Experiment 3

Student Name: Pranjal Singh **UID: 22BCS13041**
Branch: CSE **Section: FL_IOT_601/A**
Semester: 6th **DOP: 25/01/25**
Subject: CLOUD IOT EDGE ML LAB **Subject Code: 22CSP-367**

Aim: Monitor air quality using a gas sensor (MQ135) and display the data on ThingSpeak.

Objective: The objective of this experiment is to simulate a cloud scenario using MATLAB and run an algorithm to analysis the air quality variations within that environment.

Hardware Required:

1. MQ135 gas sensor
2. ESP8266/NodeMCU (or any microcontroller with Wi-Fi capability)
3. Breadboard and jumper wires
4. Power supply (5V for the sensor and microcontroller)
5. ThingSpeak account (free API key)

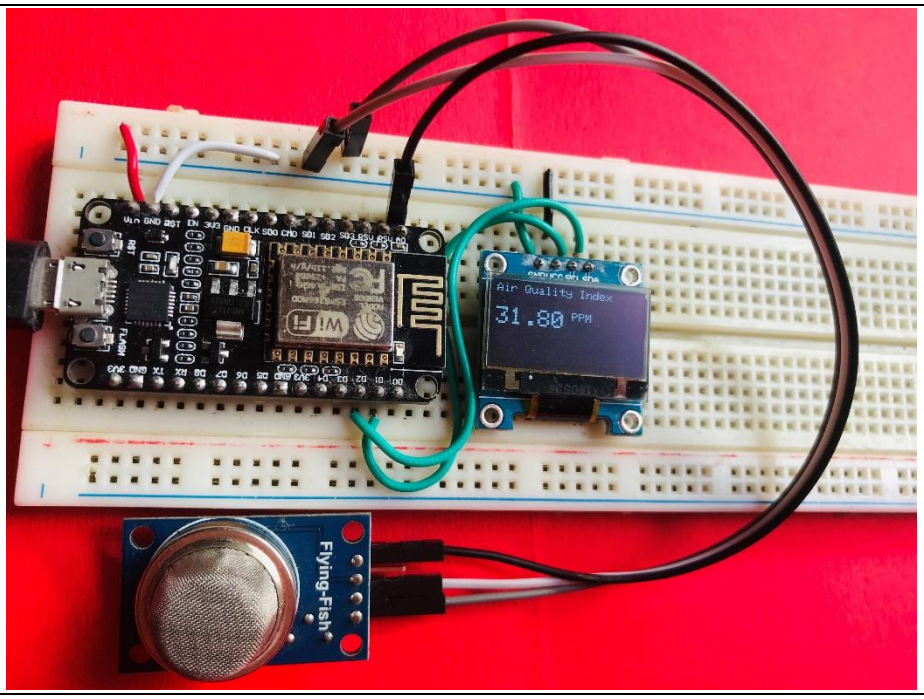
Script:

```
#include <ESP8266WiFi.h>
#include <SPI.h>
#include <Wire.h>
#include "MQ135.h"
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

String apiKey = "I1Q1QP2YFU4E22PV";
const char *ssid = "Alexahome";
const char *pass = "123456789";
const char* server = "api.thingspeak.com";

WiFiClient client;

void setup()
{
  Serial.begin(115200);
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  delay(10);
```

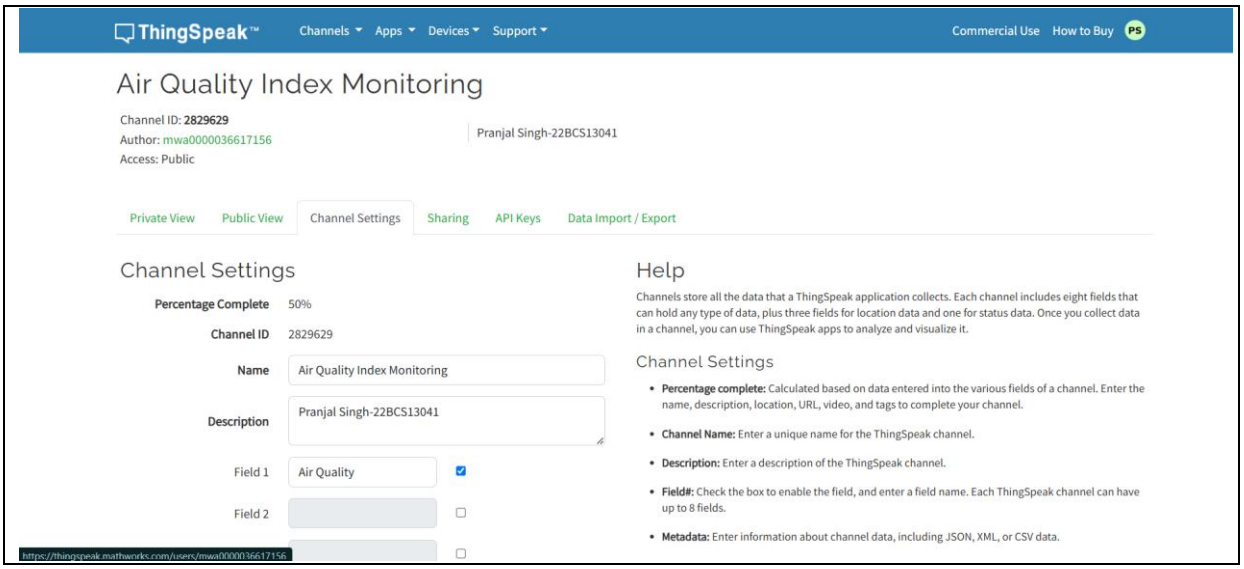
```
  Serial.println("Connecting to ");
  Serial.println(ssid);
  display.clearDisplay();
  display.setCursor(0,0);
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.println("Connecting to ");
  display.setTextSize(2);
  display.print(ssid);
  display.display();
  WiFi.begin(ssid, pass);

  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
    Serial.println("");
    Serial.println("WiFi connected");

    display.clearDisplay();
    display.setCursor(0,0);
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.print("WiFi connected");
    display.display();
    delay(4000);
}

  void loop()
  {
    MQ135 gasSensor = MQ135(A0);
    float air_quality = gasSensor.getPPM();
    Serial.print("Air Quality: ");
    Serial.print(air_quality);
    Serial.println("  PPM");
    Serial.println();

    display.clearDisplay();
    display.setCursor(0,0);
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.println("Air Quality Index");


    display.setCursor(0,20);
    display.setTextSize(2);
    display.setTextColor(WHITE);
    display.print(air_quality);
    display.setTextSize(1);
```

```
  display.setTextColor(WHITE);
   display.println(" PPM");
   display.display();
   if (client.connect(server, 80))
  {
   String postStr = apiKey;
   postStr += "&field1=";
   postStr += String(air_quality);
   postStr += "r\n";
   client.print("POST /update HTTP/1.1\n");
   client.print("Host: api.thingspeak.com\n");
   client.print("Connection: close\n");
   client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");
   client.print("Content-Type: application/x-www-form-urlencoded\n");
   client.print("Content-Length: ");
   client.print(postStr.length());
   client.print("\n\n");
   client.print(postStr);

   Serial.println("Data Send to Thingspeak");
  }
   client.stop();
   Serial.println("Waiting...");

   delay(2000);
}
```

Connection/Procedure Diagrams:



Output:



Result:

Once the code is uploaded you can open serial monitor. The Nodemcu will first start connecting to wifi network. All the happening can se observed on Serial Monitor.

Conclusion/Analysis:

The implementation of an air quality monitoring system using the MQ135 gas sensor and ThingSpeak provides a cost-effective solution for real-time pollution tracking, enabling users to monitor harmful gases like CO_2 and NH_3 remotely. The system ensures accurate data collection, cloud-based visualization, and threshold-based alerts, with future improvements focusing on enhanced sensor calibration, predictive analytics, and scalable deployment for smarter environmental monitoring.

Learning Outcomes:

i. **Understanding Air Quality Monitoring** – Gained knowledge of how gas sensors like MQ135 detect pollutants and contribute to environmental monitoring.

ii. **Hands-on IoT Implementation** – Learned to integrate the MQ135 sensor with ThingSpeak for real-time data collection and visualization

iii. **Cloud-Based Data Management** – Developed skills in sending, storing, and analyzing air quality data using IoT platforms.

iv. **Threshold-Based Alerts** – Understood how to set pollution thresholds and automate alerts for hazardous air conditions.

v. **Future Enhancements & Scalability** – Explored potential improvements like predictive analytics, advanced calibration, and large-scale deployment.