**Lab Exercise 11/21/2022**

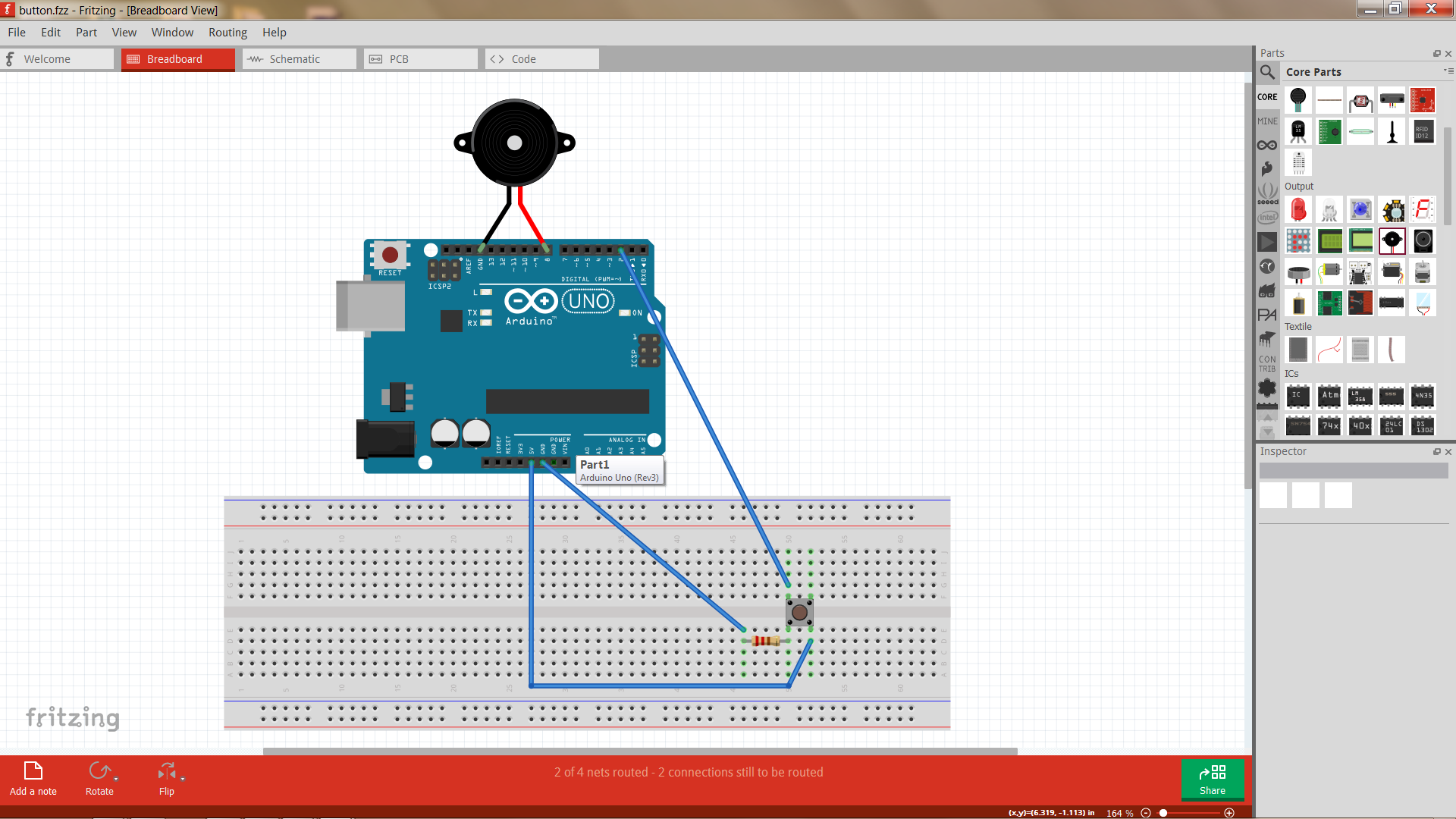
**Programming I**

**Creating a Music Box (“Ode to Joy”)**

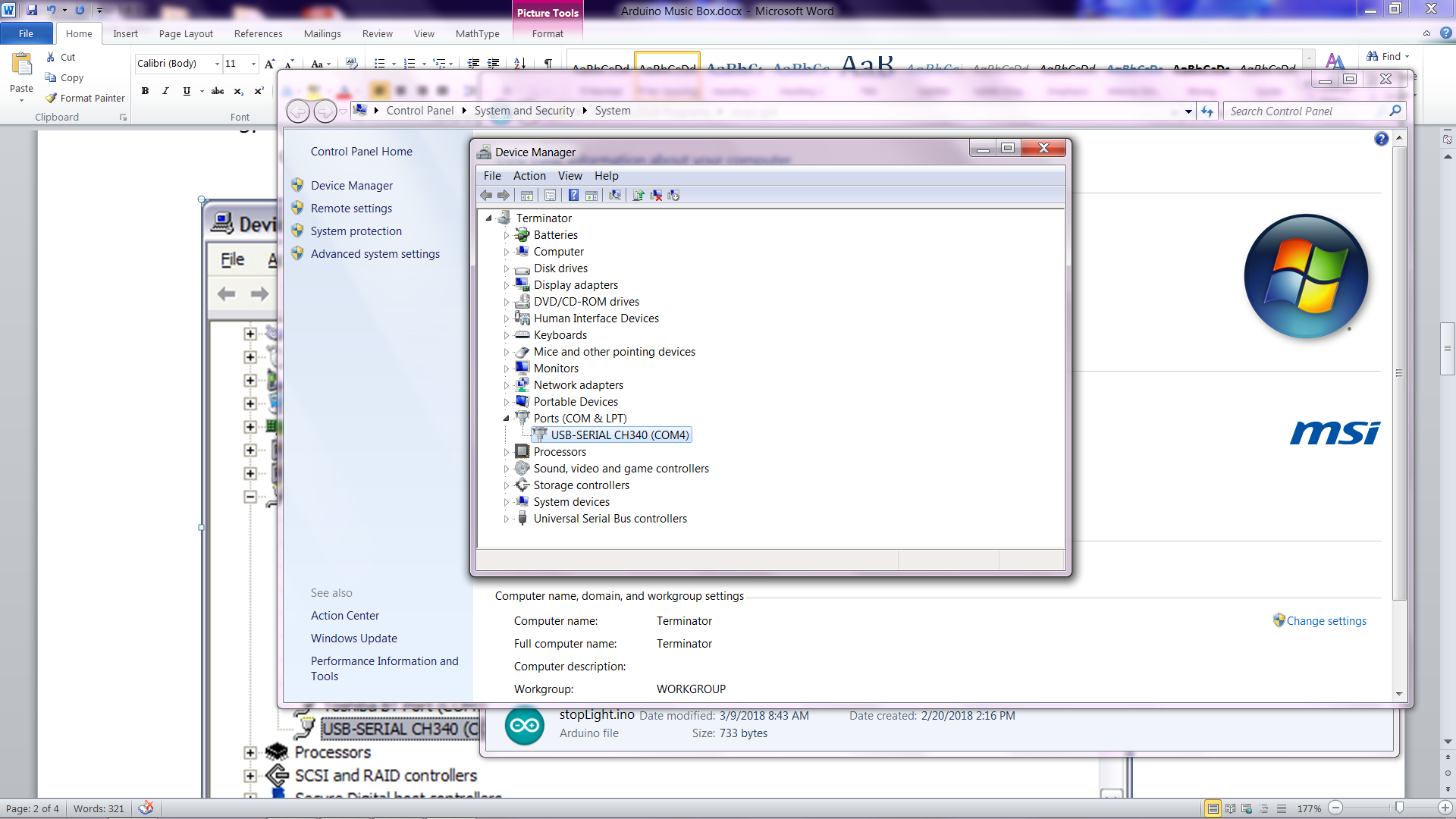
In this activity, you will require the following:

* Arduino
* Breadboard
* Pushbutton Switch
* One 10000 Ohm Resistor
* Hookup wires
* USB A-B cable

1. Connect our pushbutton to the Arduino by mounting it on the breadboard.
2. Place the 10000 Ohm resistor in the breadboard
3. Hook up the Arduino to the breadboard as in the diagram attached.
4. Connect your Arduino to your computer with the USB A-B cable.

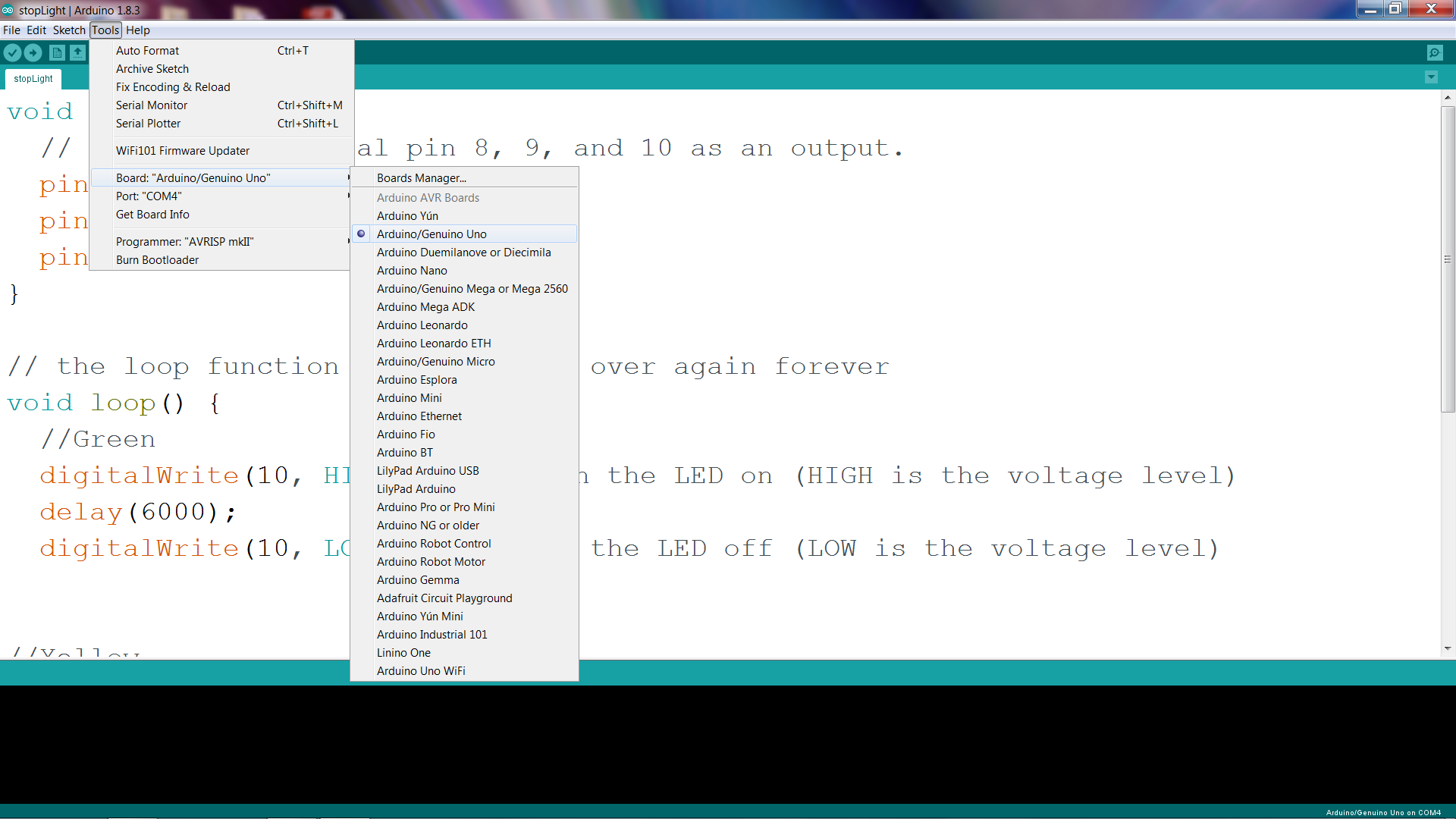


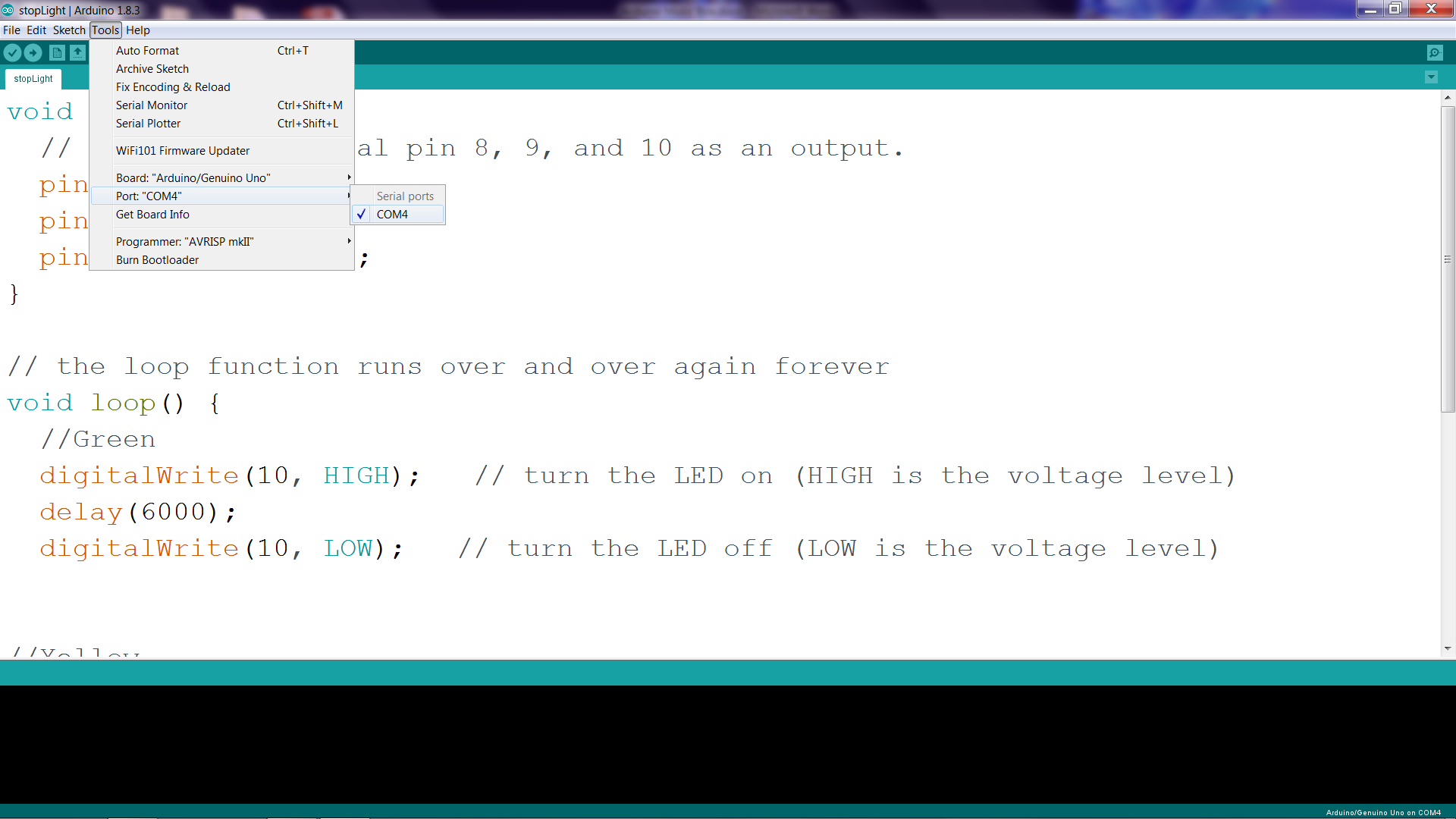
1. (Optional) Make sure Arduino is you can check out device manager to find out what Port we are connected to. You can see this in the Device Manager.



Note: Mine is connected to Port 4 but yours might be different. You should make a note of this.

1. Start your Arduino development environment.
2. On your Tools menu, check that the Board is set to Uno and the Port is set to the value you noted in step 5.





In this program, we will be using two special functions; tone and delay.

The tone function has 3 parameters; pin to send signal to, frequency in Hertz, and duration in milliseconds

The delay function has 1 parameter; time in milliseconds

Now let’s make some music by typing in the following code and running it. Press the button and see what happens. Do you recognize the song?

**Now let’s add the code**

//Plays a melody

const int buttonPin = 2;

int pbState = 0;

void setup(){

pinMode(buttonPin, INPUT);

}

void loop(){

pbState = digitalRead(buttonPin);

if (pbState == HIGH){

//First group of notes

tone(8, 247, 300);

delay(500);

tone(8, 247, 300);

delay(500);

tone(8, 262, 300);

delay(500);

tone(8, 294, 300);

delay(500);

tone(8, 294, 300);

delay(500);

//second group of notes

tone(8, 262, 300);

delay(500);

tone(8, 247, 300);

delay(500);

tone(8, 220, 300);

delay(500);

tone(8, 196, 300);

delay(500);

tone(8, 196, 300);

delay(500);

//third group of notes

tone(8, 220, 300);

delay(500);

tone(8, 247, 300);

delay(500);

tone(8, 247, 500);

delay(650);

tone(8, 220, 200);

delay(250);

tone(8, 220, 200);

delay(250);

}

}

**Stop Light Simulator**

In this lab we are going to write a program that interfaces with computer hardware to make something physical occur.

For this activity we will require the following:

1. Arduino Uno
2. Four male/male jumper wires
3. Solderless breadboard
4. Red LED, Yellow LED, Green LED
5. 220 Ω resistor
6. USB cable (A – B)

The hardware we will be using will be the Arduino which has 13 Digital IO (Input Output) pins and 6 Analog input pins which can be easily programmed to send output signals to or receive input signals.

For this activity, we will be using pins 8, 9 and 10 as well as one of the Ground pins (labeled GRD). Here is a diagram of an LED setup. To get the LED to light we need approximately 2.1 volts across it. The purpose of the resistor is to limit the current through the LED. Pin 8, 9, and 10 are capable of supplying 5 volts. If we do not use a current limiting resistor, the LED will light brightly for a while and then go dark. For a nominal brightness we want about 10 mA to flow through the LED. This means a 220 Ω resistor should do. Actually we should be OK with a resistor or 180 Ω to 550 Ω.

A computer screen shot of a computer

Description automatically generated

**Hardware Assembly Procedure**

1. Add 330 Ω resistor to the breadboard
2. Add Red LED to breadboard so Cathode of LED (shorter lead) is connected to resistor
3. Add Yellow LED to breadboard so Cathode of LED (shorter lead) is connected to resistor
4. Add Green LED to breadboard so Cathode of LED (shorter lead) is connected to resistor
5. Connect GRD on the Arduino header to unconnected lead of resistor
6. Connect Pin 8 on Arduino header to Anode of Red LED (longer lead)
7. Connect Pin 9 on Arduino header to Anode of Yellow LED (longer lead)
8. Connect Pin 10 on Arduino header to Anode of Green LED (longer lead)
9. Hook up your Arduino to your computer.

**Now let's make our stoplight simulation:**

Type in the following script:

// Stop light simulation

// the setup function runs once when you press reset or power the board

void setup() {

// initialize digital pin 8, 9, and 10 as an output.

pinMode(8, OUTPUT);

pinMode(9, OUTPUT);

pinMode(10, OUTPUT);

}

// the loop function runs over and over again forever

void loop() {

//Green

digitalWrite(10, HIGH); // turn the LED on (HIGH is the voltage level)

delay(6000);

digitalWrite(10, LOW); // turn the LED off (LOW is the voltage level)

//Yellow

digitalWrite(9, HIGH); // turn the LED on (HIGH is the voltage level)

delay(2000);

digitalWrite(9, LOW); // turn the LED off (LOW is the voltage level)

//Red

digitalWrite(8, HIGH); // turn the LED on (HIGH is the voltage level)

delay(6000);

digitalWrite(8, LOW); // turn the LED off (LOW is the voltage level)

}