

# Advanced Manufacturing Academy

ROBOTICS – 5  
Input & Sensors

College of Engineering and Technology

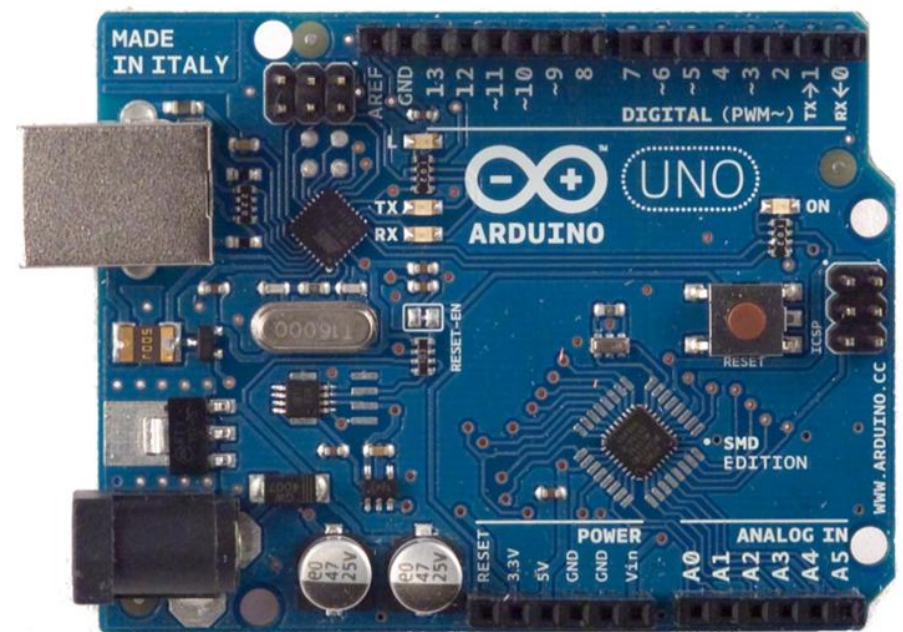
**East Carolina University**

# Our Robot – Our Plan

- We have a moving robot
- How do we know when to stop?
- Need some way to get information back to the Arduino about its environment.
- Sensors!
  - Digital inputs
  - Switch/Button
  - Read Voltages – on/off
  - Programming details – Loops/Conditionals

# Inputs to the Arduino

- Digital inputs
  - Pins 0-13 will act as digital inputs
  - Some with special functions
- Analog Inputs
  - 6 pins: A0 – A5
  - Read 0 – 5VDC
  - 10 bit Resolution



DIGITAL INPUTS

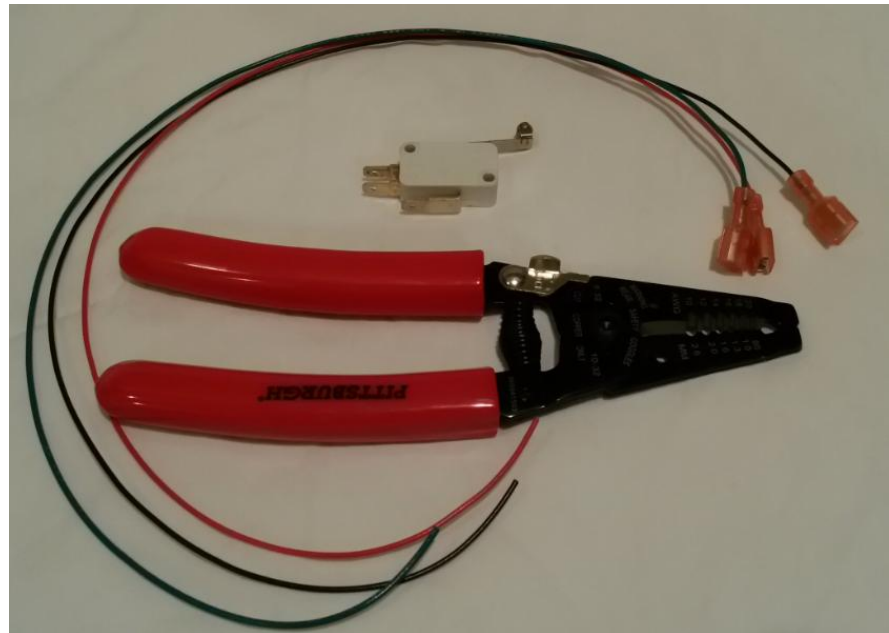
ANALOG INPUTS

# Digital Input – On/Off

- Information to the Arduino by sending
  - 5V = High = True = On
  - 0V = Low = False = Off
- Pick a digital pin – name it
- Set it to be an “input”
  - IE.. If we set pin 2
    - ✓ `Const int pin2 = 2; // const could also be int`
    - ✓ `pinMode(pin2, input); // pin2 is an input`

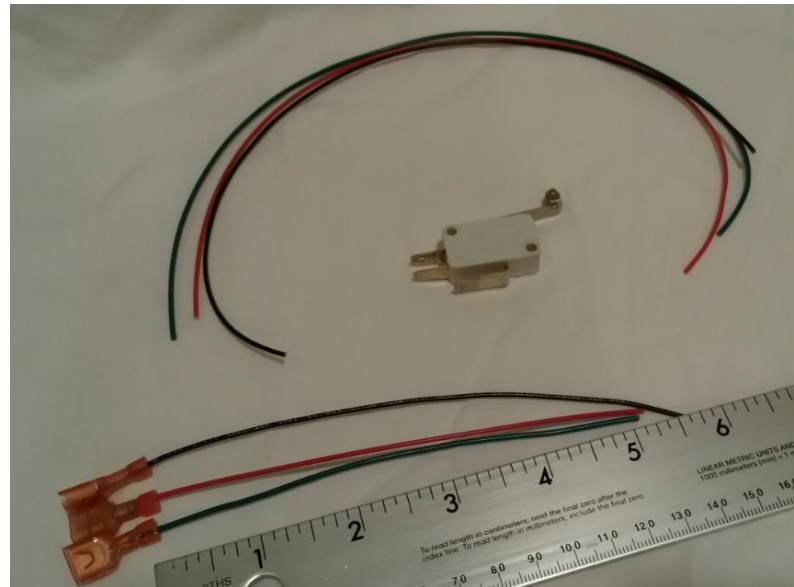
# Parts for our “Button” exercise

- Need to prep and understand parts
- You will need:
  - Wire cutter/strippers
  - Wires with female connectors
  - Switch
  - Resistor



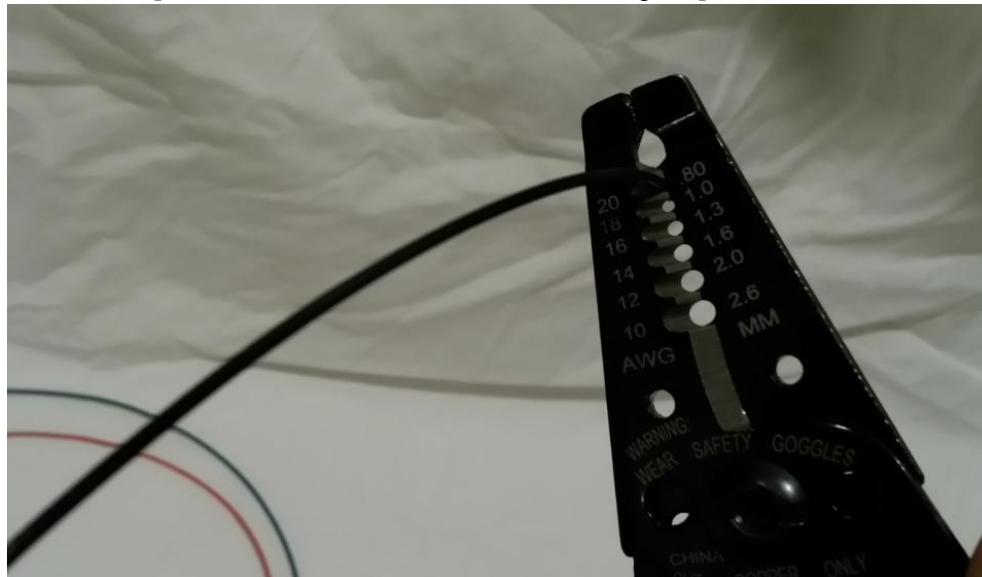
# Make Jumpers/Connectors

- Take each provided wire
  - Measure 6" down from connector
  - Cut the wire
  - You should have 1 plain wire and one with a connector for each color



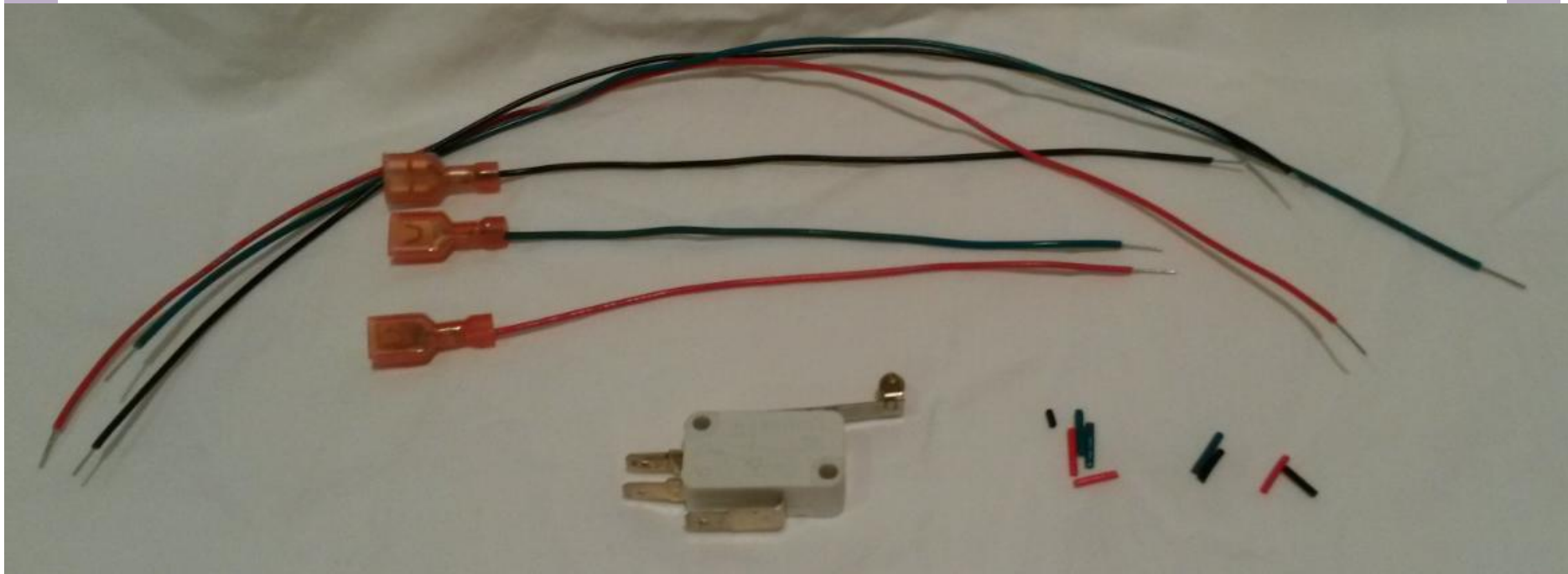
# Make Jumpers/Connector (cont)

- Need to strip  $\frac{1}{4}$ " of insulation off wires
- Strip each end of plain wires
- Strip single end off connector wire
  - Place wire into strippers – Use 20 hole
  - Close pliers and firmly pull wire



# Jumpers/Connectors Done

- Wires should look like this....
  - 3 plain wires with stripped ends
  - 3 connector wires with stripped end





# Switches

- Talk about our switch
- Switch opens and closes a circuit
- Lots of different types but same concept
- Two possible states;
  - Open = OFF
  - Closed = ON



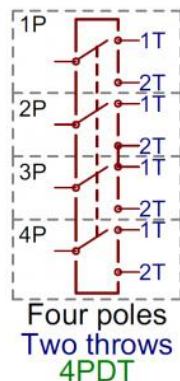
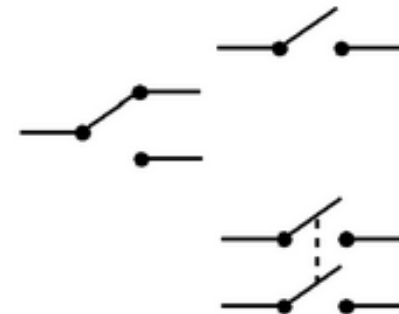
Open



Closed

# You can stack up switches

- Keep adding Poles and Throws
  - Poles – each “Circuit” through switch
  - Throws – how many ways to connect
    - ✓ Single pole single throw
    - ✓ Single pole double throw
    - ✓ Double pole single throw



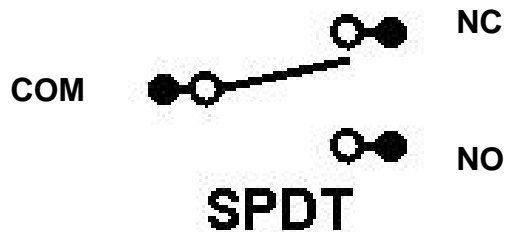
*A massive 4PDT circuit symbol, and an physically massive 4PDT toggle switch.*

<http://www.tb-training.co.uk/MarineE02.html>

<https://learn.sparkfun.com/tutorials/switch-basics/poles-and-throws-open-and-closed>

# Our Switch – The details

- Our Switch is a SPDT – Single Pole Double throw.
- It has three connections
  - COM, NC and NO?
  - There should also be a picture on the side



# Let's Try the Button Program

- Load the Button Program from desktop
  - Init section and setup

```
by Tom Igoe
This example code is in the public domain.
http://www.arduino.cc/en/Tutorial/Button
*/

// constants won't change. They're used here to
// set pin numbers:
const int buttonPin = 2;    // the number of the pushbutton pin
const int ledPin = 13;     // the number of the LED pin

// variables will change:
int buttonState = 0;       // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}
```

# Button Program (cont)

- Button Program – void loop section

```
void loop(){
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  // check if the pushbutton is pressed.
  // if it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn LED on:
    digitalWrite(ledPin, HIGH);
  }
  else {
    // turn LED off:
    digitalWrite(ledPin, LOW);
  }
}
```

# IF Command

- IF Command - A conditional statement/test
  - If Something is true, do the following  
if (someVariable > 50)

```
{
```

```
// do something here
```

```
}
```

```
if (x > 120) digitalWrite(LEDpin, HIGH);
```

```
if (x > 120)  
digitalWrite(LEDpin, HIGH);
```

```
if (x > 120){ digitalWrite(LEDpin, HIGH); }
```

```
if (x > 120){  
    digitalWrite(LEDpin1, HIGH);  
    digitalWrite(LEDpin2, HIGH);  
}
```

```
// all are correct
```

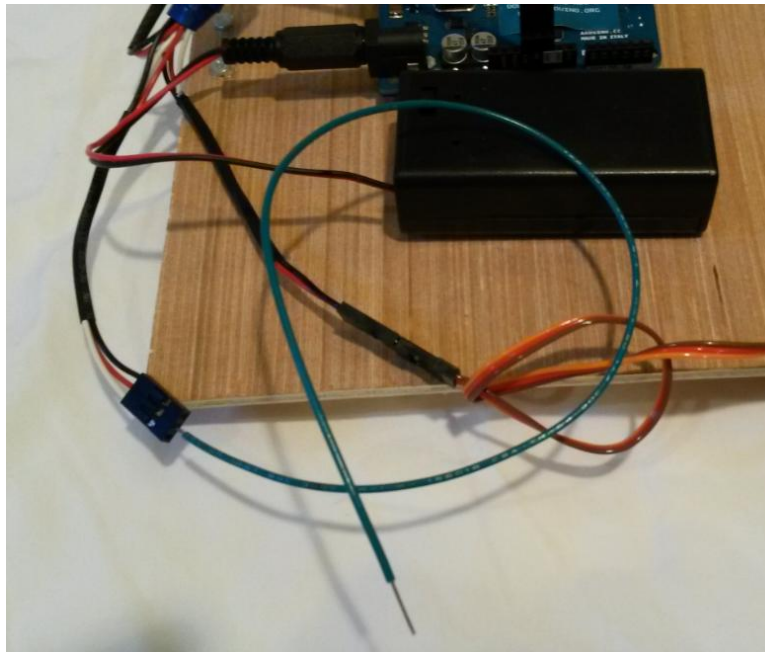
# IF - Else Command

- IF – Else – Two option conditional statement/test
  - If Something is true, do the following otherwise do the other

```
if (pinFiveInput < 500)
{
    // action A
}
else
{
    // action B
}
```

# Button Code Test

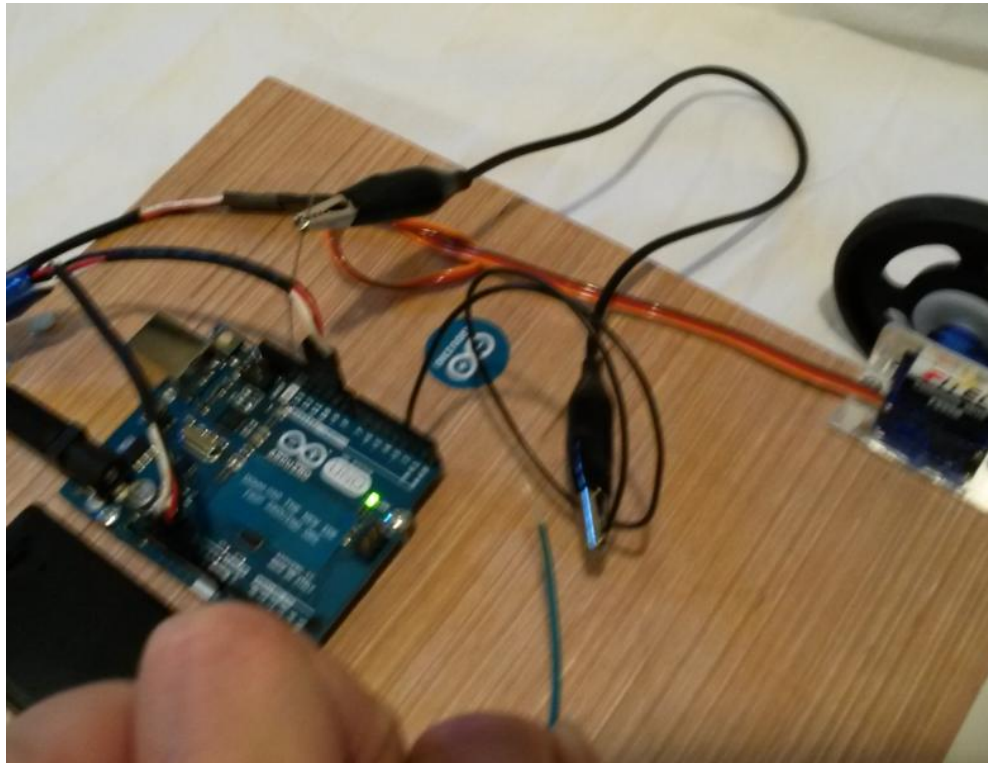
- NOW – Let's try the code
  - Load Button code to arduino
  - Install wire to red wire on pin 11
  - Touch wire to pin 2 – What happens to light?





# Button Code Bugs?

- Light Comes on and stays on?
- Light comes on and goes out?
- Not stable! Add a resistor as shown



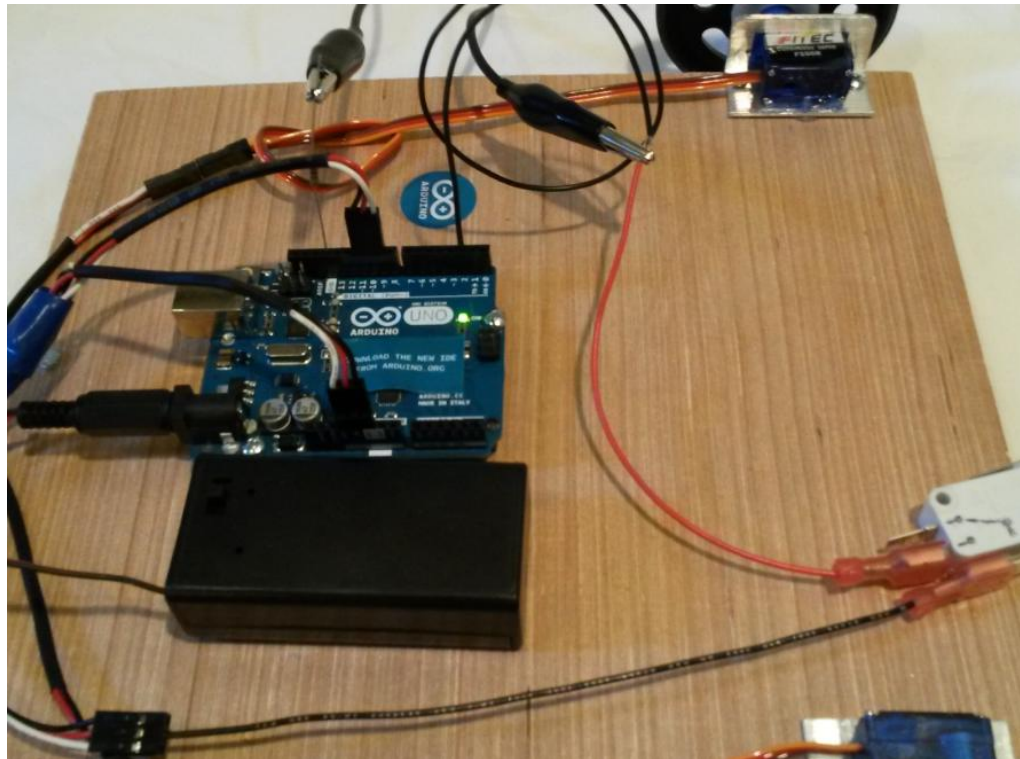
# Add The Switch!

- Not a stable way to make connections
- Add switch
  - Install black lead on “Com”
  - Install red lead on “NO”



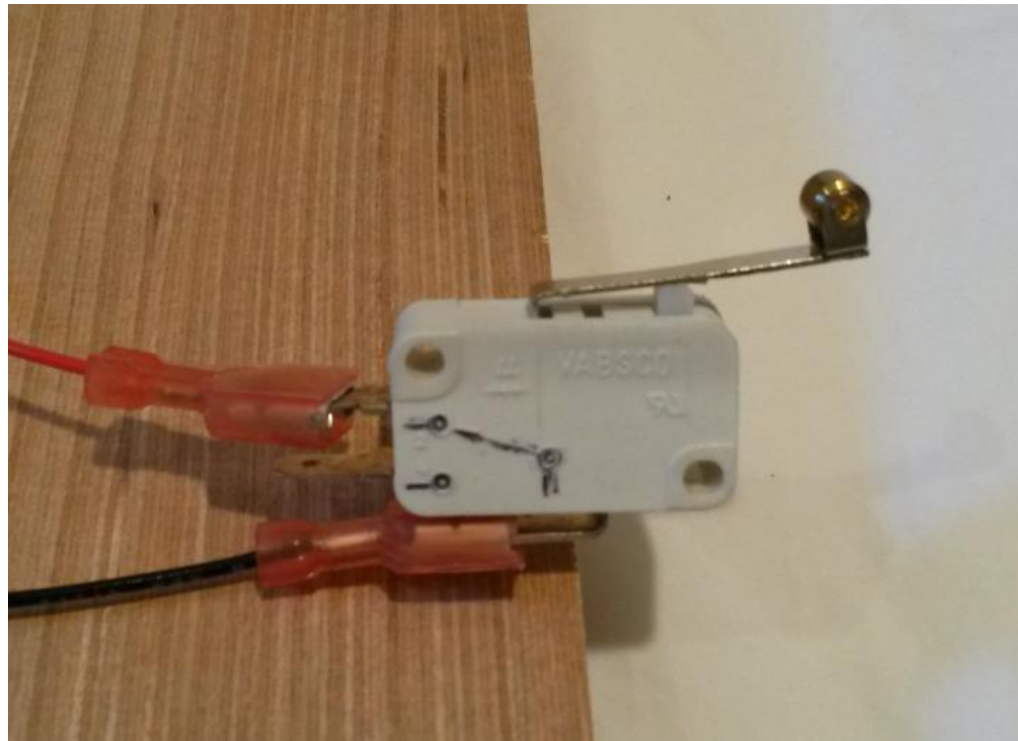
# Install Switch in Circuit

- Remove green wire from plug 11
- Install black wire on red terminal
- Put red wire and black wire in alligator clip



# Try Switch

- What happens if you switch the leads?
- Like this?
- NO vs. NC?



# CHALLENGES!

- Use switch Input to stop/start 1 servo?
  - Need servo and if-then functions added to button file...
  - Make sure to include all the parts!
- Make it run continuously?

# Build your BOT!

- Build a bot to solve the final Challenge
  - Example provided in hall
- Use all parts provided
- Hot glue and cardboard provided
- Store files on your jump drive
- Working time
  - Today
  - Short time on Friday
- Competition FRIDAY 6/26!