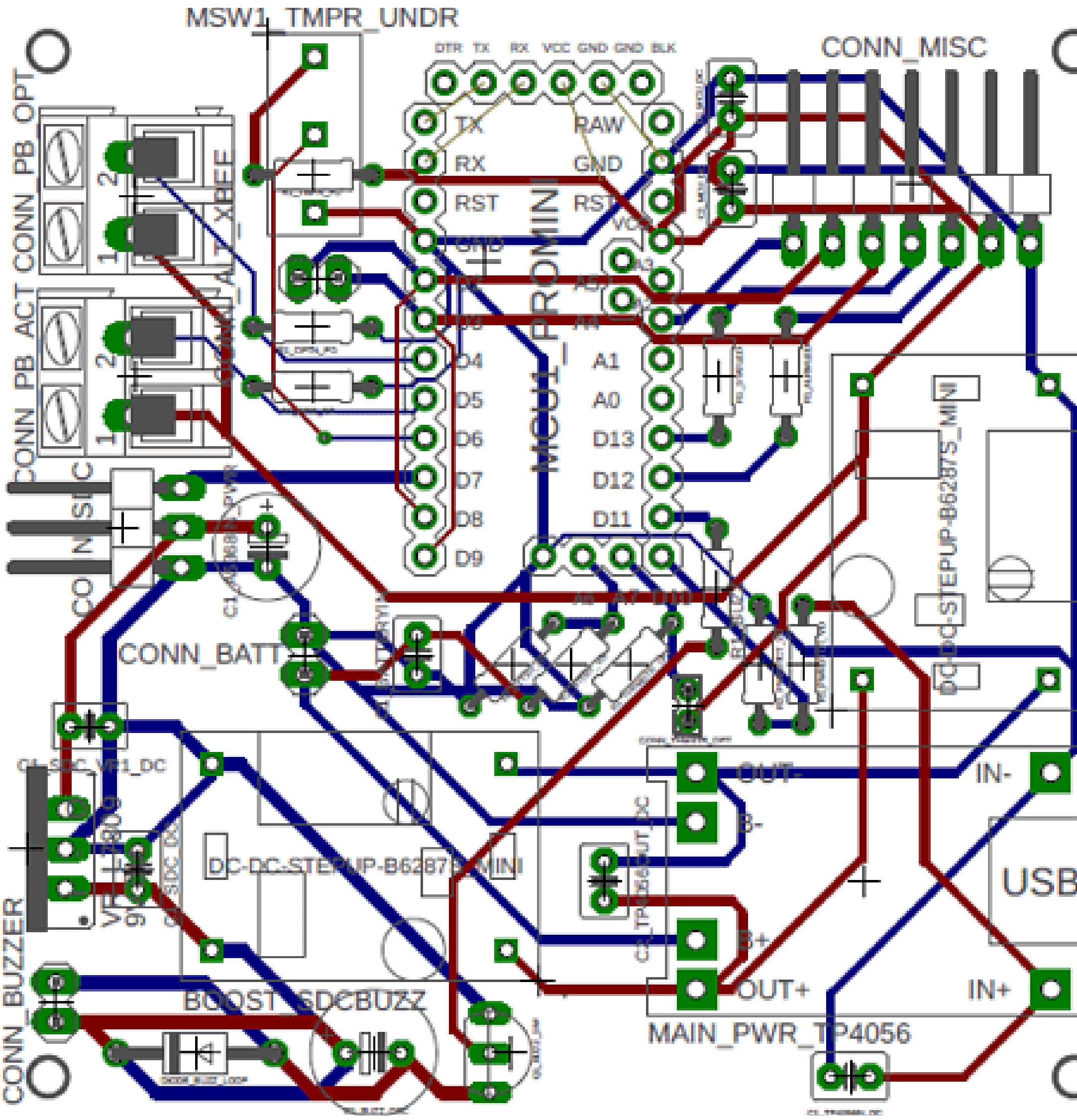
Project Report -
Fire Alarm Circuit



PREPARED FOR

Jannatun Naeem Tanin
Lecturer at DIU University

```
 /\_/\
( o.o )
 > ^ <
```

Team Members

MD Abdul Mueed Shah	241-15-655
Sabikon Nahar Anchol	241-15-703
MD SUZON MIA	241-15-717

Project Repository

https://github.com/mdsuzon-hub/Fire-Alarm

Department of Computer Science and Engineering (DIU)

17 August 2025

1. Objectives

The main objectives of this project are:

- To design and implement a fire alarm system that can detect **fire, smoke, and high temperature (≥ 40 °C)**.
- To provide **real-time alerts** using visual indicators (LEDs, OLED display), audio signals (buzzer), and long-distance notifications (smartphone via Wi-Fi).
- To differentiate between hazards with unique alert modes:
    - **Fire** → Red LED + short buzzer beeps.
    - **Smoke** → Green LED + long buzzer beeps.
    - **High Temperature (≥ 40 °C)** → Continuous buzzer + smartphone alert.
- To use **Firebase Realtime Database** for storing sensor values and enabling push notifications.
- To build the system on a **breadboard with ESP8266-12E** for processing, cloud connectivity, and mobile app integration.
- To prototype a **low-cost, portable, and reliable IoT fire alarm system** for homes, labs, and small industries.

2. Theory

Fires and smoke-related accidents cause major property losses and life risks. Early detection plays a crucial role in preventing disasters. Traditional alarms work only locally, while modern IoT-based systems can alert users remotely on their smartphones.

Working Principle

This project is based on **sensor-based hazard detection** combined with **IoT cloud services (Firebase)** for data storage and push notifications.

- **Smoke Detection:** MQ-2 detects combustible gases/smoke. Its resistance changes with gas concentration, producing an output signal to ESP8266.

- **Fire Detection:** The IR flame sensor detects infrared radiation from flames and sends a digital signal.

- **Temperature Detection:** The DHT11/LM35 sensor measures ambient temperature. If it exceeds **40 °C**, an overheating/fire condition is assumed.

- **Alerts and Indicators:**

    - **Red LED → Fire indication.**

    - **Green LED → Smoke indication.**

    - **Buzzer → Unique beep patterns for each hazard.**

    - **OLED Display → Shows real-time readings and warnings.**

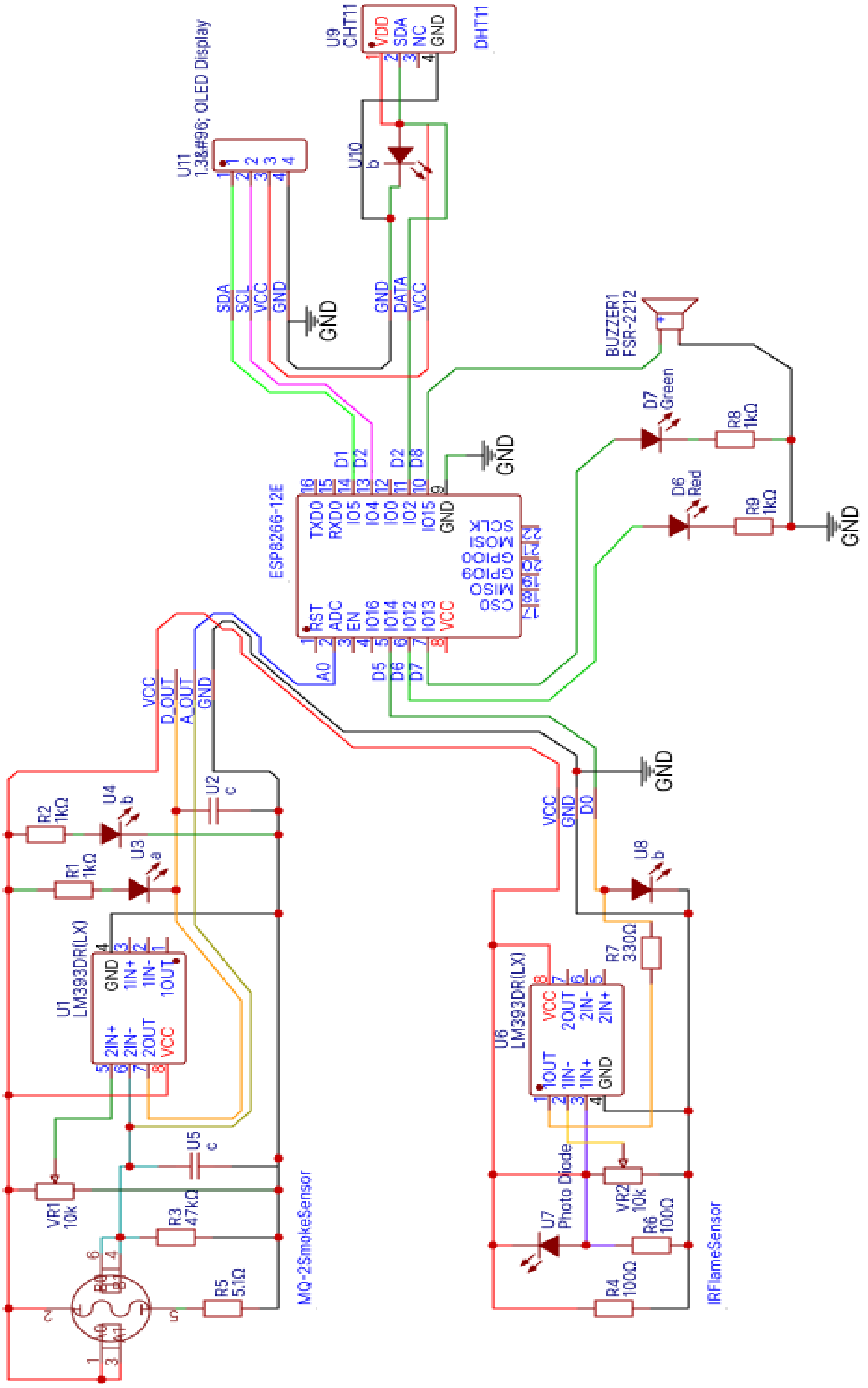
- **Smartphone Alert:** ESP8266 sends data to **Firebase Realtime Database**, which triggers **push notifications** and **alert popups** on the mobile app.

Thus, the project effectively combines **sensors + IoT cloud + smartphone alerts** to provide a modern fire alarm system.

3. Equipment Required

Component	Specification/Value	Function
ESP8266-12E / NodeMCU board	Wi-Fi Microcontroller	Main controller, IoT communication
MQ-2 Smoke Sensor	Gas/smoke sensor	Detects smoke/combustible gases
Temperature Sensor (LM35/DHT11/DHT22)	Analog/Digital	Monitors temperature, triggers alert ≥ 40 °C
IR Flame Sensor	IR-based fire sensor	Detects flame/fire presence
OLED Display (1.3-inch, SSD1306, I2C)	128×64 resolution	Displays readings and warnings
Red LED	5mm + 1kΩ resistor	Fire indicator
Green LED	5mm + 1kΩ resistor	Smoke indicator
Buzzer	Piezo, 5V	Audio alert (different beep modes)
Resistor	1kΩ	Current limiting for LEDs
Breadboard	830 tie-points	Prototyping circuit
Jumper Wires	Male-Female, Male-Male	Circuit connections
USB Cable (Type-B/Micro)	For NodeMCU	Programming + power
Power Source	5V adapter / Power bank	Power supply
Smartphone (with custom app)	Android/iOS	Displays data, receives notifications
Firebase Realtime Database	Cloud service	Stores sensor values + sends push alerts

4. Experimental Circuit



Circuit Setup and Connections:

OLED Display (SH1106, I²C)

- VCC → 3.3V
- GND → GND
- SDA → D2 (GPIO4)
- SCL → D1 (GPIO5)

DHT11 Temperature & Humidity Sensor

- VCC → 3.3V
- GND → GND
- Data → D4 (GPIO2)

Flame Sensor (Digital Output)

- VCC → 3.3V
- GND → GND
- DO → D5 (GPIO14)

Smoke Sensor (MQ-2, Analog Output)

- VCC → 3.3V (or 5V if module supports)
- GND → GND
- AO → A0

Alert Devices

- Red LED → D6 (GPIO12)
- Green LED → D7 (GPIO13)
- Buzzer → D8 (GPIO15)

ESP8266 (NodeMCU / Wemos D1 Mini)

- Powered via Micro-USB (5V) or regulated 3.3V supply.
- Provides WiFi connectivity for Firebase integration.

5. Result

The system was tested successfully and performed as expected. Hazards were detected accurately, and both local and remote alerts worked seamlessly.

Results Observed:

- **Fire Detected** → Red LED ON + Short buzzer beeps + Firebase push notification.
- **Smoke Detected** → Green LED ON + Long buzzer beeps + Firebase push notification.
- **High Temperature ≥ 40 °C** → Continuous buzzer + OLED warning + Firebase push notification.
- **OLED Display** → Showed live sensor readings and warning messages.
- **Smartphone App** → Displayed real-time values and triggered **pop-up alerts** via Firebase.

6. Conclusion

The Fire Alarm Circuit project successfully demonstrated the **integration of sensors, microcontroller, and Firebase IoT services** for real-time hazard detection. It provides both **local alerts (LEDs, buzzer, OLED)** and **remote alerts (smartphone notifications)**, making it highly effective for early warning systems.

Key takeaways:

- Reliable detection of **fire, smoke, and excessive temperature**.
- Real-time monitoring via **OLED + smartphone app**.
- Cloud-based data storage and push alerts using **Firebase Realtime Database**.
- A **low-cost, portable, and scalable system** for homes, labs, and small industries.
- Can be extended with **CO sensor, humidity sensor, or IP camera** for advanced safety.

Thus, the project meets its objective of building a **modern IoT-enabled fire alarm system**.

7. References

Datasheet of ESP8266 Wi-Fi Module – Espressif Systems.

Datasheet of MQ-2 Gas/Smoke Sensor.

Datasheet of DHT11/DHT22 / LM35 Temperature Sensor.

Datasheet of IR Flame Sensor.

SSD1306 OLED Display Documentation.

Firebase Documentation – https://firebase.google.com/docs.

Firebase Cloud Messaging (FCM) Guide.

IoT Communication Protocols (MQTT, HTTP).

Thank You