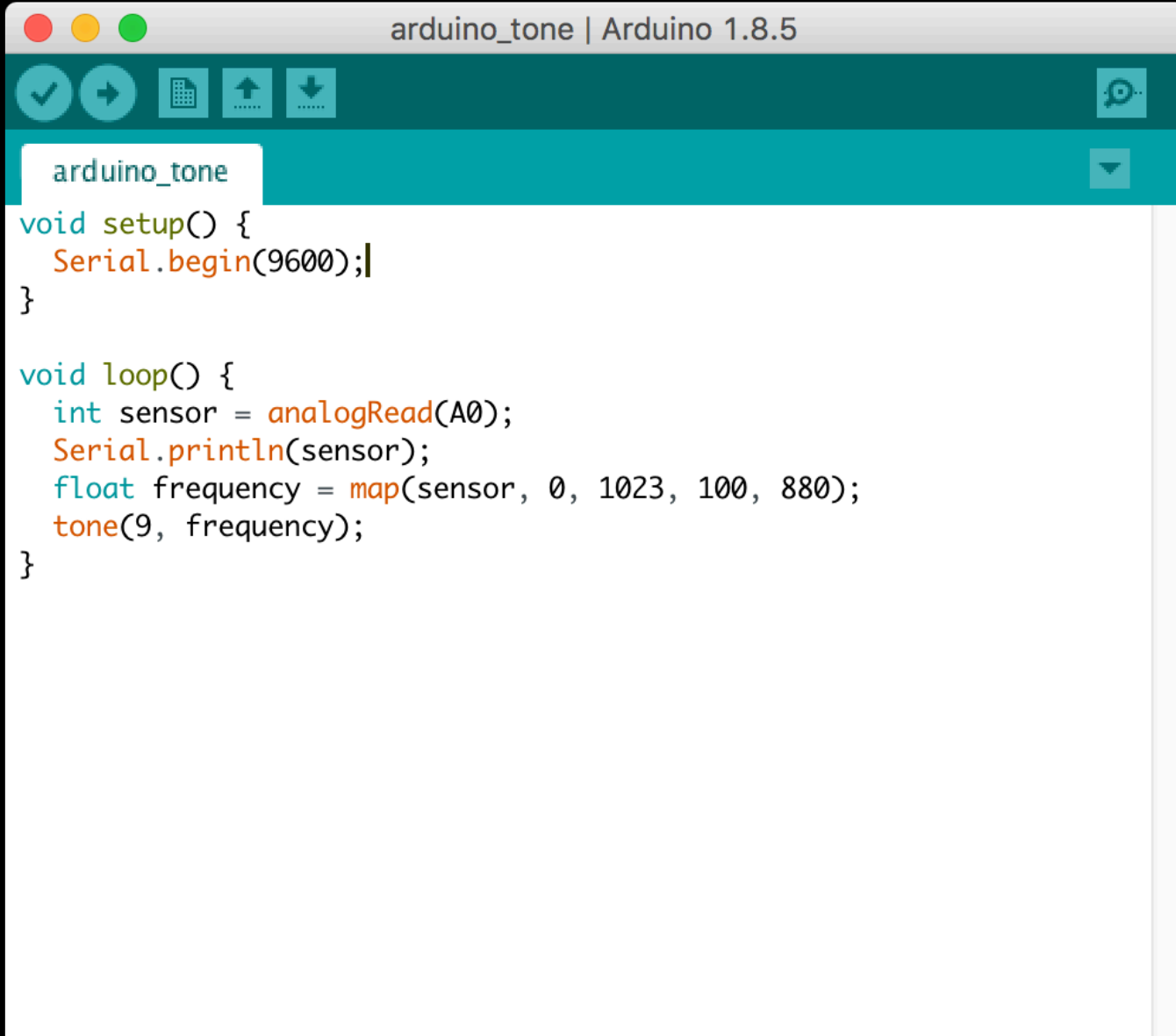
Tone



Tone

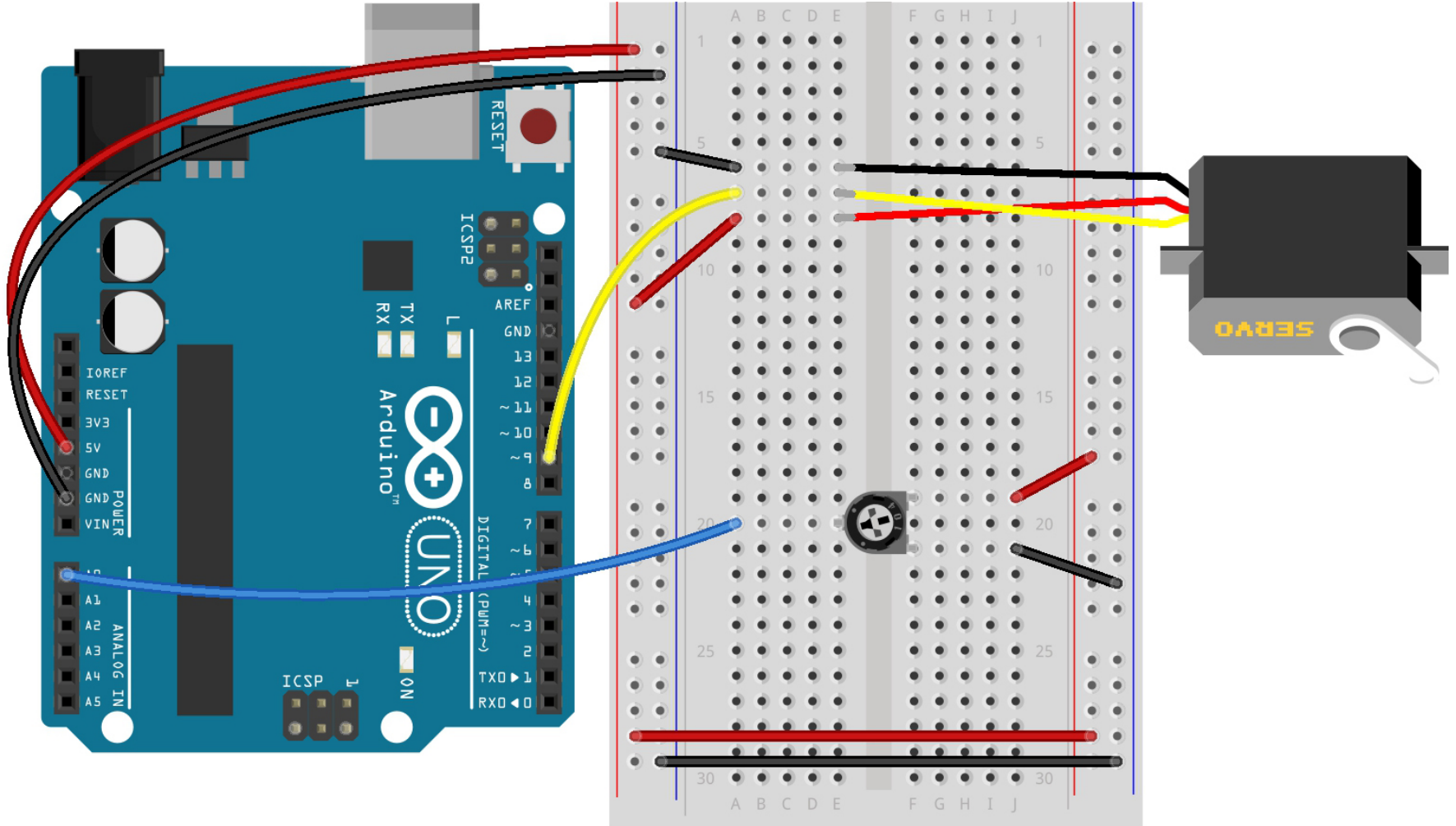


Tone

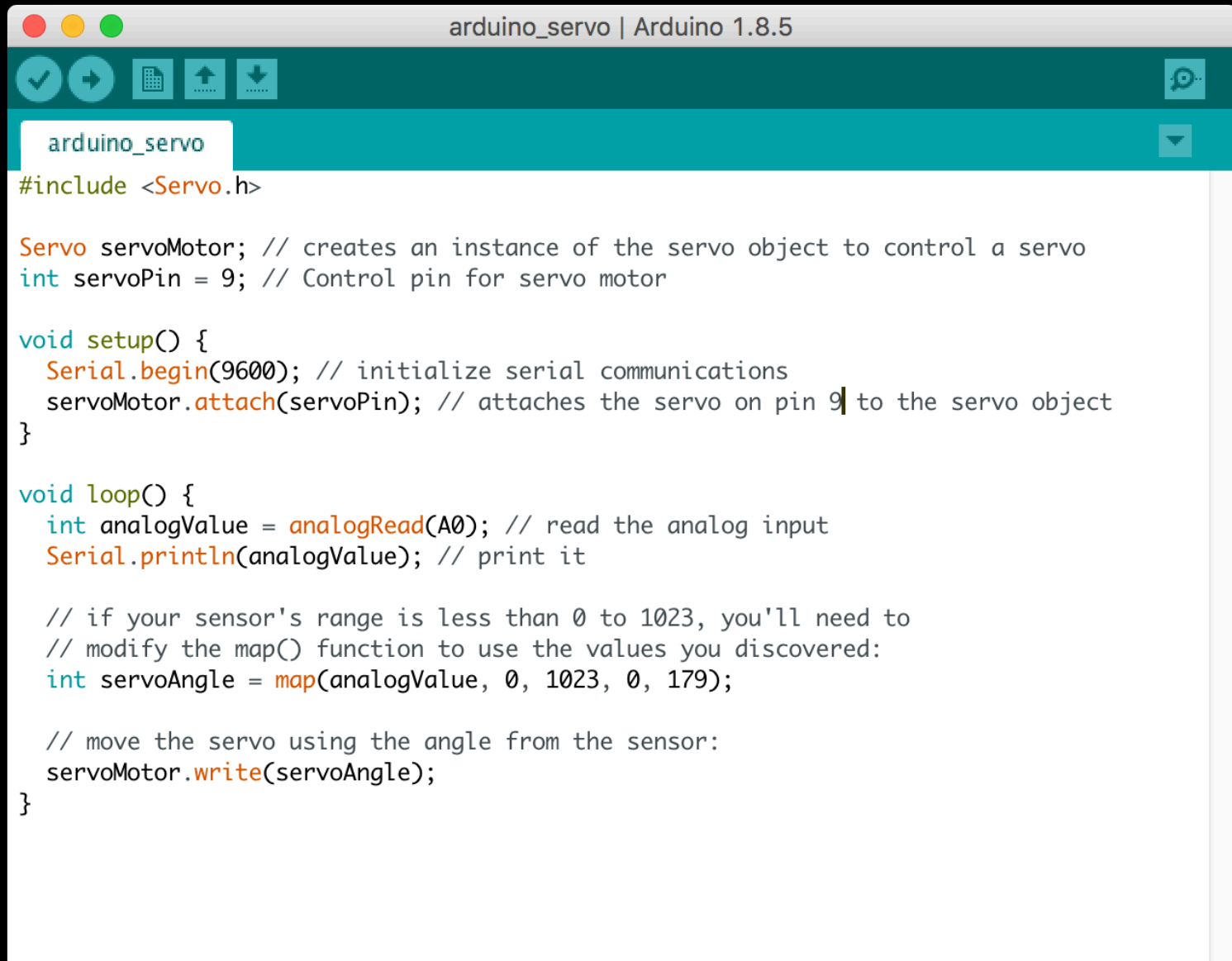


```
arduino_tone | Arduino 1.8.5  
arduino_tone  
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  int sensor = analogRead(A0);  
  Serial.println(sensor);  
  float frequency = map(sensor, 0, 1023, 100, 880);  
  tone(9, frequency);  
}
```

Servo Motor



Servo Motor



The image shows a screenshot of the Arduino IDE interface. The window title is "arduino_servo | Arduino 1.8.5". The code editor displays the following C++ code:

```
arduino_servo
#include <Servo.h>

Servo servoMotor; // creates an instance of the servo object to control a servo
int servoPin = 9; // Control pin for servo motor

void setup() {
  Serial.begin(9600); // initialize serial communications
  servoMotor.attach(servoPin); // attaches the servo on pin 9 to the servo object
}

void loop() {
  int analogValue = analogRead(A0); // read the analog input
  Serial.println(analogValue); // print it

  // if your sensor's range is less than 0 to 1023, you'll need to
  // modify the map() function to use the values you discovered:
  int servoAngle = map(analogValue, 0, 1023, 0, 179);

  // move the servo using the angle from the sensor:
  servoMotor.write(servoAngle);
}
```

DC Motor H-Bridge



L293D
L293DD

PUSH-PULL FOUR CHANNEL DRIVER WITH DIODES

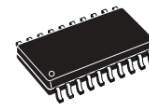
- 600mA OUTPUT CURRENT CAPABILITY PER CHANNEL
- 1.2A PEAK OUTPUT CURRENT (non repetitive) PER CHANNEL
- ENABLE FACILITY
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)
- INTERNAL CLAMP DIODES

DESCRIPTION

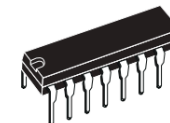
The Device is a monolithic integrated high voltage, high current four channel driver designed to accept standard DTL or TTL logic levels and drive inductive loads (such as relays solenoids, DC and stepping motors) and switching power transistors.

To simplify use as two bridges each pair of channels is equipped with an enable input. A separate supply input is provided for the logic, allowing operation at a lower voltage and internal clamp diodes are included.

This device is suitable for use in switching applications at frequencies up to 5 kHz.



SO(12+4+4)



Powerdip (12+2+2)

ORDERING NUMBERS:

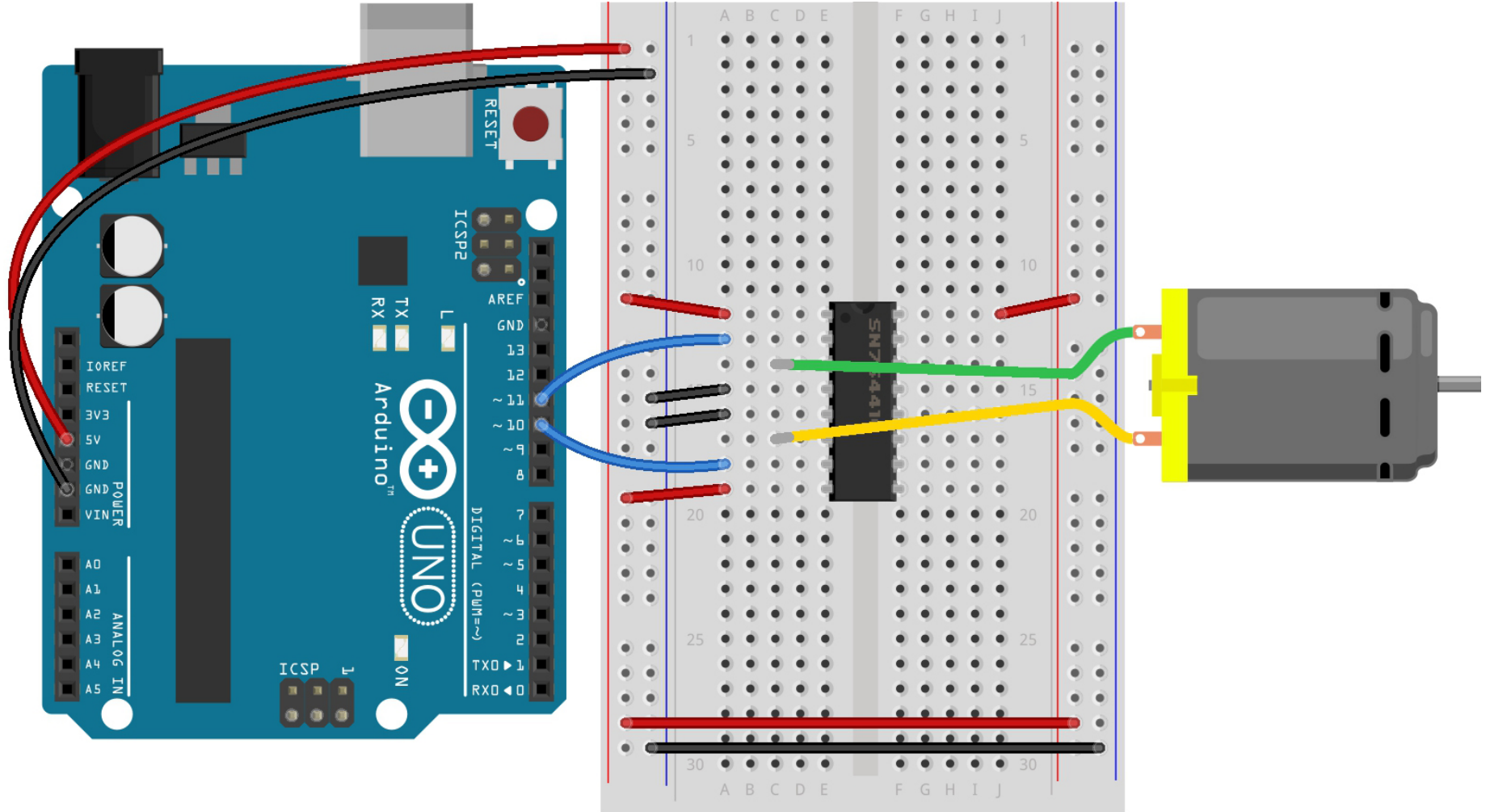
L293DD

L293D

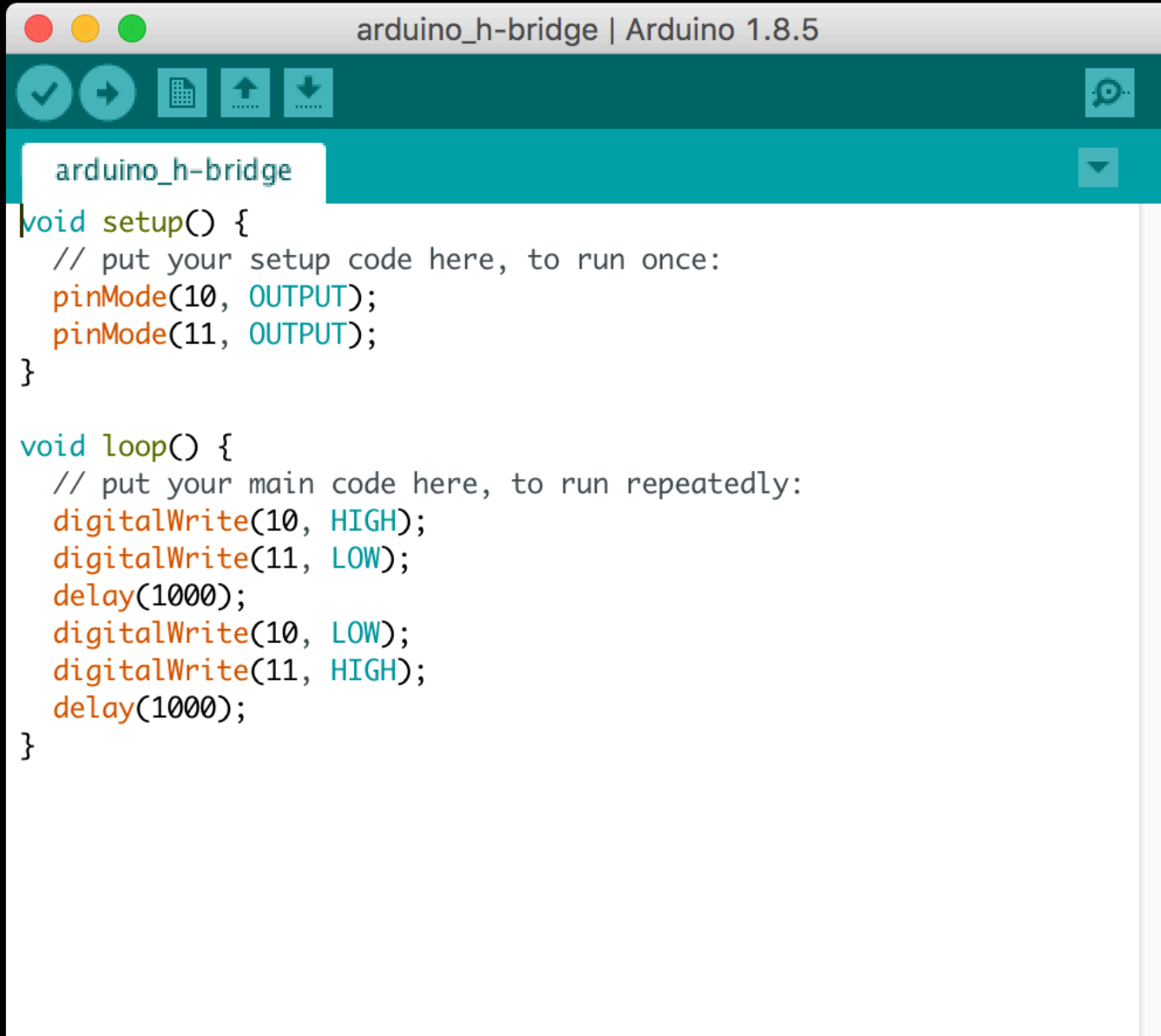
The L293D is assembled in a 16 lead plastic package which has 4 center pins connected together and used for heatsinking

The L293DD is assembled in a 20 lead surface mount which has 8 center pins connected together and used for heatsinking.

DC Motor H-Bridge



DC Motor H-Bridge

A screenshot of the Arduino IDE interface. The window title is "arduino_h-bridge | Arduino 1.8.5". The toolbar shows icons for check, run, upload, and download. The code editor displays the following C++ code:

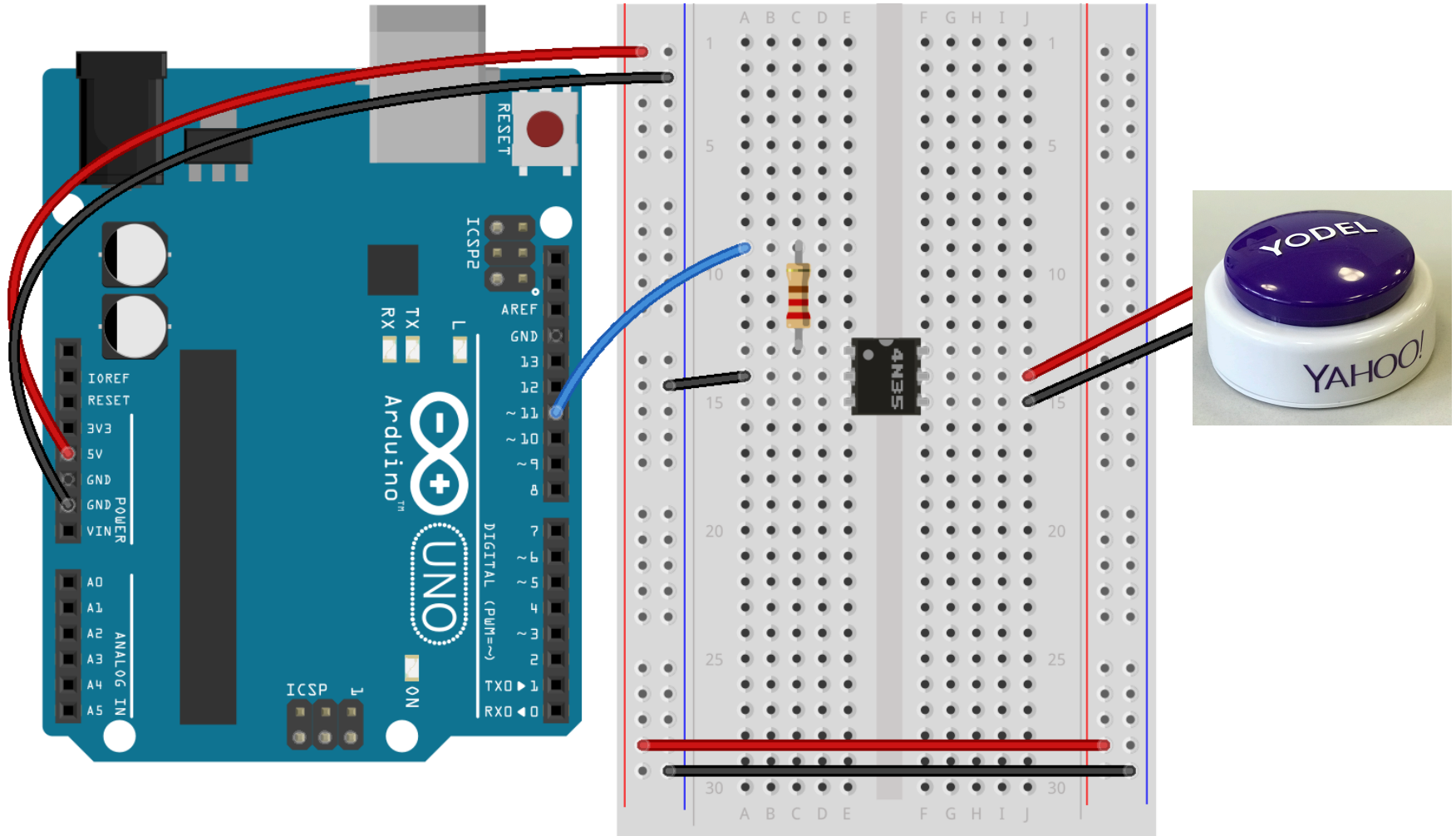
```
arduino_h-bridge
void setup() {
  // put your setup code here, to run once:
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(10, HIGH);
  digitalWrite(11, LOW);
  delay(1000);
  digitalWrite(10, LOW);
  digitalWrite(11, HIGH);
  delay(1000);
}
```


Optocoupler / Digital Switch



Optocoupler / Digital Switch



fritzing

Optocoupler / Digital Switch



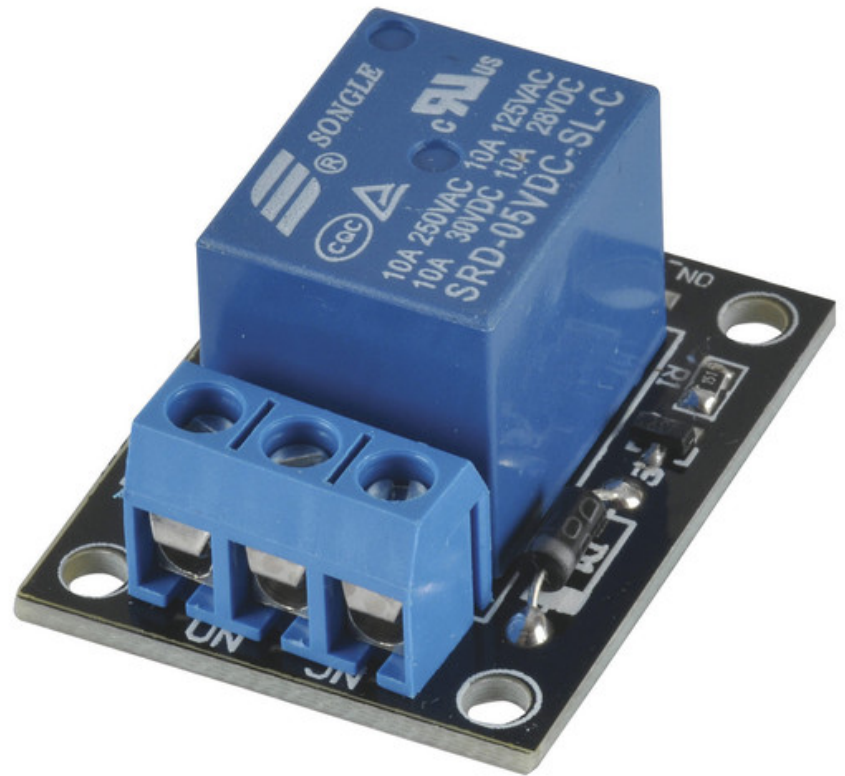
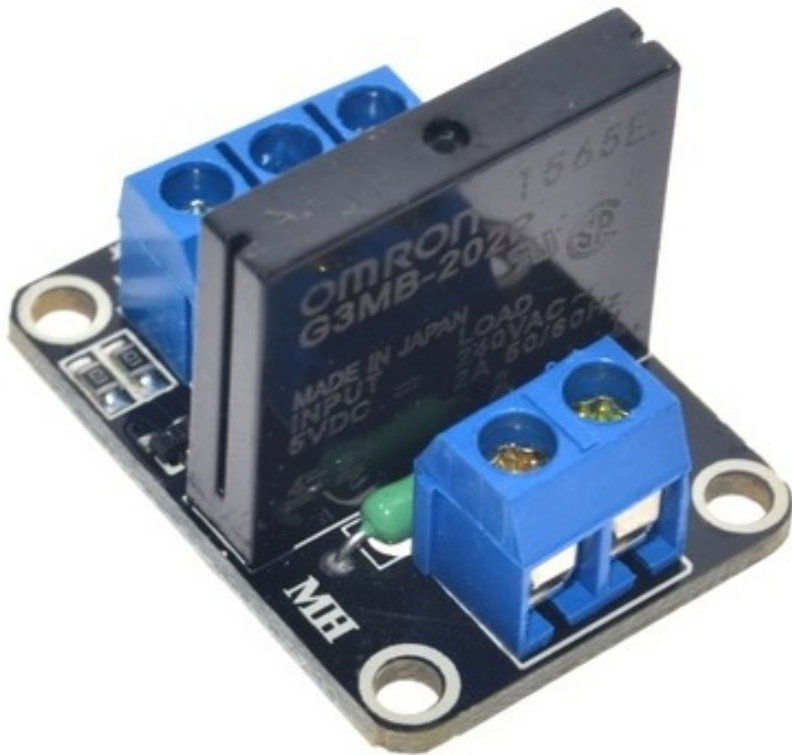
```
arduino_optocouplers | Arduino 1.8.5
arduino_optocouplers
// Lip is 1, 2, 3 side

int digiSwitch = 11;

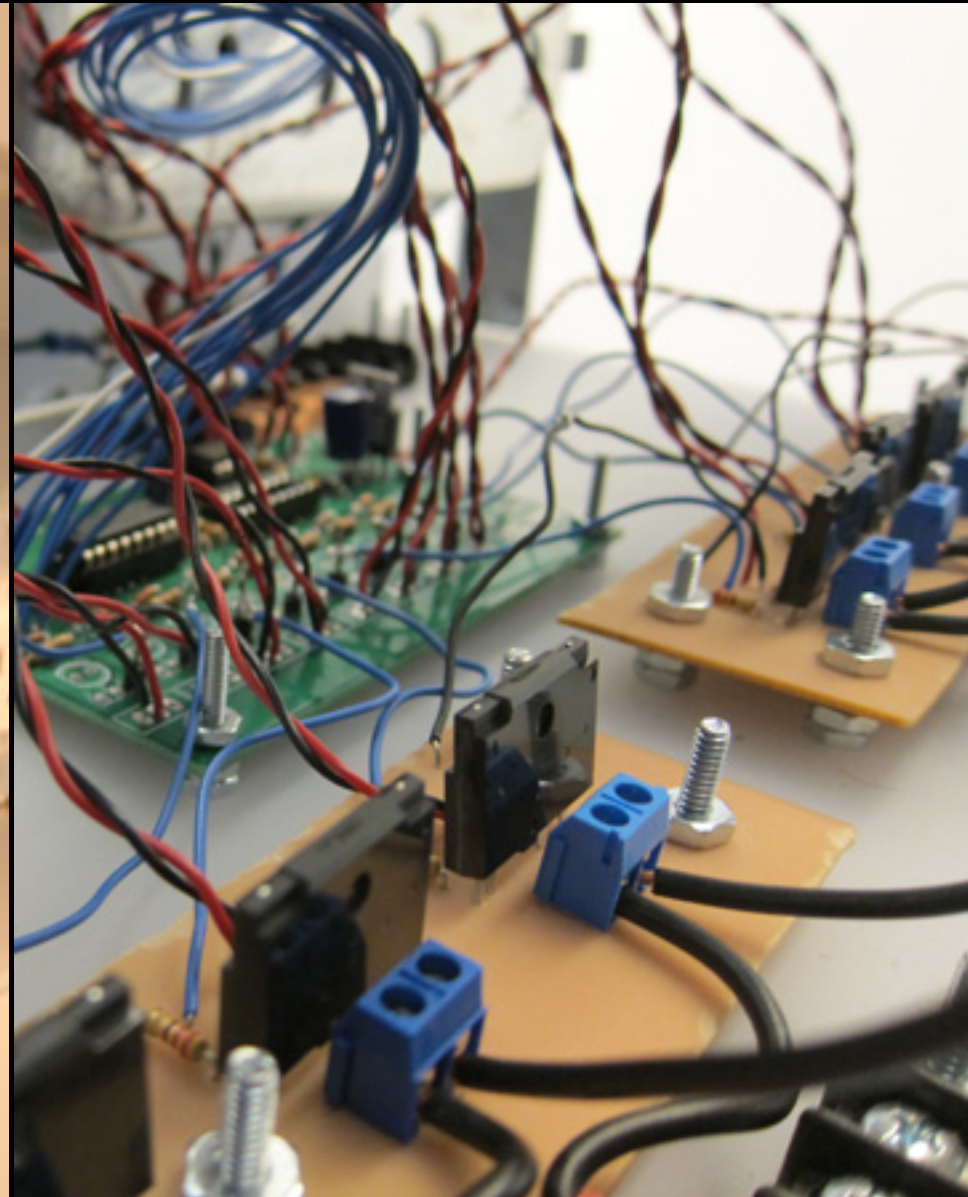
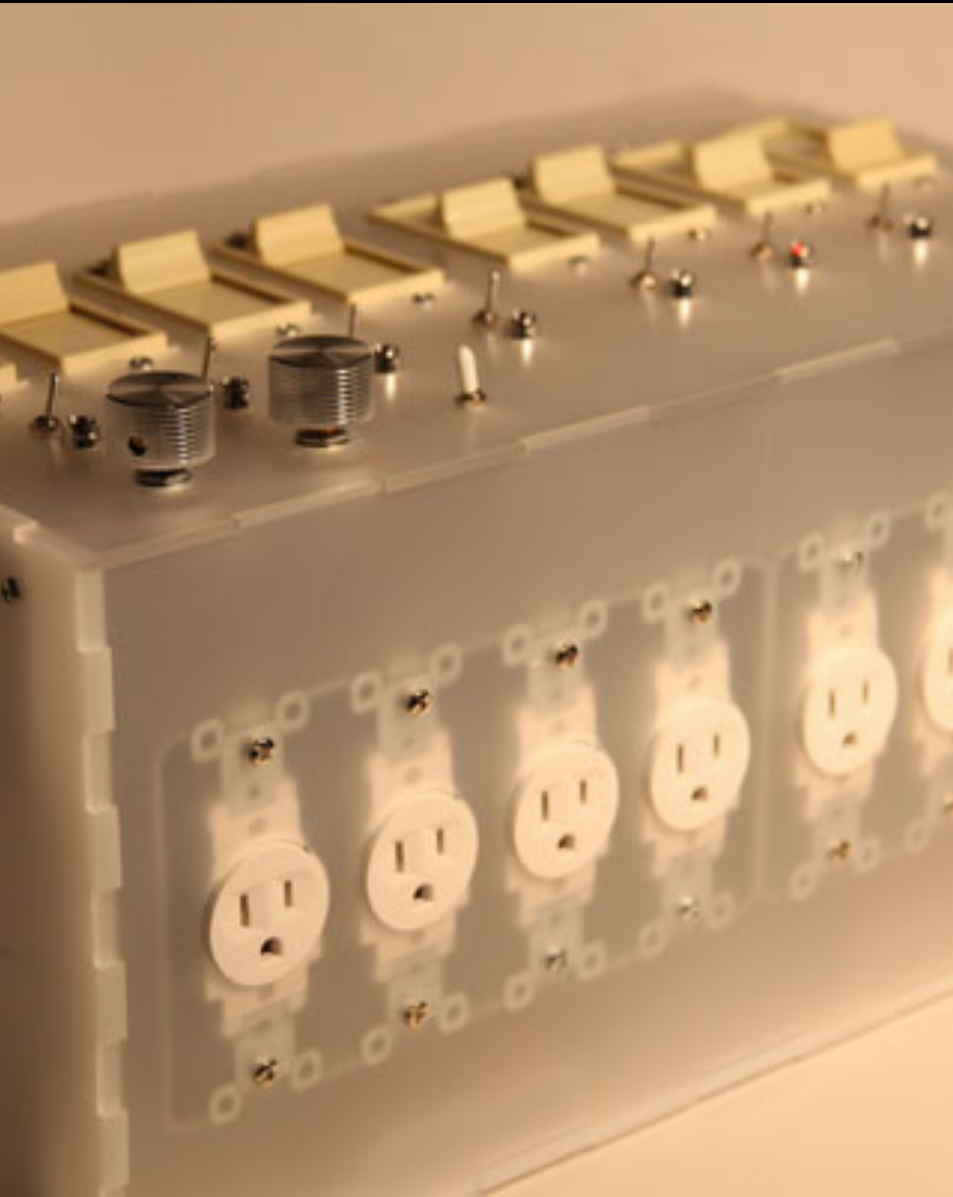
void setup() {
  // put your setup code here, to run once:
  pinMode(digiSwitch, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(digiSwitch, HIGH);
  delay(15);
  digitalWrite(digiSwitch, LOW);
  delay(200);
}
```

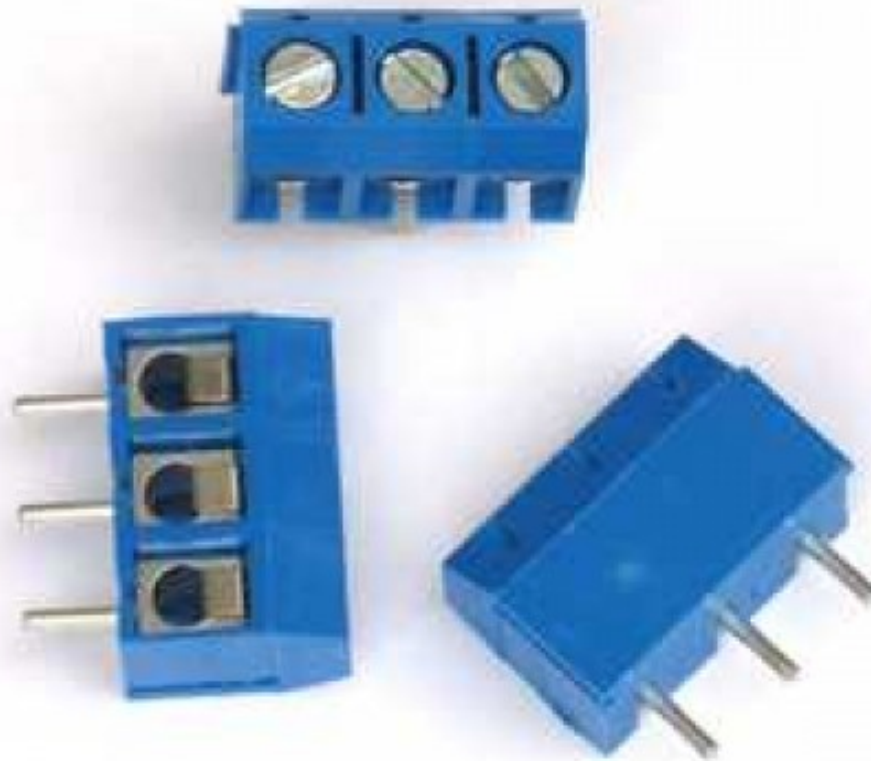
Relay



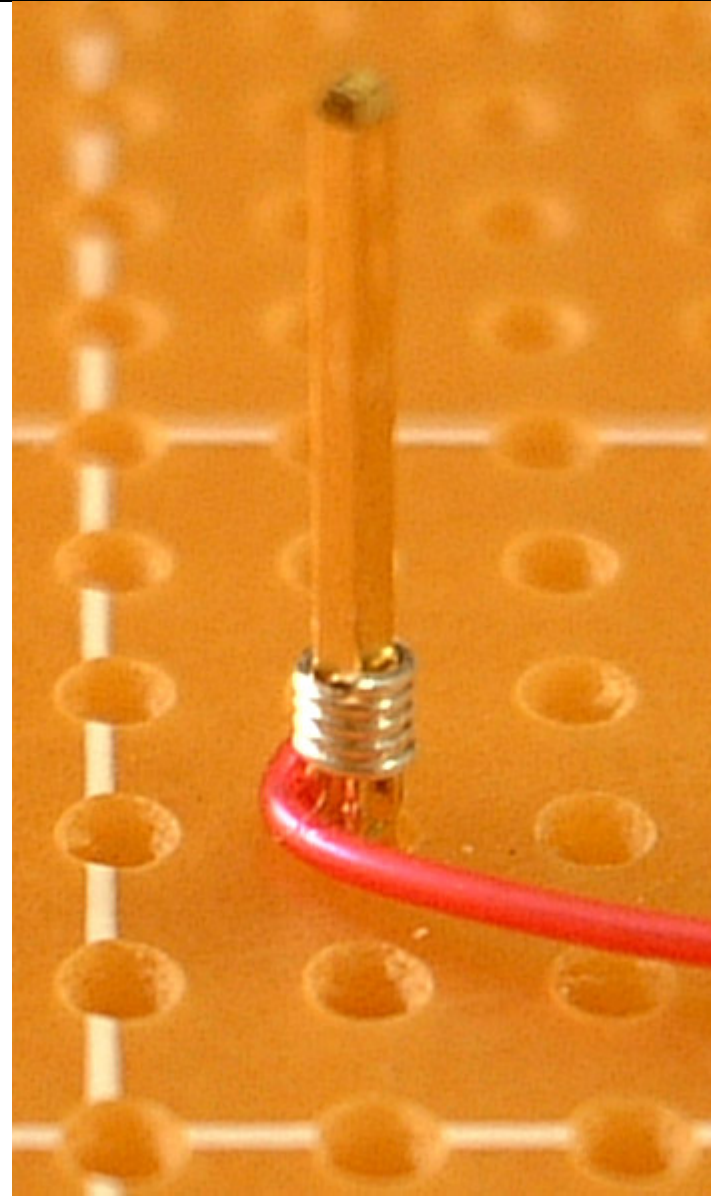
Relay



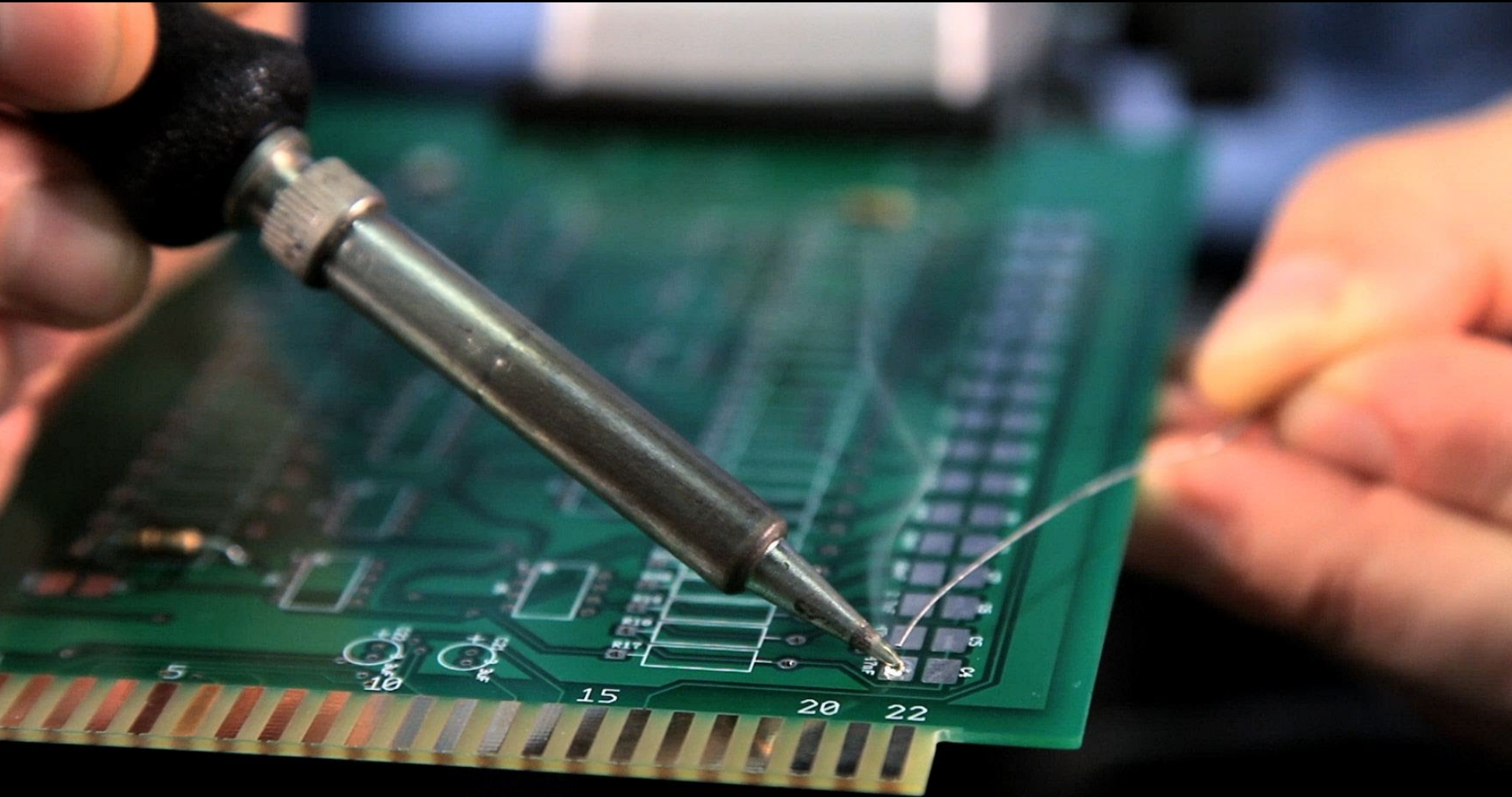
Screw Terminals



Wire Wrapping



Soldering



In Class/Homework

Start brain storming and sketching ideas for your midterm project. Think about what sensors and components you want to use as well as how you want people to interact with it.

Physical Computing

Professor Danne Woo

dwoo@qc.cuny.edu

pcomp.dannewoo.com