Portable Gas Sensor

Project Based Learning

By: Abhigyan Varma-1032232985
Akshat Agarwal – 1032240204
Abhyudeet Dawad – 1032240109

Table of Contents

1.1 Introduction

The "Portable Gas Sensor" is a compact, innovative device designed to detect gas leaks in confined spaces, particularly in environments like chemical factories where leakage risks are high. This project uses advanced gas detection technology to identify hazardous gases and alert workers to potential dangers, thereby helping to prevent direct harm. Our vision for this sensor is to offer a portable, reliable safety solution that requires minimal setup and can provide rapid, real-time gas monitoring in locations lacking extensive safety infrastructure.

1.2 Why did we choose this project?

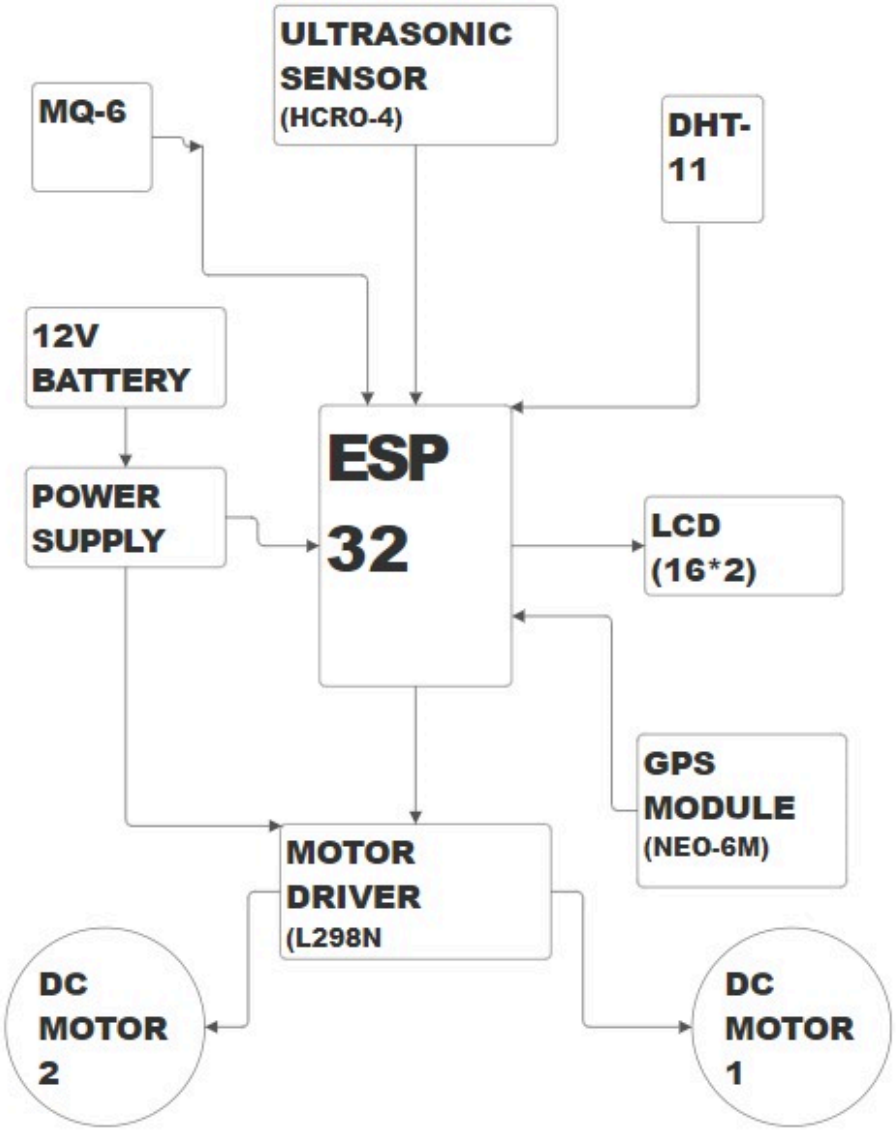
We chose this project to demonstrate how modern safety technology can address present and future industrial challenges, with an urgent need for improved worker protection. As industries evolve and expand, the demand for reliable safety solutions has intensified, particularly in hazardous environments like chemical factories. Our Portable Gas Sensor offers a proactive approach to safeguarding personnel by detecting potentially dangerous gas leaks early, with minimal disruption to daily operations. This project not only highlights the importance of advanced gas detection systems but also showcases how technological innovation can be seamlessly integrated into real-world safety practices to mitigate risks and protect lives.

2. Literature Review

Portable gas detection systems are increasingly recognized as critical safety measures in industrial and confined environments, especially in sectors with high potential for hazardous gas exposure, like chemical processing and manufacturing. For instance, studies indicate that confined spaces in chemical factories pose significant risks due to the potential accumulation of toxic gases, which can lead to severe health impacts if undetected. Our Portable Gas Sensor exemplifies how modern detection technology can effectively mitigate these dangers by providing timely alerts in gas-prone environments, proving both technically feasible and economically valuable in preventing accidents.

This portable model has demonstrated a substantial reduction in risks associated with gases like carbon monoxide, methane, and sulfur compounds, which are common in industrial settings. By providing real-time monitoring, our sensor offers flexibility and reliability, especially in scenarios where traditional fixed gas monitoring systems may be inaccessible or impractical. In emergency cases, such as gas leaks during natural disasters, this portable sensor can offer crucial, off-grid safety assurance, alerting workers to hazardous gas presence even in remote locations. The sensor’s mobility and sensitivity make it ideal for widespread use, enabling safe and continuous monitoring with minimal setup, thus supporting safer industrial practices and highlighting the importance of accessible, real-time gas detection in critical applications.

3.1 Block Diagram



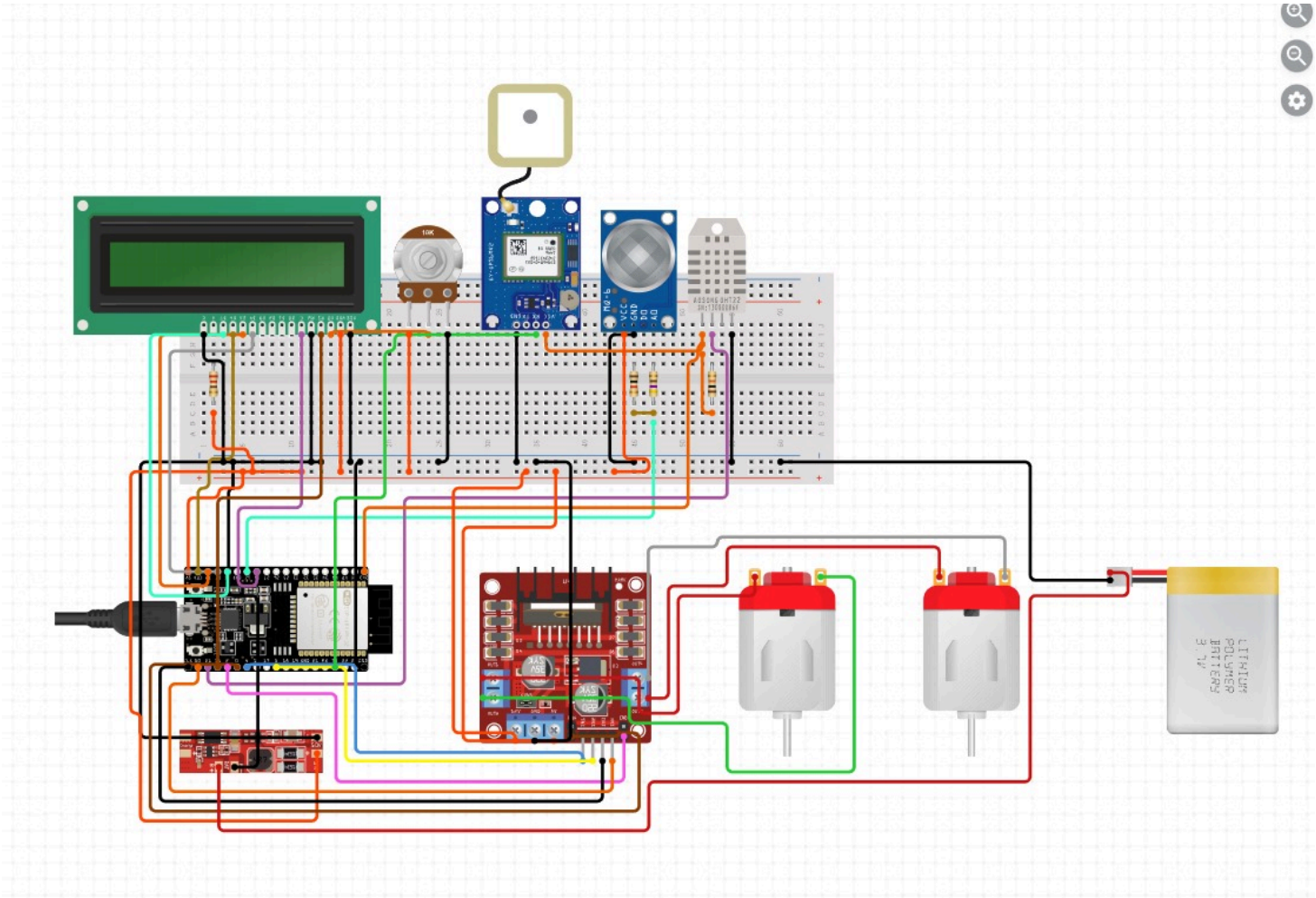
3.2 Working Mechanism

- **Motion and Navigation:** The car's movement is managed by two DC motors, controlled via an L298N motor driver. The ESP32 sends commands to the motor driver to adjust the speed and direction of the motors, allowing the car to move forward, backward, left, and right based on specific commands.

- **Obstacle Detection**: An ultrasonic sensor (HC-SR04) is used to measure the distance to any obstacles in front of the car. The ESP32 calculates this distance and, if an obstacle is detected within a certain range, it can adjust the car's path to avoid collisions. The distance is displayed on an LCD, providing real-time feedback on nearby obstacles.
- **Gas Detection**: An MQ-6 gas sensor is employed to monitor the presence of harmful gases in the environment. When gas is detected, the ESP32 displays a warning message on the LCD. This enables the car to act as a mobile gas leak detector, providing alerts to nearby personnel without needing a network connection.
- **Environmental Monitoring**: The DHT11 sensor records the ambient temperature, which is also displayed on the LCD. This additional data can help monitor environmental conditions in the car's immediate surroundings.
- **Display and Interface**: The LCD (16x2) screen shows real-time information on gas detection, distance to obstacles, and temperature. This helps users keep track of the car's surroundings and receive alerts directly from the display.

Overall, this setup makes the car an autonomous, mobile hazard-detection robot capable of moving through various environments, detecting gas leaks, avoiding obstacles, and displaying critical information on an LCD screen.

4. Circuit Diagram



5. Hardware Specifications

ESP32 Microcontroller

- Processor: Dual-core, 32-bit CPU, 160MHz or 240MHz clock speed
- Memory: 4MB Flash, 520KB SRAM
- Connectivity: Wi-Fi (802.11 b/g/n), Bluetooth (Classic
- Pins: 34 GPIO pins
- Power: 3.3V logic
- Function: Controls the operation of the LDRs, servo motor, and communicates with other components. It processes light intensity data from the LDRs and adjusts the servo motor to tilt the solar panels.

Lithium-Ion Batteries (3 units)

- Type: 18650 Lithium-ion Battery
- Voltage: 3.7V per cell
- Capacity: 2200mAh
- Function: Stores the energy collected by the solar panel. The system uses 3 batteries connected in a suitable configuration to provide stable and sufficient power to the charging system.

Seven Segment Display

- Type: 7-segment LED display
- Voltage: 5V
- Function: Displays the battery voltage level to show the charge status of the battery.

DC Motors

- Operating Voltage: Typically 3V to 12V (varies by motor type)
- Speed Range: 1000 to 30,000 RPM (varies by motor specifications)
- Torque: Ranges from 0.1 to 10+ kg·cm, depending on motor size and type
- Current Consumption: 0.1A to 3A (depends on load and voltage)
- Efficiency: Varies based on motor type and application

Ultrasonic Sensor

- Operating Voltage: 5V
- Operating Current: 15 mA (max)
- Measurement Range: 2 cm to 400 cm (1 inch to 13 feet)
- Accuracy: ±3 mm
- Frequency: 40 kHz
- Output: Digital signal (Echo and Trigger pins)

Motor Driver

- Operating Voltage: 5V to 35V
- Motor Current: 2A per channel (max)
- Logic Voltage: 5V (for control pins)
- Control: H-Bridge, supports both DC motors and stepper motors
- Power Dissipation: ~1W (depends on load)
- Interface: Control pins for direction, speed (PWM), and enable signals

GPS Module

- Operating Voltage: 3.3V to 5V
- Power Consumption: ~35 mA (depending on activity)
- Accuracy: ±2.5 meters (typical)
- Signal: GPS, GLONASS, and sometimes BeiDou
- Update Rate: 1 Hz (default), up to 10 Hz in some modules
- Interface: UART (serial communication

MQ6

- Operating Voltage: 5V
- Power Consumption: < 150 mA
- Sensing Range: 200 to 10,000 ppm for LPG, 300 to 10,000 ppm for methane
- Response Time: < 10 seconds
- Output: Analog voltage output (0-5V)
- Warm-Up Time: ~30 seconds

DHT11

- Operating Voltage: 3.3V to 5V
- Temperature Range: 0°C to 50°C (accuracy ±2°C)
- Humidity Range: 20% to 90% RH (accuracy ±5% RH)
- Sampling Rate: 1 Hz (once every second)
- Output: Digital signal (single-wire data interface)

6. Software – (Code)

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <WebServer.h>
#include <TinyGPS++.h>
#include <DHT.h>
#include <LiquidCrystal_I2C.h> // Include the LCD library

// WiFi credentials
const char* ssid = "ROHIT";
const char* password = "utlkothrud@321";

// Pins for Serial 2 (GPS) and Gas Sensor
#define RXD2 16
#define TXD2 17
#define GAS_SENSOR_PIN 35
#define TRIG_PIN 13  // HC-SR04 trigger pin
#define ECHO_PIN 12  // HC-SR04 echo pin
#define DHTPIN 14   // DHT11 data pin
#define DHTTYPE DHT11 // Define the DHT sensor type

// GPS and Motor Control
#define GPS_BAUD 9600
#define ENA 19
#define ENB 5
#define IN_1 4
#define IN_2 15
#define IN_3 18
#define IN_4 23
```

```
int speedCar = 800;
int speed_Coeff = 3;
String command;

WebServer server(80);
HardwareSerial gpsSerial(2);
TinyGPSPlus gps;
DHT dht(DHTPIN, DHTTYPE);

// LCD setup (I2C address: 0x27, 16x2 display)
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Function prototypes
void goAhead();
void goBack();
void goLeft();
void goRight();
void goAheadRight();
void goAheadLeft();
void goBackRight();
void goBackLeft();
void stopRobot();
float measureDistance();

float measureDistance() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);

float duration = pulseIn(ECHO_PIN, HIGH);
float distance = (duration * 0.0343) / 2; // Calculate distance in cm

return distance;
}

void setup() {
// Initialize Serial Monitor
  Serial.begin(115200);
```

```
// Set up motor and gas sensor pins
  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  pinMode(IN_1, OUTPUT);
  pinMode(IN_2, OUTPUT);
  pinMode(IN_3, OUTPUT);
  pinMode(IN_4, OUTPUT);
  pinMode(GAS_SENSOR_PIN, INPUT);

  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);

// Start DHT sensor
  dht.begin();

// Connect to WiFi
  WiFi.begin(ssid, password);
  Serial.print("Connecting to WiFi...");
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nConnected to WiFi.");

// Start GPS Serial 2
  gpsSerial.begin(GPS_BAUD, SERIAL_8N1, RXD2, TXD2);

// Start web server
  WiFi.mode(WIFI_AP);
  WiFi.softAP(ssid);
  IPAddress myIP = WiFi.softAPIP();
  Serial.print("AP IP address: ");
  Serial.println(myIP);

  server.on("/", HTTP_handleRoot);
  server.onNotFound(HTTP_handleRoot);
  server.begin();

// Initialize LCD
  lcd.init();
  lcd.backlight(); // Turn on the backlight
  lcd.setCursor(0, 0);
  lcd.print("Gas, Dist, Temp");
}

void loop() {
```

```
void loop() {
  server.handleClient();

  command = server.arg("State");
if (command == "F") goAhead();
else if (command == "B") goBack();
else if (command == "L") goLeft();
else if (command == "R") goRight();
else if (command == "I") goAheadRight();
else if (command == "G") goAheadLeft();
else if (command == "J") goBackRight();
else if (command == "H") goBackLeft();
else if (command == "0") speedCar = 400;
else if (command == "1") speedCar = 470;
else if (command == "2") speedCar = 540;
else if (command == "3") speedCar = 610;
else if (command == "4") speedCar = 680;
else if (command == "5") speedCar = 750;
else if (command == "6") speedCar = 820;
else if (command == "7") speedCar = 890;
else if (command == "8") speedCar = 960;
else if (command == "9") speedCar = 1023;
else if (command == "S") stopRobot();

// Check gas detection
int gasSensorValue = digitalRead(GAS_SENSOR_PIN);
  String gasStatus = (gasSensorValue == HIGH) ? "Gas Detected!" : "No Gas";

// Measure distance and temperature
float distance = measureDistance();
float temperature = dht.readTemperature();
if (isnan(temperature)) {
    temperature = 0.0; // If reading fails, display 0
  }

// Display values on LCD
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(gasStatus.substring(0, 16)); // Display gas status (first line)
  lcd.setCursor(0, 1);
  lcd.print("     D:" + String(distance) + "cm");
  lcd.setCursor(0, 1);
  lcd.print("T:" + String(temperature) + "C");
  delay(100); // Update every 2 seconds
}

void HTTP_handleRoot(void) {
if (server.hasArg("State")) {
```

```
void HTTP_handleRoot(void) {
if (server.hasArg("State")) {
   Serial.println(server.arg("State"));
 }
 server.send(200, "text/html", "");
 delay(1);
}

// Movement functions
void goAhead() {
 digitalWrite(IN_1, LOW);
 digitalWrite(IN_2, HIGH);
 analogWrite(ENA, speedCar);
 digitalWrite(IN_3, HIGH);
 digitalWrite(IN_4, LOW);
 analogWrite(ENB, speedCar);
}

void goBack() {
 digitalWrite(IN_1, HIGH);
 digitalWrite(IN_2, LOW);
 analogWrite(ENA, speedCar);
 digitalWrite(IN_3, LOW);
 digitalWrite(IN_4, HIGH);
 analogWrite(ENB, speedCar);
}

void goLeft() {
 digitalWrite(IN_1, LOW);
 digitalWrite(IN_2, HIGH);
 analogWrite(ENA, speedCar);
 digitalWrite(IN_3, LOW);
 digitalWrite(IN_4, HIGH);
 analogWrite(ENB, speedCar);
}

void goRight() {
 digitalWrite(IN_1, HIGH);
 digitalWrite(IN_2, LOW);
 analogWrite(ENA, speedCar);
 digitalWrite(IN_3, HIGH);
 digitalWrite(IN_4, LOW);
 analogWrite(ENB, speedCar);
}

void goAheadRight() {
 goAhead();
```

```
  goAhead();
  digitalWrite(IN_3, HIGH);
  digitalWrite(IN_4, LOW);
}

void goAheadLeft() {
  goAhead();
  digitalWrite(IN_3, LOW);
  digitalWrite(IN_4, HIGH);
}

void goBackRight() {
  goBack();
  digitalWrite(IN_3, HIGH);
  digitalWrite(IN_4, LOW);
}

void goBackLeft() {
  goBack();
  digitalWrite(IN_3, LOW);
  digitalWrite(IN_4, HIGH);
}

void stopRobot() {
  digitalWrite(IN_1, LOW);
  digitalWrite(IN_2, LOW);
  digitalWrite(IN_3, LOW);
  digitalWrite(IN_4, LOW);
}
```

7. Results

- **Motion & Navigation:**

The car effectively moves forward, backward, left, and right with precise control via the L298N motor driver and ESP32.
Speed and direction adjustments are responsive to commands, demonstrating reliable movement in various directions.

- **Obstacle Detection:**

HC-SR04 ultrasonic sensor accurately detects obstacles within a specified range.
Distance measurements are displayed on the LCD, enabling real-time obstacle tracking and avoidance.
The car successfully alters its path when obstacles are detected within the defined proximity, minimizing collision risks.

- **Gas Detection:**

The MQ-6 gas sensor detects the presence of harmful gases like LPG and methane.
When gas is detected, a warning message appears on the LCD, alerting the user to potential hazards.
The system works autonomously without needing a network connection, making it suitable for isolated environments.

- **Environmental Monitoring:**

The DHT11 sensor effectively measures the ambient temperature, providing additional environmental data.
Real-time temperature readings are displayed on the LCD, allowing monitoring of the car's surroundings.

- **Overall Performance:**

The system successfully integrates navigation, obstacle avoidance, gas detection, and environmental monitoring in one autonomous vehicle.
The LCD 16x2 display offers clear, real-time feedback on obstacle distance, gas detection, and temperature, ensuring the user stays informed about the car’s environment.

8. Conclusions

The project successfully demonstrated the practicality of portable gas sensors for enhancing safety in confined and industrial environments. It highlighted the effectiveness of real-time gas detection technology in protecting workers from hazardous leaks, showcasing how such devices can significantly reduce risk and improve emergency response. This Portable Gas Sensor exemplifies the potential for reliable, off-grid safety solutions, emphasizing the importance of accessible safety technology for mobile and confined applications. As industries advance, we envision such sensors becoming common in workplaces globally, ensuring safer environments and making safety measures more convenient and widespread for everyday use.

9. References

1. https://openai.com/index/chatgpt/
2. https://www.youtube.com/watch?v=RiYnucfy_rs
3. https://www.youtube.com/watch?v=rSCy1_GbC0w&t=4s