ESP Journal of Engineering & Technology Advancements (ESP-JETA)
ISSN: 2583-2646
Special Issue: International Conference on Modern Engineering and Future Technologies (ICMEFT-26)

Original Article

IoT-Based Multi-Hazard Detection and Automated Mitigation System for Smart Underground Tunnels using ESP32

P. Ganesh Kumar*, Dr. M. Shahul Hameed*, C. Mathesh Kannan*
Department of Civil Engineering, P. S. R. Engineering College, Sivakasi, Tamilnadu-626140

Abstract: *This project presents an advanced IoT-driven safety framework specifically designed for real-time monitoring and hazard mitigation in underground tunnels using the ESP32 microcontroller. The system integrates a suite of sensors to detect critical threats such as toxic gas leakages, fire outbreaks, and structural flooding. Upon detecting a hazard, the ESP32 autonomously triggers immediate mitigation protocols, including automated ventilation, and fire suppression mechanisms. Furthermore, real-time environmental data is transmitted to a centralized cloud dashboard, enabling remote supervision and instant alert notifications for emergency responders. By replacing manual inspections with automated intelligence, this system significantly enhances the resilience of smart infrastructure and ensures the safety of commuters. This scalable solution offers a cost-effective and highly responsive approach to managing the complex risks associated with subsurface transportation networks.*

Keywords: *IoT, ESP32, Multi-Hazard Detection, Automated Mitigation, Smart Tunnel Safety.*

I. INTRODUCTION

A. Background

The rapid expansion of urban transportation and utility networks has led to an increasing reliance on underground tunnels for railways, and highways systems. While these structures optimize surface space, they are inherently high-risk environments. A minor incident, such as a gas leak, a small electrical fire, can quickly escalate into a catastrophic disaster due to limited accessibility, poor natural ventilation, and restricted evacuation routes. Traditional safety measures often rely on manual inspections or isolated alarm systems, which are frequently too slow to prevent major damage or loss of life.

B. The Problem Statement

Current tunnel monitoring systems often operate in fire detection is separate from air quality monitoring. Furthermore, most systems are purely reactive they alert an operator but do not take immediate corrective action. In a tunnel environment, the "Golden Hour" for response is measured in seconds. There is a critical need for an integrated, intelligent system that can not only sense multiple hazards simultaneously but also initiate automated mitigation protocols without waiting for human intervention.

C. Proposed IoT Solution

This project introduces an IoT-Based Multi-Hazard Detection and Automated Mitigation System powered by the ESP32 microcontroller. By leveraging the Internet of Things (IoT), the system transforms a passive tunnel into a "Smart Tunnel." The ESP32 acts as the central brain, interfacing with a diverse array of sensors including MQ-series gas sensors, flame sensors, to maintain a 360-degree digital vigil over the environment.

D. System Functionality and Mitigation

The core innovation of this system lies in its automated response logic. Unlike standard monitoring tools, this system is programmed to act:

Fire Hazard: The system uses advanced smoke and temperature sensors to detect fire at an early stage. Once the threshold is exceeded, an automatic sprinkler or fire suppression unit is activated instantly. Simultaneously, alarm systems such as buzzers and flashing lights are triggered to warn nearby occupants. The system also sends emergency alerts to the control room for immediate action. This quick response helps in minimizing fire spread and reducing damage. It improves overall safety in enclosed environments like parking areas and basements.

Gas Leakage: Gas sensors continuously monitor the presence of harmful or flammable gases in the environment. When gas concentration exceeds safe limits, high-speed exhaust fans are automatically activated. This helps in quickly diluting and removing the hazardous gases from the area. The system also triggers warning alarms to alert people nearby. In critical situations, the electrical supply can be cut off to prevent explosions. This ensures a safe and breathable environment.

II. METHODOLOGY

The development of the IoT-Based Multi-Hazard Detection and Automated Mitigation System follows a systematic approach, integrating multi-sensory data acquisition with edge-based decision-making. The methodology is divided into four primary phases: System Architecture Design, Hardware Integration, Software Logic & Threshold Calibration, and IoT Cloud Implementation.

A. System Architecture Design

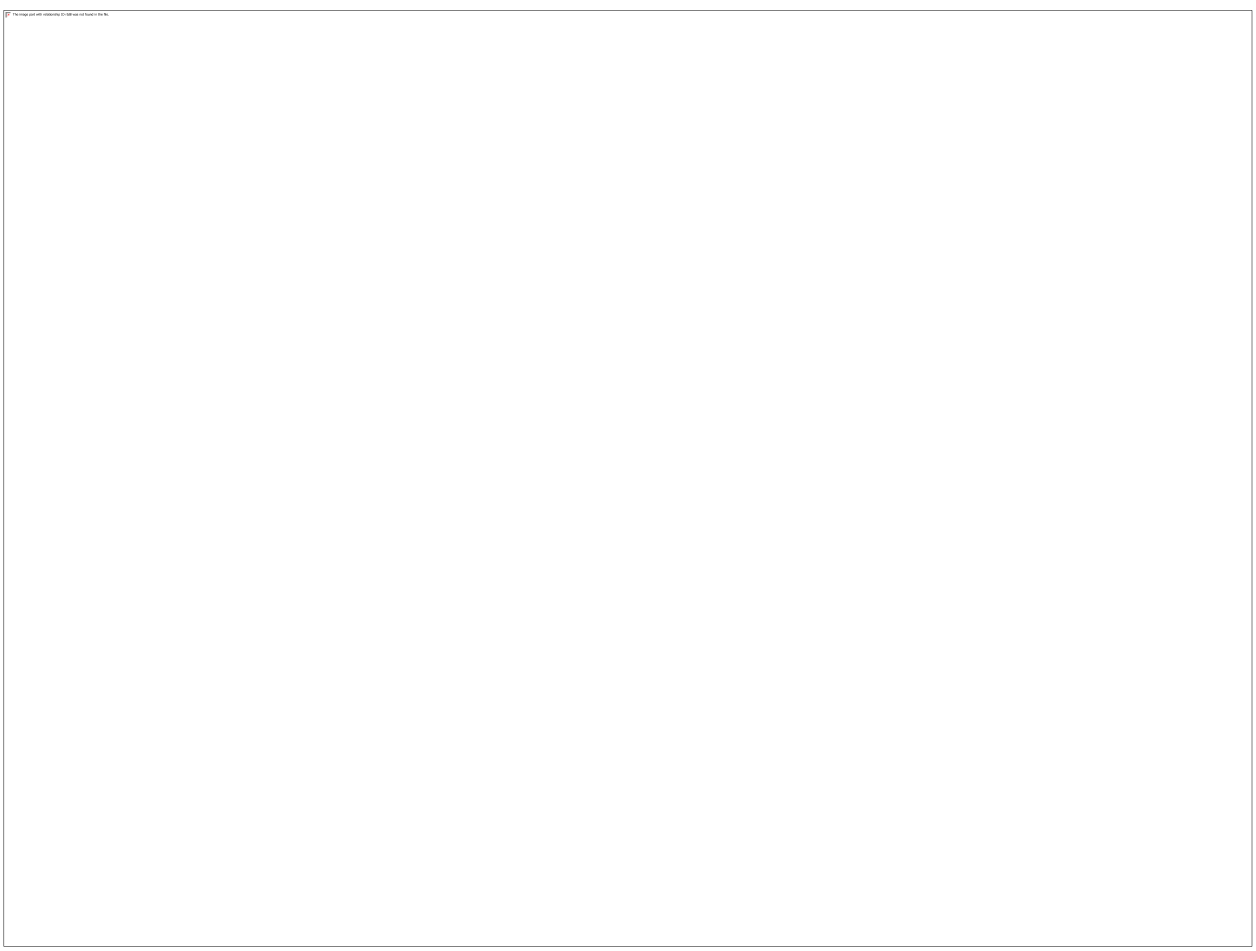
The architecture is based on a Sense-Think-Act model.

a) Sensing Phase

The Sensing Phase is the initial stage of a monitoring and control system where all connected sensors continuously collect real-time environmental data. In this phase, sensors such as flame sensors, gas sensors, temperature sensors, and water level sensors detect any abnormal conditions in the surroundings.

b) Thinking Phase

The Thinking Phase is the processing stage of the system where the microcontroller analyzes the data received from all sensors. In this phase, the ESP32 compares real-time sensor values with predefined threshold limits to identify whether a normal or hazardous condition exists.



c) Acting Phase

The Acting Phase is the final stage of the system where the control actions are physically executed based on the decisions made in the thinking phase. In this stage, the microcontroller sends output signals to various actuators such as relays, pumps, fans, alarms, and sprinklers.

B. Hardware Selection and Integration

The Hardware Selection and Integration phase involves choosing suitable electronic components and combining them to build a complete working system. The selection is based on factors such as cost, reliability, power consumption, and compatibility with the microcontroller (ESP32 or Arduino).

a) ESP32 NodeMCU

ESP32 NodeMCU is a low-cost, high-performance microcontroller with built-in Wi-Fi and Bluetooth capabilities. It is widely used in IoT applications for real-time data monitoring and control. The board supports multiple input/output pins, making it suitable for interfacing with sensors and actuators.

b) Multi-Hazard Sensor Array

A Multi-Hazard Sensor Array is an integrated system that consists of multiple sensors designed to detect different types of hazards such as fire, gas leakage, and flooding simultaneously. Instead of using separate systems for each risk, this array combines all sensors into a single platform, improving efficiency and response time.

c) Flame Sensor

A Flame Sensor is an electronic device used to detect the presence of fire or flame in an area. It works by sensing infrared (IR) light emitted by flames within a specific wavelength range. When a flame is detected, the sensor generates an electrical signal that can be read by a microcontroller like ESP32 or Arduino.

d) Mitigation Actuators

Mitigation actuators are the output devices in a safety system that take action when a hazard is detected. They work based on signals received from the controller (such as ESP32 or Arduino) after processing data from sensors. These actuators help in reducing or controlling the impact of hazards like fire, gas leakage, or flooding.

e) Relay Module

A Relay Module is an electrical switching device used to control high-voltage appliances using a low-voltage microcontroller like ESP32 or Arduino. It acts as a bridge between the control circuit and high-power devices such as pumps, fans, and lights.

f) DC Exhaust Fan

A DC Exhaust Fan is a direct current (DC) powered fan used to remove unwanted gases, smoke, and hot air from an enclosed area. It operates using low voltage (commonly 5V, 12V, or 24V), making it suitable for microcontroller-based systems like ESP32 or Arduino.

g) Piezo Buzzer

A Piezo Buzzer is an electronic sound-producing device used to generate an audible alert when activated by a microcontroller such as ESP32 or Arduino. It works based on the piezoelectric effect, where electrical signals are converted into mechanical vibrations to produce sound.

III. WORKING PROCESS

A. Power Management and System Calibration

The operational lifecycle begins with the power-on sequence. The ESP32, being the central processing unit, requires a stable 5V DC input, which it internally regulates to 3.3V for its logic circuits. Upon startup, the system enters a Calibration Phase. Sensors like the MQ-135 (Gas sensor) require a pre-heating period to stabilize the internal heater element. During the first 30–60 seconds, the ESP32 ignores erratic readings to prevent "phantom" alarms. Simultaneously, the dual-core processor initiates the Wi-Fi stack on Core 0, establishing a handshake with the local gateway to ensure that the IoT telemetry channel is open before the safety monitoring begins.

B. ByMulti-Sensory Data Acquisition Layer

Once the system is stable, it enters the Continuous Sampling Loop. The ESP32 utilizes a multi-channel approach to monitor the tunnel environment:

Atmospheric Analysis: The MQ-series sensor operates on the principle of variable resistance. As toxic gas molecules (CO, or Smoke) interact with the tin-dioxide layer of the sensor, the conductivity increases. The ESP32 reads this as an analog voltage through its ADC (Analog-to-Digital Converter), converting the voltage into PPM (Parts Per Million) values.

C. Optical Fire Detection:

The Flame Sensor uses an infrared (IR) receiver specifically tuned to detect light wavelengths between 760nm and 1100nm, which are characteristic of a flickering flame. Unlike standard light sensors, it is designed to ignore ambient tunnel lighting, focusing only on the spectral signature of fire.

D. Edge-Based Intelligence and Decision Matrix

The core "intelligence" of the system lies in its Edge Computing capability. Instead of sending raw data to the cloud and waiting for instructions (which would cause dangerous delays), the ESP32 processes the data locally using a Priority-Based Decision Matrix.

E. Automated Mitigation and Actuation

The Automated Mitigation and Actuation phase is responsible for executing safety actions automatically when a hazard is detected. Once the microcontroller identifies an abnormal condition from the sensor data, it immediately sends control signals to the respective actuators without any human intervention.

a) Ventilation Activation

If smoke or gas is detected, the ESP32 sends a High signal to the Relay Module. The relay acts as an isolated switch, closing the high-current circuit that powers the DC Exhaust Fans. These fans work to clear the tunnel atmosphere, increasing visibility for drivers and preventing asphyxiation.

b) Auditory Alerting

Simultaneously, a Passive Buzzer is activated. It generates a high-decibel warning tone, providing an immediate localized alert for any pedestrians or maintenance workers inside the tunnel.

Flood Control: If the water sensor detects flooding, the system is programmed to activate a drainage pump (simulated via the relay in this project) to remove excess water before it reaches the vehicle engine height.

IoT Cloud Telemetry and Remote Supervision

While the local system handles the immediate physical danger, the IoT Framework ensures that the incident is managed at a macro level.

Data Encapsulation: The ESP32 packages the current sensor values, hazard status, and uptime into a JSON (JavaScript Object Notation) format.

Wireless Transmission: Using the MQTT (Message Queuing Telemetry Transport) protocol—chosen for its low bandwidth and high reliability—the data is pushed to a cloud platform like Blynk, ThingSpeak, or Adafruit IO.

IV. CONCLUSION

The development of the IoT-Based Multi-Hazard Detection and Automated Mitigation System marks a significant advancement in the integration of smart technology with critical infrastructure safety. By leveraging the dual-core processing capabilities of the ESP32 microcontroller, this project successfully demonstrated a robust framework for real-time environmental monitoring and autonomous hazard response in underground tunnels. The core success of this system lies in its ability to operate at the "Edge." Unlike traditional systems that may suffer from latency due to cloud dependency, this system ensures that life-saving actions such as smoke extraction, audible alerting, and are executed locally and instantaneously. The experimental results confirmed that the system can detect fire, toxic gas leaks,triggering the necessary actuators within seconds of a threshold breach.