#include <Servo.h>

const int mq2Pin = A0; // Analog pin connected to MQ-2 sensor

const int fanPin = 10; // Digital pin connected to fan control MOSFET gate

const int buzzerPin = 8; // Digital pin connected to buzzer

const int servoPin = 9; // Digital pin connected to servo motor control signal

Servo myservo; // Create a servo object

int sensorValue = 0; // Variable to store sensor reading

int threshold = 90; // Threshold value for gas detection (adjust based on calibration)

bool alarmActive = false; // Flag to indicate active alarm state

void setup() {

Serial.begin(9600); // Initialize serial communication for debugging

pinMode(mq2Pin, INPUT); // Set MQ-2 sensor pin as input (analog)

pinMode(fanPin, OUTPUT); // Set fan control MOSFET gate pin as output

pinMode(buzzerPin, OUTPUT); // Set buzzer pin as output

pinMode(servoPin, OUTPUT); // Set servo motor control pin as output

myservo.attach(servoPin); // Attach servo to pin

myservo.write(90); // Set servo to initial position (adjust as needed)

}

void loop() {

sensorValue = analogRead(mq2Pin); // Read analog value from MQ-2 sensor

Serial.print("Sensor Value: ");

Serial.println(sensorValue);

if (sensorValue > threshold && !alarmActive) {

// Gas detected above threshold, activate alarm

digitalWrite(fanPin, HIGH); // Turn on fan

tone(buzzerPin, 1000); // Activate buzzer with 1 kHz tone

myservo.write(0); // Set servo to activate position (adjust as needed)

alarmActive = true; // Set alarm active flag

int count = 1;

while(count==1){

delay(1000);

}

} else if (sensorValue <= threshold && alarmActive) {

// Gas level below threshold, deactivate alarm

digitalWrite(fanPin, LOW); // Turn off fan

noTone(buzzerPin); // Deactivate buzzer

myservo.write(90); // Set servo back to initial position

alarmActive = false; // Reset alarm active flag

}

delay(500); // Delay between readings (adjust as needed)

}