

# **EMBEDDED FIRE DETECTION AND ALARM NOTIFICATION SYSTEM**

## **TEAM MEMBERS:**

Joseph Manuel Thomas R

Ram Prasanth S

Manoj N

Revanth Kumar M G

**Aim:**

To detect the fire and notify the user by alarm using embedded system.

**Tools / Hardware Required:**

- ESP32-S NodeMCU
- Flame Sensor
- Temperature Sensor (DHT22)
- Buzzer
- Jumper Wires
- Arduino IDE

**Theory:**

Fire detection systems are essential for safety and early hazard prevention. This project includes:

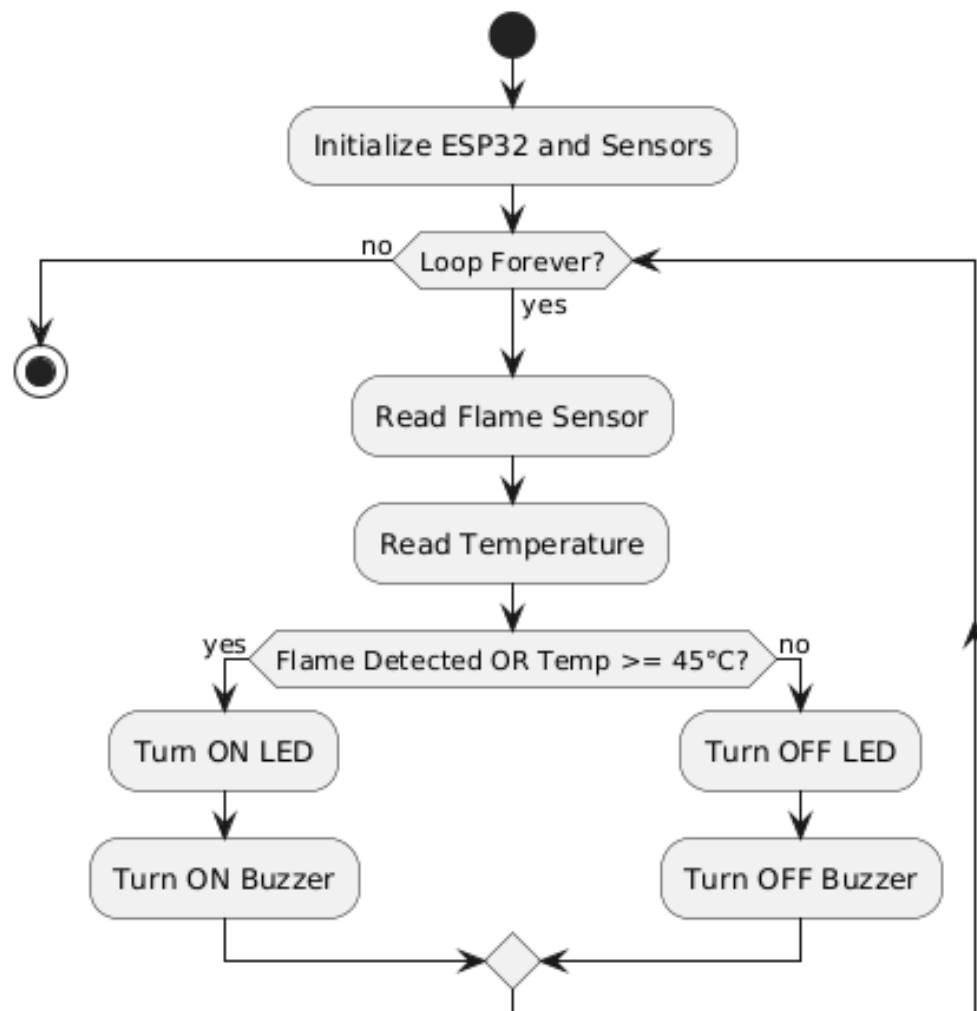
- ESP32-S NodeMCU
  - A powerful Wi-Fi-enabled microcontroller board.
  - Supports IoT applications for remote monitoring and alerts.
- Flame Sensor
  - Detects presence of fire or flame using infrared light.
  - Digital output pin gives HIGH or LOW based on flame intensity.
- DHT22 Sensor
  - Measures temperature and humidity.
  - Provides digital signal output for precise readings.
  - Temperature range:  $-40$  to  $+80$  °C, Humidity: 0–100% RH.
- LED & Buzzer
  - Provide visual and audio alarms when fire is detected or abnormal temperature rise occurs.

The system continuously monitors flame presence and temperature/humidity. If fire is detected or if temperature exceeds a threshold, the LED and buzzer are activated, and notifications can be sent via ESP32's Wi-Fi capability.

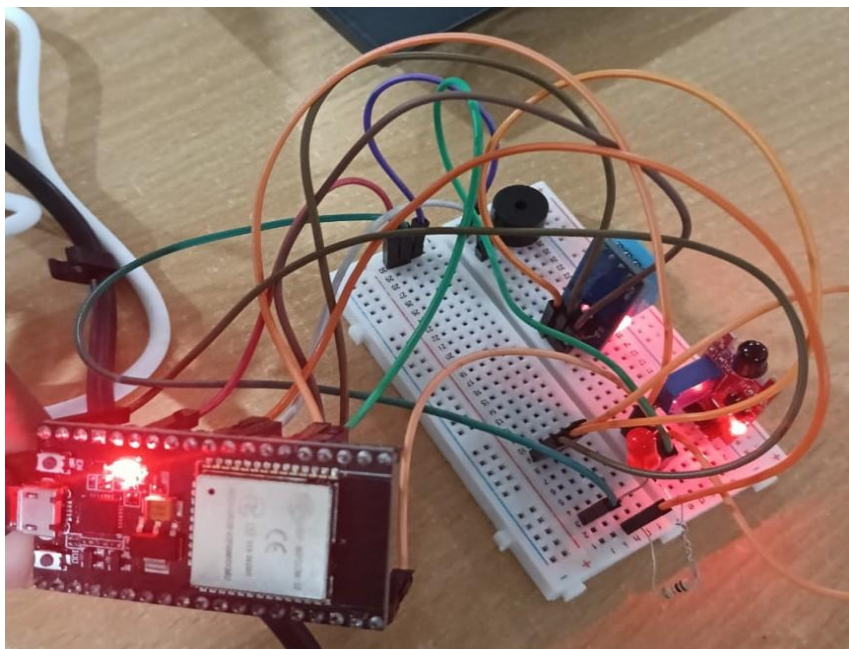
**Pin Connections:**

| Component       | Pico W Pin |
|-----------------|------------|
| Flame Sensor D0 | GPIO 27    |
| DHT22 Data Pin  | GPIO 4     |
| LED             | GPIO 26    |
| Buzzer          | GPIO 25    |
| VCC             | 5V         |
| GND             | GND        |

## Flowchart:



## Circuit Diagram:



## Program:

```
#include <DHT.h>
#define FLAME_DO_PIN 15 // Flame sensor digital output pin
#define DHT_PIN 14 // DHT11 data pin
#define BUZZER_PIN 18 // Buzzer pin
#define LED_PIN 2 // On-board LED
#define DHTTYPE DHT11
DHT dht(DHT_PIN, DHTTYPE);

const float TEMP_HIGH_C = 45.0;
const float TEMP_WARN_C = 30.0;
bool flameDetected = false;
float temperature = 0.0;
float humidity = 0.0;

void setup() {
  Serial.begin(115200);
  pinMode(FLAME_DO_PIN, INPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  pinMode(LED_PIN, OUTPUT);
  dht.begin();
  Serial.println("🔥 Fire Detection + Alarm System (ESP32)");
}

void loop() {
  flameDetected = (digitalRead(FLAME_DO_PIN) == LOW);
  temperature = dht.readTemperature();
  humidity = dht.readHumidity();
  if (isnan(temperature) || isnan(humidity)) {
    Serial.println("⚠ Failed to read from DHT11 sensor!");
    delay(2000);
    return;
  }

  bool alarm = false;
  String reason = "";
  if (flameDetected) {
    alarm = true;
```

```

    reason = "FLAME";
} else if (temperature >= TEMP_HIGH_C) {
    alarm = true;
    reason = "TEMP HIGH";
}

if (alarm) {
    digitalWrite(LED_PIN, HIGH); // short beep
    delay(200); // urgent beep
} else {
    digitalWrite(LED_PIN, LOW);
    delay(500);
}

Serial.print("Flame: ");
Serial.print(flameDetected ? " 🔥 DETECTED" : "No Flame");
Serial.print(" | Temp: ");
Serial.print(temperature);
Serial.print(" C | Hum: ");
Serial.print(humidity);
Serial.print(" % | Alarm: ");
Serial.println(alarm ? reason : "OFF");
}

```

### Result:

Thus the embedded system to detect the fire and notify the user is designed.