SMART FIRE DETECTION SYSTEM

Project Exhibition – 1

Submitted in partial fulfillment for the award of the degree of

Bachelor of Technology

In

Electronics and Communication ENGINEERING

(Ai and Cybernetics)

Submitted to

VIT BHOPAL UNIVERSITY (M.P.)

Submitted by

GOKUL-24BAC10062

HARSHVARDHAN BORGOHAIN-24BAC10024

NAVYA NARULA-24BAC10014

Under the Supervision of

DR.ABHAY VIDHYARTHI

SCHOOL OF ELECTRICAL & ELECTRONICS ENGG.

VIT BHOPAL UNIVERSITY (BHOPAL)

(M.P.)-466114

September 2025

VIT BHOPAL UNIVERSITY, BHOPAL

SCHOOL OF ELECTRICAL & ELECTRONICS ENGINEERING

DECLARATION

I hereby declare that the Dissertation entitled "**SMART FIRE DETECTION SYSTEM**" is my own work conducted under the supervision of *ABHAY VIDHYARTHI* Assistant Professor Grade 1, SEEE at VIT Bhopal University, Bhopal.

I further declare that to the best of my knowledge this report does not contain any part of work that has been submitted for the award of any degree either in this university or in other university / Deemed University without proper citation

GOKUL-24BAC10062

HARSHVARDHAN BORGOHAIN-24BAC10024

NAVYA NARULA-24BAC10014

VIT BHOPAL UNIVERSITY, BHOPAL

SCHOOL OF ELECTRICAL & ELECTRONICS ENGINEERING

CERTIFICATE

This is to certify that the work embodied in this Project Exhibition - II report entitled "**SMART FIRE DETECTION SYSTEM**" has been satisfactorily completed by Mr. **GOKUL**, **HARSHVARDHAN BORGOHAIN**, **NAVYA NARULA** Registration No. 24BAC10062, 24BAC10024, 23BAC10014 in the School of Electrical & Electronics Engineering of Electronics and communication engineering with specialization in AI and Cybernetics at VIT Bhopal University, Bhopal. This work is a bonafide piece of work, carried out under our guidance in the School of Electrical & Electronics Engineering for the partial fulfilment of the degree of Bachelor of Technology.

DR.ABHAY VIDHYARTHI

Senior Assistant Professor

Forwarded by

Dr. Soumitra K Nayak

Program Chair

Approved by

Dr. Suresh M

Dean of SEEE

ACKNOWLEDGEMENT

In the first place I would like to record my gratitude to"**DR.ABHAY VIDYARTHI**" for his supervision, advice, and guidance from the very early stage of this thesis work as well as giving me extraordinary experiences throughout the work. Above all and the most needed, he provided me unflinching encouragement and support in various ways. His words have always inspired me to work in an efficient and comprehensive way. I would like to thank him for his constant encouragement that enabled me to grow as a person. His presence has definitely improved me as a human being.

I express my gratitude towards "**Dr. M SURESH**", Professor and Dean of School of Electrical and Electronics Engineering Department, VIT Bhopal University, Bhopal, for providing me all the help and permitted me to work in the laboratory with no time limits.

I shall ever remain indebted to "Dr. **T B Sridharan**", Pro Vice Chancellor, **VIT Bhopal University**, for providing me institutional and administrative facilities during my project work at VIT.

GOKUL-24BAC10062

HARSHVARDHAN BORGOHAIN-24BAC10024

NAVYA NARULA-24BAC10014

Abstract:

This paper presents an abstract for a smart fire detection system that leverages a multi-sensor approach and machine learning to provide rapid, accurate, and low-cost fire alerts. Traditional smoke detectors often suffer from false alarms and limited range, making them less effective in modern, complex environments. Our proposed system addresses these limitations by integrating sensors for temperature, smoke (particulate matter), and carbon monoxide (CO).

➢ Table of Contents   Page no

➢ Contents   Page no

CHAPTER1: INTRODUCTION

MOTIVATION:

Fire safety is essential for protecting lives and property. Traditional detection methods, which rely on sensors and alarms, often produce delayed alerts, which can lead to increased response times and greater potential loss. The limitations of these older methods highlight the need for a more responsive and integrated system

General Theory related to topic:

This project leverages the power of the Internet of Things (IoT) to enhance fire detection by providing real-time monitoring and instant alerts. This approach significantly reduces response times and improves overall emergency preparedness. Automated alerts from fire and smoke detection systems are crucial for effective emergency management as they minimize potential damage and provide real-time communication.

Objective

The objective of this project is to create an effective and responsive fire and smoke detection system. This system will utilize Arduino for sensor integration, Python for data processing, and the Twilio API to send immediate message alerts to users, thereby enhancing safety and response efficiency during emergencies

Brief Methodology:

The methodology involves three main steps:

1. System Design: Identify and integrate the necessary components, including smoke and flame sensors, an Arduino board, and the Twilio API.

2. Software Implementation: Develop and upload code to the Arduino to detect fire and smoke, and create a Python script to process sensor data and send alerts via the Twilio API.

3. Testing and Calibration: Conduct comprehensive testing to verify the system's performance, focusing on detection accuracy, false alarms, and the reliability of the message alerts. We will also calibrate the sensors to optimize performance and minimize false alarms.

Chapter 2: LITERATURE SURVEY

The literature survey for this project involves a review of previous studies on fire and smoke detection systems, IoT-based safety solutions, and the use of communication APIs like Twilio for emergency alerts. Thegoal is to understand existing technologies, identify their limitations, and justify the unique contribution of our proposed system.

[1] Mahor (2010): This study discusses the effectiveness of fire detection technologies in industrial settings, highlighting the importance of system reliability.

[2] Sharma (2016): This paper presented IoT-enabled safety systems with cloud-based monitoring but lacked real-time SMS alert integration.

[3] Verma and Singh (2019): They reviewed sensor-based fire detection systems and suggested integration with mobile communication APIs for better responsiveness.

From these studies, it is clear that while IoT-based fire detection systems exist, there remains a gap in terms of real-time SMS-based alerts that can ensure immediate communication. Our project addresses this gap by integrating Arduino, Python, and Twilio for timely and reliable notifications.

Chapter 3: PROBLEM FORMULATION AND PROPOSED METHODOLOGY:

3.1 Components Required

The system is built on three key components:
-Arduino Board: Central processing unit integrating sensor data.
-Smoke and Fire Sensors: Crucial for early hazard detection.
-Twilio API: Enables instant SMS notifications.

3.2 Circuit Diagram and Connections

The sensors are connected to the Arduino Uno with proper wiring for accuracy. The circuit ensures robust detection, and a backup power supply provides continuous operation even during outages.

3.3 Block Diagram

Input: Flame and smoke sensors → Processing: Arduino Uno → Data Handling: Python script → Output: Buzzer, LED, and Twilio SMS alerts.

3.4 Software Implementation

Arduino Programming: The microcontroller code detects fire/smoke and sends data to the connected system.
Python and Twilio Integration: Python scripts process Arduino inputs and interact with Twilio API to send SMS alerts. Twilio account credentials are securely configured for communication.

Chapter 4: RESULTS AND DISCUSSION

4.1 System Testing

Test scenarios involved simulating smoke and fire conditions. The system successfully detected hazards and triggered local alarms (buzzer/LED) while simultaneously sending SMS alerts.

4.2 Calibration of Sensors

Sensor sensitivity was fine-tuned to reduce false positives. Regular maintenance checks ensured long-term accuracy and reliability.

4.3 Real-World Applications

-Home Safety: Protecting residential spaces.
-Industrial Safety: Providing timely warnings in factories and offices.
-Public Buildings: Enhancing fire safety compliance.

4.4 User Feedback and Improvements

User trials indicated satisfaction with real-time alerts. Feedback suggested adding customizable notification settings and AI-driven predictions, which will be explored in future improvements.

Chapter 5: CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

This project successfully developed a fire and smoke detection system using Arduino, Python, and Twilio API. The integration of IoT with SMS alerts provides an efficient and cost-effective solution for enhancing fire safety.

5.2 Future Enhancements

-Incorporating AI/ML for predictive fire detection.
-Integrating with smart home systems for automated responses.
-Expanding system to include monitoring of air quality and other environmental factors.

References:

[1] Mahor, A (2010). Fire detection technologies in industrial applications. International Journal of Safety Engineering. vol. 7, no. 3

[2] Sharma, R (2016). IoT-enabled safety monitoring systems. IEEE IoT Journal. vol. 12, no. 1,

[3] Verma, P. and Singh, A. (2019). Sensor-based fire detection systems. Journal of Emerging Technologies. vol. 9, no. 3, pp. 112–117.

ARDIUNO CODE:

```
// Include the necessary libraries for the LCD and DHT11 sensor.
// You must install the "LiquidCrystal I2C" library by Frank de Brabander via the Library Manager.
// You also need the "DHT sensor library by Adafruit".
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>

// Define the pins for the hardware components
#define DHTPIN 7       // Digital pin for the DHT11 sensor data pin
#define BUZZER_PIN 8   // Digital pin for the buzzer's I/O pin
#define MQ2_PIN A0     // Analog pin for the MQ2 sensor's A0 pin

// Define thresholds for the alarm
// Adjust these values based on your testing and environment
const int TEMP_THRESHOLD = 30;  // Temperature in Celsius
const int SMOKE_THRESHOLD = 300; // Analog smoke value (0-1023)

// Initialize the LCD with its I2C address and dimensions (16 columns, 2 rows).
// A common I2C address is 0x27, but if it doesn't work, try 0x3F.
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Initialize the DHT11 sensor
#define DHTTYPE DHT11
```

```
DHT dht(DHTPIN, DHTTYPE);

// A variable to store the state of the alarm
bool alarmActive = false;

void setup() {
// Strt the serial communication for debugging
Serial.begin(9600);

// Set up the LCD
lcd.init();
lcd.backlight(); // Turn on the backlight
lcd.clear();
lcd.print("Fire Detection");
lcd.setCursor(0, 1);
lcd.print("System Ready");

// Initialize the DHT sensor
dht.begin();

// Set the buzzer pin as an output
pinMode(BUZZER_PIN, OUTPUT);
digitalWrite(BUZZER_PIN, LOW); // Ensure buzzer is off initially
}

void loop() {
```

```
// Read sensor values
float temperature = dht.readTemperature(); // Read temperature in Celsius
int smokeValue = analogRead(MQ2_PIN);    // Read smoke level from MQ2

// Check if sensor readings are valid
if (isnan(temperature)) {
Serial.println("Failed to read from DHT sensor!");
lcd.clear();
lcd.print("DHT Sensor Error");
lcd.setCursor(0, 1);
lcd.print("Check connections");
delay(2000);
} else {
// Check if the alarm conditions are met
if (temperature > TEMP_THRESHOLD || smokeValue > SMOKE_THRESHOLD) {
// Alarm condition met
alarmActive = true;
} else {
// Alarm condition not met
alarmActive = false;
}

// Update the display based on the alarm state
if (alarmActive) {
// Display the alarm message
Serial.println("!!! FIRE ALARM !!!");
```

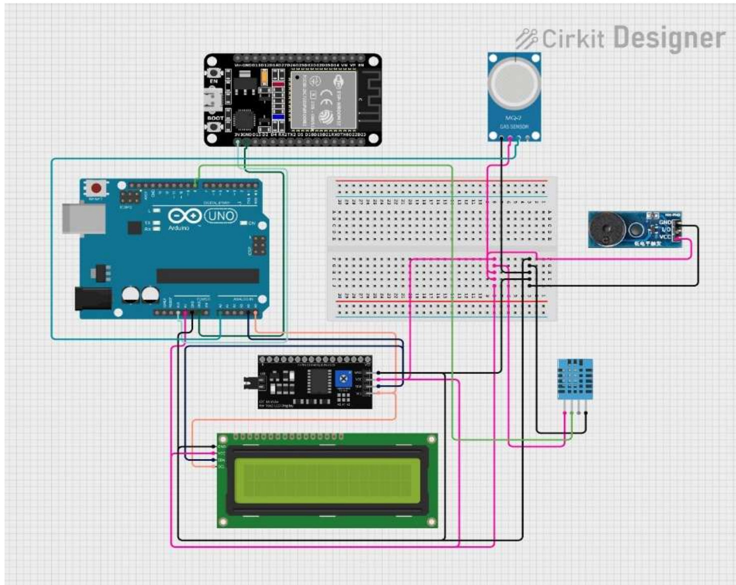
```
lcd.clear();
lcd.print("Fire Detected!");
lcd.setCursor(0, 1);
lcd.print("T:");
lcd.print(temperature);
lcd.print(" S:");
lcd.print(smokeValue);

// Flash the buzzer on and off to create a pulsing sound
digitalWrite(BUZZER_PIN, HIGH);
delay(500);
digitalWrite(BUZZER_PIN, LOW);
delay(500);

} else {
// Display normal status
Serial.println("System Normal");
lcd.clear();
lcd.print("Temp:");
lcd.print(temperature);
lcd.print(" C");
lcd.setCursor(0, 1);
lcd.print("Smoke:");
lcd.print(smokeValue);
digitalWrite(BUZZER_PIN, LOW); // Keep the buzzer off
}
```

```
}
// Delay for a short period before the next loop iteration
delay(1000);
    }
```

➢ CIRCUIT DIAGRAM:



➢ REFERENCE IMAGE :