Air Quality Monitoring System

Phase_3 : Project implementation part 1

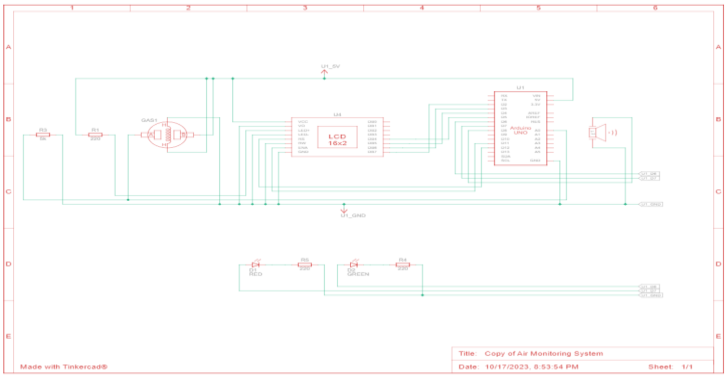
Abstract:

In this preliminary stage of development, the fundamental groundwork has been completed. We have opted for the Arduino Mega as the central microcontroller for our project. To quantify the presence of specific gases in the atmosphere, particularly measuring parts per million (ppm), we have incorporated the MQ-135 gas sensor into our setup. The decision to utilize the Arduino Mega stems from its robust capabilities and versatile features, making it well-suited for the intended application. Its ample processing power and numerous input/output pins provide the necessary foundation to support our gas sensor and other associated components. The MQ-135 gas sensor, a pivotal element in our system, serves the purpose of detecting and quantifying various gases, including but not limited to carbon dioxide (CO_2), ammonia (NH_3), and methane (CH_4). By employing this sensor, we aim to gain precise measurements of gas concentrations in the air, which is vital for various environmental monitoring and safety applications. This initial phase sets the stage for the subsequent stages of development, where we will delve deeper into the intricacies of code development, calibration, and integration of the sensor data into a comprehensive monitoring and control system.

Components Used:

Name	Quantity	Component
U1	1	Arduino Uno R3
U4	1	LCD 16 x 2
R1	1	220 Ω Resistor
GAS1	1	Gas Sensor
R3	1	5 kΩ Resistor
PIEZO1	1	Piezo
D1	1	Red LED
R4 R5	2	220 Ω Resistor
D2	1	Green LED

Circuit Diagram:



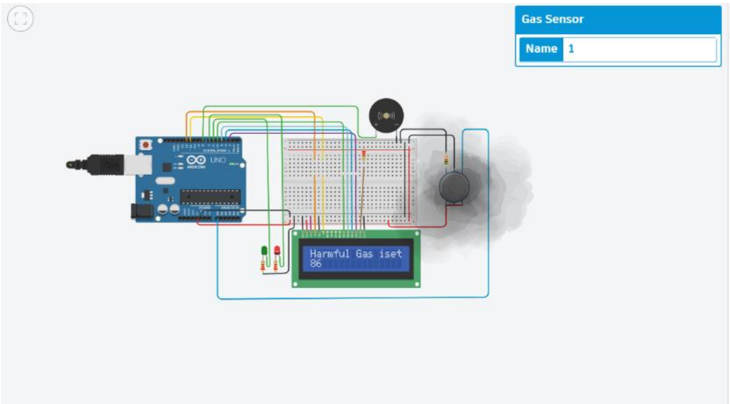
Sample Code:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

int sensor = A0;
int val = 0;
int limit = 40;

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

  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT);
  digitalWrite(6, LOW);
  digitalWrite(7, LOW);



  lcd.begin(16, 2);
```

```
  lcd.print("Air Monitoring");
  lcd.setCursor(0, 1);
  lcd.print("System");
  delay(500);
  lcd.clear();
}



void loop() {

  val = analogRead(sensor);
  val = map(val, 306, 750, 0, 100);
  Serial.println(val);
  lcd.setCursor(0,0);
  lcd.print("Particle contnet in air is ");
  lcd.setCursor(0, 1);
  lcd.print(val);


    if (val > limit) {

      lcd.setCursor(0, 0);
      lcd.print("Harmful Gas is");
      lcd.setCursor(0, 1);
      lcd.print(val);
      digitalWrite(7, HIGH);
      digitalWrite(6, LOW);
      tone(8, 1000);
      delay(100);
```

```
    noTone(8);
    delay(100);
  }

  else {
    lcd.clear();
    noTone(8);
    digitalWrite(7, LOW);
    digitalWrite(6, HIGH);
  }
}
```

Sample Output:



Explanation:

In order to bring our initial concept to life, we've harnessed the capabilities of Tinkercad's circuit simulation tools. Within this virtual environment, we've designed a system where, if the concentration of air impurities surpasses a predefined threshold, several responsive actions occur.

When the level of impurities in the air reaches this critical point, a red indicator light begins to flash, and a continuous alarm emanates from a buzzer, effectively signaling a hazardous condition. To provide a quantitative measure of air quality, we've also incorporated an LCD display that showcases the precise parts per million (ppm) reading of air impurities.

Furthermore, we've enriched our system with a DTH sensor, which enables the real-time measurement of air humidity and temperature. By capturing this additional data, we can gain valuable insights into the overall air quality. This multifaceted approach ensures that our system not only detects pollution levels but also offers comprehensive environmental monitoring, allowing us to keep a vigilant eye on air quality in real time.

Future Development:

In the upcoming project phases, the information gathered from the sensors will be seamlessly transmitted to cloud-based platforms for further analysis and processing. Depending on the results of this data analysis, instantaneous notifications can be dispatched to the general public, ensuring real-time awareness of air quality conditions. Moreover, we are envisioning the development of a user-friendly mobile application that will empower users to remotely monitor air quality metrics. To enable wireless data transmission, we will leverage the capabilities of an ESP32 sensor in conjunction with the Arduino UNO. For data storage and management in the cloud, we will explore various cloud platforms such as ThingSpeak, Beceptor, and others, which will serve as repositories for the vast amount of sensor data. The final phase of our project is focused on consolidating historical records and training a machine learning model. This ML model will bring automation to the prediction of future air quality conditions. It will possess the intelligence to trigger real-time alerts in situations where harmful gases are detected, ensuring a proactive response to potential hazards.