PROJECT: AIR QUALITY MONITORING

PHASE-2:INNOVATION

INTRODUCTION:

Implementing air quality monitoring using ESP32 and wokwi. Here is the sample code and step by step procedure to implement this project. Combining the MQ135 gas sensor and the DHT sensor (DHT22, for instance) for an Internet of Things (IoT) air quality monitoring project can provide real-time data that can be accessed remotely. Here's a high-level overview of how to create an IoT air quality monitoring system using these sensors:

Hardware Required:

1.ESP32 board

2.Air Quality sensors-MQ135

4.Temperature, Humidity using DHT sensor

4.Timer

5.Wokwi account

6.Internet connection for Wokwi stimulation

Step 1: Setting up Wokwi environment

1.Visit Wokwi website that is https://wokwi.com/.

2. Create an account if you don't have one already.

3. Once logged in, click on "Create a new simulation".

4. Select "ESP32 Kit" as your board.

Step 2: Connecting the Hardware in Wokwi

1. In the Wokwi simulator, you can add components like the ESP32, MQ135 for air quality, and DHT senor for temperature and humidity by dragging them from the components panel onto the virtual breadboard.

2. Connect the components using virtual jumper wires. Connect the power and ground pins

appropriately.

3. Connect the air quality sensor(MQ135) and DHT senor for temperature and Humidity to an analog input pin on the ESP32.

4. Connect the buzzer/speaker to a digital output pin on the ESP32 kit.

Step 3: Writing the Code

Here is the simple Arduino code for Air Quality Monitoring. This code reads temperature, humidity, and air quality from virtual sensors and prints the values to the serial monitor.

```
#include <Adafruit_Sensor.h>
#include <DHT.h>

#define DHTPIN 2        // Pin connected to the DHT sensor
#define DHTTYPE DHT22    // DHT sensor type (DHT11, DHT22, or DHT21)
#define MQ135PIN A0      // Analog pin for MQ135 sensor

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  dht.begin();
}

void loop() {
  float temperature = dht.readTemperature(); // Read temperature in °C
  float humidity = dht.readHumidity();      // Read humidity in %

  // Read analog value from MQ135 sensor
  int airQuality = analogRead(MQ135PIN);

  // MQ135 sensor provides a voltage value; convert it to PPM (parts per million)
  // Replace these values with the calibration data for your specific sensor.
  // The values below are for reference and might not be accurate for your sensor.
  float sensorVoltage = (5.0 / 1024.0) * airQuality;
  float ppm = 60 * pow(sensorVoltage / 5.0, -2.5);
  Serial.print("Temperature (°C): ");
  Serial.println(temperature);
  Serial.print("Humidity (%): ");
  Serial.println(humidity);
  Serial.print("Air Quality (PPM): ");
  Serial.println(ppm);

  // You can add code here to send data to a display, IoT platform, or data logger.
  // Make sure to replace the calibration data with values appropriate for your sensor.

  delay(5000);  // Delay for 5 seconds (adjust as needed)
}
```

Step 4: Stimulating the Air quality monitoring

1. In the Wokwi simulator, click on the "Start Simulation" button to run your project.

2. Simulate the air quality sensor by clicking on it and changing its value to simulate temperature and humidity.

3. Observe the reading of the temperature and humidity.

Step 5: Real Implementation (optional)

To implement this project in the real world, you will need to:

1. First Purchase the necessary hardware components ESP32 kit and virtual sensors (DHT sensor for temperature and humidity and MQ135 for air quality).

2. Set up your ESP32 development environment (Arduino IDE or PlatformIO).

3. Connect the real hardware following the same connections as in the Wokwi simulation.

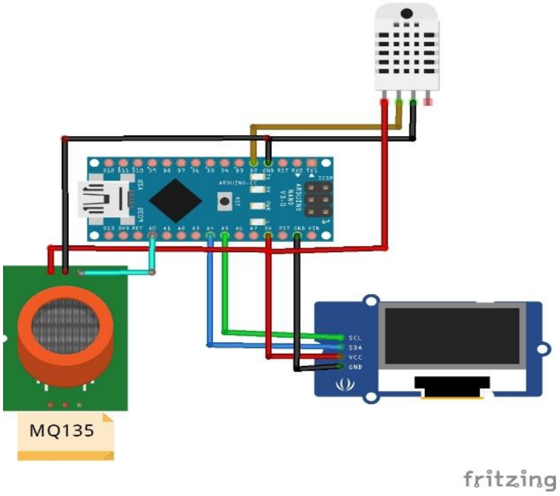
4. Upload the Arduino code to your ESP32.

5. Deploy the system in the area you want to monitor for temperature and humidity.

6. Consider adding additional features like sending alerts to a server or smartphone.

7. Give internet acces and note down the reading of Temperature and Humidity

CIRCUIT DIAGRAM:



APPLICATIONS:

They provide valuable data on various aspects of air quality, including the presence of harmful gases, temperature, and humidity. Here are some applications of the MQ135 and DHT sensors in air quality monitoring:

1. **Indoor Air Quality (IAQ) Monitoring:**

- Home and office environments can suffer from poor air quality due to factors like volatile organic compounds (VOCs) from paints and cleaning products, CO2 buildup, and temperature and humidity fluctuations. The combination of the MQ135 and DHT sensors can help monitor these parameters to maintain a healthy indoor environment.

2. **Smart HVAC Control:**

- Air conditioning and heating systems can benefit from real-time monitoring of temperature and humidity to maintain comfort and energy efficiency. The MQ135 can also detect the presence of harmful gases, enabling the system to take action in case of gas leaks.

3. **Environmental Monitoring:**

- These sensors are often used in environmental monitoring projects. Researchers and environmentalists use them to monitor air quality in various locations, such as parks, forests, and urban areas. It helps in assessing the impact of pollution on the environment.

4. **Industrial Safety:**

- In industrial settings where the presence of harmful gases can pose significant risks, the MQ135 sensor can provide early detection and warnings. Temperature and humidity data can be critical for safety in certain industrial processes.

5. **Air Purifiers and Ventilation Systems:**

- Air purifiers can be equipped with MQ135 sensors to measure air quality and activate the purification process when pollution levels are high. Ventilation systems in factories, offices, and homes can adjust airflow based on temperature and humidity readings.

6. **Greenhouses and Agriculture:**

- Greenhouses require precise control over temperature and humidity for optimal plant growth. MQ135 sensors can detect gases that could be harmful to plants, while DHT sensors can provide climate data for controlling the environment.