ROCO222: Intro to sensors and actuators

Lecture 4

Arduino RC servo control

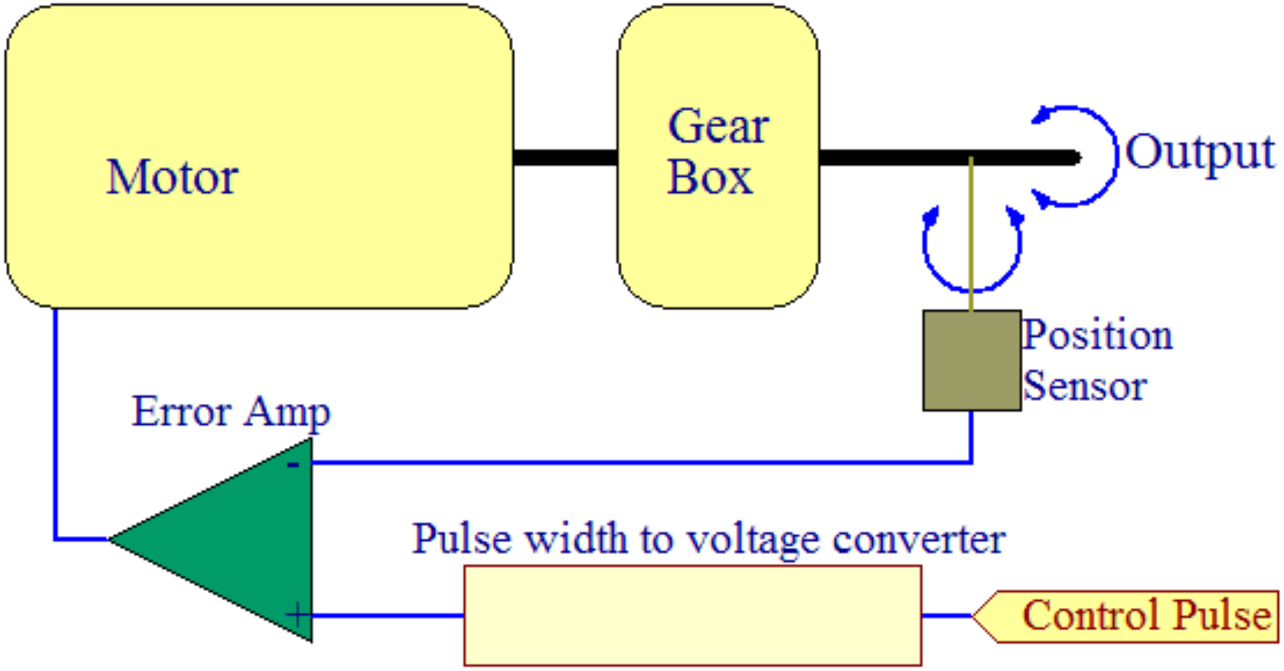
Arduino RC servos

- Small electric motor driving a gear train
- Potentiometer position measurement
- Small
- Cheap

- Servo Motors are electronic devices that convert digital signal to rotational movement.
- Standard servos that their rotation is limited to maximum of 180 degrees in each direction
- Continuous Rotation Servos that can provide rotation unlimitedly in both directions

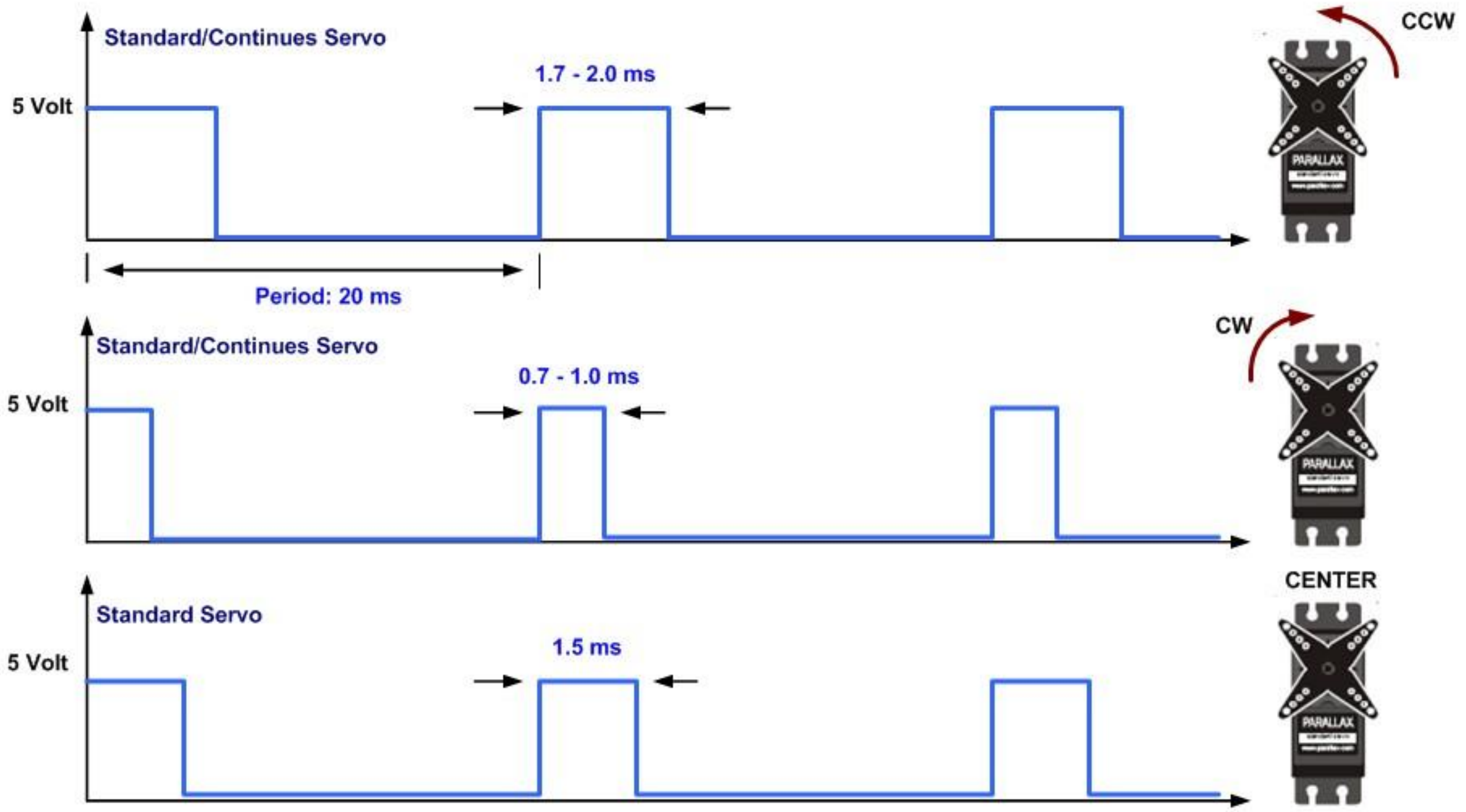
RC servo operation

- Output position compared to the commanded position
- Gives rise to error signal in appropriate direction
- Error drives the electric motor
- When becomes zero servo stops moving



Pulse width control of RC servo

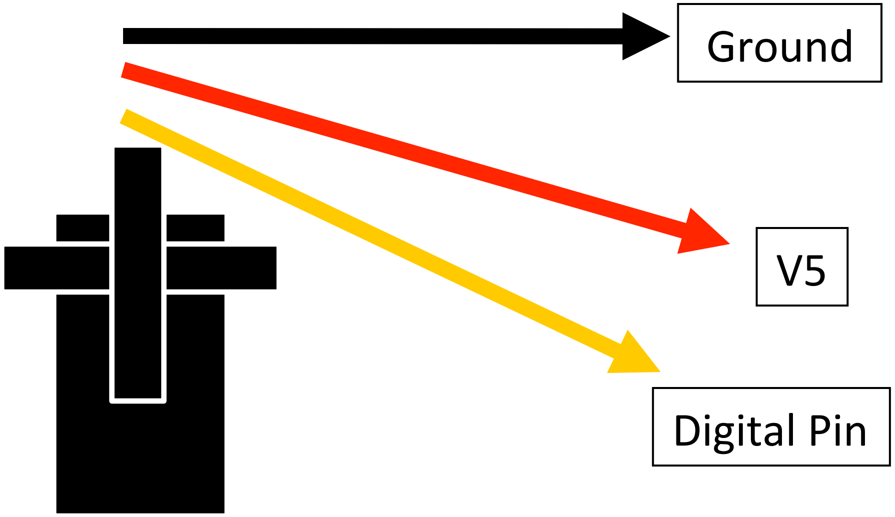
- Commanded signal encoded using PWM
- The pulse duration determines the position of the shaft

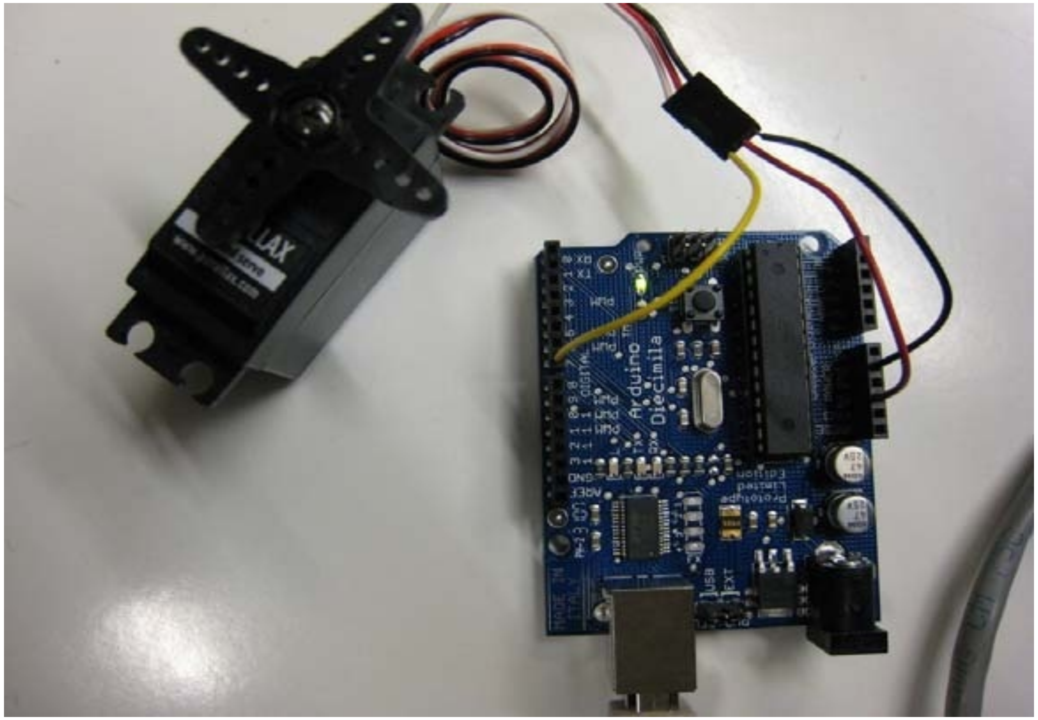


Connecting Arduino to RC servo

Here:

- Black is ground
- Red is connected to 5V
- Yellow wire (sometimes white) is set to the digital pin





Arduino RC servo circuit

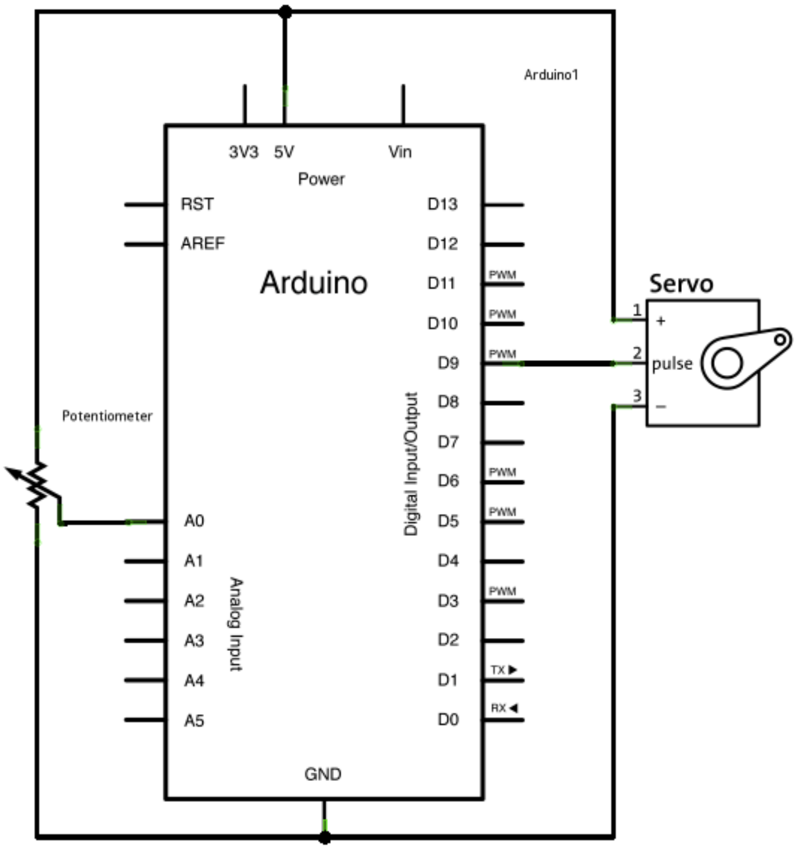
- Servo motors have three wires: power, ground, and signal.

- The power wire is typically red, and should be connected to the 5V pin on the Arduino board.

- The ground wire is typically black or brown and should be connected to a ground pin on the Arduino board.

- The signal pin is typically yellow or orange and should be connected to pin 9 on the Arduino board.

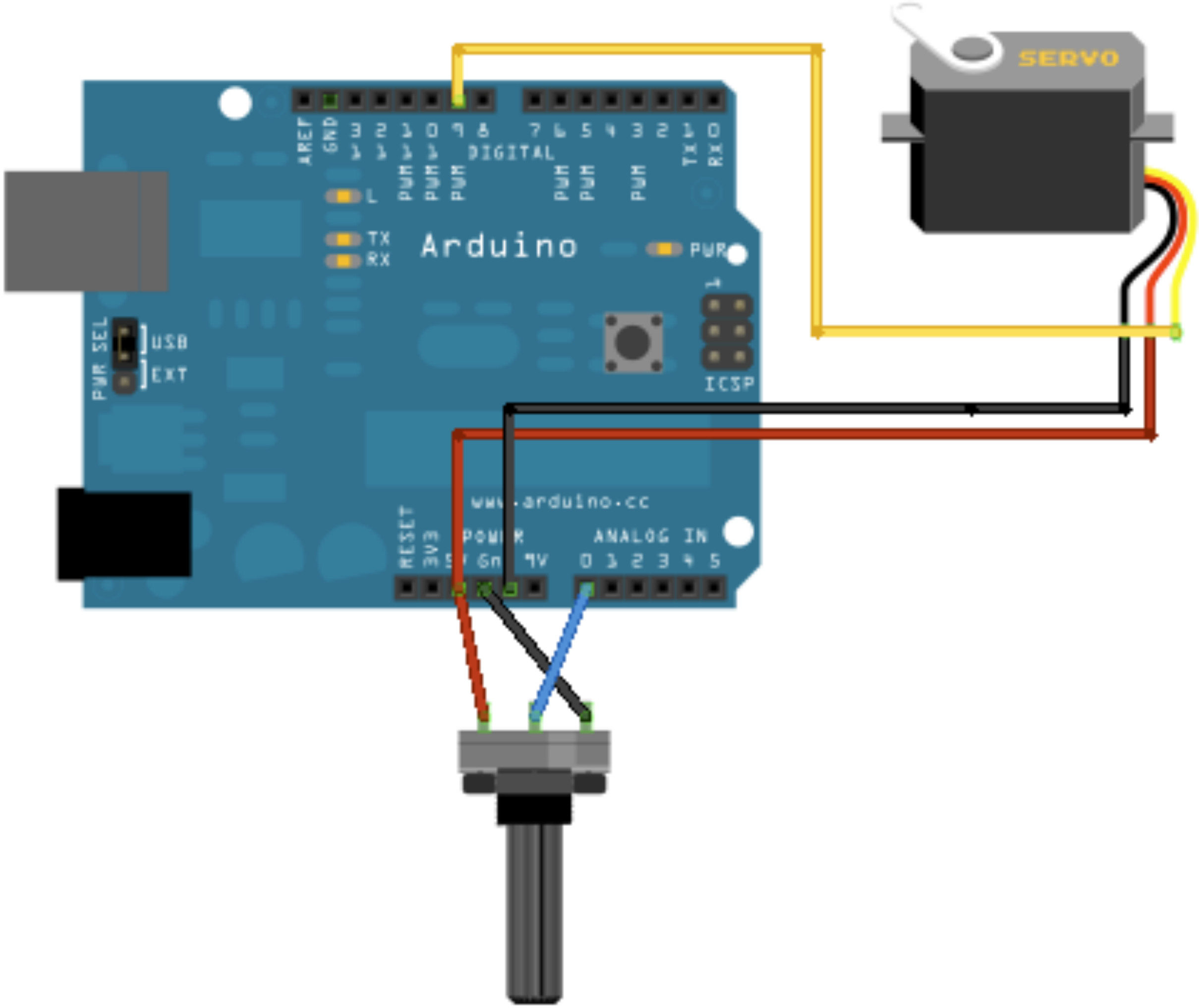
- The potentiometer should be wired so that its two outer pins are connected to power (+5V) and ground, and its middle pin is connected to analog input 0 on the Arduino.



http://arduino.cc/en/Tutorial/Knob

Servo follow-potentiometer example

- Arduino Board
- (1) Servo Motor
- (1) Potentiometer
- hook-up wire



http://arduino.cc/en/Tutorial/Knob

Servo follow-potentiometer example

```
// Controlling a servo position using a potentiometer (variable resistor)
// by Michal Rinott http://people.interaction-ivrea.it/m.rinott
#include <Servo.h>

Servo myservo;  // create servo object to control a servo
int potpin = 0;  // analog pin used to connect the potentiometer
int val;    // variable to read the value from the analog pin

void setup()
{
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}

void loop()
{
  val = analogRead(potpin);            // reads the value of the potentiometer (value between 0 and
1023)
  val = map(val, 0, 1023, 0, 179);     // scale it to use it with the servo (value between 0 and 180)
  myservo.write(val);                  // sets the servo position according to the scaled value
  delay(15);                           // waits for the servo to get there
}
```

http://arduino.cc/en/Tutorial/Knob

Servo ramp example

```
// Sweep by BARRAGAN http://barraganstudio.com  This example code is in the public domain.
#include <Servo.h>
Servo myservo;                          // create servo object to control a
int pos = 0;                            // variable to store the servo position
void setup()
{
  myservo.attach(9);                    // attaches the servo on pin 9 to the servo object
}

void loop()
{
  for(pos = 0; pos < 180; pos += 1)     // goes from 0 degrees to 180 degrees
  {                                     // in steps of 1 degree
    myservo.write(pos);                           // tell servo to go to position in variable 'pos'
    delay(15);                          // waits 15ms for the servo to reach the position
  }
  for(pos = 180; pos>=1; pos-=1)        // goes from 180 degrees to 0 degrees
  {
    myservo.write(pos);                  // tell servo to go to position in variable 'pos'
    delay(15);                          // waits 15ms for the servo to reach the position
  }
}
```

http://arduino.cc/en/Tutorial/Knob

Sweep code running