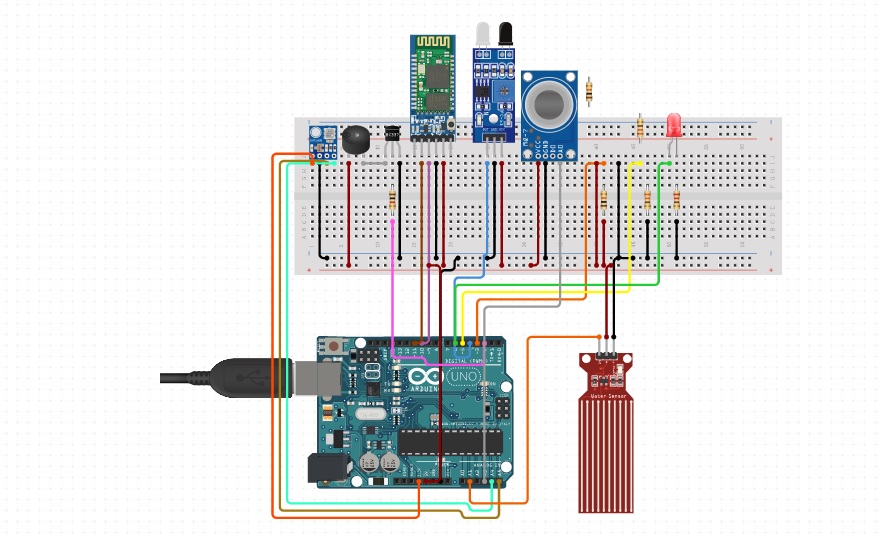
**WEATHER MONITORING SYSTEM USING ARDUINO**

Weather Monitoring has been increasingly becoming more and more important. This may be due to the conditions the nature is showing to all of us human beings down here. We are designing a Weather Monitoring System using IoT sensors, so that we can keep an eye in the changes in the weather with accurate readings and quick readings. Arduino is an open source platform available that enables us to prepare short and quick IoT projects for real time data. Arduino also has its own board – Uno, Mega and many more, and even has its own IDE for coding for the connections.

We are going to use the HC-05 Bluetooth Module to display the readings of the different sensors namely, DHT11, MQ-7, and MPX10DP, etc. Mainly, the sensors calculate the data readings and send them to the Arduino micro-controller and it respectively then shows the output on the Mobile App Screen. Arduino IDE is an Embedded C software.

IoT is the future technology and in very much development. It has high scopes in future revolutionary technology. Smart Home, Smart Dustbin, Smart Watch, and many others – these all are examples of IoT Embedded Systems (ES).

**CIRCUIT DIAGRAM**



**EXPLANATION:-**

The connections are pretty simple to connect where the sensors are connected to the Vcc, GND and their respective analog or digital input pins. Bluetooth module is connected to the ‘Tx’ & ‘Rx’ pins of the board. We even connect LEDs and a Buzzer to get active after some threshold value is reached and the value should not get further increased. The project can be implemented for many industrial, household applications which enables us to monitor the weather.

## **REQUIREMENT OF HARDWARE & SOFTWARE**

**1. DHT11 – Humidity & Temperature Sensor -**

The **DHT11** can be used as a sensor or even as a module. It performs exactly the same in both ways, but the choice depends on the user’s application and usage of the sensor or module. It calculates the Humidity and Temperature in the atmosphere and gives its readings.

**DHT11 Specifications:**

1. Operating Voltage: 3.5V to 5.5V
2. Operating current: 0.3mA (measuring) 60uA (standby)
3. Temperature Range: 0°C to 50°C



**2. MQ-7 – CO Gas Sensor –**

The **MQ-7 CO Gas Sensor** is an easy to use Carbon Monoxide Sensor. It can identify CO in the atmosphere in the range of 20 to 2000 ppm. The sensor has an accurate reading and a very quick response. CO is a tasteless, colourless, odourless, flammable gas. Carbon Monoxide poisoning is a fatal disease that has been seen in many countries.

**Specifications:**

1. Operating Voltage: 3.3 – 5V.
2. Type: Analog Sensor.
3. **Temperature:**20°C ± 2°C
4. **Humidity:**65% ± 5% RH

****

**3.MPX10DP – Pressure Sensor –**

**The MPX10DP is a Pressure Sensor for showing the environmental changes in pressure. It shows very good values and is responsive with just a little delay in its calculations.**

**Specifications:**

1. **Pressure Type: Differential**
2. **Sensitivity: 3.5mV/kPa**
3. **Operating Pressure: Min: 0kPa, Max: 10kPa**
4. **Supply Voltage: Min: 3 V, Max: 6 V**
5. **No. of Pins: 4**
6. **Operating Temperature: Min: -40 C, Max: 125 C**

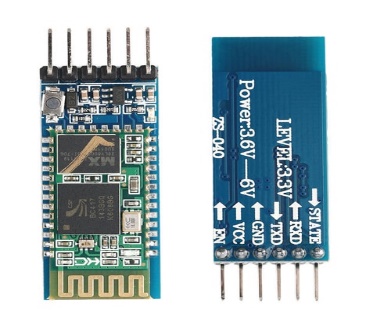
****

**4. HC-05 Bluetooth Module –**

The **HC-05 Bluetooth module** can act as a communication medium between two micro-controllers or between a micro-controller and any device with Bluetooth functionality like smartphones, laptop, etc. There are other similar modules like HC-02, HC-04, HC-06, etc.

**HC-05 Default Settings:**

1. Default Bluetooth Name: “HC-05”
2. Default Password: 1234 or 0000
3. Default Mode: Data Mode
4. Data Mode Baud Rate: 9600.



**5. MQ-2 – Gas Sensor –**

The Grove - Gas Sensor (MQ2) module is useful for gas leakage detection (home and industry). It is suitable for detecting H2, LPG, CH4, CO, Alcohol, Smoke or Propane. Due to its high sensitivity and fast response time, measurement can be taken as soon as possible. The sensitivity of the sensor can be adjusted by potentiometer.

## **Features:**

## Wide detecting scope

## Stable and long lifetime

## Fast response and High sensitivity

****

**6. Water Sensor –**

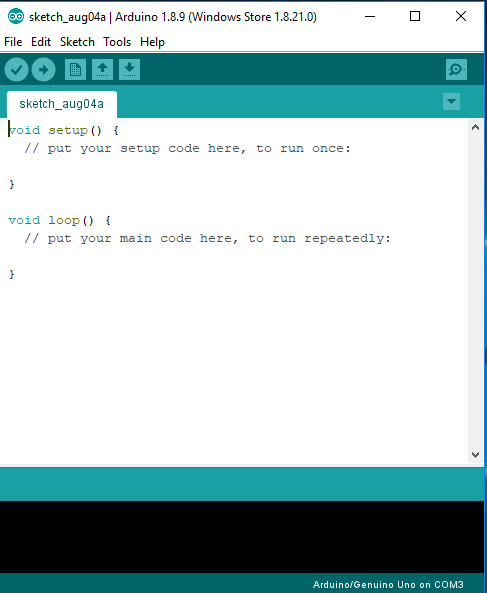
**Water sensor brick** is designed for water detection, which can be widely used in sensing rainfall, water level, and even liquid leakage. Connecting a water sensor to an Arduino is a great way to detect a leak, spill, flood, rain, etc. It can be used to detect the presence, the level, the volume and/or the absence of water. While this could be used to remind you to water your plants, there is a better Grove sensor for that. The sensor has an array of exposed traces, which read LOW when water is detected.

1. Working voltage: dc 3-5V
2. Working current: <20mA
3. Sensor type: simulation

****

**7. Arduino Software (Version 1.8.9) –**

**Arduino software** (Figure 7) is used to put the instructions of the code written to the microcontroller. The IDE is written in ‘Embedded C’ language for communicating with micro-controllers. When signal data arrives the Arduino the pin which corresponds to the particular input is set to high. We have used Arduino IDE version 1.8.9 for writing program for Arduino. It is recommended to keep on checking their official website for any updates in the software as it will help in further projects with less errors and bugs present in the software. There are two functions pre-defined in the software, which are setup () and loop () respectively. The program or condition or any line of code written in setup () section will be only executed once in the entire run of the code. Whereas, the code written in the loop section will run continuously till you remove the power supply or stop the program manually. It is very helpful for real time data applications.

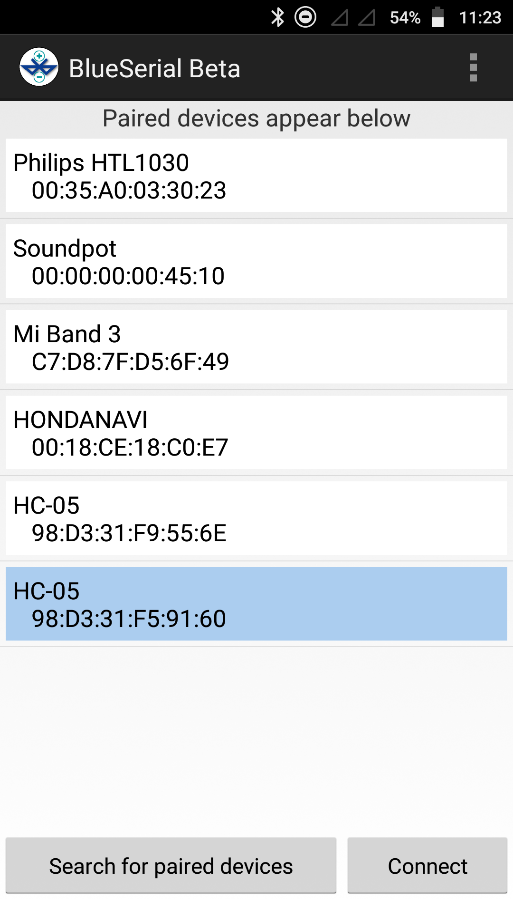


**EXPLANATION OF CODE: -**

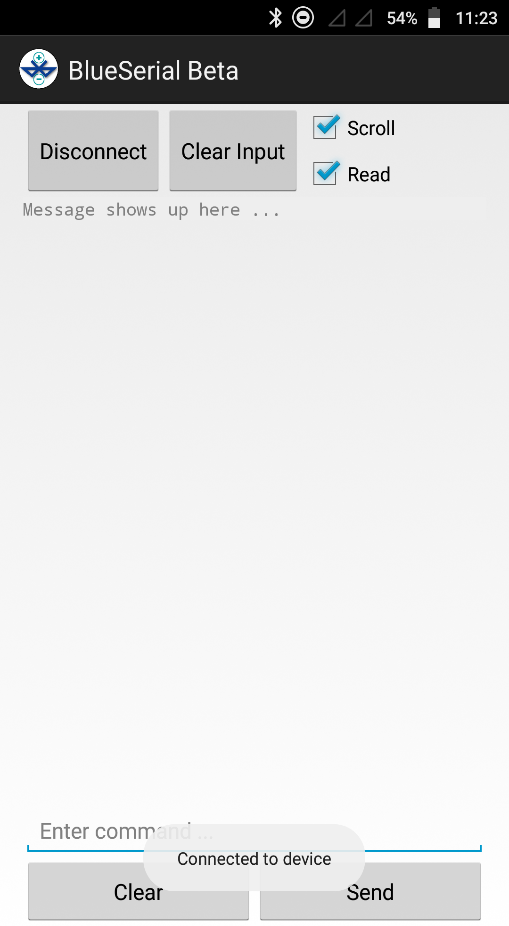
The weather monitoring system is quite accurate with the sensor readings and displaying it on the Bluetooth app on any smartphone. The code is very straightforward and easy to understand and comprehend. We have included the necessary libraries like SoftwareSerial for Bluetooth, dht for DHT11 sensor and we have defined the dht pin as well to any analog pin which you would want to connect. We then create the necessary variables for the input of the different sensors. We have also created variables to store the pin names or numbers, so that we can easily call the pin without writing and remembering the name or number again and again. We then declare the Bluetooth Rx & Tx pins for our module to get connected. In the setup () section, we declare all the pins we are going to use as input or outputs in our project. All input pins are the sensor pins and the output pins are the LED’s or buzzer pins. We also begin the serial communication at a baud rate of 9600 and begin the Bluetooth communication at the same 9600 baud rate. We can change the baud rate according to our requirements. In the loop () section, we firstly write the code for reading the different sensors i.e. using analogRead for analog sensors and digitalRead for digital sensors. We then check that the Bluetooth is available to pair and is connected or not. It will take a while for the Bluetooth to show up in your smartphone. We firstly have to pair the HC-05 Bluetooth module with our smartphone with ‘0000’ or ‘1234’ as the default password set. As we come to know that the Bluetooth is connected and is working properly, we start reading the different sensor readings and will only print it when we press a character on our smartphone Bluetooth app. In this case, we have ‘v’ as the character to be pressed in our app and the list of all the different sensor readings will come with their static names. This is the Weather Monitoring System.

**RESULT**

1. Open BlueSerial Beta App and see the nearby paired devices.



2. Connect to the HC-05 Bluetooth module.



3. Enter the character ‘v’, press ‘Send’ and you see the output.

