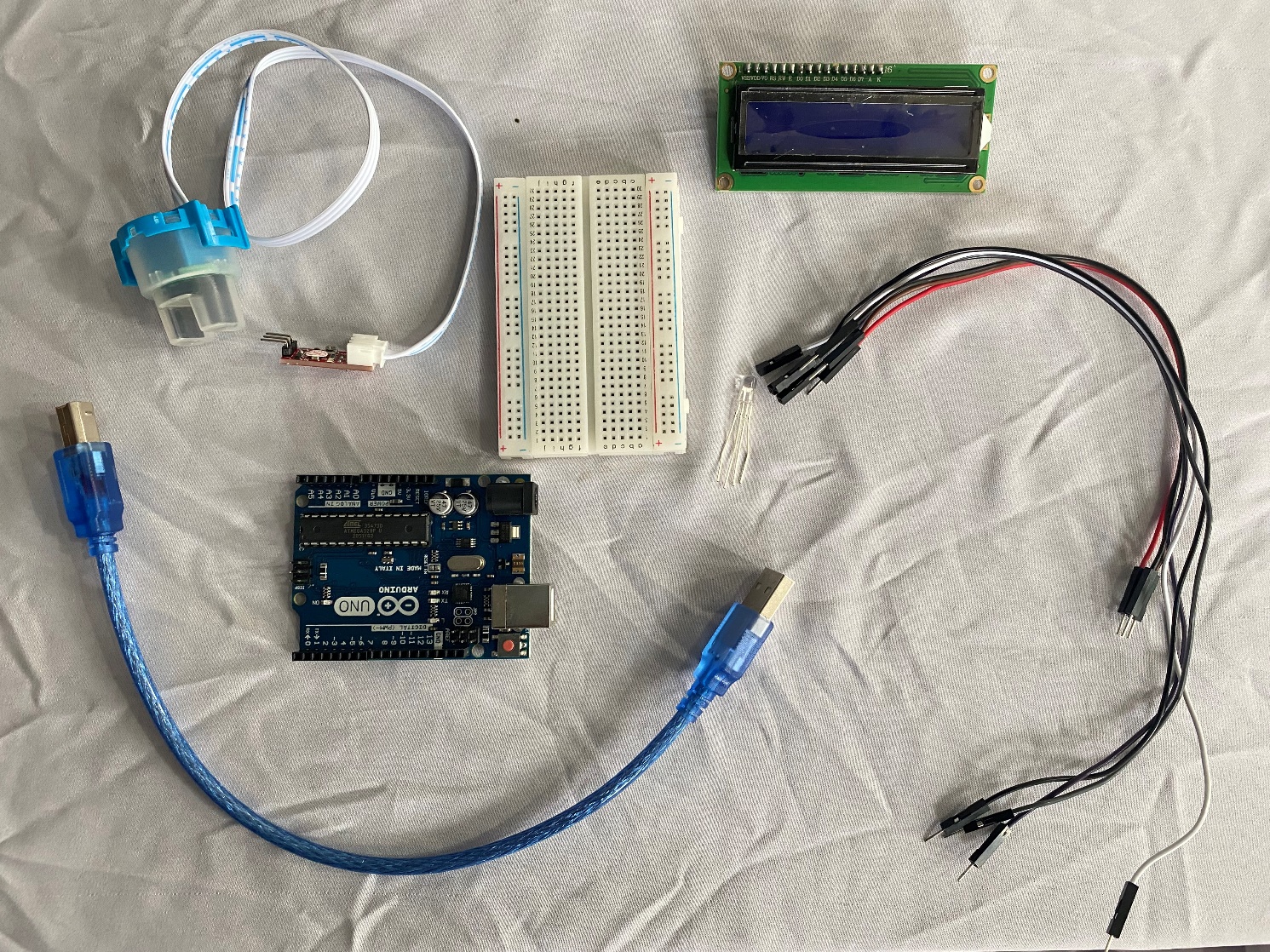
# **Components needed for making Water Quality Monitoring System**

## Hardware



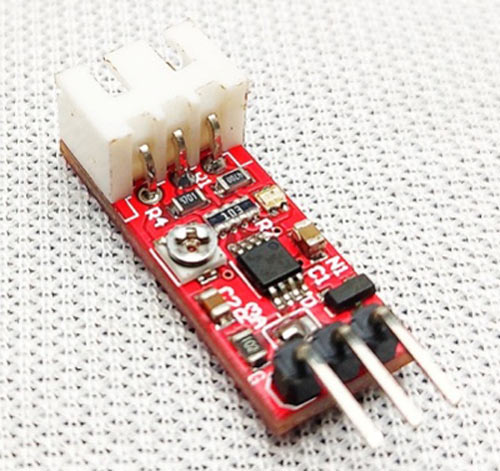
1. Turbidity Module
2. Arduino Uno (ATmega328P)
3. Breadboard
4. Jumper wires
5. RGB LED (Common Cathode)
6. 16\*2 LCD ( I2C )
7. Arduino IDE

## **Overview of Turbidity Sensor**



*Figure1: Turbidity Module*

As you can see, there are three pieces to this turbidity sensor module. A waterproof lead, a driver circuit, and a connecting wire.



*Figure 2: Driver Circuit*

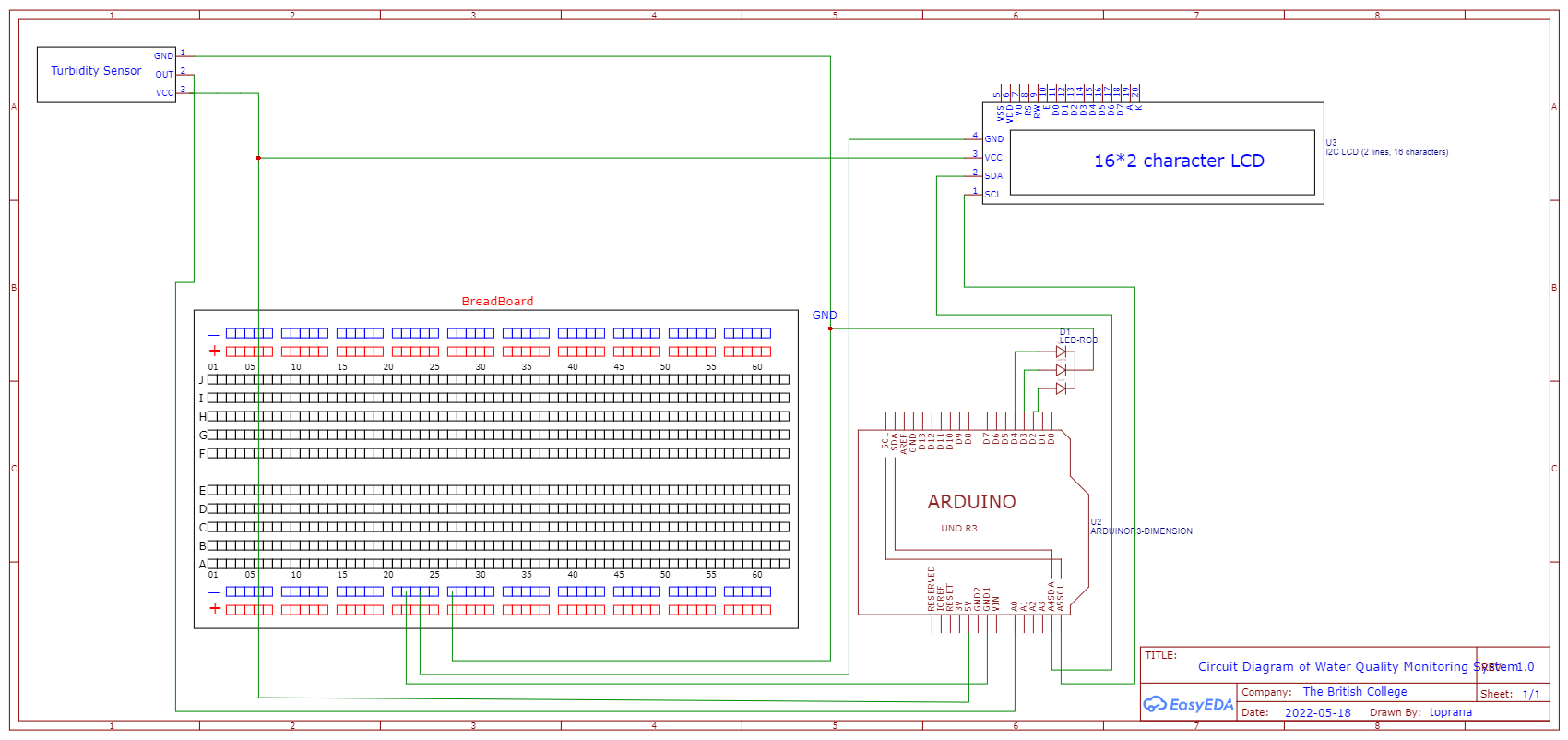
Driver circuit has three pins, VCC, ground (GND), and output. VCC connects to the 5V and ground to ground. The output of this module is an analog value i.e. it changes according to the light intensity. So connect output to A0 of Arduino.

## **Key Features of Turbidity Module**

* Operating Voltage: 5VDC
* Current: 30mA (MAX).
* Operating temperature: -30 ° C to 80 ° C.
* Compatible with Arduino, Raspberry Pi, AVR, PIC, etc.

# Circuit Diagram for Connecting a Turbidity Sensor to an Arduino

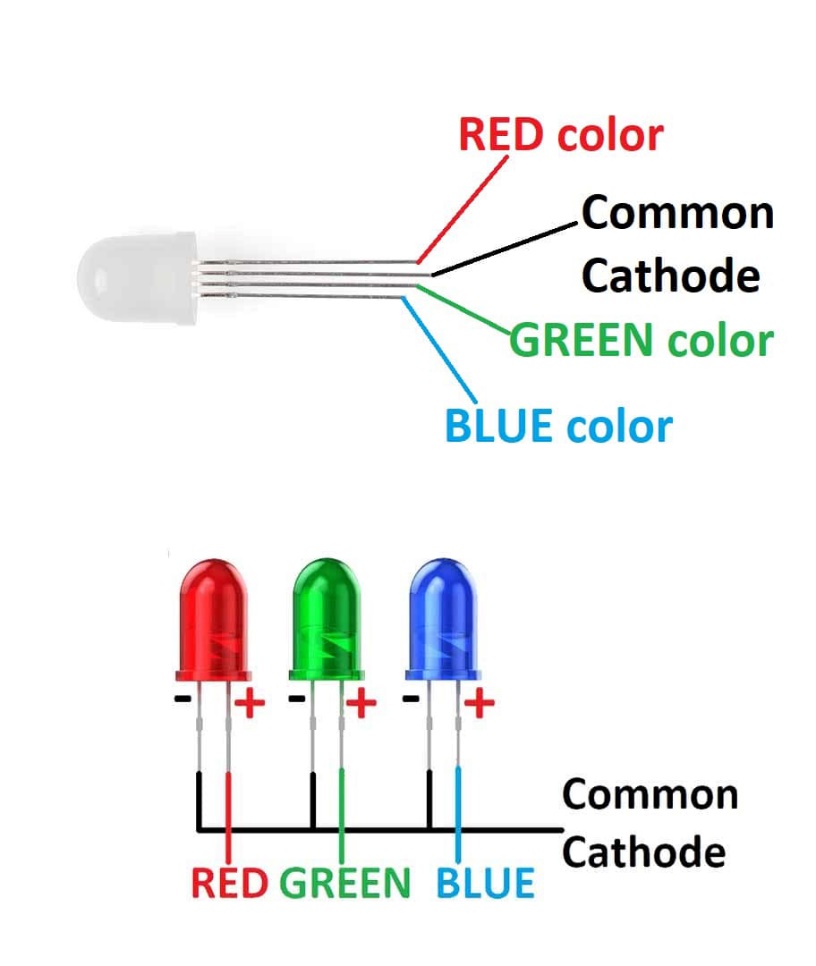
The complete schematic to connect the Turbidity sensor to Arduino is shown below. This circuit diagram contains the details connection of all components to each other.



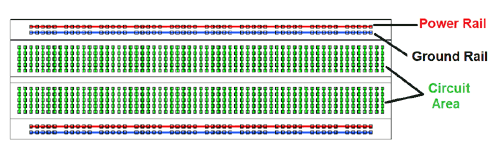
*Figure3: Schematic Diagram for connection*

This is very simple schematic diagram. The turbidity sensor output is analog, therefore it's connected to the Arduino's A0 pin. Now let us add 16x2 I2C LCD Display to display the Turbidity Value in LCD Screen. Connect the SDA & SCL Pin of I2C LCD to Arduino A4 (SDA) & A5 (SCL) Pin as shown in figure 3.

The RGB LED connected to digital pin D4, D3, and D2 of Arduino. Similarly, connect Common Cathode (GND) to GND.

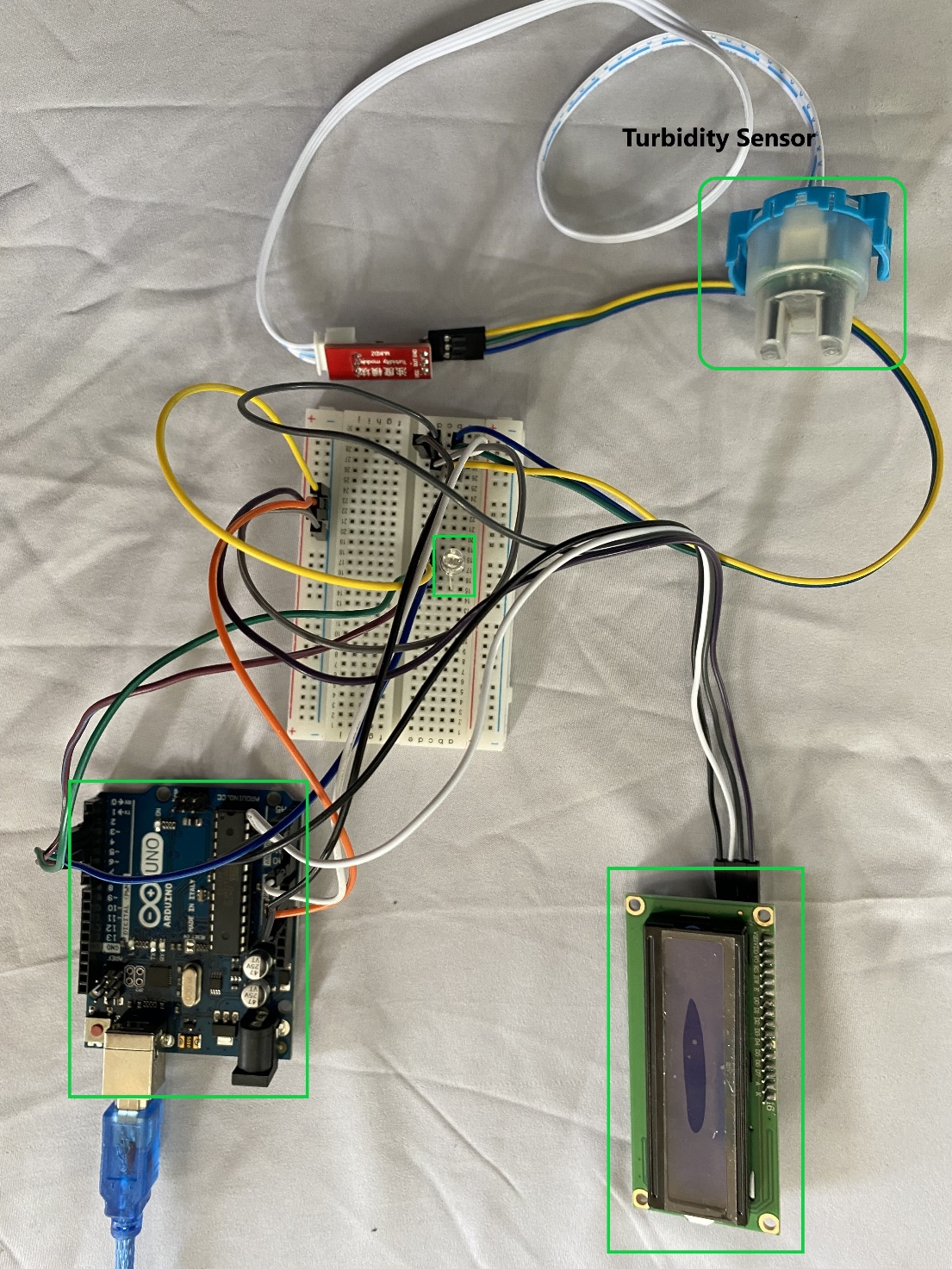


Now connect GND1 of Arduino, GND of 16\*2 LCD, GND of Turbidity Sensor, & GND of RGB LED to Ground rail of Breadboard (horizontal blue- also known as Bus Strips)



Similarly, connect 5V of Arduino Uno, VCC of Turbidity Module & VCC of 16\*2 LCD to Circuit Area of Breadboard (all align in one vertical line aka Terminal strips)

After you've completed the connections, your hardware should look like this which is given below in figure 5.



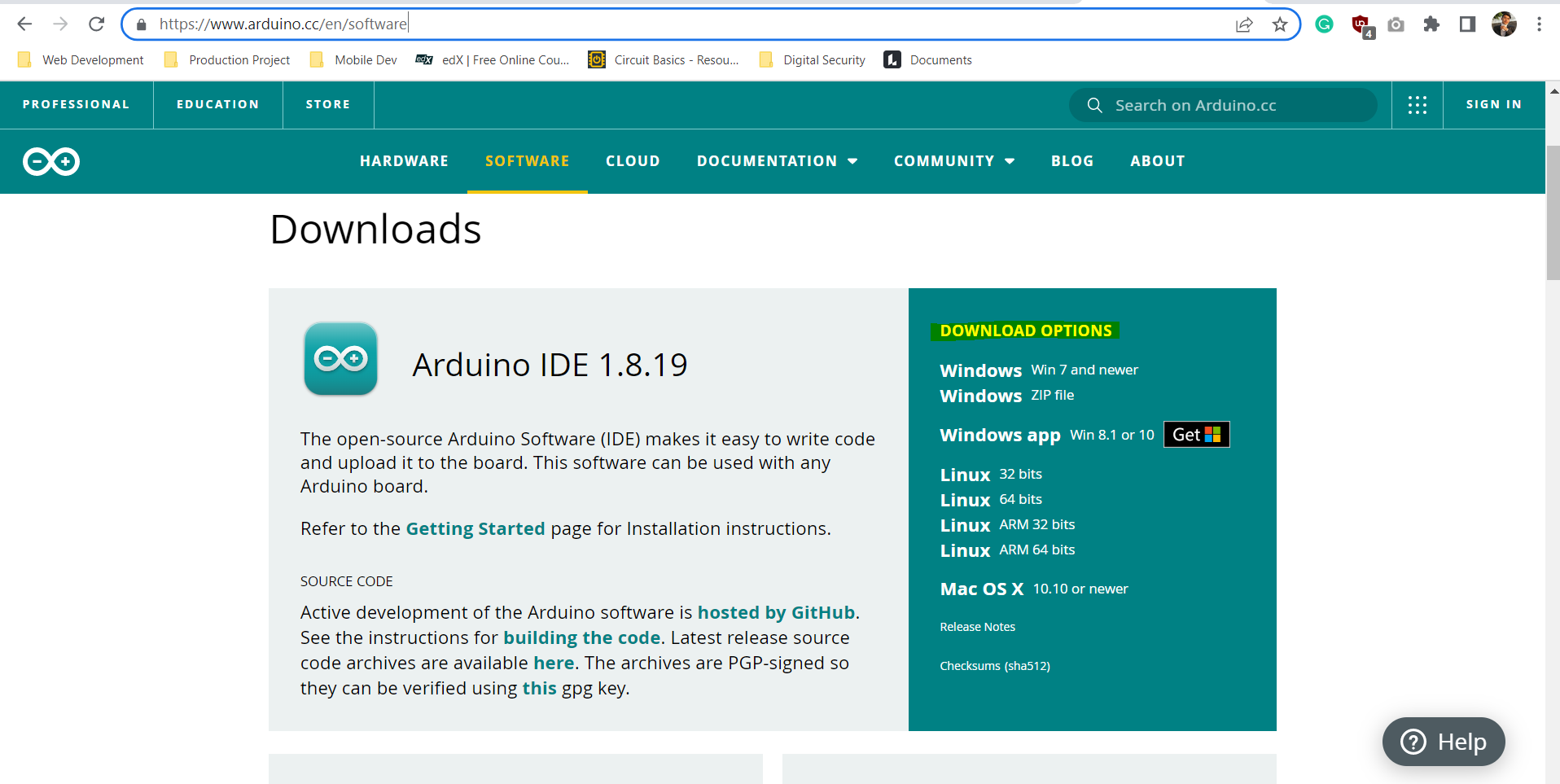
*Figure 5: Overview of the System*

## **Software Installation**

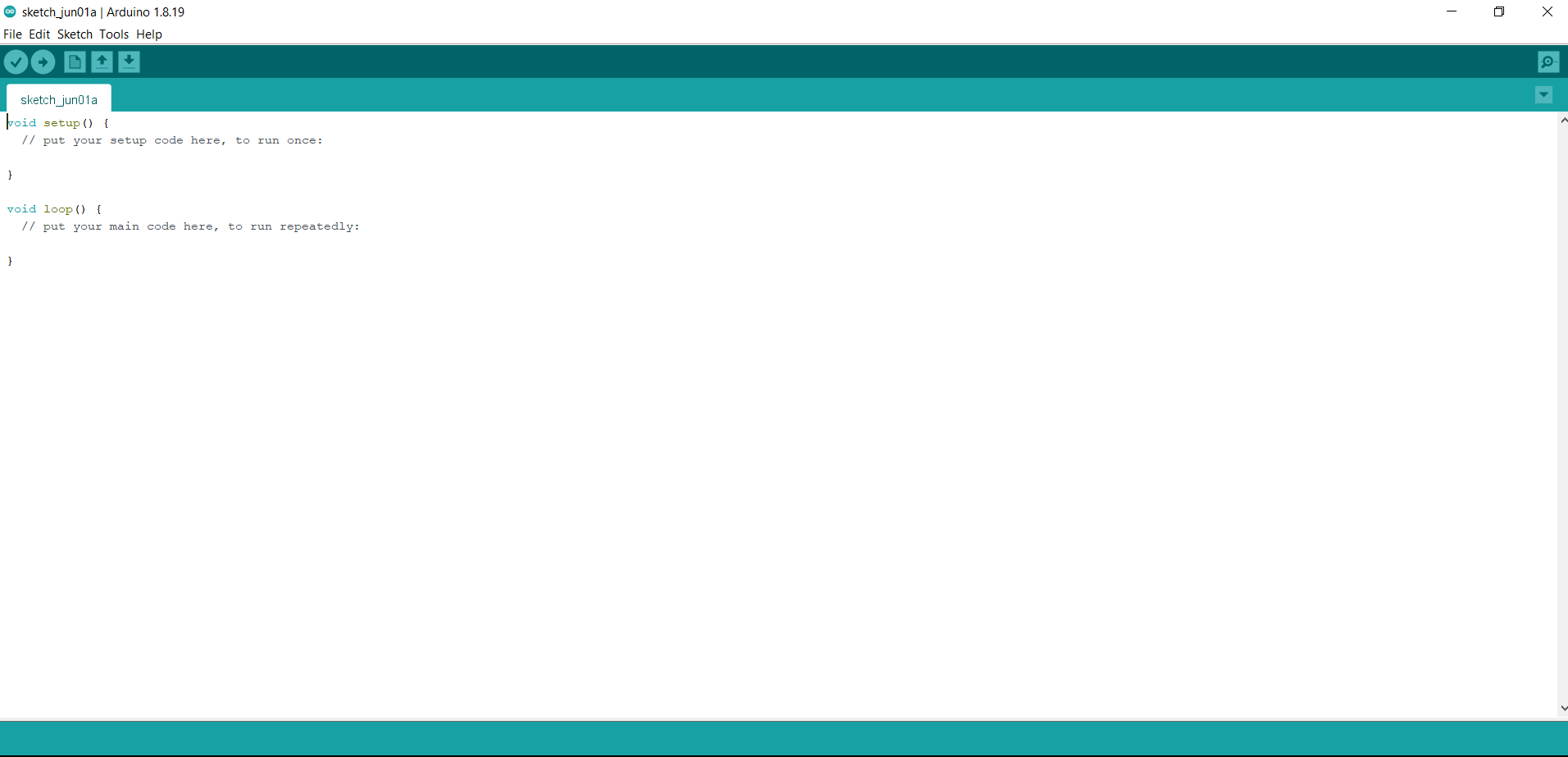
* 1. Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board. Visit the below sites to download Arduino IDE and download it.

<https://www.arduino.cc/en/software>



After downloading it, your IDE will look like this in below.



The goal is to show turbidity values ranging from 0 to 100. That example, for clear liquids, the meter should read 0 and for severely turbid liquids, it should read 100.

Simply copy the below codes to Arduino IDE.

*#include <LiquidCrystal\_I2C.h>*

*LiquidCrystal\_I2C lcd(0x27, 2, 16);*

*int sensorPin = A0;*

*void setup()*

*{*

*Serial.begin(9600);*

*lcd.init();*

*lcd.backlight();*

*pinMode(2, OUTPUT);*

*pinMode(3, OUTPUT);*

*pinMode(4, OUTPUT);*

*}*

*void loop() {*

*int sensorValue = analogRead(sensorPin);*

*Serial.println(sensorValue);*

*int turbidity = map(sensorValue, 0, 619, 100, 0);*

*delay(100);*

*lcd.setCursor(0, 0);*

*lcd.print("turbidity:");*

*lcd.print(" ");*

*lcd.setCursor(10, 0);*

*lcd.print(turbidity);*

*delay(100);*

*if (turbidity < 20) {*

*digitalWrite(2, HIGH);*

*digitalWrite(3, LOW);*

*digitalWrite(4, LOW);*

*lcd.setCursor(0, 1);*

*lcd.print(" Status: CLEAR ");*

*}*

*if ((turbidity > 20) && (turbidity < 50)) {*

*digitalWrite(2, LOW);*

*digitalWrite(3, HIGH);*

*digitalWrite(4, LOW);*

*lcd.setCursor(0, 1);*

*lcd.print(" Status: CLOUDY ");*

*}*

*if (turbidity > 50) {*

*digitalWrite(2, LOW);*

*digitalWrite(3, LOW);*

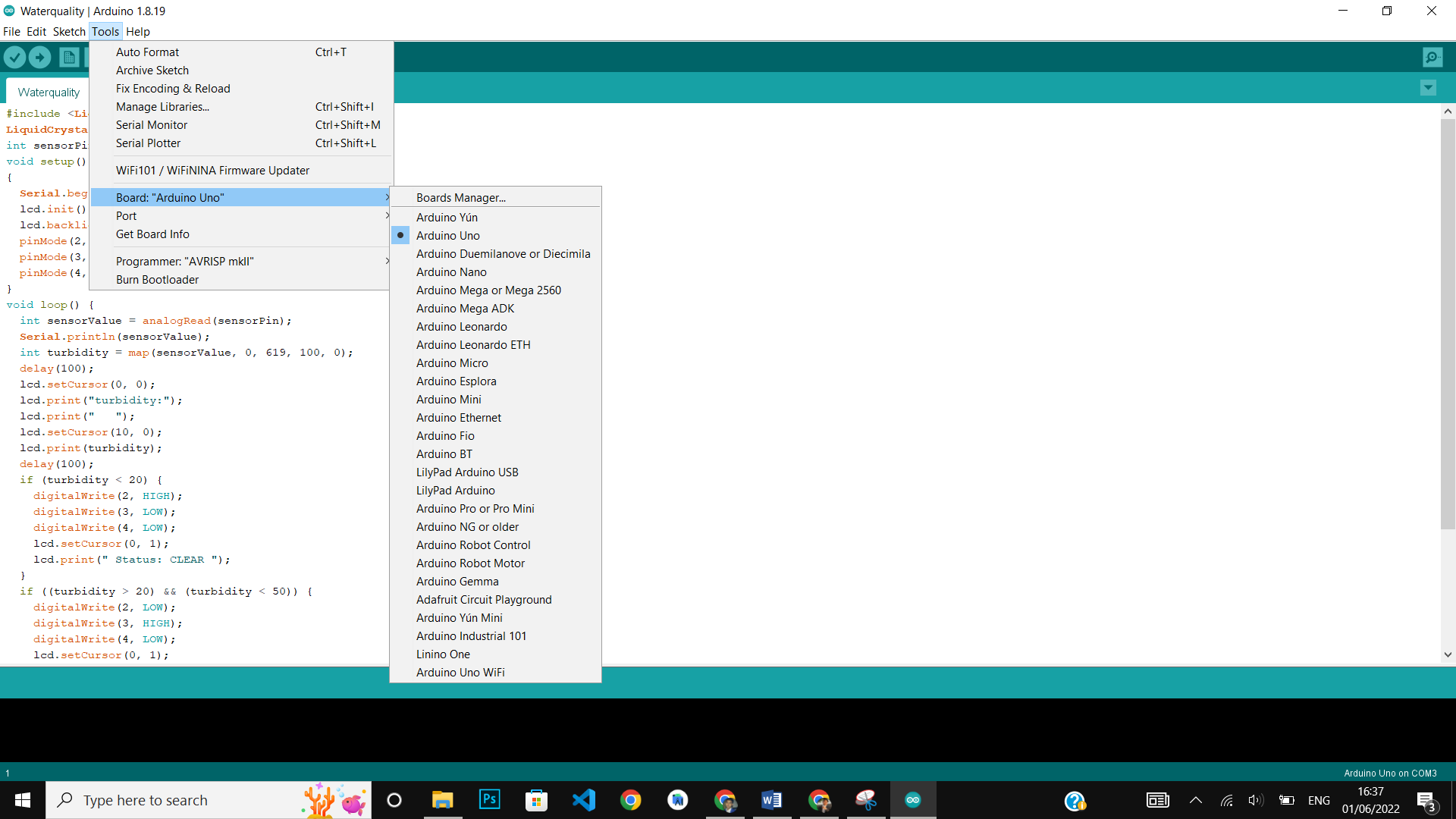
*digitalWrite(4, HIGH);*

*lcd.setCursor(0, 1);*

*lcd.print(" Status: DIRTY ");*

*}*

*}*



Check these setting before running the code.

Don’t forget to connect Arduino Uno to your PC and after that simply run your code by upload and it will work and read the turbidity as shown below:

