Wi-Fi Air Quality Monitor using ESP8266 & Arduino

Project Overview

This **Wi-Fi Air Quality Monitor** uses sensors to measure **air pollutants (PM2.5, CO2, VOCs, and temperature/humidity)** and sends the data to a **web dashboard** for remote monitoring. It can help track air pollution levels indoors or outdoors, making it useful for schools, offices, and homes.

Objectives

✅ **Measure real-time air quality parameters** (CO2, PM2.5, VOCs, temperature, humidity).
✅ **Send data wirelessly** to a cloud dashboard (ThingSpeak, Blynk, Firebase).
✅ **Display air quality on an OLED/LCD screen** for local monitoring.
✅ **Trigger alerts** when pollution levels exceed safe limits.

Components Required

1. **ESP8266 (NodeMCU or ESP-01)** – Microcontroller with Wi-Fi capability.
2. **MQ135 Gas Sensor** – Measures air quality (CO2, NH3, alcohol, benzene, smoke).
3. **DHT11/DHT22 Sensor** – Measures temperature and humidity.
4. **OLED Display (0.96" I2C) / 16x2 LCD with I2C** – Displays real-time air quality data.
5. **Jumper Wires & Breadboard** – For circuit connections.
6. **5V Power Supply (USB Adapter/3.3V Regulator)** – Powers the ESP8266 and sensors.

How the System Works

1. **Sensors collect air quality data** (CO2, PM2.5, VOCs, temperature, and humidity).
2. **ESP8266 processes the data** and uploads it to the cloud.
3. **Users access real-time air quality reports** on a web dashboard.
4. The **OLED/LCD screen displays live air quality readings** locally.
5. If air pollution exceeds safe limits, **alerts (buzzer, LED, notifications) are triggered**.

Circuit Diagram

Connections for MQ135 Sensor:

MQ135 ESP8266 (NodeMCU)
VCC 3.3V / 5V
GND GND
A0 A0 (Analog Input)

Connections for DHT11 Sensor:

DHT11 ESP8266
VCC 3.3V / 5V
GND GND
DATA D4

Connections for OLED Display (I2C):

OLED (I2C) ESP8266
VCC 3.3V / 5V
GND GND
SDA D2
SCL D1

Arduino Code for Wi-Fi Air Quality Monitor

This code reads air quality data and uploads it to **ThingSpeak**.

```
#include <ESP8266WiFi.h>
#include <DHT.h>

#define DHTPIN D4
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

const char* ssid = "Your_WiFi_Name";
const char* password = "Your_WiFi_Password";
const char* server = "api.thingspeak.com";
String apiKey = "Your_ThingSpeak_API_Key";

WiFiClient client;
int MQ135_PIN = A0;

void setup() {
    Serial.begin(115200);
    dht.begin();
```

```
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(1000);
        Serial.println("Connecting to Wi-Fi...");
    }
    Serial.println("Connected!");

}

void loop() {
    float temp = dht.readTemperature();
    float hum = dht.readHumidity();
    int air_quality = analogRead(MQ135_PIN);

    if (isnan(temp) || isnan(hum)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    Serial.print("Temp: ");
    Serial.print(temp);
    Serial.print("°C, Humidity: ");
    Serial.print(hum);
    Serial.print("%, Air Quality: ");
    Serial.println(air_quality);

    if (client.connect(server, 80)) {
        String data = "GET /update?api_key=" + apiKey + "&field1=" +
String(temp) + "&field2=" + String(hum) + "&field3=" + String(air_quality);
        client.print(data);
        client.stop();
    }

    delay(30000); // Upload data every 30 seconds
}
```

How to Use the System

1. **Upload the code** to ESP8266.
2. **Connect to Wi-Fi** and check the IP address in the Serial Monitor.
3. **Visit the ThingSpeak dashboard** to view real-time air quality data.
4. The **OLED/LCD screen** will display air quality readings locally.
5. If pollution levels are too high, **LED/Buzzer alerts will activate**.

Features & Benefits

✅ **Wi-Fi Connectivity** – Monitor air quality remotely.
✅ **Real-Time Updates** – Sensor data updates every 30 seconds.
✅ **Cloud Storage** – Logs air quality for historical analysis.

✅ **Local Display** – View readings without needing an app.
✅ **Air Quality Alerts** – Buzzer, LED, or notifications for unsafe levels.

Future Enhancements

SMS/Email Alerts when air pollution exceeds safe limits.
Mobile App Integration for real-time monitoring.
Battery-Powered Version for portable use.
CO & NO2 Sensors for advanced air quality analysis.

Would you like me to add **mobile app support** or **email alerts** for air pollution warnings?