Phase2: Innovation

Air pollution is one of the biggest threats to the present-day environment. Everyone is being affected by air pollution day by day including humans, animals, crops, cities, forests and aquatic ecosystems. Besides that, it should be controlled at a certain level to prevent the increasing rate of global warming.

The IoT-based air pollution monitoring system would not only help us to monitor the air quality but also be able to send alert signals whenever the air quality deteriorates and goes down beyond a certain level.

The following simple flow diagram (as shown in Fig. 1) indicates the working mechanism of the IoT-based Air Pollution Monitoring System.

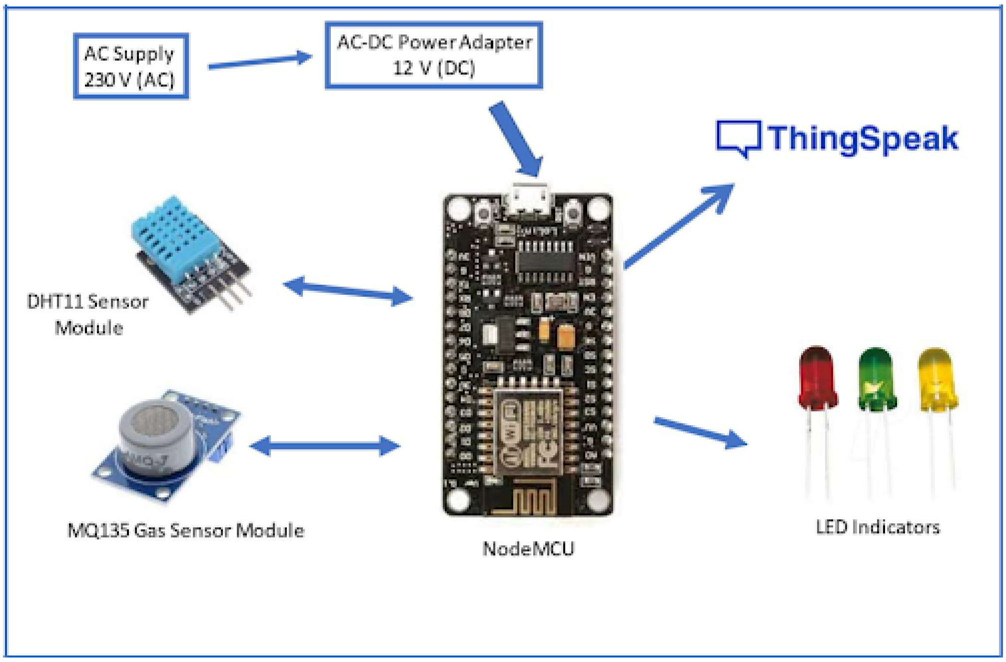

Fig.1. IoT-based Air Pollution Monitoring System

Components Used

Hardware Components

1. **NodeMCU V3**
2. **DHT11 Sensor Module**
3. **MQ-135 Gas Sensor Module**

4. Veroboard(KS100)
5. Breadboard
6. Connecting Wires
7. AC-DC Adapters
8. LEDs emitting green, yellow and red colours
9. Resistors

SOFTWARE COMPONENTS

1. ThinkSpeak Cloud
2. Arduino IDE

NodeMCU V3

NodeMCU V3 is an open-source ESP8266 development kit, armed with the CH340G USB-TTL Serial chip. It has firmware that runs on ESP8266 Wi-Fi SoC from Espressif Systems. Whilst cheaper, CH340 is super reliable even in industrial applications. It is tested to be stable on all supported platforms as well. It can be simply coded in Arduino IDE. It has a very low current consumption between 15 µA to 400 mA. The pinout Diagram of NodeMC3 is shown in Fig. 2.

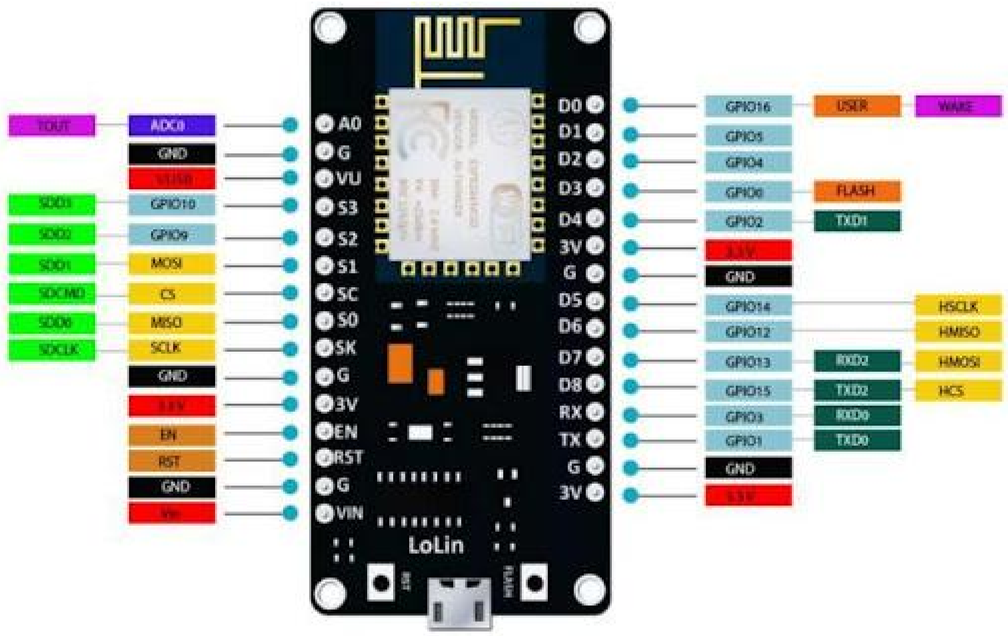

Fig. 2.1 (Pinout Diagram of NodeMCU V3)

DHT11 Sensor Module

The DHT11 is a temperature and humidity sensor that gives digital output in terms of voltage. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air.
As shown in Fig. 3 we need to supply a 5V (DC) voltage to the Vcc pin and ground it to the GND pin. The sensor output can be easily read from the Data pin in terms of voltage (in digital mode).

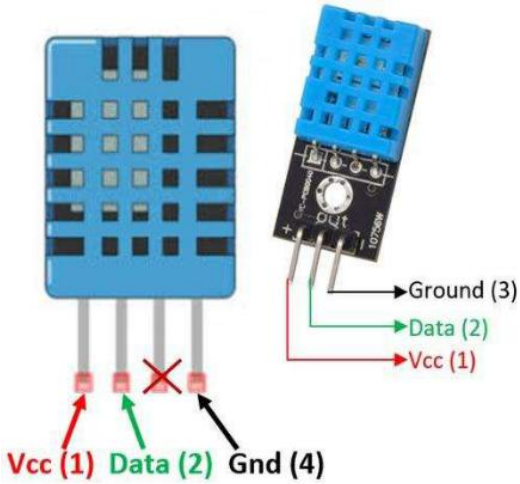

Fig 3(Pinout Diagram of DHT11sensor)

Humidity Measurement:
The humidity sensing capacitor has two electrodes with a moisture-holding substrate as a dielectric between them as shown in Fig 4. Change in the capacitance value occurs with the change in humidity levels. The IC measure processes these changed resistance values and then converts them into digital form.

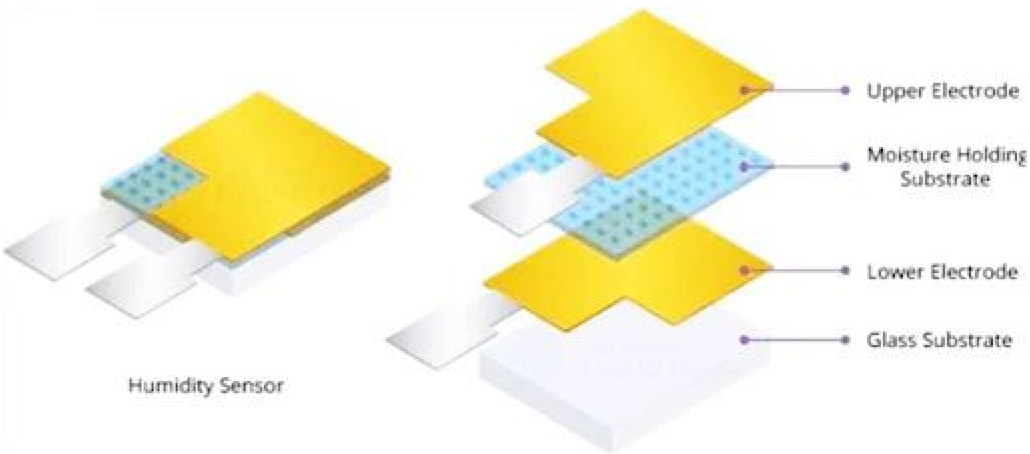

Fig 4(The structure of the humidity sensor)

MQ-135 Gas Sensor Module

The material of MQ135 is SnO_2, it is a special material: when exposed to clean air, it is hardly being conducted, however, when put in an environment with combustible gas, it has **pretty performance of conductivity**

In a normal environment, the environment that doesn't have detected gas sets the sensor's output voltage as the reference voltage, the analog output voltage will be about 1V when the sensor detects gas, and harmful gas concentration increases by 20ppm per voltage increase by 0.1V.

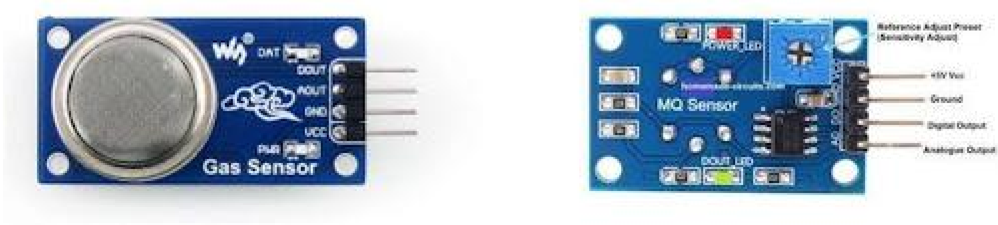

Fig 5 (MQ-135 Gas Sensor Module)

ThingSpeak Cloud

ThingSpeak is open-source software written in Ruby which allows users to communicate with internet-enabled devices. It facilitates data access, retrieval, and logging of data by providing an API to both the devices and social network websites.

ThingSpeak was originally launched by ioBridge in 2010 as a service in support of IoT applications. ThingSpeak has integrated support from the numerical computing software MATLAB Fig 6.

ThingSpeak Cloud from MathWorks, allowing ThingSpeak users to analyze and visualize uploaded data using MATLAB without requiring the purchase of a MATLAB license from MathWorks.

Working Procedures

NodeMCU plays the main controlling role in this project. It has been programmed in a manner, such that, it senses the sensory signals from the sensors and shows the quality level via led indicators. The DHT11 sensor module is used to measure the temperature and the humidity of the surroundings.With the help of the MQ-135 gas sensor module, air quality is measured in ppm. These data are fed to the ThinkSpeak cloud over the internet. We have also provided LED indicators to indicate the safety levels.

STEP 1. Firstly, the calibration ofthe MQ-135 gas sensor module is done. The sensor is set to preheat for 24 minutes. Then the software code is uploaded to the NodeMCU followed by the hardware circuit to calibrate the sensor has been performed.

STEP 2. Then, the DHT11 sensor is set to preheat for 10 minutes.

STEP 3. The result of calibration found in STEP 1 is used to configure the final working code.

STEP 4. The final working code is then uploaded to the NodeMCU.

STEP 5. Finally, the complete hardware circuit is implemented.

Final Hardware Model

The following steps were performed to execute the project

STEP 1 : The Vcc pin of the MQ-135 gas sensor module and DHT11 sensor module was connected via Veroboard with an adapter delivering around 5V.

STEP 2 : The Gnd pin of the MQ-135 gas sensor module, DHT11 sensor module and the cathode of the LED indicators was connected via Veroboard with the Gnd pin of the NodeMCU
.

STEP 3 : The analog DATA pin of the MQ-135 gas sensor module was connected with the A0 Pin of the NodeMCU.

STEP 4 : The DATA pin of the DHT11 sensor module was connected with the D0 pin of the NodeMCU**.**

STEP 5 : The anode of the three LED indicators (green, yellow, and red) were connected to the D2, D3, and D4 pins of the NodeMCU respectively.

STEP 6 : The software code to execute the project was then uploaded to the NodeMCU.

STEP 7 : The setup was then powered with 9V DC via AC-DC adapter.

It can be now turned ON/OFF as per the requirements. Fig 7 represents the circuit diagram of the setup.

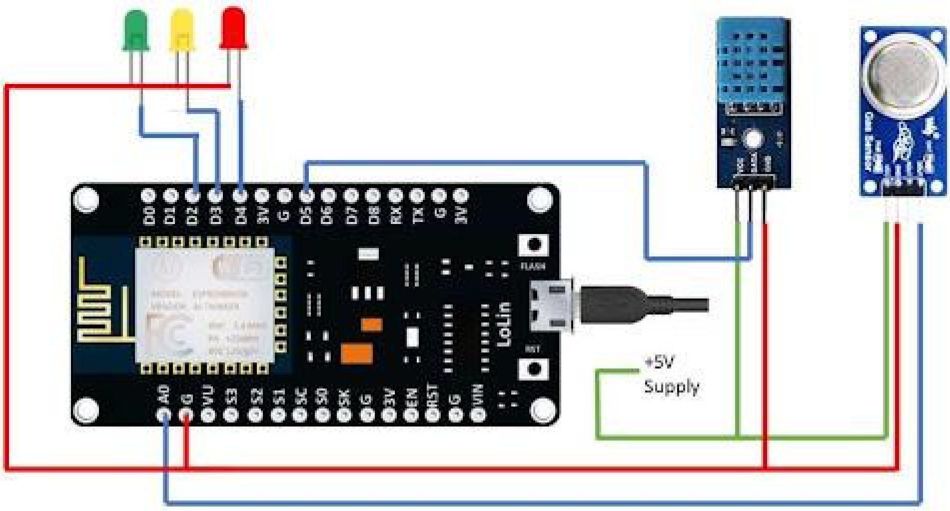

Fig. 7 (Circuit Diagram of the setup)