**ESP32-based Fire Alarm System**

**🔥 Fire Alarm System with ESP32, Blynk, I2C LCD, and Flame Sensor**

**📘 1. Project Overview**

**This project is a smart fire detection system using an ESP32 microcontroller, flame sensor, buzzer, red/green LEDs, 16x2 I2C LCD, and Blynk IoT platform. When a fire is detected, the system:**

* **Activates a blinking red LED and buzzer**
* **Displays warning messages on the LCD**
* **Sends a real-time update to the Blynk web dashboard**
* **Shows “Normal” status with a green LED when no fire is detected**

**🔩 2. Components Used**

| **Component Name** | **Quantity** |
| --- | --- |
| **ESP32 Dev Module** | **1** |
| **Flame Sensor** | **1** |
| **Red LED** | **1** |
| **Green LED** | **1** |
| **Buzzer** | **1** |
| **16x2 LCD (I2C)** | **1** |
| **Jumper Wires** | **As needed** |
| **Breadboard** | **1** |
| **USB Cable (Micro)** | **1** |

**🔌 3. Wiring Connections**

| **Component Name** | **Pin No** | **Destination Component** | **Pin No** | **Special Remark** |
| --- | --- | --- | --- | --- |
| **Flame Sensor** | **D0** | **ESP32** | **GPIO34** | **Digital Output** |
| **Red LED** | **+ve** | **ESP32** | **GPIO25** | **Fire Indicator** |
| **Green LED** | **+ve** | **ESP32** | **GPIO26** | **Normal Indicator** |
| **Buzzer** | **+ve** | **ESP32** | **GPIO27** | **Fire Alert** |
| **LCD (I2C) SDA** |  | **ESP32** | **GPIO21** | **I2C SDA** |
| **LCD (I2C) SCL** |  | **ESP32** | **GPIO22** | **I2C SCL** |
| **All Grounds** | **GND** | **ESP32** | **GND** | **Common Ground** |
| **All VCCs** | **VCC** | **ESP32** | **5V** | **Power Supply** |

**🧠 4. Working Principle**

* **The flame sensor detects infrared radiation from fire.**
* **When fire is detected (LOW signal), the system:**
  + **Blinks red LED**
  + **Turns ON buzzer with blinking effect**
  + **Turns OFF green LED**
  + **Sends alert to Blynk Web Dashboard**
  + **Displays “FIRE DETECTED” on LCD**
* **When no fire is detected (HIGH signal):**
  + **Turns ON green LED**
  + **Turns OFF red LED and buzzer**
  + **Sends normal status to Blynk**
  + **Displays “No Fire Detected” on LCD**

**📲 5. Blynk Dashboard Setup**

1. **Go to** [**https://blynk.cloud**](https://blynk.cloud/)
2. **Create a new template:**
   * **Template Name: Fire Alarm System**
   * **Board: ESP32 Dev Board**
3. **Add Datastream:**
   * **Name: fire\_status, Virtual Pin: V0, Type: Boolean**
4. **Create a device from the template.**
5. **Add Widgets to Web Dashboard:**
   * **LED → V0 (shows red for fire, gray for normal)**
   * **Optional: Value Display (text status)**

**💻 6. Arduino Code**

**✅ Fully integrated with:**

* **Blynk Cloud**
* **Blinking red LED and buzzer**
* **Real-time LCD status**

**🔽** [**Scroll up for full code**](https://chatgpt.com/c/6878866a-ce74-8013-b8c9-76ddd9c3b5eb#scroll-to-the-code-block-above) **or use this filename: Fire\_Alarm\_Blynk\_LCD.ino**

**🧪 7. Testing Procedure**

| **Test Scenario** | **Expected Output** |
| --- | --- |
| **Flame Sensor = HIGH (No Fire)** | **Green LED ON, LCD shows “No Fire Detected”, V0 = 0** |
| **Flame Sensor = LOW (Fire Detected)** | **Red LED and Buzzer blink, LCD shows “FIRE DETECTED!”, V0 = 1** |

**🔒 8. Safety Precautions**

* **Do not expose actual fire directly to the sensor.**
* **Use an IR lighter or candle for safe flame testing.**
* **Ensure components are not overheated or overpowered.**

**🧾 9. Advantages**

* **Real-time IoT monitoring with Blynk**
* **Audible and visual alert system**
* **User-friendly LCD feedback**
* **Safe flame detection test with visual indicators**

**📌 10. Future Enhancements**

* **Add SMS/Email alerts using Blynk Automations**
* **Interface with fire extinguisher system**
* **Include temperature and smoke sensors (e.g., MQ2, DHT11)**
* **Add battery backup and enclosure for portability**

**Let me know if you'd like this documentation exported as Word, PDF, or Excel pin mapping.**