MIRACLE
SOFTWARE SYSTEMS

Building Fun Circuits with Arduino and Sketch IDE

Mini Lab | Digital Summit 2019

Building Fun Circuits with Arduino and Sketch IDE

Introduction

Come and play with few IoT devices and try to control them with a microprocessor like Arduino. In this mini lab you will learn how to program with Arduino IDE and building your own circuit in order to control the sensors.

Pre-Requisites

All attendees must have their workstation (with Internet) to participate in the lab (Both PC and MAC are compatible). The following pre-requisites will help you to make the Hands-on Lab experience easier.

- Arduino IDE

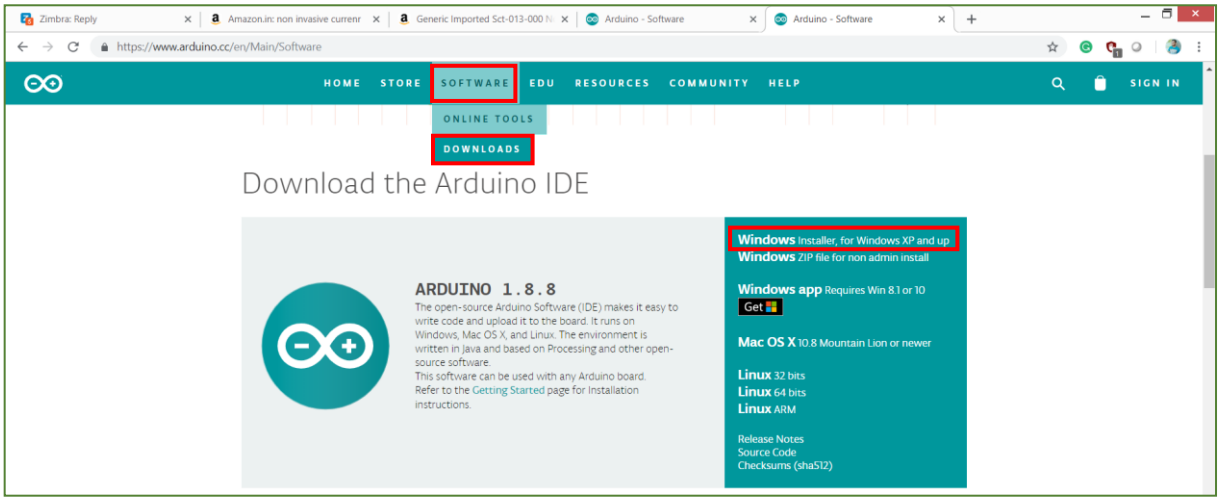
Lab Steps

So, let us get started with the lab!

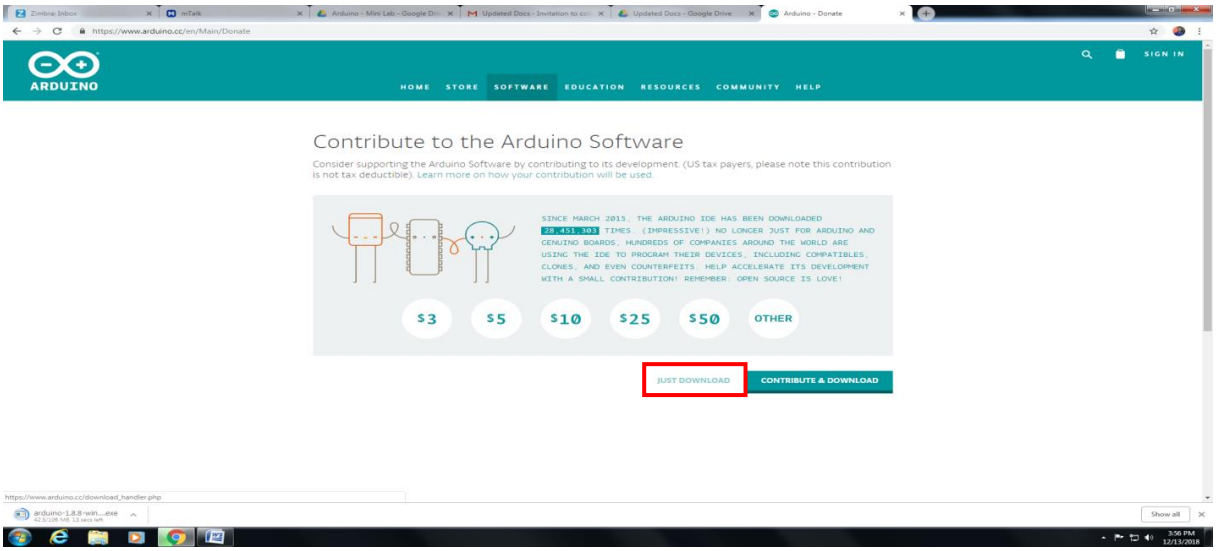
Step #1 | Installing Arduino IDE

To install the Arduino IDE, click on the below link and follow this process as shown below. Select Software tab from the top menu and click on Downloads to download the IDE file.

https://www.arduino.cc/en/Main/Software



After selecting the download file for windows, please select **JUST DOWNLOAD** as shown in the below image.



Upon successful download of the file, please click on the .exe file to install the Arduino IDE on your computer/pc.

Arduino UNO

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

Step #2 | Connecting Ultrasonic sensor and Buzzer

In this project you're going to build a simple circuit with an Arduino and Ultrasonic Sensor that can measure the distance between object and sensor. An LED will light up when it measures the distance and defines some threshold value for the distance and trigger buzzer.

Parts required are as follows,

- Arduino Uno
- Micro USB cable
- Ultrasonic sensor
- LED
- Buzzer
- Jumper cables
- Bread board

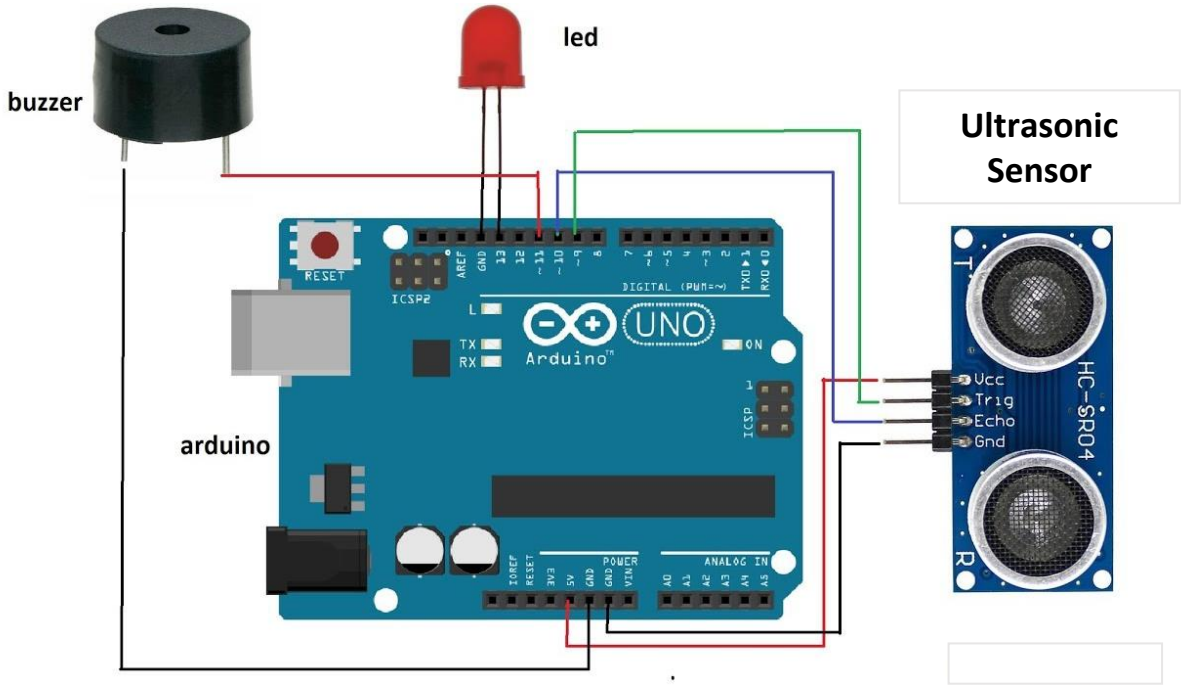
HC-SR04 Ultrasonic Sensor

The ultrasonic sensor uses sonar to determine the distance to an object. Here's what happens, the transmitter (trig pin) sends a signal: a high-frequency sound. When the signal finds an object, it is reflected and the transmitter (echo pin) receives it.

Pin Out

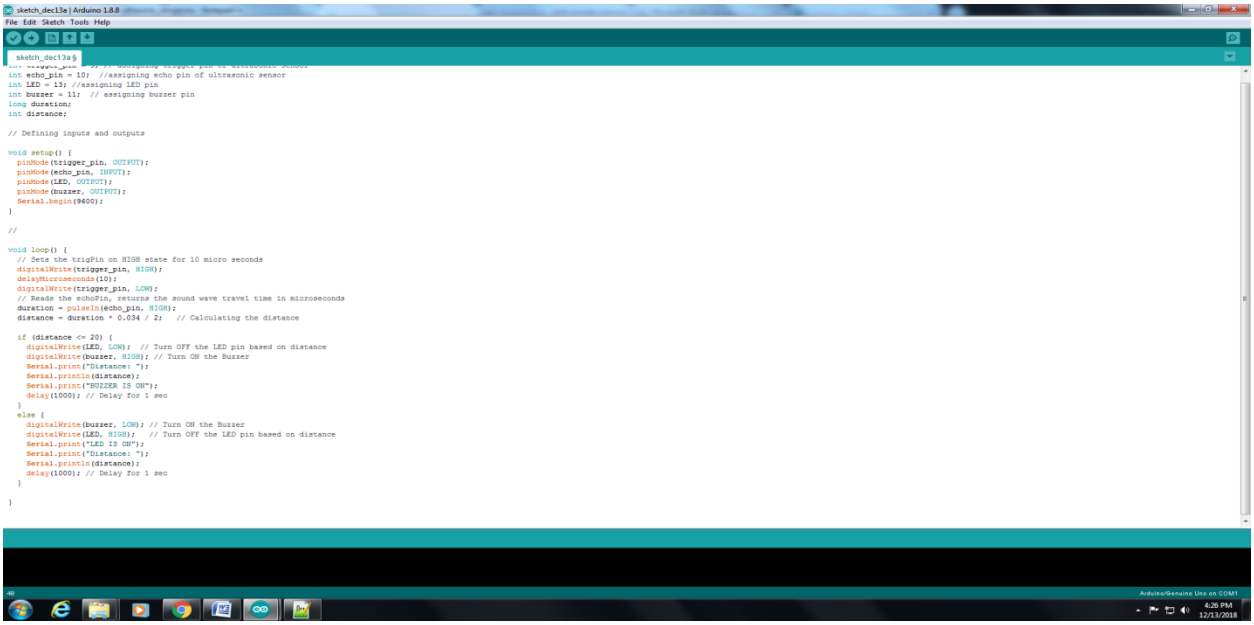
- Vcc - connect to 5v
- Trig - connect to Arduino digital pin (input)
- Echo - connect to Arduino digital pin (output)
- GND - connect to ground

Schematic Diagram



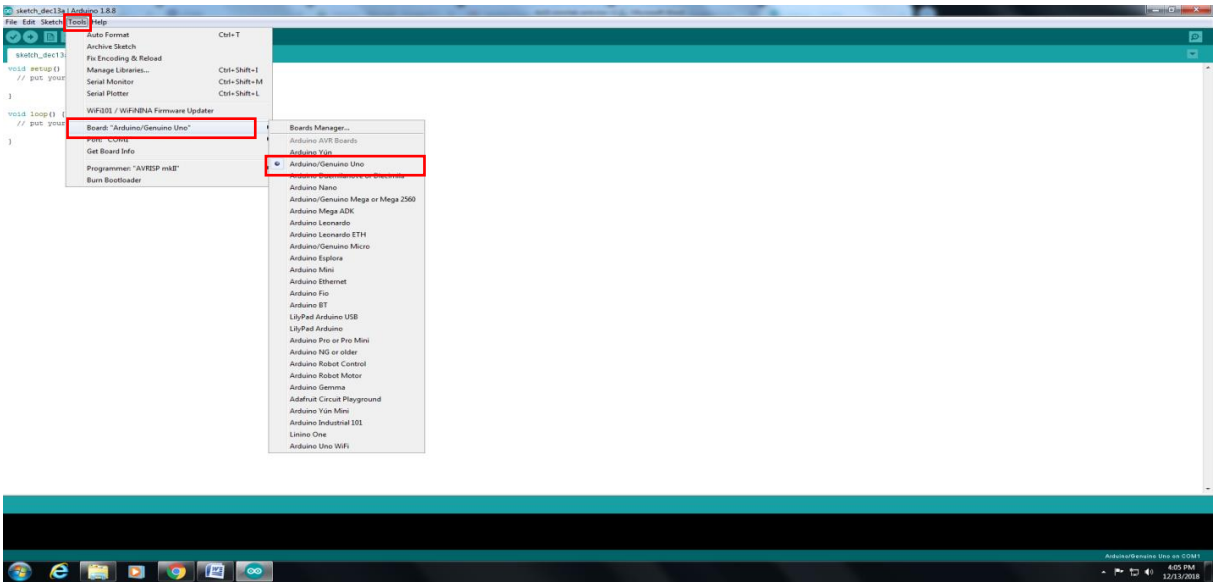
Connect the sensors accordingly. After connecting the sensors to the Arduino, open Arduino IDE, which we had installed previously in **Step #1**. Connect the Arduino board to your PC/laptop using an USB Cable.

Code Snippet

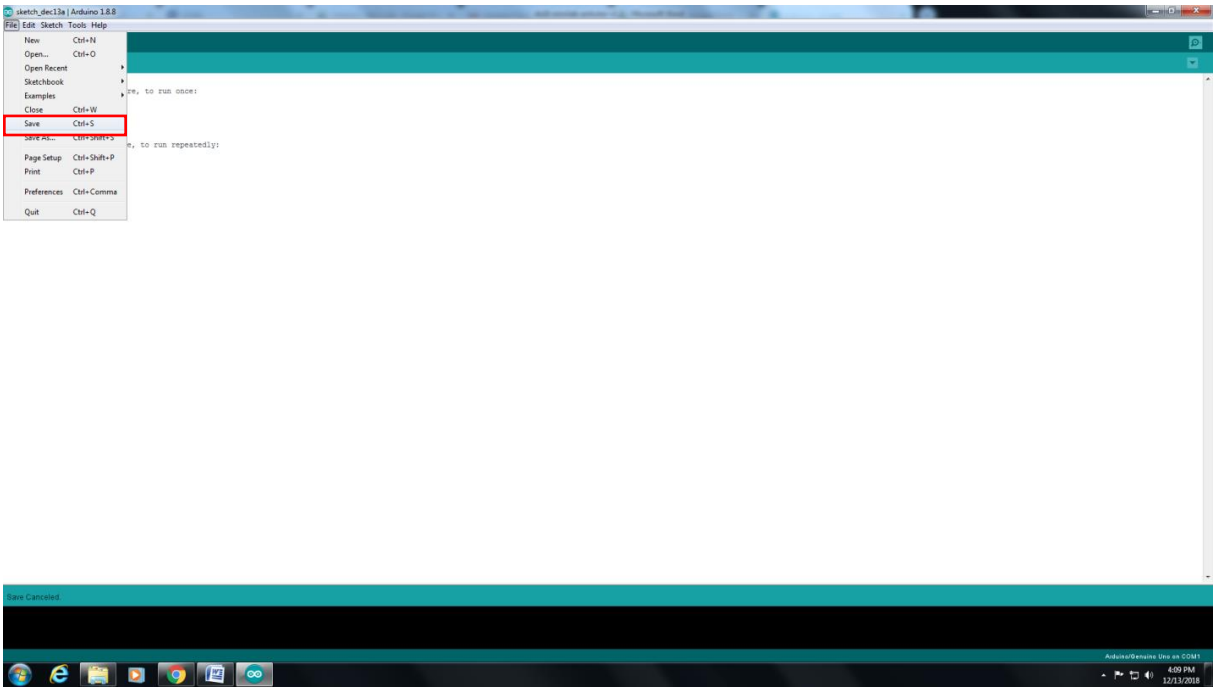


```
int echo_pin = 10;  //assigning echo pin of ultrasonic sensor
int LED = 13; //assigning LED pin
int buzzer = 11;  // assigning buzzer pin
long duration;
int distance;

// Defining inputs and outputs

void setup() {
  pinMode(trigger_pin, OUTPUT);
  pinMode(echo_pin, INPUT);
  pinMode(LED, OUTPUT);
  pinMode(buzzer, OUTPUT);
  Serial.begin(9600);
}

//

void loop() {
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigger_pin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigger_pin, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echo_pin, HIGH);
  distance = duration * 0.034 / 2;  // Calculating the distance

  if (distance <= 20) {
    digitalWrite(LED, LOW);  // Turn OFF the LED pin based on distance
    digitalWrite(buzzer, HIGH); // Turn ON the Buzzer
    Serial.print("Distance: ");
    Serial.println(distance);
    Serial.print("BUZZER IS ON");
    delay(1000); // Delay for 1 sec
  }
  else {
    digitalWrite(buzzer, LOW); // Turn ON the Buzzer
    digitalWrite(LED, HIGH);  // Turn OFF the LED pin based on distance
    Serial.print("LED IS ON");
    Serial.print("Distance: ");
    Serial.println(distance);
    delay(1000); // Delay for 1 sec
  }

}
```

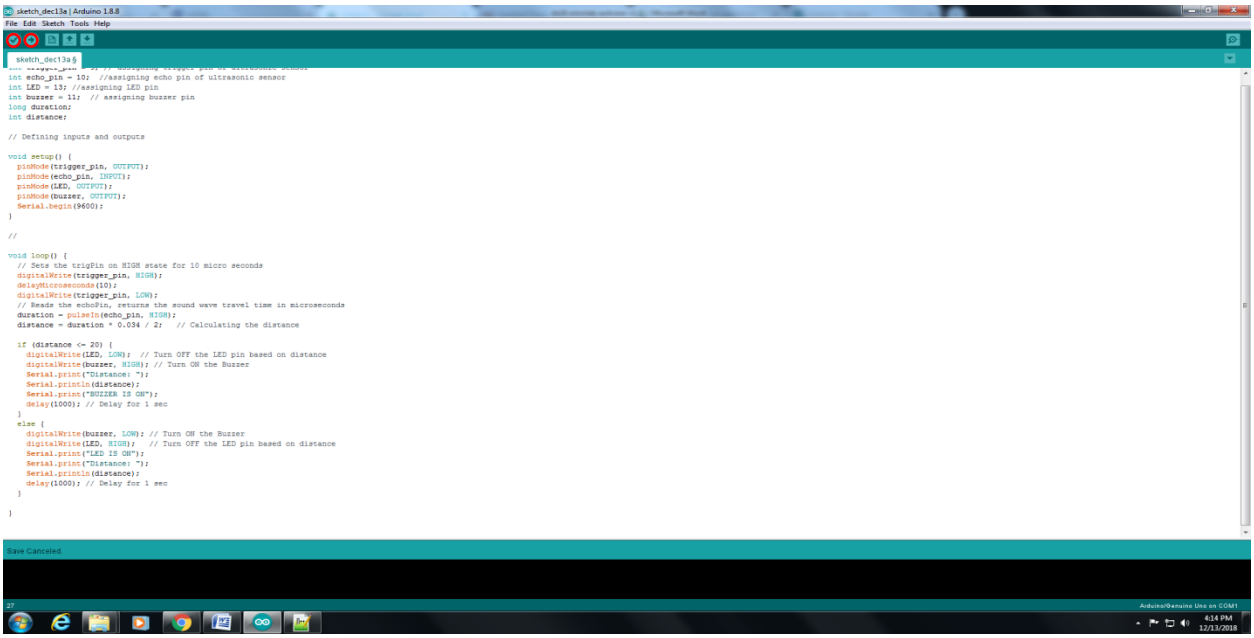
To deploy the code onto the board, we need to set the Board as **Arduino UNO** and port as **COM** port listed in the **port**, which are listed in the **Tools**.



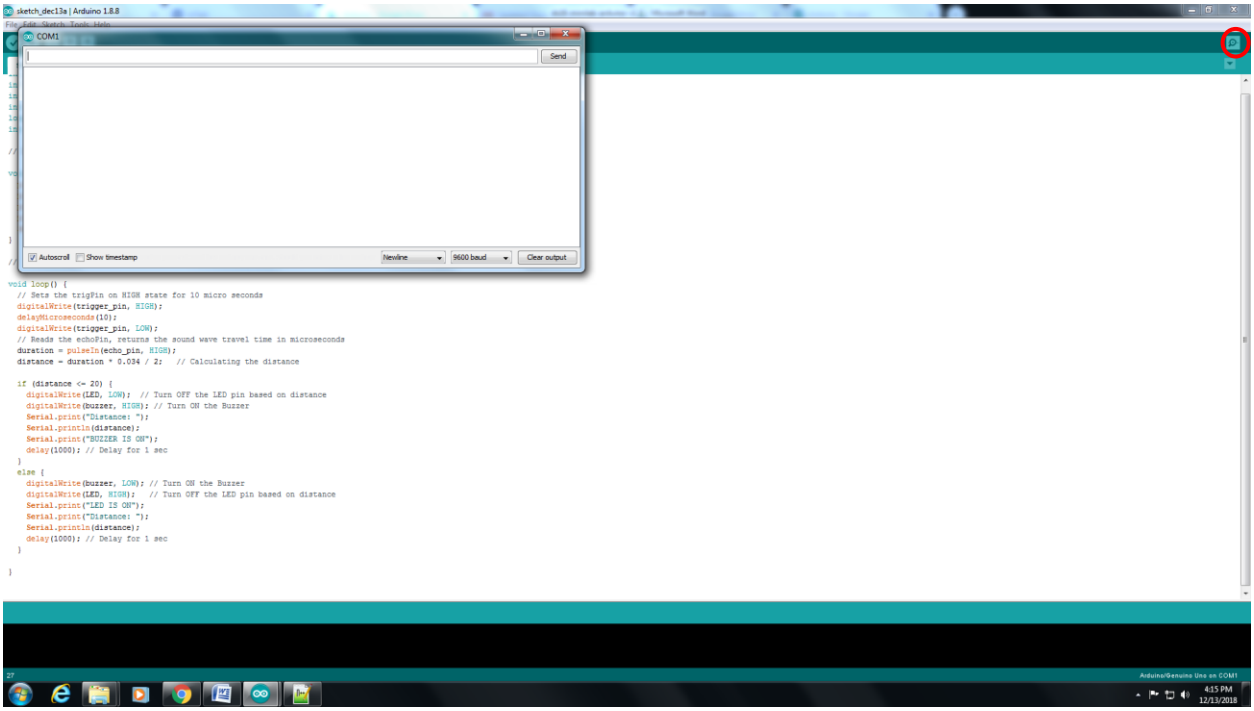
Open **Arduino** folder which is there in **Arduino - Mini Lab** folder and select **ultrasonic_sensor.ino** file and copy that code and paste it in Arduino IDE and save it with any name. To save the code click on **File** and click on **Save**.



Now click on **Verify** to verify if the code is error free and then click on **Upload** button to deploy the code into the Arduino board.



To view the output we need to go to **Serial Monitor**.



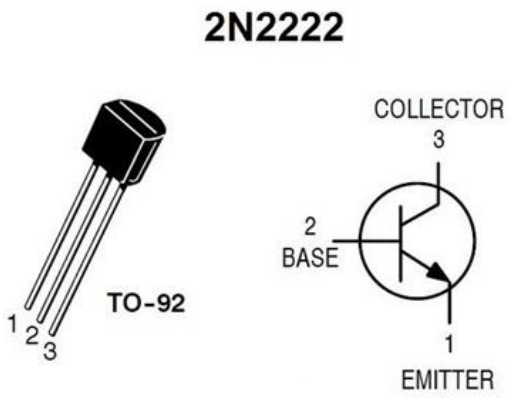
Step #3| Controlling DC Motor with Arduino

In this part, we are controlling the FAN which is connected to motor by using push button switch.

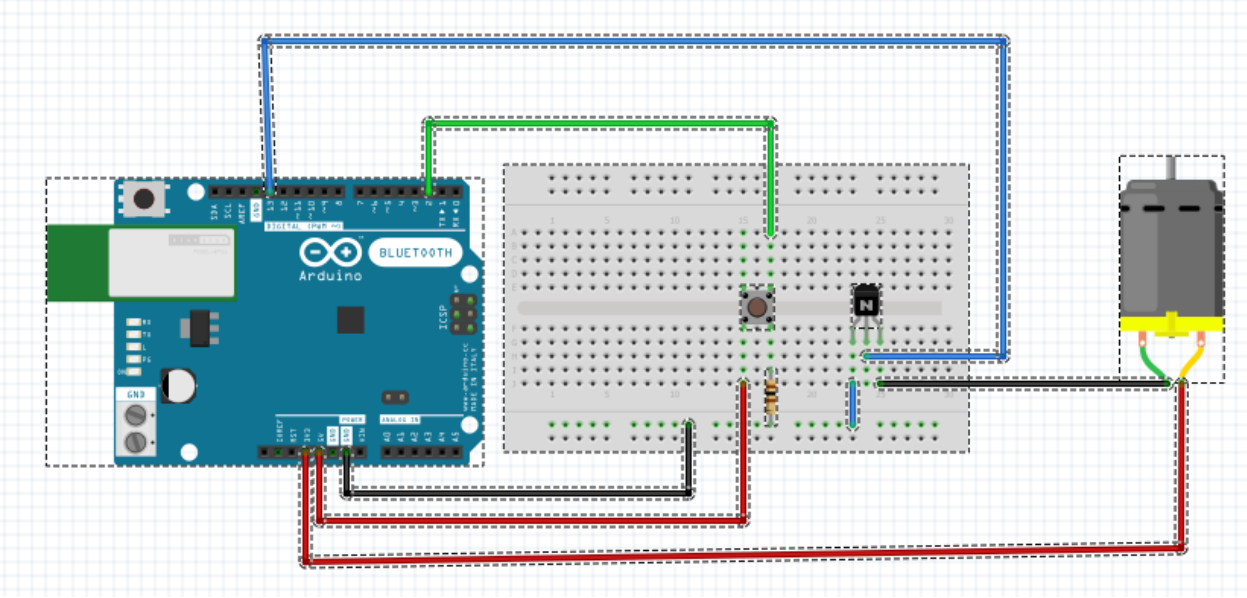
Parts required are as follows,

- Arduino Uno
- Micro USB cable
- DC motor 5v
- Push button Switch
- Jumper cables
- Bread Board
- Transistor
- Transistor (npn)

A transistor is a semiconductor device used to amplify or switch electronic signals and electrical power. It is composed of semiconductor material usually with at least three terminals for connection to an external circuit.

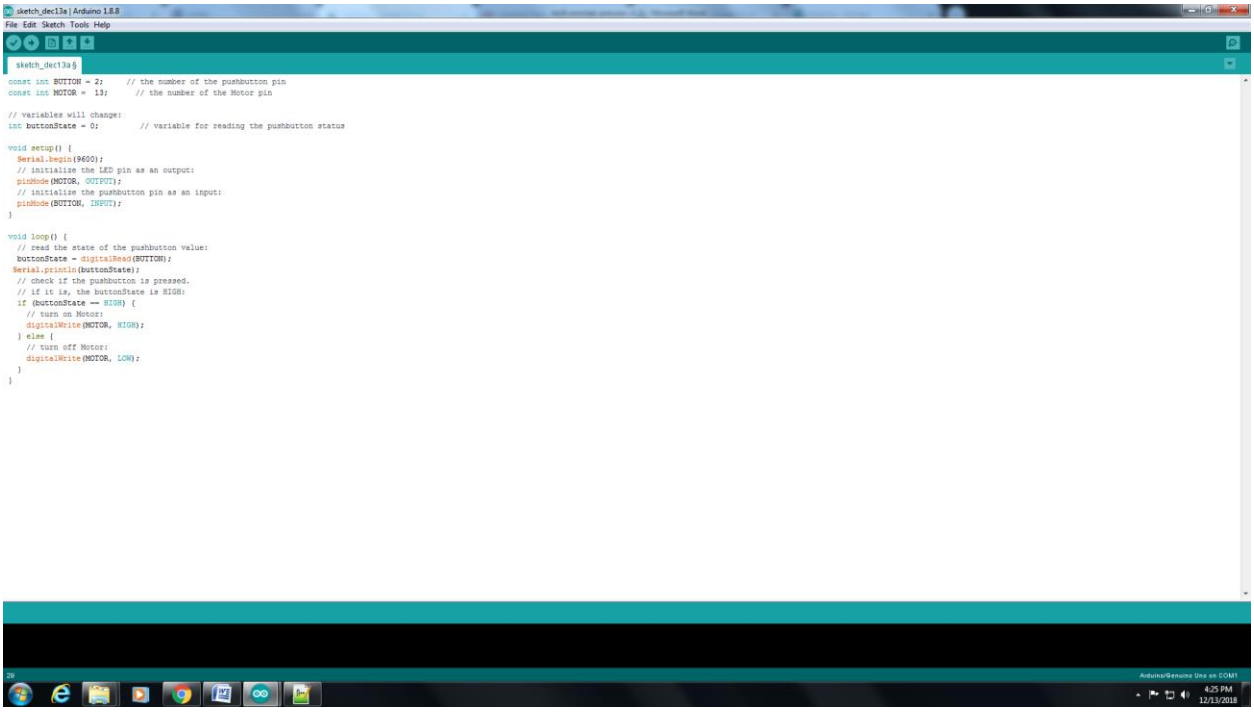


Schematic Diagram



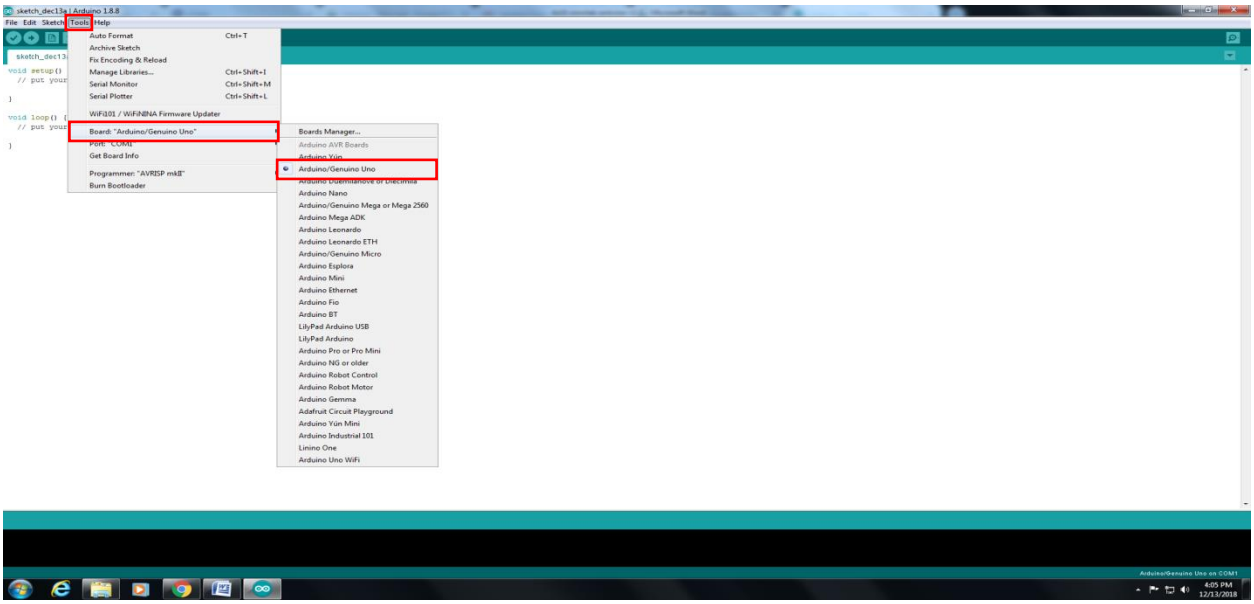
Connect the sensors accordingly. After connecting the sensors to the Arduino, open Arduino IDE, which we had installed previously in **Step #1**. Connect the Arduino board to your PC/laptop using an USB Cable.

Code Snippet

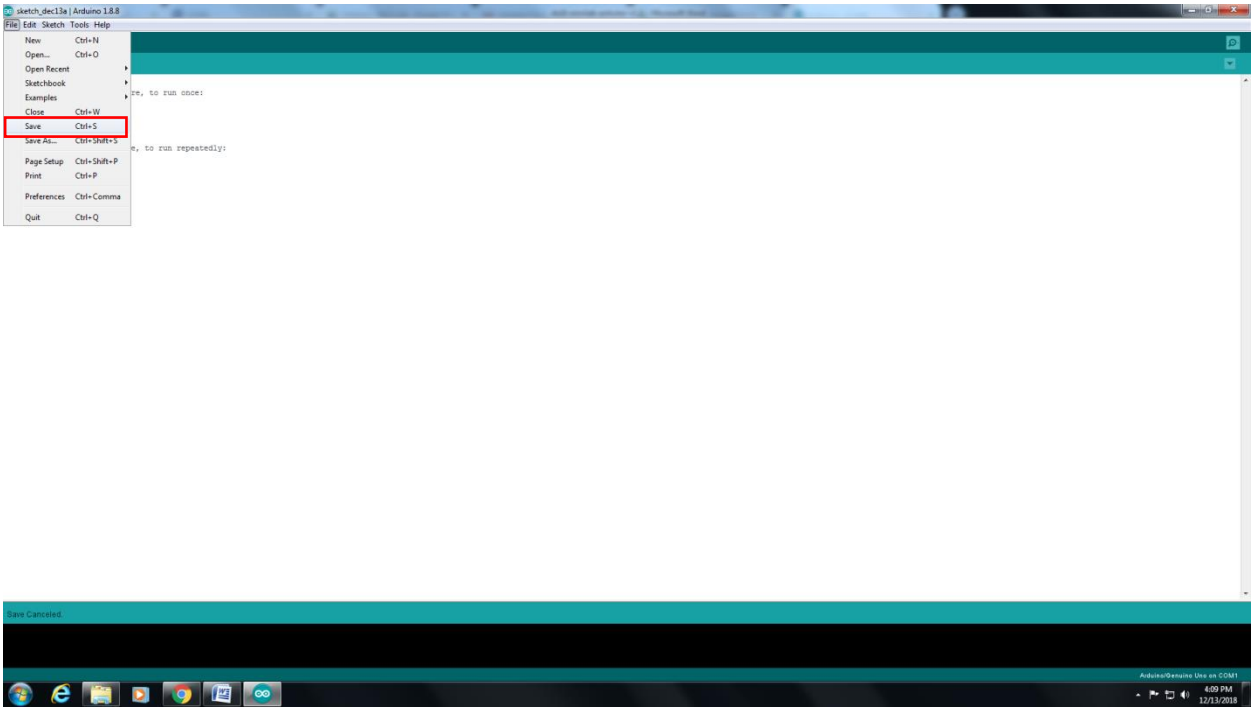


```
const int BUTTON = 2;     // the number of the pushbutton pin
const int MOTOR =  13;     // the number of the Motor pin

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

void setup() {
  Serial.begin(9600);
  // initialize the LED pin as an output:
  pinMode(MOTOR, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(BUTTON, INPUT);
}

void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(BUTTON);
Serial.println(buttonState);
  // check if the pushbutton is pressed.
  // if it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn on Motor:
    digitalWrite(MOTOR, HIGH);
  } else {
    // turn off Motor:
    digitalWrite(MOTOR, LOW);
  }
}
```

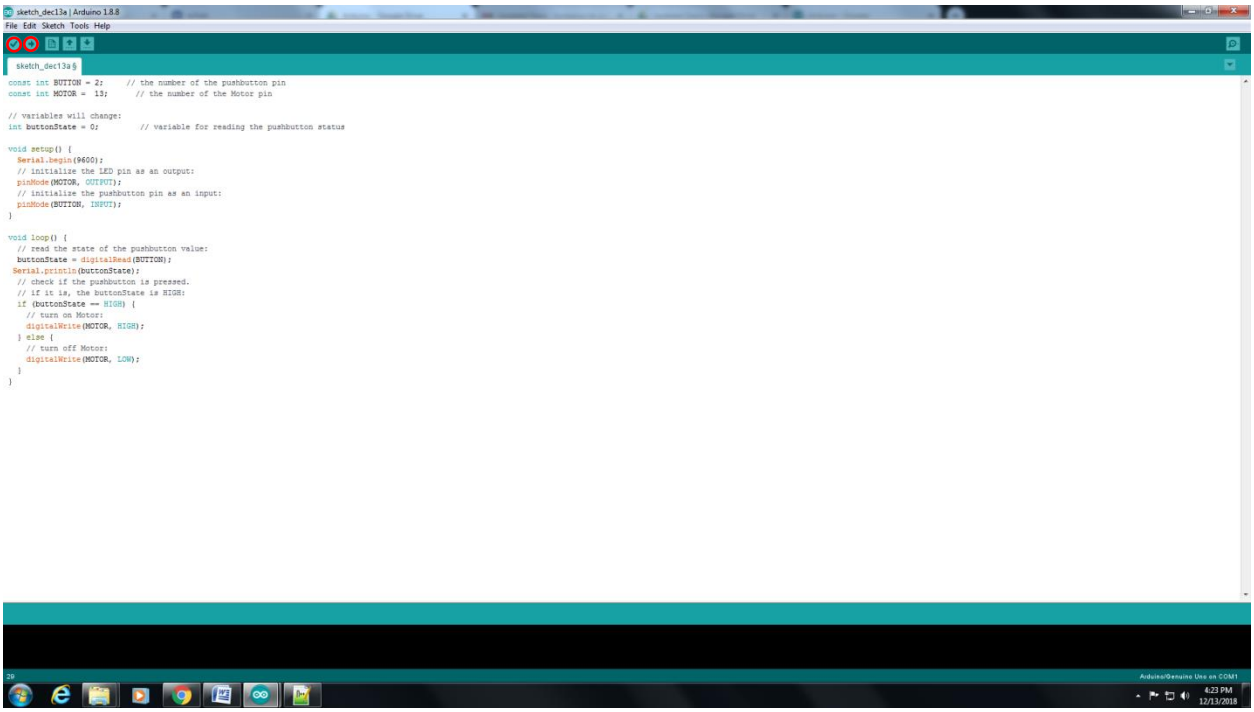
To deploy the code onto the board, we need to set the Board as **Arduino UNO** and port as **COM** port listed in the **port**, which are listed in the **Tools**.



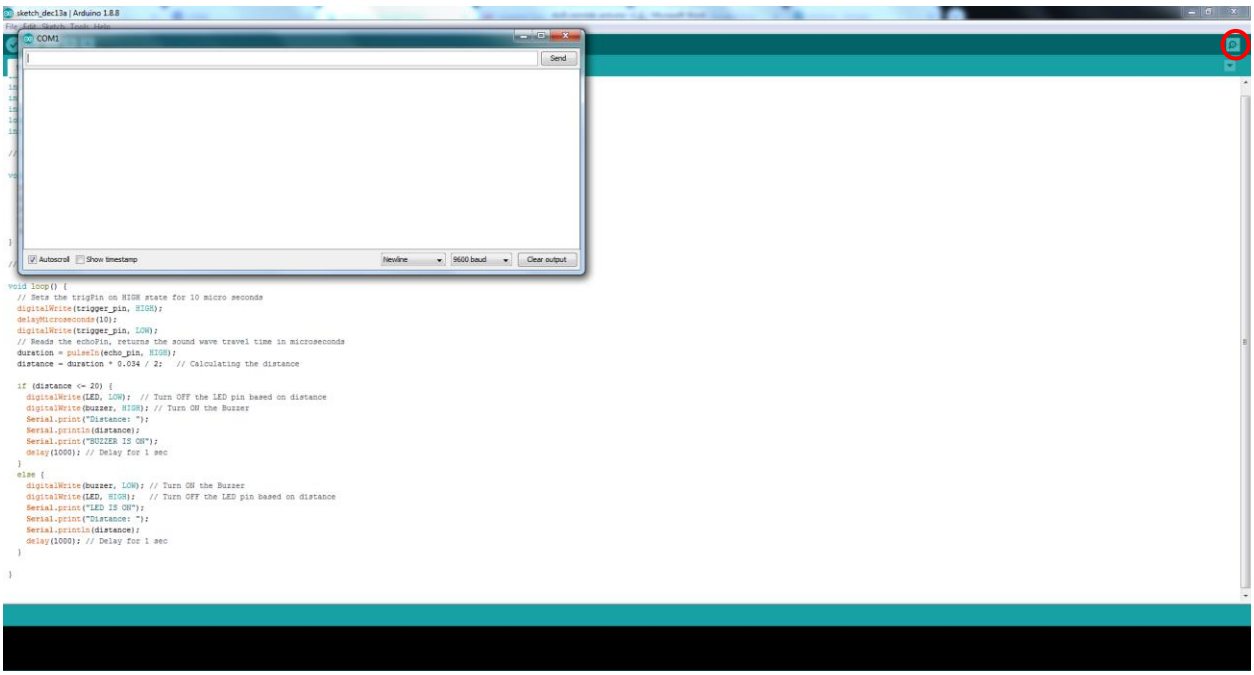
Open **Arduino** folder which is there in **Arduino - Mini Lab** folder and select **motor.ino** file and copy that code and paste it in Arduino IDE and save it with any name. To save the code click on **File** and click on **Save**.



Now click on **Verify** to verify if the code is error free and then click on **Upload** button to deploy the code into the Arduino board.



To view the output, we need to go to **Serial Monitor**.



For any questions regarding the lab, please feel free to reach out to **innovation@miraclesoft.com**. **We hope you enjoyed this!**