Wi- Fi Based Forest Fire ALERT SYSTEM without LCD

A PROJECT REPORT

SUBMITTED BY

ASHA M	**912222106006**
DIVYADHARSHINI J	**912222106010**
MALINI M	**912222106301**
HEMADHARSHINI S	**912222106012**
KIRUTHIKA M	**912222106017**
NIMITHA B	**912222106026**

In practical fulfillment of the requirements for the

DEPARTMENT OF

ELECTRONICS AND COMMUNCATION ENGINEERING

SOLAMALAI COLLEGE OF ENGINEERING

(Approved by AICTE, New Delhi and Affiliated to Anna University, Chennai)

An ISO 9001: 2015 Certified Institution

Solamalai Knowledge Park,

S.V. Raja Nagar, Veerapanjan, Madurai -625020

(Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai)
An ISO 9001:2015 Certified Institution | Recognized by UGC under 2(f)
Solamalai Knowledge Park, S.V.Raja Nagar, Veerapanjan, Madurai - 625020

BONAFIDE CERTIFICATE

Certified that this EIOT LAB report on **"WIFI BASED FOREST FIRE ALERT SYSTEM WITHOUT LCD"** to the bonafide record of work done by **ASHA M (912222106006), DIVYADHARSHINI J (912222106010), MALINI M (912222106301), HEMADHARSHINI S (912222106012), KIRUTHIKA M (912222106017), NIMITHA B (912222106026)** practical fulfillment for the award of "**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**" by the **ANNA UNIVERSITY**, Chennai.

SIGNATURE	**COURSE COORDINATOR**
Mr. B. MAHESH, M.E, (Ph.D.)	Mr. M.G. RAJENDRAKUMAR, ME, (Ph.D.)
HEAD OF THE DEPARTMENT	**ASSISTANT PROFESSOR**
Department of ECE,	Department of ECE,
Solamalai College of Engineering,	Solamalai College of Engineering,
Madurai-625020.	Madurai-625020.

Submitted for the board examination held on ________________

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express the gratitude to the people who have been instrumental in the successful completion of this Naan Mudhalvan.

We take immense pleasure in thinking respected **Mr. V.S.P. SOLAMALAI PITCHAI B.Sc.,** Chairman, Solamalai College of Engineering for having permitted us to carry out this report.

We express our thanks and gratitude to our beloved Associate Director **Dr. S. NAGARAJAN M.E., Ph.D.** and Principal **Dr. M. SANTHI M.E., Ph.D.** for extending his support and permit us to do this Naan Mudhalvan.

We are highly indebted to **Mr. B. MAHESH M.E., (Ph.D.)** Head of the Department Electronics and Communication Engineering. their guidance and constant supervision as well as for providing necessary information regarding the Naan Mudhalvan & also for their support in completing the report.

We also express our sincere thanks to our esteemed under guided Naan Mudhalvan Course Coordinator **Mr. M.G. RAJENDRAKUMAR, M.E., (Ph.D.)** for his valuable suggestions while doing this report.

We would like to express our gratitude to all staffs of **SOLAMALAI COLLEGE OF ENGINEERING** for their kind Co-operation and encouragement which help using completion of this report.

Finally, yet important, we would like to express our heartfelt thanks to our beloved parents for their blessings, our classmates for their help and wishes for the successful completion of this **EIOT LAB REPORT**.

ABSTRACT

Forest fires are one of the most devastating natural disasters, leading to significant environmental and economic damage. Early detection and timely alerts are crucial in preventing the spread of such fires. This project presents a Wi-Fi based forest fire detection and alert system designed without the use of an LCD, making it cost-effective and suitable for remote locations.

The system employs **temperature and gas sensors (such as DHT11 and MQ series)** to continuously monitor environmental conditions. When the sensor readings cross predefined threshold levels indicating a potential fire, the data is transmitted through a **Wi-Fi module (ESP8266)** to a cloud-based platform or mobile app. This enables real-time alerts to forest authorities or responsible personnel via the internet, without the need for an LCD or physical display.

By eliminating the LCD, the system is made simpler, more power-efficient, and easily deployable in inaccessible forest areas. The project demonstrates an effective, low-cost IoT-based solution that contributes to the early detection and prevention of forest fires, thus helping protect wildlife, vegetation, and human lives.

INTRODUCTION

Forest fires are one of the most serious natural disasters that pose a threat to the environment, biodiversity, and human life. These fires can start due to natural causes like lightning, but more often they are the result of human negligence. Once ignited, forest fires can spread rapidly, especially in dry seasons, destroying large areas of forest and wildlife in a short period. Early detection and rapid response are essential to minimize damage and control the spread of fire.

Traditional methods of forest fire detection such as watch towers, satellite imaging, or human patrols are often slow, expensive, and inefficient. These methods may not detect a fire at its early stage, leading to delayed action and severe damage. To overcome these limitations, the use of modern technology like the Internet of Things (IoT) has become a promising solution.

This project focuses on developing a Wi-Fi based forest fire detection alert system that continuously monitors environmental parameters such as temperature and gas concentration. The system is designed **without an LCD display to reduce cost and power consumption, making it ideal for deployment in remote forest areas** where resources are limited.

The core of the system is a **Wi-Fi-enabled microcontroller (ESP8266), which gathers data from sensors like the DHT11 temperature sensor and MQ-series gas sensor.** If the sensor readings exceed certain thresholds indicating fire conditions, the system activates an alert mechanism and sends the information through the internet to a mobile app or cloud server for remote monitoring.

METHODOLOGY

This section outlines the step-by-step approach followed to design and implement the Wi-Fi Based Forest Fire Detection Alert System Without LCD. The methodology is divided into the following key stages:

1.Hardware Setup

The hardware setup involves selecting and assembling the required electronic components:

- **ESP8266 Node MCU** acts as the central controller with **built-in Wi-Fi capability.**
- **DHT11 sensor** measures **the temperature** and humidity of the environment.
- **MQ-2 or MQ-135 gas sensor detects** smoke and **gases that indicate fire.**
- A **buzzer** is connected for **immediate local alerts**.
- All components are powered and interconnected on a breadboard or PCB using jumper wires.

2.Software Development

The software is developed using the Arduino IDE, where:

- Libraries for the sensors and Wi-Fi module are included.
- Threshold values for temperature and gas levels are set in the code.
- If sensor readings exceed limits, the system triggers the buzzer and sends data via Wi-Fi.
- The program is uploaded to the ESP8266 through a USB connection.

3.Data Processing and Display

- Sensor data is continuously read and processed by the ESP8266.
- If abnormal readings are detected, they are converted into useful alerts.
- Instead of displaying the data on an LCD, it is sent to a web server or mobile application for remote monitoring.

4. Web Server Development

- A cloud platform such as Thing Speak, Blynk, or a custom-built web server is used.
- ESP8266 sends data to the server using HTTP or MQTT protocols.
- Users can view live sensor data and receive fire alerts on a smartphone or computer via the internet.

5.Alert Mechanism

When fire conditions are detected:

- The buzzer sounds as a local warning.
- A notification or email alert is sent through the cloud server.

6.Testing and Validation

- The system is tested in various simulated conditions with increased temperature and smoke to check response accuracy.
- Data logs are verified on the cloud platform.
- The sensitivity and accuracy of the sensors are calibrated during testing.

7.Future Enhancements

- Integration with solar power systems for long-term forest deployment.
- Adding GPS modules for exact fire location tracking.
- Using AI/ML algorithms to predict fire risks based on sensor trends.
- Expanding the system into a mesh network for wider forest coverage.

EXISTING WORK

In recent years, several forest fire detection systems have been developed using various technologies such as satellite monitoring, drone surveillance, and wireless sensor networks (WSN). However, many of these methods have limitations like high cost, delayed data transmission, or dependency on human monitoring.

Some existing systems use Arduino or Raspberry Pi boards with temperature, smoke, and flame sensors. These systems often include LCD displays to show real-time sensor data, which increases power consumption and makes the setup more complex for remote areas.

Other IoT-based models send alerts via GSM modules or Bluetooth, which have range limitations or incur extra operational costs. While cloud-based solutions exist, they are sometimes not optimized for low-power, low-cost field deployment.

Your proposed system improves upon existing models by:

- Eliminating the LCD, reducing cost and power usage.
- Using Wi-Fi (ESP8266) for real-time alerts and cloud integration.

Traditional Methods

Earlier, forest fire detection heavily relied on:

1.Watchtowers and Fire Lookouts:

- Human observers stationed in towers manually monitor forest areas and report visible smoke or fire.

2.Satellite Imaging:

- Government agencies use satellites to monitor large forest regions and detect heat signatures or smoke. However, this method has delayed response times and is dependent on weather conditions and satellite availability.

3.Helicopter or Drone Patrols:

- These provide an aerial view of the forest to detect fires. While effective, they are expensive and not feasible for continuous monitoring.

These methods have several limitations:

- High operational and maintenance costs
- Delay in detection and alerting
- Dependence on human intervention or favorable weather

Modern IoT-Based Systems

To overcome the above drawbacks, recent advancements have focused on sensor-based automated systems using microcontrollers, sensors, and communication modules.

1.Arduino-Based Systems with LCD

Many projects use Arduino boards integrated with temperature and gas sensors. These systems often include an LCD screen to display real-time sensor data. However, such systems:

- Require human presence to read the display
- Consume more power due to the LCD
- May not be practical for remote, unattended locations

2.GSM-Based Alert Systems

Some systems use GSM modules (SIM800/SIM900) to send SMS alerts when a fire is detected. These are reliable for alerting users without internet access, but:

- Require mobile network coverage
- Have ongoing SIM card costs
- Offer slower data transfer compared to Wi-Fi

3.Bluetooth and Zigbee Modules

Bluetooth-based systems can send fire alerts to nearby mobile devices. Zigbee is also used in mesh networks for sensor communication. But:

- Bluetooth has very limited range
- Zigbee requires complex networking and additional devices

4.Cloud-Connected IoT Platforms

Recent systems use ESP32 or ESP8266 with Wi-Fi connectivity to send data to cloud platforms like Thing Speak or Blynk. These platforms allow remote monitoring through mobile apps or dashboards.

PROPOSED WORK

proposed system aims to develop a Wi-Fi based forest fire detection and alert system without using an LCD, making it **more power-efficient and cost-effective** for remote forest monitoring.

The system uses sensors like DHT11 to monitor temperature and MQ-series gas sensors to detect smoke or harmful gases. These sensors are connected to an ESP8266 Node MCU, which processes the data and connects to the internet via Wi-Fi.

If the readings cross predefined thresholds, the system:

- Triggers a buzzer for immediate local alerts.
  Sends real-time data to a cloud platform or mobile app for remote monitoring.
- By removing the LCD, the system becomes simpler, consumes less power, and is more suitable for deployment in large or remote forest areas.
- The proposed work enhances existing systems by offering a reliable, low-cost, and internet-enabled solution for early forest fire detection.

Key Features of the Proposed System:

1. No LCD Display:

- Most existing systems use LCDs to display sensor data locally. In remote forest areas, LCDs are unnecessary, consume power, and are hard to read. The proposed system eliminates this component to reduce cost, complexity, and power consumption.

2. Sensor-Based Monitoring:

- The system uses a DHT11 sensor to monitor temperature and humidity, and an MQ-series gas sensor (like MQ-2 or MQ-135) to detect smoke, CO, or other fire-related gases.
- These sensors are reliable for detecting early signs of forest fires, especially when temperature and gas levels increase rapidly.

3. ESP8266 Wi-Fi Microcontroller:

- The Node MCU ESP8266 module is used as the central controller.
- It reads data from the sensors and connects to the internet using its built-in Wi-Fi feature.
- The use of ESP8266 allows low-power operation and easy programming via Arduino IDE.

4. Cloud Communication and Alerts:

- When the system detects abnormal readings (e.g., temperature > 50°C or gas concentration above a threshold), it:
- Activates a buzzer as a local alert.
- Sends data to a cloud platform (like Blynk, ThingSpeak, or a custom web server).
- Triggers mobile notifications or emails to alert forest personnel or emergency services.

5. Real-Time Remote Monitoring:

- Since the data is sent via Wi-Fi to the cloud, users can monitor the environment remotely using a smartphone, tablet, or computer.
- This allows for faster responses even if the site is far from any human monitoring team.

6. Low-Cost and Scalable Design:

- The system is built using affordable components that can be easily sourced.
- It can be replicated and installed in multiple forest areas, creating a network of smart detection nodes.

Advantages of the Proposed System

- **Real-time monitoring** via Wi-Fi
- **Cost-effective** due to elimination of display units
- **Easy to install** and operate
- Portable and **low power consumption**

REQUIREMENTS

1. Hardware Requirements:

- Microcontroller Board
- Temperature Sensor
- Gas/Smoke Sensor
- Wi-Fi Module
- Buzzer/Alarm – To give local alert
- LEDs – For visual indication (optional)
- Power Supply – Battery or USB
- Connecting Wires – For circuit setup
- Breadboard or PCB – For component placement

2. Software Requirements:

- Arduino IDE – For writing and uploading code
- IoT Platform (Optional) – For remote monitoring (e.g., Blynk, Thing Speak, Firebase)
- Internet Connection – For Wi-Fi connectivity and data transfer.

3. Functional Requirements:

- Detect high temperature and smoke/gas levels in the environment.
- Trigger alarm (buzzer) when thresholds are exceeded.
- Send alert messages via WiFi to a smartphone or cloud platform.
- Provide real-time monitoring (if integrated with IoT platform).

4. Non-Functional Requirements:

- Should be reliable and responsive to environmental changes.
- Scalable to monitor large forest areas using multiple units.

IMPLEMENTATION DETAILS

1. Sensor Integration:

- Temperature Sensor (e.g., LM35/DHT11) is connected to the Analog/digital pin of the microcontroller to read ambient temperature.
- Smoke/Gas Sensor (e.g., MQ-2/MQ-135) detects the presence of smoke or harmful gases and sends Analog data to the microcontroller.

2. Microcontroller Unit:

- A Node MCU (ESP8266) or ESP32 is used as the main controller.
- It processes sensor data and checks if the values cross the predefined threshold (e.g., temperature > 50°C or gas level > safe limit).
- If any sensor value exceeds the threshold, it activates the buzzer and sends an alert through Wi-Fi.

3. Wi-Fi Communication:

- The microcontroller connects to a Wi-Fi network using its inbuilt Wi-Fi module.
- Data is sent to a remote server or IoT platform like Blynk, Thing Speak, or Firebase for real-time monitoring.
- A notification or alert is triggered on the user's smartphone or computer if danger is detected.

4. Alert System:

- A buzzer is used to provide a local sound alarm.
- LED indicators may be used to show status (e.g., green for safe, red for fire detected).
- Alerts can also be shown on a mobile app or web dashboard (if integrated).

5. Power Supply:

- The system is powered using a USB cable, battery pack, or solar power (for real-time deployment in forests).
- A voltage regulator may be used to ensure stable power to the sensors and microcontroller.

6. Programming Logic (in Arduino IDE):

- Read sensor values at intervals.
- Compare readings with preset thresholds.
- If values exceed limits:
- Trigger buzzer
- Send alert message through Wi-Fi

DATA COLLECTION

1. Source of Data:

The sensors connected to the microcontroller collect environmental data such as:

- Temperature (using DHT11/LM35)
- Smoke or Gas levels (using MQ-2/MQ-135)

2. Data Acquisition Process:

- Sensors continuously monitor the surrounding environment.
- Sensor readings are taken at regular time intervals (e.g., every 1–5 seconds).
- The microcontroller reads this data in real time via analog/digital input pins.

3. Data Filtering and Threshold Check:

- Raw data is processed by the microcontroller.
- It checks whether the temperature or smoke levels exceed the predefined safety thresholds.

Example:

Temperature > 50°C

Smoke level > 300 ppm

If threshold is crossed, it is considered critical data.

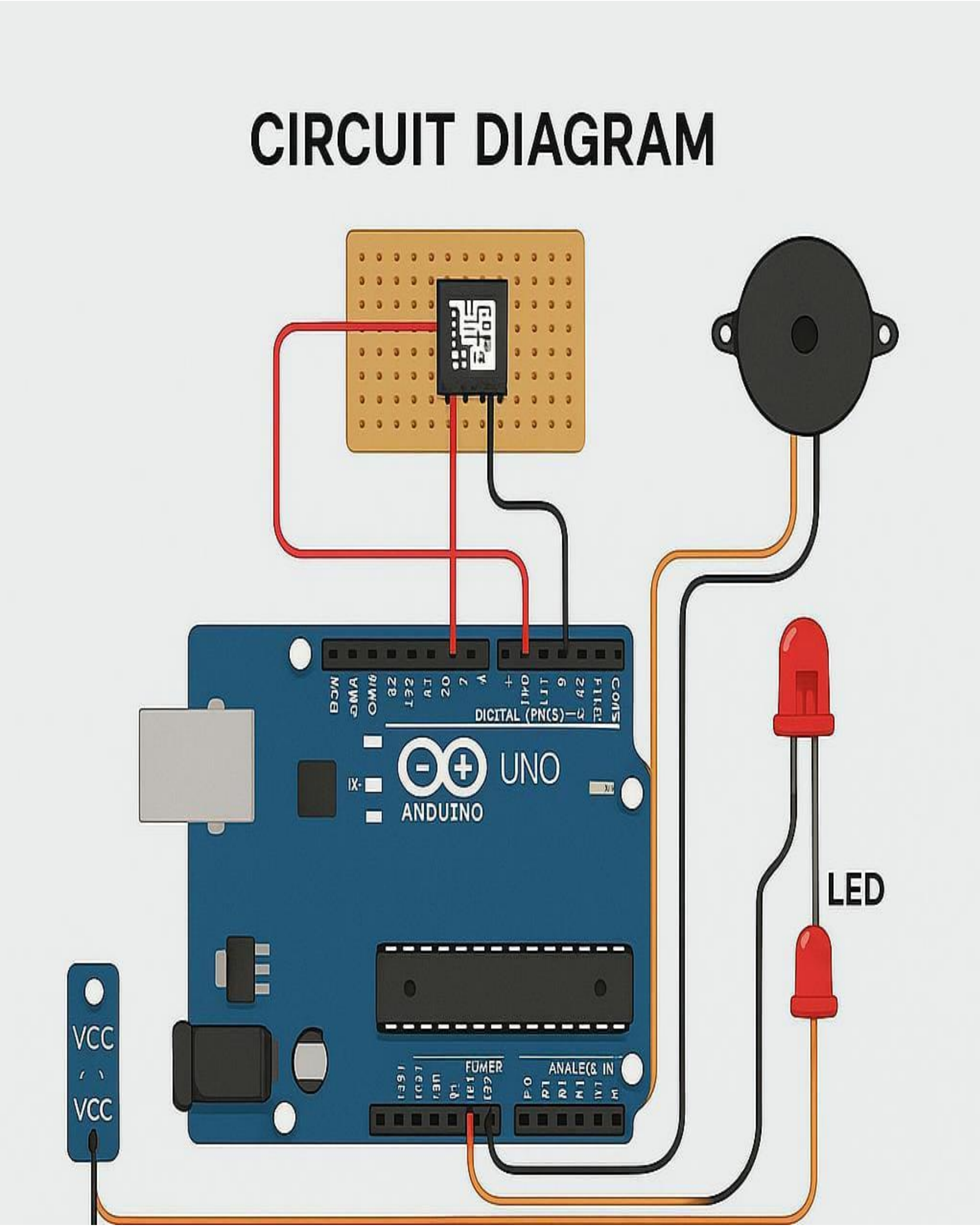
4. Data Transmission (via Wi-Fi):

- The collected data is sent to an IoT platform or cloud server using Wi-Fi.
- Platforms like Thing Speak, Blynk, or Firebase can store and display this data.
- Real-time monitoring dashboards or mobile apps can be used to visualize and track the data remotely.

5. Alert-Based Data Collection:

- When danger is detected.
- System stores timestamped data.
- Sends alert notifications along with current readings.
- Records event details (temperature, smoke level, time)

CIRCUIT DIAGRAM



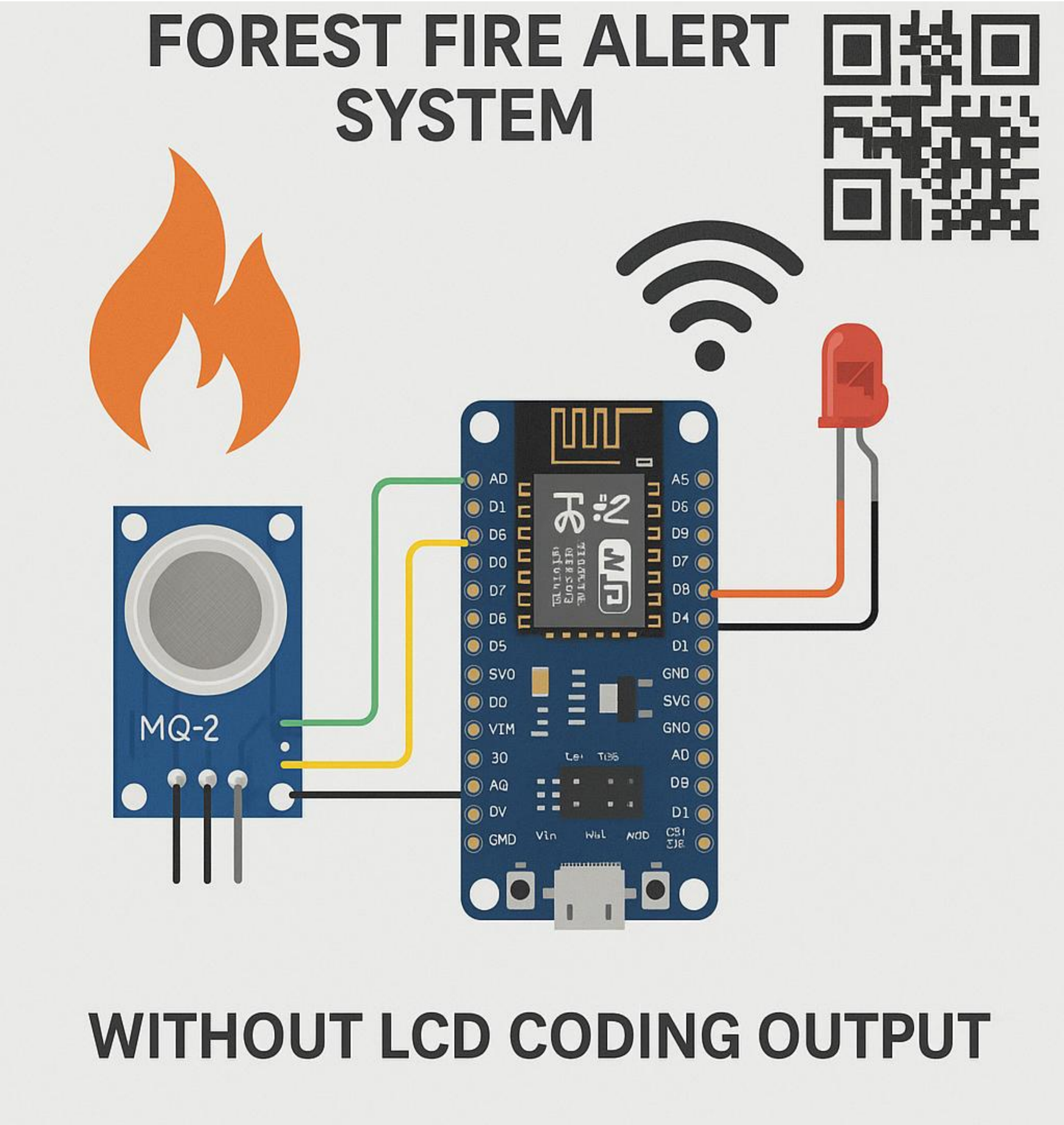
CODING

```
/*
  WiFi Based Forest Fire ALERT SYSTEM without LCD
*/

int flameSensorPin = A0;    // select the input pin for Sensor

const int buzzerPin = 6;
const int ledPin = 13;

int flameValue = 0;  // variable to store the value coming from the sensor1

void setup()
{
  Serial.begin(115200);

  // declare the ledPin as an OUTPUT:
  pinMode(buzzerPin, OUTPUT);
  pinMode(ledPin, OUTPUT);

  digitalWrite(buzzerPin, LOW);
  digitalWrite(ledPin, LOW);

  wificonfig();
  delay(3000);
  lcd.clear();

  digitalWrite(ledPin, HIGH);
  delay(100);
}

void loop()
{
```

```
  Start:
  delay(10);

  Serial.println("AT+CIPSEND=0,22");
  delay(500);
  Serial.println("                    ");
  delay(100);

  // read the value from the sensor:
  flameValue = analogRead(flameSensorPin);
  delay(10);

  if(flameValue < 300)
  {
    digitalWrite(buzzerPin, HIGH);
    delay(100);
    Serial.println("AT+CIPSEND=0,25");
    delay(500);

    Serial.println("FIRE DETECTED @ FOREST ; ");
    delay(1000);
  }

  if(flameValue > 500)
  {
    digitalWrite(buzzerPin, LOW);
    delay(100);
  }
}

void wificonfig()
{
  Serial.println("AT");
  delay(300);
```

```
  Serial.println("ATE0");
  delay(300);
  Serial.println("AT+CWMODE=1");
  delay(300);
  Serial.println("AT+CIPMUX=1");

  delay(300);
  Serial.println("AT+CIPSERVER=1,80");
  delay(300);
  Serial.print("AT+CWJAP=");
  Serial.write(0x22);
  Serial.print("fire");//user name- minimum length name
  Serial.write(0x22);
  Serial.write(0x2C);
  Serial.write(0x22);
  Serial.print("monitor1234");//password- numbers
   delay(1000);
}
```

OUTPUT



CONCLUSION

Forest fires are among the most destructive natural disasters, causing serious harm to the environment, wildlife, and even human settlements. Traditional detection systems often rely on manual observation or expensive satellite surveillance, which may not be fast enough to prevent large-scale damage. In this context, the proposed Wi-Fi-Based Forest Fire Detection Alert System Without LCD offers a modern and practical solution that leverages IoT technology for early detection and instant alerting.

This project successfully demonstrates a simple, reliable, and cost-effective method to detect forest fire conditions by using sensors that monitor temperature and harmful gases in the environment. The integration of the ESP8266 Node MCU microcontroller with DHT11 and MQ-series gas sensors allows the system to collect real-time environmental data and transmit it wirelessly through a Wi-Fi connection.

One of the major highlights of this system is the elimination of the LCD display, which makes the design more power-efficient and reduces overall cost. Instead of local display output, the system sends alerts directly to a cloud platform or mobile application, allowing forest officers and emergency response teams to take timely action from any location.

The project not only contributes to environmental protection but also shows how IoT can be effectively applied in real-world applications to improve disaster management systems. With further enhancements like GPS integration, solar-powered operation, and AI-based prediction models, the system can be scaled and deployed on a large scale in forest areas across the globe.

REFERENCE

1**. Arduino.cc**. (n.d.). Arduino Uno Rev3. Retrieved from **https://www.arduino.cc/en/Main/ArduinoBoardUno**

2. ESP8266 Community. (n.d.). ESP8266 Wi-Fi Module. Retrieved from **https://www.espressif.com/en/products/hardware/esp8266/overview**

3. Chatterjee, S., & Bandyopadhyay, S. (2018). Design and Implementation of IoT-based Forest Fire Detection System Using Arduino and Wi-Fi Module. International Journal of Advanced Research in Computer Science, 9(1), 42-48.

4. Sahu, N., & Sahoo, S. (2017). IoT Based Forest Fire Detection System using Wireless Sensors. International Journal of Advanced Research in Computer Science, 8(3), 97-102.

5. DHT11 Sensor Documentation. (n.d.). DHT11 Temperature and Humidity Sensor. Retrieved from **https://www.electronicwings.com/nodemcu/dht11-temperature-and-humidity-sensor-interfacing-with-nodemcu**

6. Google Firebase. (n.d.). Firebase Realtime Database. Retrieved from **https://firebase.google.com/docs/database**

7. Kumar, R., & Gupta, A. (2019). Wireless Sensor Networks for Forest Fire Detection and Monitoring: A Review. Journal of Environmental Science and Engineering, 5(1), 34-40.

8. Soni, S., & Shah, P. (2020). Design of IoT-Based Forest Fire Detection System Using Sensors and Cloud Computing. International Journal of Computer Science and Information Security, 18(5), 56-60.