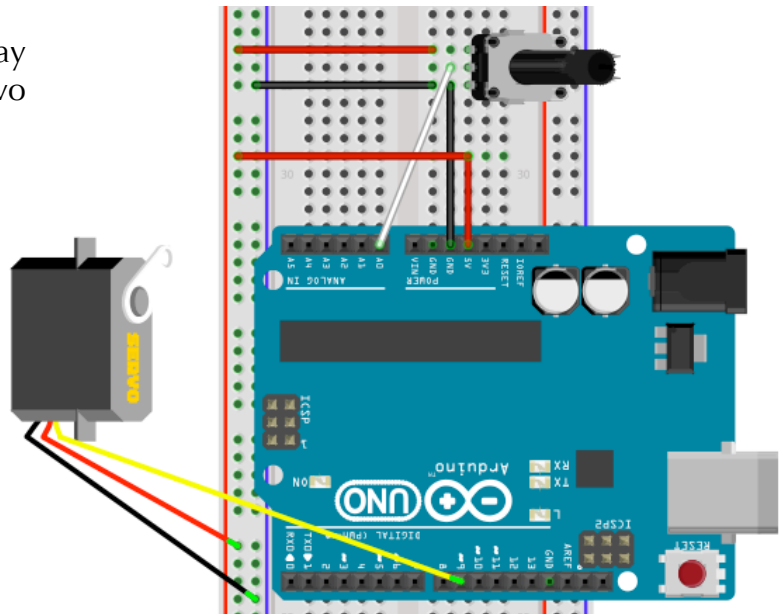


Activity #12: Working with a Servo Motor and Range Sensor

CHALLENGE: Figure out a way to control a (continuous) servo motor using a potentiometer

SETUP:



CODE:

```
#include <Servo.h> // Use Servo library, included with IDE

Servo myServo; // Create Servo object to control the servo
int potPin = 0; // Connect potentiometer to Analog Pin 0
int servoPin = 9; // Connect Servo motor to Digital Pin 9

int potVal = -1;
int servoSpeed = 0;

int potLow = 90; //Lowest reading from potentiometer.
int potHigh = 1023; //Highest reading from potentiometer.
int servoLow = 1300; //Setting for full clockwise.
int servoHigh = 1700; //Setting for full counterclockwise.

void setup() {
  myServo.attach(servoPin); // Servo is connected to digital pin 9
  Serial.begin(9600);
}

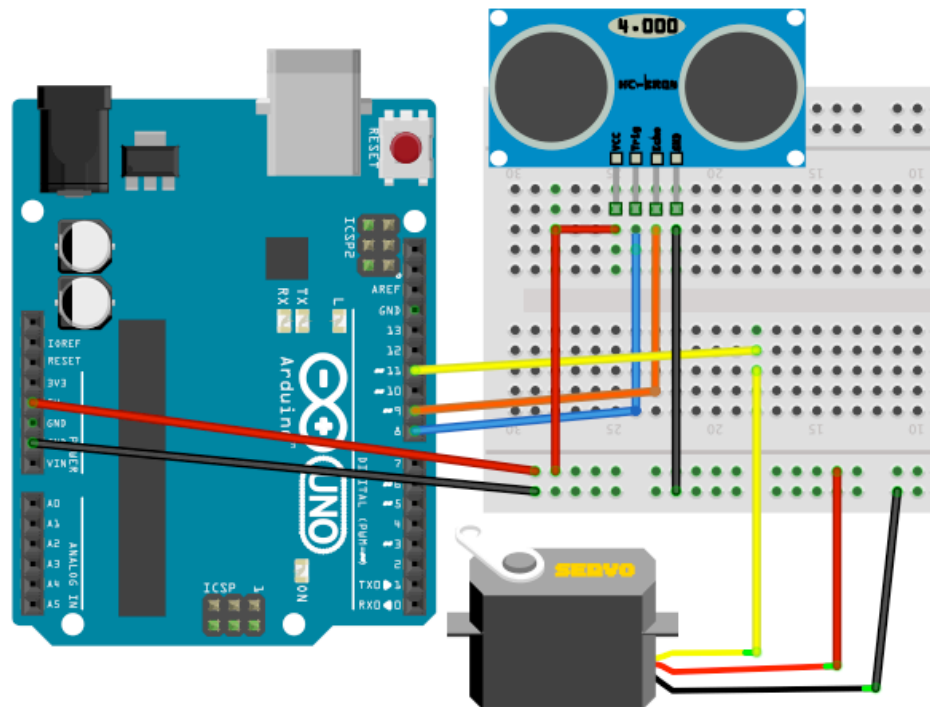
void loop() {
  potVal=analogRead(0);
  Serial.println(potVal);
  delay(500);

  servoSpeed = map(potVal, potLow, potHigh, servoLow, servoHigh);
  Serial.println(servoSpeed);

  myServo.writeMicroseconds(servoSpeed);
}
```

CHALLENGE: Figure out a way to control a (continuous) servo motor using a ultrasonic range finder

SETUP:



CODE:

```
/* Using the HC-SR04 (Ultrasonic Range Sensor) for Arduino to drive a Continuous Servo Motor
(original project from http://www.swanrobotics.com)
*/
```

```
#include <Servo.h> // Use Servo library, included with IDE
```

```
Servo myServo; // Create Servo object to control the servo
int servoSpeed;
```

```
const int TriggerPin = 8; //Trig pin
const int EchoPin = 9; //Echo pin
const int ServoPin = 11; //Servo pin
long targetDistance = 0;
```

```
void setup() {
  pinMode(TriggerPin,OUTPUT); // Trigger is an output pin
  pinMode(EchoPin,INPUT); // Echo is an input pin

  myServo.attach(ServoPin);

  Serial.begin(9600); // Serial Output
}
```

```
void loop() {
```



```
targetDistance = getDistance();

if (targetDistance > 100) {
    myServo.writeMicroseconds(1700); //Spin Counterclockwise
}
else if (targetDistance < 50) {
    myServo.writeMicroseconds(1300); //Spin Clockwise
}
else {
    myServo.writeMicroseconds(1500); //Don't Spin
}

delay(1000); // Wait to do next measurement
}

long getDistance() {
    digitalWrite(TripPin, LOW);
    delayMicroseconds(2);
    digitalWrite(TripPin, HIGH); // Trip pin to HIGH
    delayMicroseconds(10); // 10us high
    digitalWrite(TripPin, LOW); // Trip pin to HIGH

    long Duration = pulseIn(EchoPin,HIGH); // Waits for the echo pin to get high
    // returns the Duration in microseconds
    long Distance_mm = calcDistance(Duration); // Use function to calculate the distance

    Serial.print("Distance = "); // Output to serial
    Serial.print(Distance_mm);
    Serial.println(" mm");

    return Distance_mm;
}

long calcDistance(long time) {
    // Calculates the Distance in mm
    // ((time)*(Speed of sound))/ toward and backward of object * 10

    long DistanceCalc; // Calculation variable
    DistanceCalc = ((time / 2.9) / 2); // Actual calculation in mm
    //DistanceCalc = time / 74 / 2; // Actual calculation in inches
    return DistanceCalc; // return calculated value
}
```