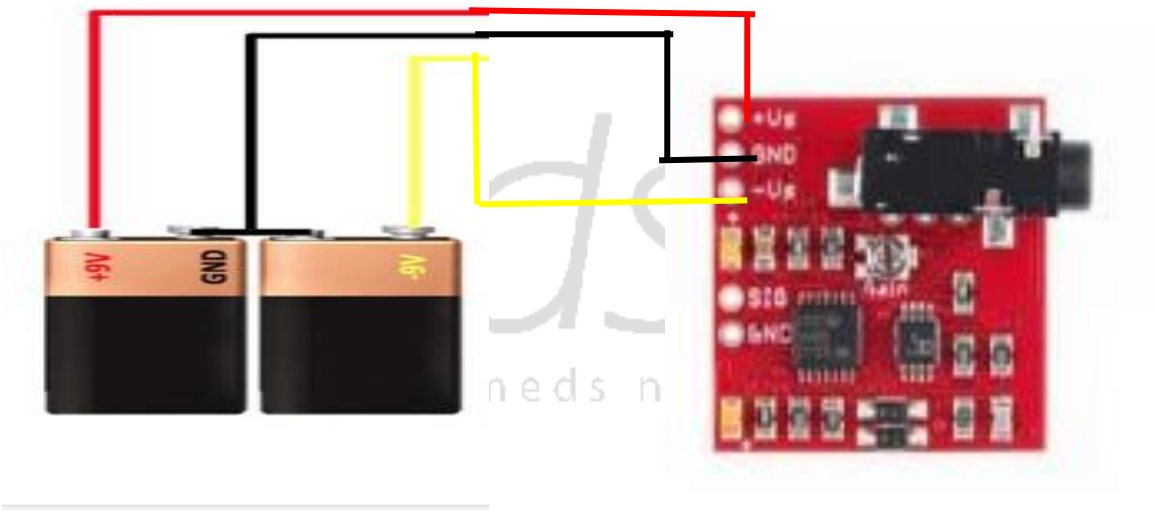
Bionic Arm

Medsby's bionic arm is a custom-made 3D-designed and 3D-printed prosthetic limb. It is controlled by motors for pronation and supination motions, which are in turn governed by the muscle movements of the user.

Components in bionic Arm

- EMG sensor
- Arduino UNO
- Servo Motor

Connection of Battery to the EMG sensor



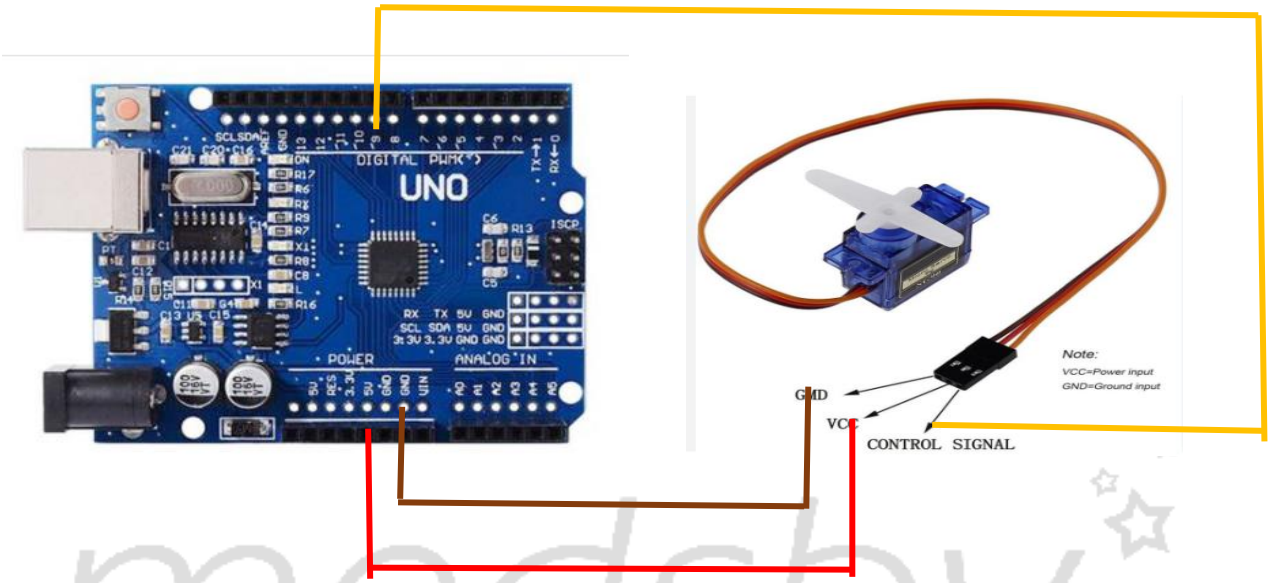
Step 1: Sort the positive and negative ends of the battery cap. Then, use this sorted battery cap to connect two 9V batteries together. Finally, connect the other end of the sorted battery cap to the ground pin of the EMG sensor.

Step 2: Now, connect the positive terminal of the battery to the +Vs of the EMG sensor using a separate battery cap. Cut or insulate the black end, ensuring only the positive end is in contact.

Step 3: Similarly, use another separate battery cap to connect the negative terminal of the battery to the -Vs of the EMG sensor. Cut or insulate the positive end, ensuring only the negative end is in contact.

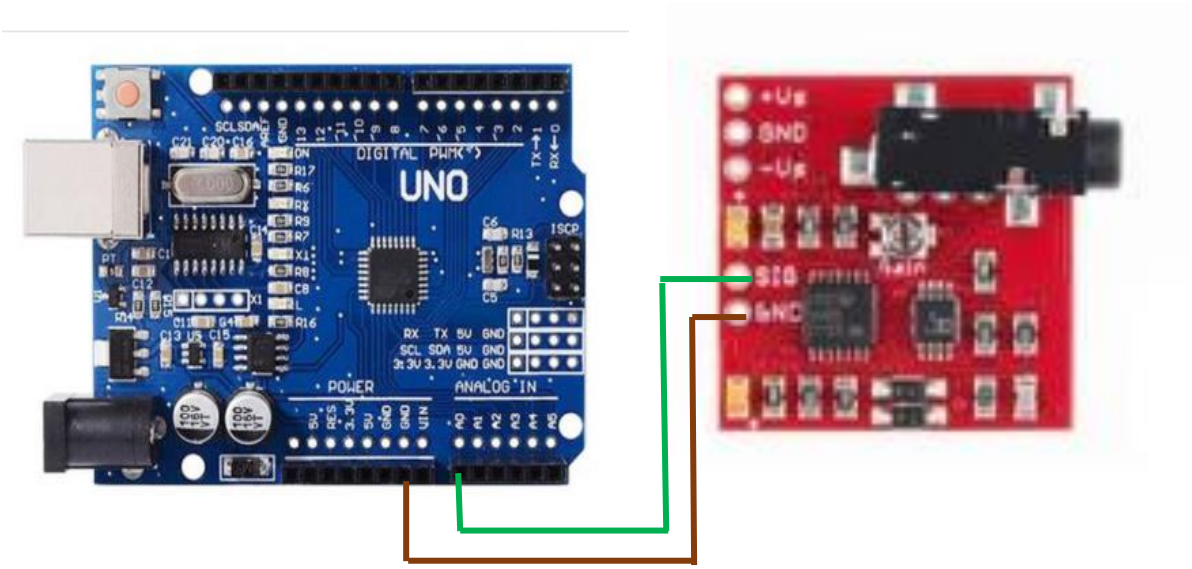
Step 4: The remaining two pins of the EMG sensor are the output and ground. Connect the output to the A0 pin of the Arduino Uno, and connect the ground to the ground pin of the Arduino Uno.

Connection of servo to the Arduino UNO



Servo Motor	Arduino UNO
Ground(Brown)	Ground
Vcc(Red)	5V
PWM(Orange)	9th Pin

Connection of Arduino to the EMG sensor

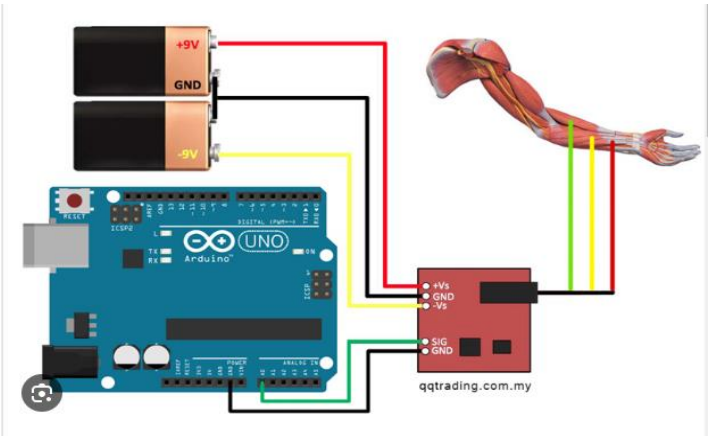


EMG Sensor	Arduino Uno
Ground	Ground
Signal	A0

Final Setup

The electrodes from the emg sensor should be positioned based on color. The reference electrode, designated as green, should be positioned in the elbow bone as a reference. The other two electrodes, designated as red and yellow, should be positioned in the upper arm at a distance to measure the muscle's electric potential in relation to the electromyograph.

.



Program for a single servo

```
#include <Servo.h>

const int analogInPin = A0;
Servo myservo;
int sensorValue = 0;
int pos = 0;

void setup() {
  myservo.attach(9);
  Serial.begin(9600);
}

void loop() {
  sensorValue = analogRead(analogInPin);
  Serial.println(sensorValue);


  if (sensorValue >= 20) {
    Serial.println("success");
    for (pos = 20; pos <= 160; pos += 1) {
      myservo.write(pos);
      delay(5);
    }
    for (pos = 160; pos >= 0; pos -= 1) {
      myservo.write(pos);
      delay(15);
    }
  }
}
```

Program for two servos

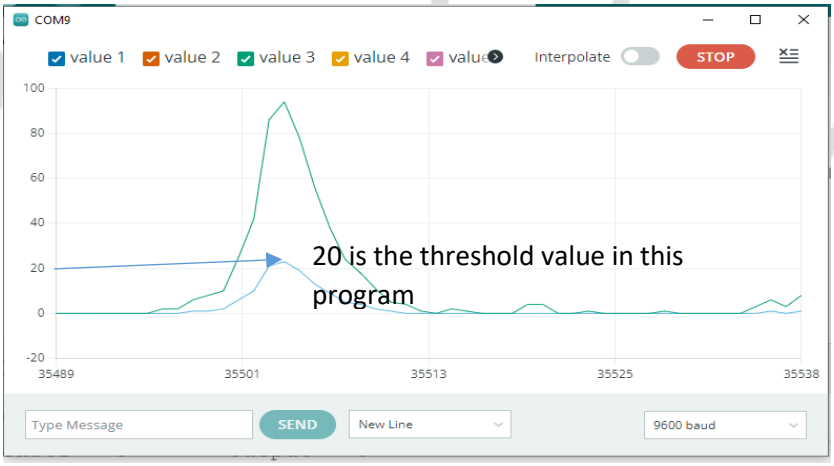
```
#include <Servo.h>

Servo servo1;
Servo servo2;

void setup() {

  servo1.attach(9);
  servo2.attach(10);
}

void loop() {

  for (int angle = 0; angle <= 180; angle++) {
    servo1.write(angle);
    delay(15);
  }


  for (int angle = 0; angle <= 180; angle++) {
    servo2.write(angle);
    delay(15);
  }


  for (int angle = 180; angle >= 0; angle--) {
    servo1.write(angle);
    delay(15);
  }


  for (int angle = 180; angle >= 0; angle--) {
    servo2.write(angle);
    delay(15);
  }
}
```

Program for Bionic Arm controlled by Emg Sensor

```
#include <Servo.h>

const int analogInPin = A0;
Servo myservo;
int sensorValue = 0;
int pos = 0;
bool rotateClockwise = true;

void setup() {
  myservo.attach(9);
  Serial.begin(9600);
}

void loop() {
  sensorValue = analogRead(analogInPin);
  Serial.println(sensorValue);

  if (sensorValue >= 20) {
    Serial.println("Threshold crossed");
    if (rotateClockwise) {
      for (pos = 160; pos >= 0; pos -= 1) {
        myservo.write(pos);
        delay(5);
      }
    } else {
      for (pos = 0; pos <= 160; pos += 1) {
        myservo.write(pos);
        delay(5);
      }
    }
    rotateClockwise = !rotateClockwise;
    delay(1000);
  }
}
```

Output

EMG wave at rest



Peak of the EMG wave

This graph represents the EMG wave at rest. When stress is applied, the wave will peak. If it surpasses the threshold level, it triggers the servo to rotate, initiating the pronation and supination movement.