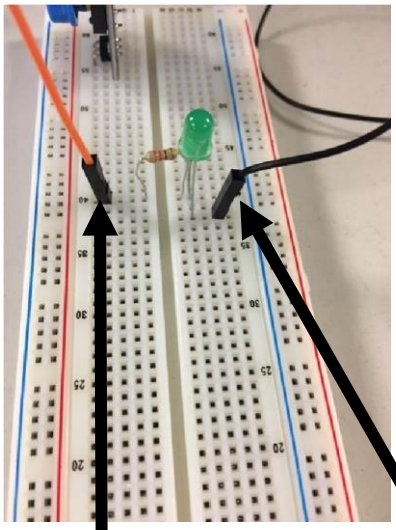
Program 3: Push Button

For this Module we will alter the TinkerCAD code as shown on Page 2 (Line of Code: if (buttonstate == LOW) and use a Button Module

Wire an LED Light



Orange Wire to Digital Port 12

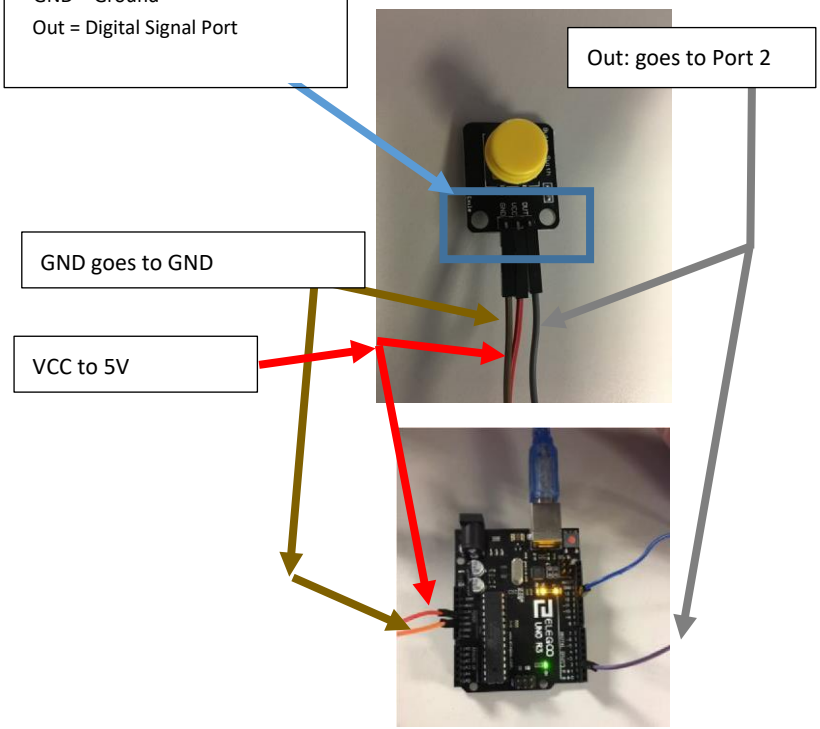
Black Wire to Ground

Wiring Diagram Option 1

Sample: Button Module Direct Wiring to Arduino

WARNING: Be Sure to Look at Each Button Module for Pin Definitions

Button Module Board has each Pin labeled to define each pin
- VCC = Volt
- GND = Ground
- Out = Digital Signal Port



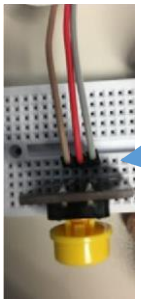
Wiring Diagram Option 2

Direct Wiring to Arduino

Brown Wire: Ground (GND) to Arduino Ground (GND)

Red Wire: Voltage (VCC) to Arduino 5V

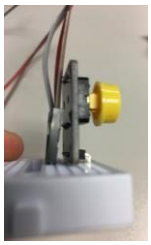
Grey Wire: Out to Arduino Port 13



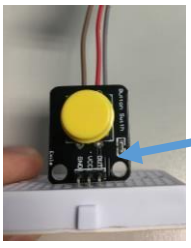
Align each wire in the same row as the pin connection

Top

Back



Side



Shows what each pin represents

Front

Write the following Program

```
int buttonpin = 2;            //Digital Port for Button to sense being pressed
int ledpin = 12;              // Digital Port to Turn LED ON/OFF

int buttonstate = 0           //Variable buttonstate is assign 0 = OFF/Not Pressed and 1 =ON/Pressed State

void setup() {
  // put your setup code here, to run once:
  pinMode (ledpin, OUTPUT);     //Sends only output instructions to ledpin (Port 2)
  pinMode (buttonpin, INPUT);   //Recieves only input instructions from button (Pressed v Not Pressed)
}

void loop() {
  // put your main code here, to run repeatedly:
  buttonstate = digitalRead (buttonpin);      //digitalRead reads the signal coming from the button and
                                              //assigns 0 = OFF/Not Pressed and 1 =ON/Pressed State to Variable buttonsate
  //Module PushButton
  //LOW State = Button Pressed
  //HIGH State = Button Not-Pressed
  if (buttonstate == LOW)                     //Condition sees if button has been pressed
  {
    digitalWrite (ledpin,HIGH);                    //Button Pressed LED turns ON
  }
  else
  {
    digitalWrite (ledpin, LOW);               //Button Not-Pressed LED turns OFF
  }
}
```

Assignment: Push Button

1. Modify the program so the serial monitor screen shows
    a. Text On (Button Pressed) and Off (Button Released)
    b. Counter showing how many times the button has been released
2. Limit of On/Off
    a. Maximum presses = 5
    b. Once Limit is met do the following
        i. Stop the light from turning on and off when button is pressed
        ii. Serial Port Print “ALL DONE!”
        iii. Only print “ALL DONE!” twice on separate lines (Serial.println)