Design and Implementation of an Internet of Things (IoT) Based Automatic Fire Fighting Sprinkler System

Abode Harry Ojata[1], and Okpanachi Benjamin[2]

[1]Department of Science Laboratory Technology (SLT), Auchi Polytechnic, Auchi, Edo State, Nigeria.
[2]Department of Physics with Electronics, Auchi Polytechnic, Auchi, Edo State, Nigeria.
*Corresponding author email: Email: abode_harry@yahoo.com; +2348060165109

Direct Research Journal of Engineering and Information Technology

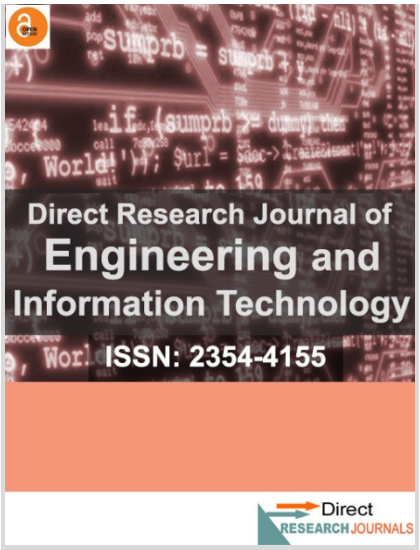
Vol. 14(1), Pp. 12-26, January 2026,
Author(s) retain the copyright of this article
This article is published under the terms of the Creative Commons Attribution License 4.0.
Research Article
ISSN: 2354-4155
https://journals.directresearchpublisher.org/index.php/drjeit; https://www.ajol.info/index.php/drjeit

Received 5 December 2025, Accepted 10 January 2026, Published 20 January 2026

ABSTRACT

The frequent incidence of fire accidents in homes, work places, industrial buildings and the limitations of conventional fire suppressor system demand the development of intelligent and responsive fire-fighting solutions. This research work, design and Implementation of an IoT based automatic fire fighting sprinkler system is aimed at improving response time, reliability, and remote monitoring compared to conventional fire protection systems. The system employs a combination of temperature, smoke and flame sensor interfaced with a microcontroller (ESP8266) to continuously monitor environmental conditions. When fire indicators exceed preset safety thresholds of above 60^{0}C, the controller automatically triggers a sprinkler mechanism through a relay- driven solenoid valve to suppress the fire. In addition, the system transmits real-time alerts and sensor data to a cloud platform (Bylink) through a wireless communication module, allowing users to monitor conditions and receive notifications from remote locations. The prototype was developed using avoidable and energy efficient components and evaluated under controlled fire simulation. The result shows that the system responds promptly to fire events which activates the sprinkler effectively, and provides reliable real- time monitoring. The constructed IoT based fire- fighting sprinkler system demonstrates improved efficiency, scalability, and suitability for home and industrial fire safety applications.

Keywords: *Fire detection, Internet of things (IoT), Automatic fire sprinkler system, embedded system, GSM alert system*



Citation: Abode, H. O., & Okpan, A. B. (2026). Design and Implementation of an Internet of Things (IoT) Based Automatic Fire Fighting Sprinkler System. *Direct Research Journal of Engineering and Information Technology*, 14(1), Pp.12-26.
https://doi.org/10.26765/DRJEIT24316899

INTRODUCTION

Every day we hear of buildings being destroyed leading to loss of lives and properties. Often times, these destructions are caused by fire outbreak which could have been managed before becoming uncontrollable. In Borno state, Nigeria, for example Between November 2023 and March 2024, 80 reported fire outbreaks affected a total of 8,814 households (HHs), representing 44,095 individuals across 35 IDP sites. 27 causalities were reported during these incidents. Cases recored represent an increasing trend, with February recording the highest incidents of 32 and March recording 23 incidents. The fire incident destroyed 8,857 shelters (emergency or self-made/makeshift units), with most of the affected population losing their belongings, including registration, food ration, and/or biometric identification cards, and food and non-food items. Fire hazards continue to present

significant threats to human life, infrastructure, and economic assets across residential, commercial, and industrial environments. Despite stringent building safety standards and the widespread adoption of traditional fire-fighting sprinkler systems, many fire incidents still result in extensive damage due to delayed detection and response, inadequate situational awareness, and limited automation in conventional systems (Issaka et al., 2025). Conventional sprinkler systems typically rely on thermal activation mechanisms that may not detect fire early enough or provide real-time information to stakeholders, thereby leading to slower intervention times and greater losses.

The emergence of the Internet of Things (IoT) defined as a network of interconnected sensing and communication devices capable of real-time data collection, processing, and transmission has revolutionized the design of safety and emergency response systems. IoT-enabled fire safety systems integrate sensors, microcontrollers, communication modules, and cloud analytics, enabling real-time monitoring, intelligent decision-making, and automated actuation of fire-fighting components such as sprinklers (Mohamad & Khalid, 2024; Issaka et al., 2025). By leveraging IoT technologies, fire suppression systems can detect the earliest physical indicators of fire—such as smoke, heat, and flame while minimizing false alarms and optimizing resource deployment.

Recent research has demonstrated significant advancements in IoT-based fire detection and suppression architectures. For instance, a study by Mohamad and Khalid (2024) developed an IoT-based fire safety detection and suppression control system that utilized multiple sensors to actively detect fire conditions and trigger both alerts and sprinkler actuation, thereby enhancing situational awareness and response effectiveness. Similarly, Issaka et al. (2025) implemented a scalable IoT-integrated fire detection and suppression system with real-time monitoring and actuator control for water discharge, significantly improving detection accuracy through sensor fusion techniques and automated sprinkler control.

The adoption of IoT in fire-fighting sprinkler systems enhances early fire detection, remote monitoring, automated control, and real-time alerting, thereby improving overall system reliability and response time (Al-Fuqaha et al., 2015; Ray & Doe, 2018). An IoT-based fire-fighting sprinkler system typically incorporates sensors such as temperature, smoke, flame, and gas sensors interfaced with a microcontroller (e.g., Arduino or ESP32). These sensors continuously monitor environmental conditions and transmit data to cloud platforms via wireless technologies such as Wi-Fi or GSM. When abnormal fire indicators exceed predefined thresholds, the system autonomously activates the sprinkler mechanism and simultaneously sends alerts to building occupants and emergency responders through mobile or web applications (Chauhan et al., 2019; Kumar & Lee, 2023).

Compared to conventional systems, IoT-enabled fire sprinkler systems offer advantages such as reduced response latency, improved situational awareness, remote accessibility, and data logging for post-incident analysis. Furthermore, intelligent control algorithms can ensure targeted water discharge, minimizing water damage and improving sustainability (Zeng et al., 2025). These features make IoT-based fire-fighting solutions particularly suitable for smart buildings and smart cities, where safety, efficiency, and automation are critical requirements. Therefore, the design and implementation of an IoT-based fire-fighting sprinkler system is a significant step toward enhancing modern fire safety infrastructure. The aim of this study is focused on developing a cost-effective, reliable, and scalable IoT-enabled sprinkler system that can detect fire incidents promptly, initiate automatic fire suppression, and provide real-time notifications to occupants and property owners

Objectives

The research was aimed at the following purposes

1. To design an embedded fire detection system using temperature sensors
2. To integrate GSM and IoT communication modules
3. To construct and demonstrate a prototype sprinkler system
4. To evaluate the system's functional response during fire simulation

LITERATURE REVIEW

The integration of Internet of Things (IoT) technologies with fire protection systems has gained increasing research attention due to its potential to improve detection accuracy, automation, and real-time remote monitoring compared to traditional fire safety systems (Alqourabah et al., 2020; Issaka et al., 2025). Traditional sprinkler systems rely largely on localized thermal or pressure activators that may lack advanced situational awareness and communication capabilities; by contrast, IoT-enabled systems leverage networked sensors and controllers to provide proactive, intelligent fire response mechanisms.

Conventional Fire Sprinkler Systems

Conventional fire sprinkler systems have long been recognized as one of the most effective active fire protection measures for buildings, particularly in residential, commercial, and industrial environments (Figure 1). These systems operate based on fixed mechanical and thermal principles, activating automatically when a predetermined temperature threshold is exceeded. Unlike modern smart or IoT-based

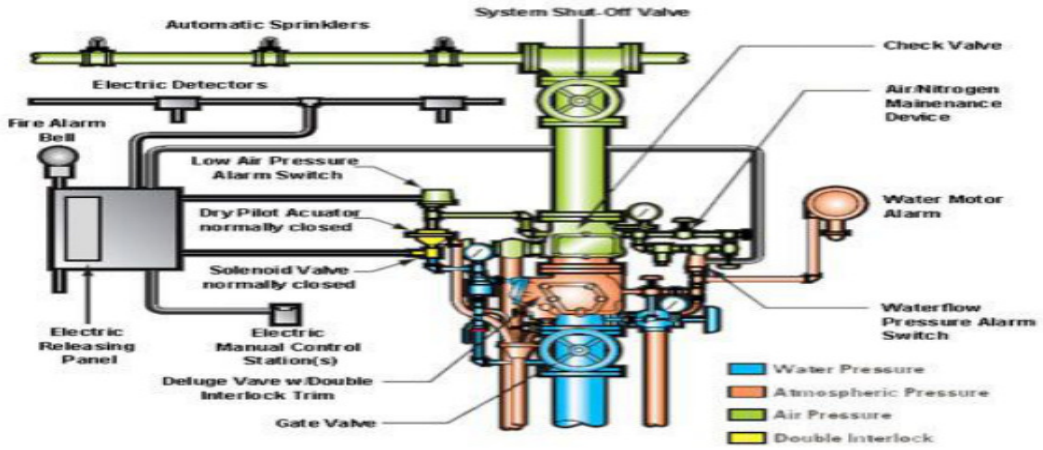


Figure 1: Schematic diagram of a conventional sprinkler system (Adapted from NFPA 13, 2022)

fire suppression systems, conventional sprinkler systems rely on heat-sensitive elements and hydraulic water distribution networks to control or extinguish fires at their early stages (Hurley et al., 2016). Extensive empirical research and fire incident statistics demonstrate the effectiveness of conventional fire sprinkler systems in controlling fires and enhancing occupant safety. According to reports by the National Fire Protection Association, sprinklered buildings experience significantly lower fire fatality rates and reduced property losses compared to unsprinklered structures (Ahrens, 2021).

IoT Fire Detection and Sprinkler Control

Modern IoT-based fire systems typically consist of distributed sensor nodes (smoke, heat, flame) connected to microcontrollers or IoT modules that communicate events to cloud servers or local dashboards. For instance, Issaka et al. (2025) describe a system integrating an ATmega328p microcontroller, RF transceivers, flame and smoke sensors, and a Node MCU (ESP8266) for data transmission, where the system uses sensor fusion to minimize false alarms and triggers water pumps and sprinklers upon confirmed fire detection. The system also supports real-time monitoring via Wi-Fi and a remote web server, illustrating how IoT integration enhances both detection and actuation functions (Issaka et al., 2025).

Similarly, the design presented by Juliaz and Maulana (2025) implements a Node MCU ESP8266-based fire detector integrated with water sprinklers, LCD displays, and relay circuits. Their research highlights how the combination of IoT communication and sensor feedback facilitates rapid automated response while sending alerts to administrators, demonstrating the practical viability of IoT-based firefighting hardware.

Sensor and False Alarm Mitigation

A key challenge in automated fire systems is balancing sensitivity with reliability. Multi-sensor fusion has been used in IoT designs to reduce false triggers that occur with single-modality sensors. Alqourabah et al. (2020) implemented a combination of heat, smoke, and flame detectors with algorithmic filtering and GSM communication to objectively evaluate fire presence before activating a water sprinkler subsystem. By processing multiple inputs and validating fire likelihood through logical checks, their system improved responsiveness and reduced erroneous activations, which is essential in automated sprinkler applications where unnecessary water discharge can cause significant damage.

GSM Alert System

GSM (Global System for Mobile Communications) alert systems are communication-based notification platforms that use cellular networks to transmit alerts, warnings, or status updates to predefined recipients in real time. These systems are widely adopted in security monitoring, healthcare, industrial automation, disaster management, and safety-critical applications due to their broad network coverage, reliability, and low deployment cost. SMS remains the most widely used communication method in GSM alert systems because of its simplicity, low cost, and independence from data services. Studies show that SMS-based alerts offer acceptable latency for most safety and monitoring applications, typically ranging from 5 to 15 seconds depending on network conditions (Patel et al., 2019). Voice call alerts are sometimes employed in critical scenarios where immediate attention is required, while packet-based GSM data services (GPRS/EDGE) are used in systems that require continuous monitoring and data logging. GSM alert systems have been extensively applied across various domains. In security and surveillance, they are used for intrusion detection and access control, where alerts are sent to property owners upon unauthorized entry (Singh et al., 2021). In healthcare, GSM-based patient monitoring systems transmit vital signs and emergency notifications to caregivers, improving response time and

patient safety (Kumar & Lee, 2020). Industrial applications include equipment fault monitoring, power failure alerts, and environmental hazard detection. In fire safety systems, GSM alert modules are integrated with smoke or temperature sensors to notify occupants and emergency responders when fire conditions are detected (Abdullahi et al., 2022). These applications highlight the versatility and scalability of GSM alert systems.

Design Considerations and System Implementation

Design methodologies in IoT fire systems emphasize modularity, scalability, and integration with cloud infrastructure. A study by International Journal of Trend in Research and Development (2025) outlined the implementation of a smart home automatic fire protection system using LM35 temperature sensors, relay circuits, and IoT connectivity to monitor room temperatures and actuate pumps for water flow, highlighting how embedded sensor systems can be adapted for smart building contexts. Ahmed et al., (2015), published in the springer international journal 'Design and fabrication of an Automatic Sprinkler fire Fighting System' The project integrated a microcontroller into a smoke detector circuitry and other components for safety purpose. The system activates an alarm to alert the occupants of the building once it detects a smoke or fire in the room and automatically releases water through the water valves. This work integrates, Ahmed et al. (2015) which underline the broad design spectrum of IoT fire protection systems from hardware integration and real-time communications to simulation, validation, and sensor fusion strategies. While promising results support IoT- enhanced detection and automated sprinkler activation, issues such as network reliability, energy consumption, and cyber security remain essential considerations for deployment in safety-critical environments.

COMPONENTS AND METHODOLOGY

Components Used (HARDWARE)

a) Aduino Microcontroller
b) ESP8226 Wi-Fi Module
c) diodes (Current diodes)
d) Infra-red Sensor (IR)
e) Smoke sensor(MQ- 135)
f) Solenoid Valve
g) Sprinkler head
h) . Relay (12volts, 12amps single pole).
i) Battery (12V/7A).
j) Charge controller.
k) . Variable resistor
l) . Resistors (4.7KΩ and 100Ω)
m) Capacitors (10µf 25V, 2,200µf 50v, 10µf-1000µf).
n) Water Collectors.
o) . Transistor (NPN).
p) Voltage regulators{ LM 7812(12Volts) & LM 7805(5Volts)}
q) LED (Light Emitting Diode)
r) Connecting wires.
s) Pumping machine.
t) Water tank/container.
u) Pipes
v) Air sucker(exhaust Fan)

Arduino Microcontroller

The Arduino Uno which makes use of the ATmega3289P microchip controller is the most well-known Arduino series microcontroller. It is used in this study as a main controller that controls the entire system including data communication flow and transmission. Furthermore, the Arduino Uno reduces power consumption by allowing current to flow through the sensor and entire system. It coordinates and controls the Wi-Fi module used for internet connectivity. The Arduino Uno is a free computing platform that enables the communication of various modules

ESP8226 Wi-Fi Module

The Wi-Fi ESP8266 module is a 32-bit controller-integrated wireless enable system on-chip (SOC). The Wi-Fi module sends real time information when fire is detected by the infrared sensor to the web server regularly. This allows the occupants of the building get notification in time to exit the building and also notify fire fighters to respond early enough to completely put out the fire. It offers Wi-Fi networking solutions for various Internet of Things (IoT) applications. It uses the full TC/IP protocol stack and supports the 802.11/b/g/n, 2.4 GHz, WI-Fi, P2P, and WPA/WPA2 protocols. The Wi-Fi module now features a GIPO pin to interface with various sensing devices.

Solenoid Valve

Ian automatic fire sprinkler system, the solenoid valve plays a crucial role. It's an electrically controlled valve that regulates water flow to the sprinkler system. When fire is detected, the fire alarm system sends an electrical signal to the solenoid valve. This makes the solenoid valve to open, allowing water to flow into the sprinkler head

Infrared Sensor (IR)

The IR sensor detects the heat or flames (Infrared radiation) emitted by the fire. When a fire is detected, the IR sensor sends a signal to the systems control panel. The control panel processes the signal and triggers the sprinkler system to activate

Smoke Sensor (MQ-135)

They are used for early detection of fire.

When smoke enters the sensor, it triggers an alarm. The sensor sends a signal to the system control panel. The control panel then processes the signal and then triggers the sprinkler system to activate, releasing water to suppress the fire.

Buzzer Chip and DC Power Cable

A buzzer is a sound notification alarm system activated when fire, heat above 60^0C, are detected. This helps to notify the occupants of the building in time. The system is powered by a 12V battery connected to a solar panel for constant supply of power.

METHODOLOGY

The steps necessary to develop the system are explained in this section. The design element and the study of each component that make up this device are the key topics covered in this section. It describes the standard by which the circuit's part is selected to guarantee the system's dependability and correct operation. Additionally, it demonstrates the connections between the circuit's various components. The design of this work is such that three white casing representing room, A, B and C respectively. They also represent source of fire. The aim of this study is improving response time, reliability, and remote monitoring compared to conventional fire protection systems. The system is made up of fire detection section, a sprinkler which opens up once the solenoid valve is activated to put out the fire once it is detected in each of the rooms (casing A, B and C).The system WI-FI module sends real time information when fire is detected by the infrared sensor to the web server and subsequently, to the owner of the building. The systems are connected to the same Arduino Nano board. The automatic IoT based fire sprinkler system involves an Infrared sensor that uses thermal detection technique to detect heat emitted by objects or flame. The Infrared is placed about 20cm away from the rooms. Once the fire starts from any of the room, the pumping machine inside the water container is activated and water flows through the pipes connected to each room (casing representing the rooms as well as fire source). Each room has a sprinkler head connected to a solenoid, this helps in distributing water directly to the rooms to put out the fire. The response time at which the water gets to each room is about 6.0seconds.With the buzzer and LED present, occupants are alerted whenever fire is detected. The methodology of this work is the process of designing and developing the hardware and software components of an IoT automatic fire fighting System, as well as the interaction and communication mechanisms between the system and owner of the building .The common steps involved here include power specification determination, circuit design, microcontroller and infrared sensor, microcontroller and GSM module etc.

Power Specification

A direct current DC-DC converter battery of 12V/7A was used. A DC-DC converter is an electronic device that changes one level of direct current (DC) voltage to another DC voltage level so that battery or DC power source can safely and efficiently power different loads or charge other batteries. In the first level, the DC-DC provides a constant voltage of 5V and a current of 5A to help power the microcontroller, chips etc. While in the second level, it uses a 12V and current of 7A to power the, solenoid valve, pumping machine, LED, Buzzer, sprinkler etc. Figure 2 shows the power circuit of the system. A 100Ω resistor is connected to the converter output, the high brightness indicator LED and the ground. This resistor helps to limit excessive current to reach the LED. The high brightness LED (Red and Blue) indicator has a forward voltage drop of (Red is 1.8V, Blue is 3.0V), which means that it needs at least 3.0V across its terminals to turn on. This LED (high brightness) has maximum current rating of 30mA,
Calculation of the Resistance value
The resistor value is therefore,

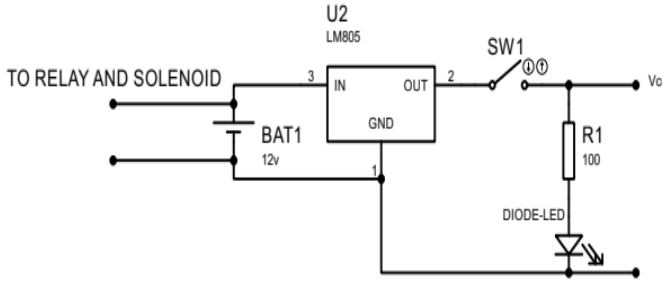


Figure 2: Circuit diagram of the powr section.

I_{min} = 20mA, V_d =2.0V

R= $\frac{V}{I} = \frac{2}{0,02} = 100\Omega$

Since, the calculated value of the resistor is equal to the maximum rating of the resistor, this shows the current through the LED indicator is the same with that of maximum rating.

Design of Circuit

Circuit Design

This subsection explains the various electronic components that were put together to make up the entire system. The block diagram of the system is a visual representation that shows the parts of the components and how they connected or interact with each other. The blocks diagram in the Figure 3 below, shows the interconnection

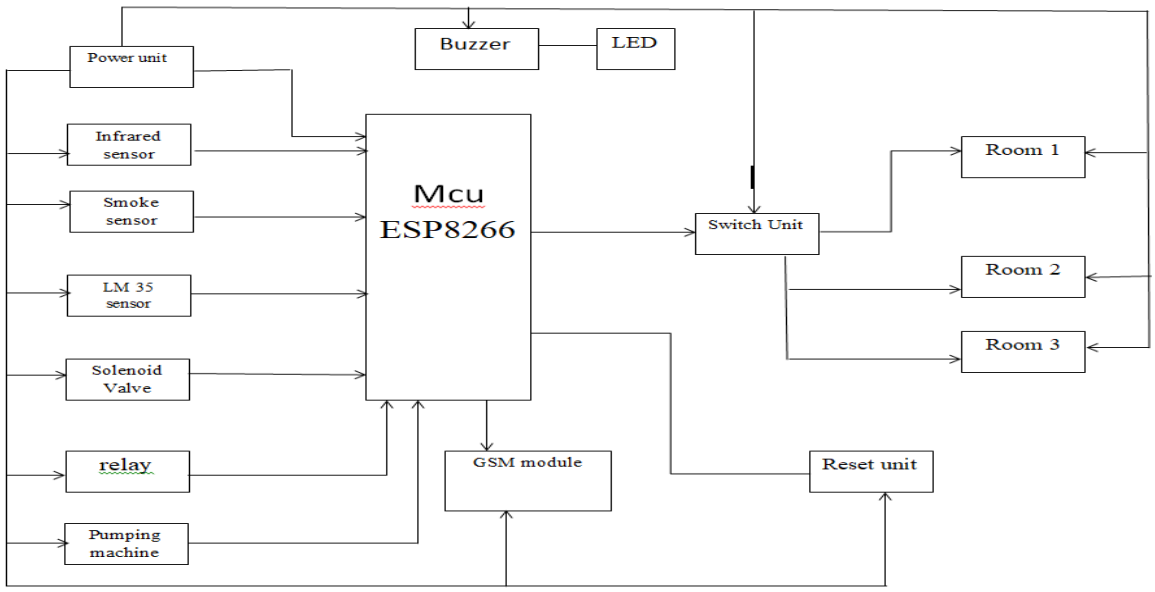


Figure 3: Block diagram for the automatic fire sprinkler fighting system.

of the units that makes up the IoT based automatic fire fighting system.

Microcontroller Section

The ESP8266 is a low-cost, low-power system-on-chip (SoC) microcontroller developed by Espressif Systems, primarily designed for wireless Internet of Things (IoT) applications. It integrates a 32-bit Tensilica L106 RISC processor operating at clock speeds of up to 80 MHz (and optionally 160 MHz), along with a full TCP/IP protocol stack and IEEE 802.11 b/g/n Wi-Fi functionality on a single chip, making it suitable for standalone embedded and networked systems (Espressif Systems, 2023). The ESP8266 includes on-chip peripherals such as general-purpose input/output (GPIO) pins, universal asynchronous receiver-transmitter (UART), serial peripheral interface (SPI), inter-integrated circuit (I²C – software implemented), pulse-width modulation (PWM), timers, and a 10-bit analog-to-digital converter (ADC). It supports external Flash memory for program and data storage, allowing flexible firmware development using environments such as the Arduino IDE, ESP-IDF, and Lua-based NodeMCU firmware (Kolban, 2022). One of the key advantages of the ESP8266 is its low power consumption and support for multiple power-saving modes, including deep sleep, which is critical for battery-powered IoT systems. These features enable efficient real-time data acquisition and wireless transmission while maintaining energy efficiency (Kolban, 2022). Due to its integrated Wi-Fi capability, the ESP8266 eliminates the need for external communication modules, reducing system complexity and overall cost.

Power supply unit

This unit is consisting of a DC-DC 12V/7A battery, charge controller, a solar panel which helps to supply constant power to the system. The battery having a current rating of 7A means it can deliver up to 7A of current to units such as the, relay, solenoid valve, pumping machine, buzzer, etc. The DC 12V/7A is converted to 5V/5A used to power the microcontroller and other chips without overheating or damaging them.

Solenoid Valve

The solenoid valve acts as an electrically controlled switch for water. When the IoT system detects fire (via sensors), the microcontroller sends an electrical signal to the solenoid. This opens the valve, allowing water to flow to the sprinklers. Once the fire threat is over or the system is turned off, the valve closes, stopping the water flow. The solenoid valve is connected to the microcontroller/relay module, which receives inputs from: Smoke sensors (detect smoke concentration), temperature sensors (detect high heat), and flame sensors (detect flame presence. This ensures automatic activation without manual intervention.

Sensor LM35

The heat sensor is typically a device that responds to rise in temperature. The heat sensor used IN4148 signal diode is a low voltage silicon diode, whose output voltage is

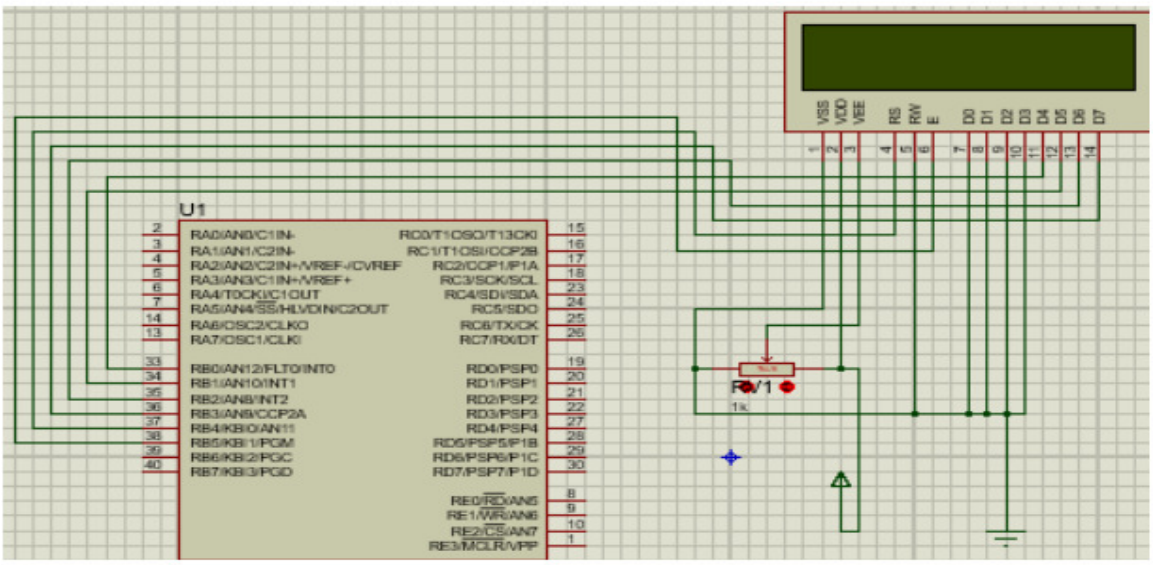


Figure 4: Schematic diagram of the microcontroller interface with LED

linearly proportional to the Celsius temperature. (Mohamad & Khalid, 2024). The IN4148 does not require any external calibration or trimming to provide typical accuracies. It is used with a single power supply, as it draws only 60UA from its supply. Other types of heat sensors include the LM35DZ, thermistors (NTC and PTC).

Microcontroller and GSM Module

The GSM Module in (Figure 3), used in this work is SM900A GSM module type. The GSM module is connected directly to the pin 0 (RX) and pin 1 TX serial port (SPI) of the microcontroller. This GSM module gets a high state signal from the microcontroller anytime there is a high temperature of about 165ºF occasioned by fire. Upon receiving a high state signal from the microcontroller, the GSM module calls and sends an SMS to the owner of the property, to alert them about a fire outbreak, so they can alert fire fighters to respond in time to completely put out the fire. A solenoid converts electrical energy into mechanical work through electromagnetic forces. For example, solenoids are often used as a valve to actuate a push or pull force on a magnetic component in a device. Solenoids can also act as a switch in electromechanical devices

Microcontroller and LED

The LED panel used in this work are high power LEDs (1watt, Red and Blue colours)) display type. This was interfaced directly to the microcontroller by connecting it to pin A4, pin 2, pin 3, pin 4, pin 5 and pin 6 of the microcontroller. The LED in conjunction with the buzzer alerts the occupants of the building. During implementation of this work, we programmed the LED panel activates once fire is detected and stops when the fire is OFF. Diagram is shown in (Figure 4).

Calibration Procedure

Calibration was carried out to ensure that the sensing units, actuation components, and communication modules of the IoT-based automatic fire sprinkler system operated within their specified accuracy limits. The procedure covered temperature sensors, smoke sensors, flame sensors, water-pressure components, solenoid valves, and the controller decision thresholds. All calibration activities were conducted under controlled laboratory conditions (Figure 5).

Software Development

An Arduino C language is written in the Arduino integrated development environment (IDE) for the proposed IoT based automatic fire fighting sprinkler system. The Arduino software is compatible with Windows, Mac OS X, and Linux. The Arduino code is designed to carry out the following instructions: (a) create an Arduino C code for the position identification module, (b) automate the opening solenoid valve (c) automate the pumping machine the pump water from the water (c) create Arduino create Arduino C code for the level identification module (d) create an Arduino C code for the alarm and LED notification system, (e) assign a unique IP address to each Room (f) create a communication link between the Wi-Fi module and each of the room. The system's code is written in Arduino C and runs on the Arduino IDE. Proteus software is also used for circuit design.

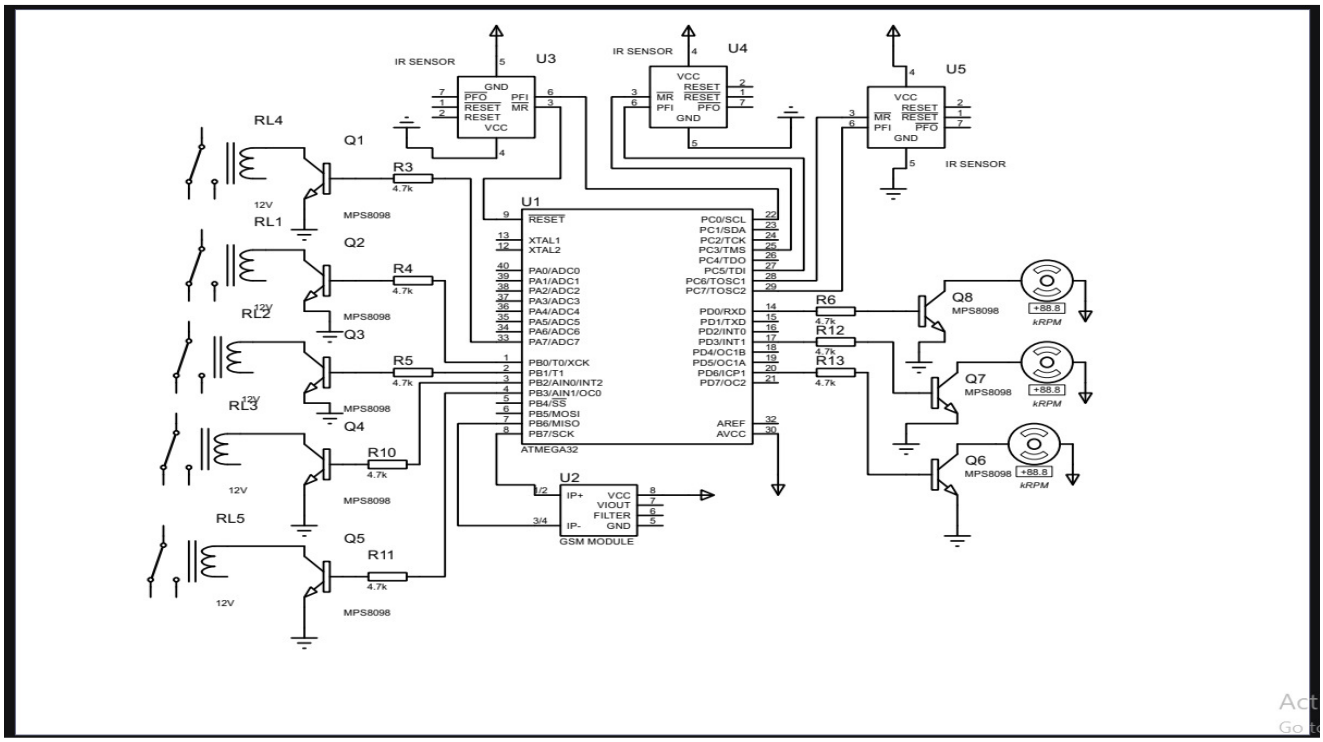


Figure 5: Circuit diagram for the entire automatic fire sprinkler system

The web user interface (Web UI) is written in the Java programming language.

Flowchart Description

Below is the fowchart for the IoT based automatic fire fighting sprinkler fighting system. See Source Code in Appendix

A [Start] --> B[System Initialization
• Configure pins
• Initialize sensors
• Setup valve/relay
• Setup buzzer/display]
B --> C[Connect to Wi-Fi / IoT Cloud]
C --> D{Connection Successful?}
D -- No --> C
D -- Yes --> E[Read Temperature Sensor]
E --> F[Read Smoke/Flame Sensor]
F --> G{Sensor Fault Detected?}
G -- Yes --> H[Log Error / Raise Fault Alert]
H --> E
G -- No --> I[Compare Temperature with Threshold]
I --> J[Compare Smoke/Flame with Threshold]
J --> K{Fire Detected?}
K -- Yes --> L[Activate Alarm/Buzzer]
L --> M[Turn ON Solenoid Valve
(Start Sprinkler)]
M --> N[Send IoT Alert
• App Notification
• SMS/Email
• Cloud update]
N --> O{Safe Condition Restored?}
O -- No --> M
O -- Yes --> P[Turn OFF Sprinkler and Alarm]
P --> Q[Log Event to Cloud]
K -- No --> R[Keep Sprinkler OFF]
R --> Q[Log Sensor Data to Cloud]
Q --> S[Delay / Wait Small Time]
S --> E

IMPLEMENTATION AND RESULT

System Implementation

The system implemantation was carried out using the circuit in (Figure7). This was connected to an Android App Interface. The three rooms(Room 1, Room 2 , Room 3) in the system where connected separately and tested. So that once all the units have given the desired results, they are then integrated to form the IoT based fier fighting sprinkler system. The response in Fgure 6a,6b (Android App interface) shows normal when there is no fire in any of the rooms. Figure 6c,6d indicates response when fire is detected in Room 1. Similar conditions applicable for other rooms.

TESTING AND RESULT

Assembly and testing

The various components used were tested using a multimeter for functionality and then soldered on a

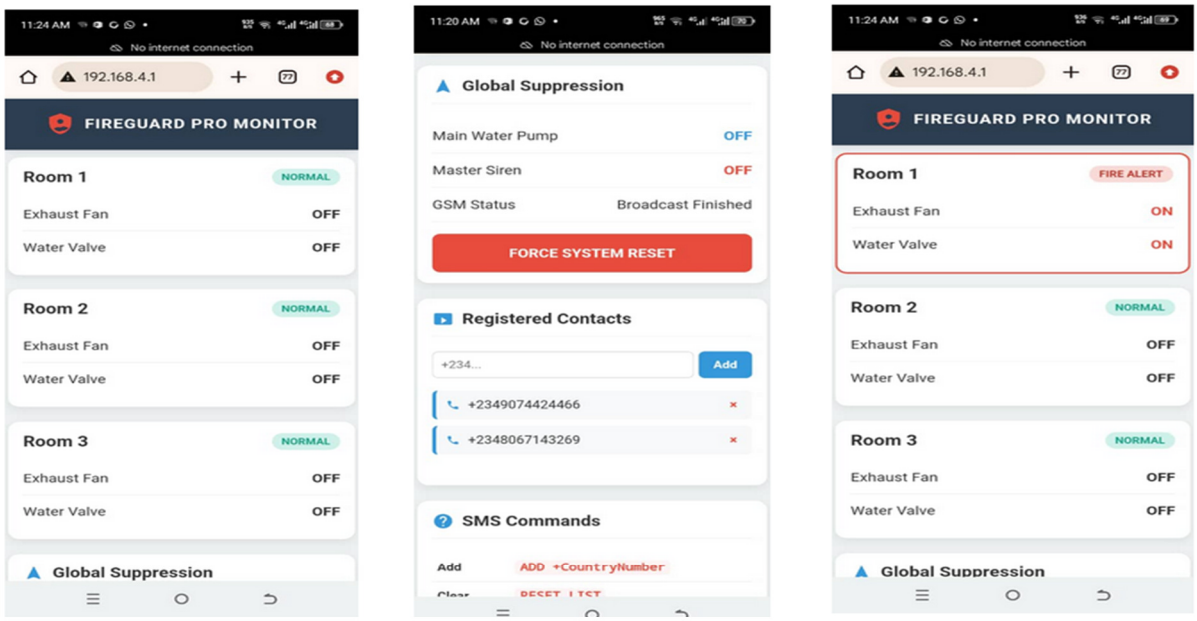


Figure 6: Android App showing the response in each room.(6a,6b and 6c).

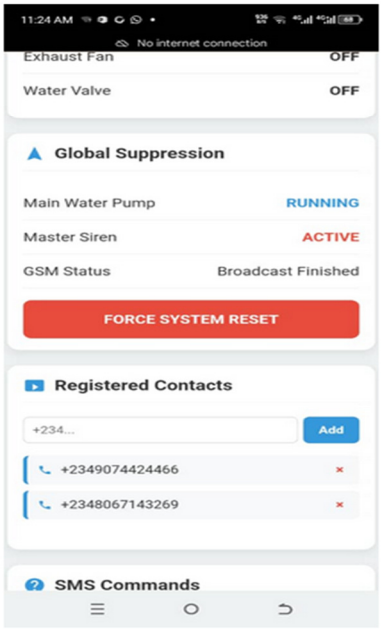


Figure 6d: Android App showing the response.

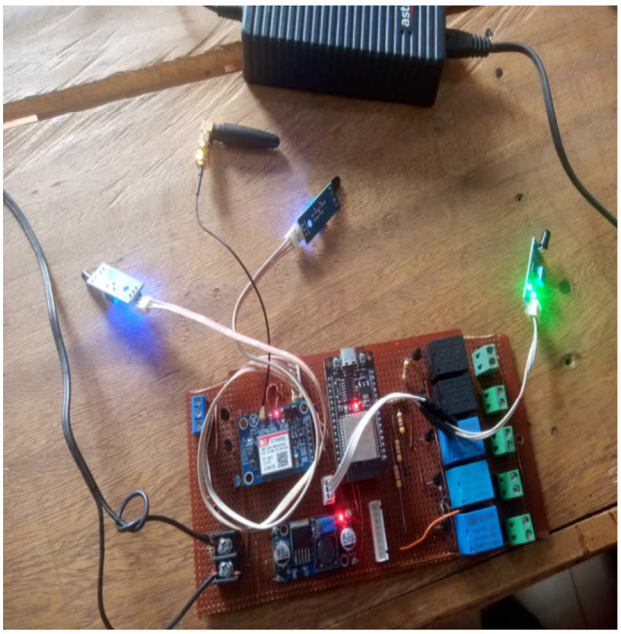


Figure 7: Shows circuit connected to Android APP

veroboard.Afeter which the output signal was tested.The output generated a signal which shows power is supplied to the circuit. The completed work was placed inside an adaptable box and then placed on metallic frame housing all the other hardwares. Figures 8 and 9, shows the circuit inside the box. Figure 10 shows the sprinkler head that helps to distribute water in a spray pattern. The droplets cool the flames and dispatch oxygen. Thus, causing the surrounding temperaature to drop. Figure 11 shows the solenoid valve connected to one of the rooms. It helps to control the discharge of water . Th microcontroller outputs logic, HIGH/LOW to the relay circuits which in turn energies the solenoid coil, opens the valve and allows pressurizes water to flow into the sprinkler head. Figure 12 shows the charge controller mounted . This is used in this work to manage charging and discharging of the battery

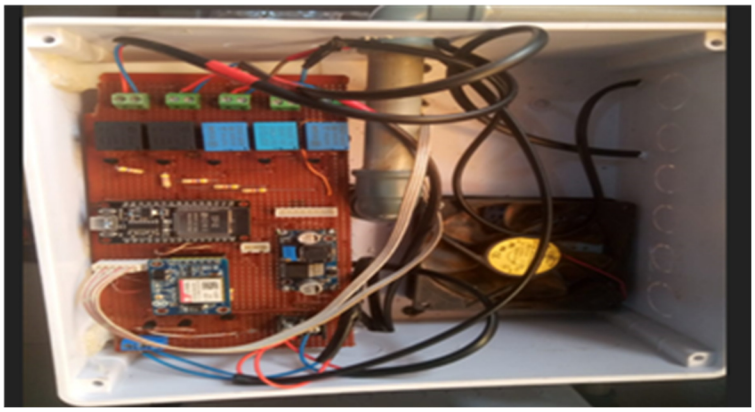


Figure 8: Circuit assembled inside an adaptable box

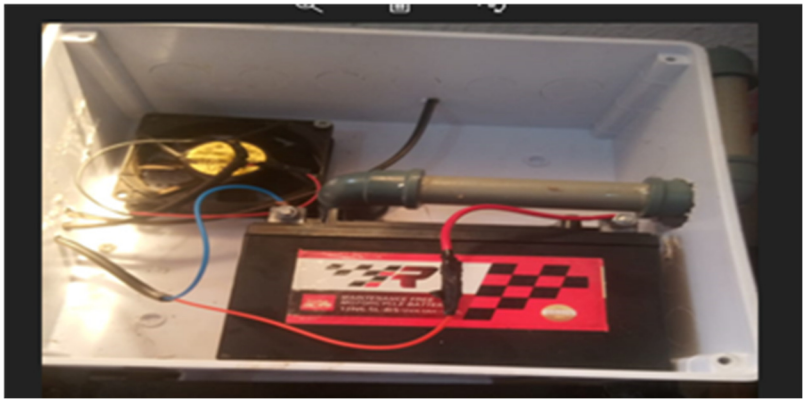


Figure 9: The DC-DC 12V/7A battery and exhaust fan inside an adaptable box

Figure 10: The prinklres head.

since the system made use of solar panel. It helps to protect the battery, extend its lifespan, and ensure reliable power to the system. Figure 13 indicates the alarm section . This part is made of siren , which sounds whenever fire, smoke, or high temperaute above 60^0C is dtetected . It also contains indicator lihts (red and blue) this is to assist those who have hearing impairment.

Figure 14 shows the pumping machine placed inside a 10 litres tank.It ensures that water is delivered at the required pressure and flow rate to extinquish or control a fire when sprinklers are activated. It boosts pressure so that water can reach all the sprinkler heads. Figure 15 shows the prototype of the system, having different sections connected with pipes to each room (rooms A, B, C,).

Figure 12: Charge controller mounted

Figure 11: The section showing the Solenoid valves mounted

Figure 13: Section showing the Alarm (Siren) and LED ON

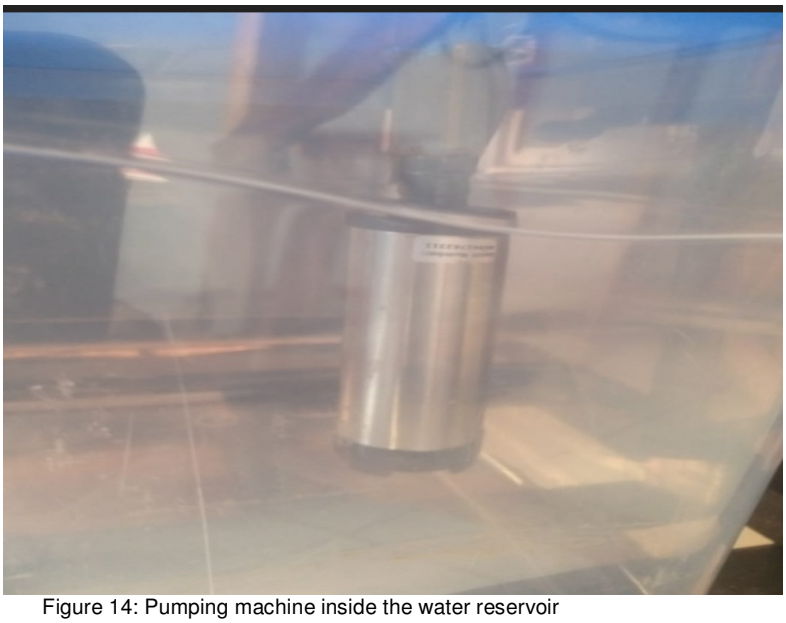


Figure 14: Pumping machine inside the water reservoir

Figure 15: A view of the prototype of the constructed Automatic fire sprinkler with IOT

Table 1: Normal condition for all the rooms.

Room	Condition	exhaust fan	water valve	Main water pump	master siren	GSM set up
Room 1	Normal	OFF	OFF	OFF	OFF	OFF
Room 2	Normal	OFF	OFF	OFF	OFF	OFF
Room 3	Normal	OFF	OFF	OFF	OFF	OFF

Table.2: Condition of Room 1

Room	Condition	exhaust fan	water valve	Main water pump	master siren	GSM set up
Room 1	Fire Alert	ON	ON	RUNNING	ACTIVE	Broadcaste calling
Room 2	Normal	OFF	OFF	OFF	OFF	OFF
Room 3	Normal	OFF	OFF	OFF	OFF	OFF

Table 3: condtion for room 2.

Room	Condition	Exhaust fan	Water valve	Main water pump	Master siren	GSM set up
Room 1	Normal	OFF	OFF	OFF	OFF	OFF
Room 2	Fire Alert	ON	ON	RUNNING	ACTIVE	Broadcaste calling
Room 3	Normal	OFF	OFF	OFF	OFF	OFF

The three rooms are represented by three white casing.which act also as fire source. Sprinklers are attached to each of the rooms.

System Testing

The system was tested after connection of all sections were done with. all connections checked using a digital multimeter to detect open circuits, short circuits, or incorrect wiring. Components such as sensors, relays, and power supply terminals were verified for correct polarity.This was to ensure continuity and proper connection of the system. Functional Test was carried out by Simulated fire conditions by :exposing the temperature sensor to controlled heat, and introducing non-toxic test smoke near the smoke sensor. Charcoal was also used to also act as a source of fire by igniting it. The response time was recorded for 5 trials. Instruments such as, Digital multimeter, Variable DC power supply, ,Temperature source (heat gun or hot air), Artificial smoke source/tester, Stopwatch for response time measurement were used for the test.

RESULTS

The result obtained after conecting the IoT based automatic fire figthing sprinkler system after using a fire source in each of the rooms is shown in (Tables 1-7).

Table 4:Condtion for room 3.

Room	Condition	Exhaust fan	Water valve	Main water pump	Master siren	GSM set up
Room 1	Normal	OFF	OFF	OFF	OFF	OFF
Room 2	Normal	OFF	OFF	OFF	OFF	OFF
Room 3	Fire Alert	ON	ON	RUNNING	ACTIVE	Broadcaste fcalling

Table 5: The latency experienced at every successful .trials for Room 1

Number of Trial (N)	Fire sprinkler latency(sec)
0	0
1	05
2	06
3	05
4	07
5	06
Total average	5,8

Table 6: The latency experienced at every successful .trials for Room 2

Number of Trial (N)	Fire sprinkler latency(sec)
0	0
1	06
2	06
3	07
4	05
5	06
Total average	6.0

Table7: The latency experienced at every successful .trials for Room 3

Number of Trial (N)	Fire sprinkler latency(sec)
0	0
1	07
2	06
3	06
4	07
5	06
Total average	6,4

DISCUSSION

The IoT based automatic fire fighting system as observed from table 1-4 above, shows the functionality and workability of the system when switched ON with a fire source. The infrared sensor detects the flame all the rooms (A, B, C), once detected, the exhaust fan, water valve (solenoid). Pumping machine, master siren, LEDs, sprinkler head and GSM module all become active. The latency (time delay for the valve to open) was noted after 5 successive trials, the average response time for the three rooms is 6.07seconds. The rooms (A, B, C) are 15cm apart, hence, the difference in the response time for each of the rooms. The upper bound recommended for this type of system safety is 20seconds. Therefore, the latency of the constructed system is 30.35% of the maximum allowable system latency.

Average latency of the system $= \frac{5.8+6.0+6.4}{3} = 6.07$sec.

This is the average response of the three rooms (A, B, C)

$$\text{Latency} = \frac{\text{Average}}{20} \times 100 = 30.35\%$$

Conclusion

The design and implementation of an IoT-based automatic fire sprinkler system has demonstrated that integrating sensing, control, and wireless communication technologies can significantly enhance early fire detection and rapid response. By combining temperature/smoke sensing units with a microcontroller, solenoid-actuated sprinkler mechanism, and remote monitoring through IoT connectivity, the developed system is capable of automatically activating fire suppression while simultaneously notifying users of emergency conditions. Experimental tests showed that the system responded within a short time interval once threshold values were exceeded, confirming its reliability for real-time application in residential and small-scale commercial environments. These systems have proven to effectively detect and

suppress fires, significantly reducing property damage and saving lives. A well-designed sprinkler system, tailored to the specific needs of building and its occupants, provides reliable fire protection and ensures compliance with local fire safety regulations

Author's Contributions

Conceptualization of design, Abode, H.O. Methodology, Abode, H.O, Testing and Assessment, Abode, H.O., Okpanachi, B, Writing original draft, Abode, H.O. Review and Editing, Abode H.O

Acknowledgement

This research was supported by the Tertiary Education Trust fund (TETFUND), Nigeria. The authors wish to express their gratitude to TETFUND for their kind approval

REFERENCES

Abdullahi, U. I., Zhang, W., Cao, Y., & Irankunda, G. (2025). Integrating IoT technology for fire risk monitoring and assessment in residential building design. Journal of Buildings, 15(8), 1346. https://doi.org/10.3390/buildings15081346

Abdullahi, M., Musa, A. A., & Sadiq, I. O. (2022). Design and implementation of a GSM-based fire alert system. *International Journal of Safety and Security Engineering*, 12(3), 345–352. https://doi.org/10.18280/ijsse.120309

Ahmed,A.,Mansor,A.,Albagul,A.(2015), Design and Fabrication of an Automatic Sprinkler Fire Fighting System,"In Design and Modelingof Mechanical Systems-11. Lecture notes in Mechanical Engineering. Vol 789,pp 42-45, Springer .https//doi.org/10.1007/978-3-319-17.527-0-5

Ahrens, M. (2021). U.S. experience with sprinklers. Quency, MA: National Fire Protection Association (NFPA). Research Report. https//www.nfpa.org/News-and-Research/Data- research-and-tools/Suppression/US-Experience-with sprinklers

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2023). Internet of Things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 25(1), 1–49. https://doi.org/10.1109/COMST.2022.3201964

Alqourabah, H., Muneer, A., & Fati, S. M. (2020). A smart fire detection system using IoT technology with automatic water sprinkler. *International Journal of Electrical and Computer Engineering (IJECE),*11(4),2994-*30*.https//doi.org/10.org/10.11591/ijece.v11i4

Chauhan, R., Singh, A., & Kumar, N. (2019). Microcontroller-based automatic fire detection and sprinkler system using wireless sensor networks. *International Journal of Engineering Research & Technology,* 8(6), 102–107.

Espressif Systems. (2023). ESP8266EX datasheet. Espressif Systems Inc. https://www.espressif.com

Hurley, M. J., Gottuk, D., Hall Jr., J. R., Harada, K., Kuligowski, E., Puchovsky, M., Torero, J. L., Watts Jr., J. M., & Wieczorek, C. (Eds.). (2016). SFPE Handbook of Fire Protection Engineering (5th ed.). Society of Fire Protection Engineers / Springer. https://link.springer.com/book/10.1007/978-1-4939-2565-0

Issaka, D., Sam-Okyere, Y. A., & Osei-Kwame, E. (2025). Internet of Things (IoT) based fire detection and suppression system. *Journal of Applied Engineering, Innovation, and Technology*, 2(2), 43–60. https://doi.org/10.62777/aeit.v2i2.70

\Juliaz, D. A., & Maulana, H. (2025). Design of fire detector with water sprinkler based on Internet of Things (IoT). *Hanif Journal of Information Systems*, 2(2), 62–69.

Kolban, N. (2022). Kolban's Book on ESP8266. & ESP32 (ebook). Published by Leanpub. https//leanpub.com/ESP8266_ESP32

Kumar, S., & Lee, J. H. (2023). Smart fire detection and suppression system using IoT and cloud computing. Sensors, 23(4), 1987. https://doi.org/10.3390/s23041987

Kumar, P., & Lee, H. J. (2020). GSM-based healthcare monitoring system for elderly patients. Sensors, 20(6), 1719. https://doi.org/10.3390/s20061719

Mohamad, N. F., & Khalid Isa. (2024). A IoT Based Fire Safety Detection and Suppression Control System. Journal of Evaluation in Electrical and Electronic Engineering, 5(1), pp298-309 https//doi,org/10.30880/eee.2024.05.01.038

National Fire Protection Association (2022).NFPA 13: Standard for the installation of sprinkler systems. National Fire Protection Association

Patel, D., Shah, K., & Mehta, R. (2019). Performance analysis of SMS-based alert systems over GSM networks.Wireless Personal Communications, 107(2), 803–818. https://doi.org/10.1007/s11277-019-06273-4

Ray,J.M., & Doe. P.Q.(2018).Automatic design methods for fire sprinkler systems in commercial buildings. .*Journal of Fire Protection Engineering*. 32(4),215-230. https//doi.org/10.1234/fireeng.20118.03245

Singh, A., Verma, P., & Yadav, R. (2021). Hybrid GSM-IoT based intelligent alert system for smart environments. *Journal of Ambient Intelligence and Humanized Computing*, 12, 9875–9886. https://doi.org/10.1007/s12652-020-02654-3

Zeng, y., liu,x., ding,,y., zheng.z., zhang.t.,huang,x., and lu,x.(2025).AI-powered automatic desgn of fire sprinkler layout for random buildig floorplans. *Journal of Infrastructure Intelligence and Resilence, 4(4). Doi: 10.1016/j.iintel.2025.100167*

APPENDIX

FULL SOURCE CODE

```
 // Fire alarm system ESP8266 + 3 Sensors + 5 Relays + GSM + SMS fallback + Water Valves #include <SoftwareSerial.h>
// ====== SENSOR PINS ======
#define IR1_PIN D1
#define IR2_PIN D2
#define IR3_PIN D3
// ====== RELAY PINS ======
#define RELAY1_PIN D5
#define RELAY2_PIN D6
#define RELAY3_PIN D7
#define RELAY4_PIN D8
#define RELAY5_PIN D0
// ====== GSM PINS ======
#define GSM_RX_PIN D4
#define GSM_TX_PIN D3
SoftwareSerial gsmSerial(GSM_RX_PIN, GSM_TX_PIN);
const char ownerNumber[] = "+2348012345678"; bool sensorTriggered[3] = {false, false, false};
void setup() {
  Serial.begin(115200);                gsmSerial.begin(9600);
delay(300);
  pinMode(IR1_PIN, INPUT);
pinMode(IR2_PIN, INPUT);
  pinMode(IR3_PIN, INPUT);
  pinMode(RELAY1_PIN, OUTPUT);
  pinMode(RELAY2_PIN, OUTPUT);
  pinMode(RELAY3_PIN, OUTPUT);
  pinMode(RELAY4_PIN, OUTPUT);
  pinMode(RELAY5_PIN, OUTPUT);
  digitalWrite(RELAY1_PIN, LOW);
  digitalWrite(RELAY2_PIN, LOW);
  digitalWrite(RELAY3_PIN, LOW);
  digitalWrite(RELAY4_PIN, LOW);
  digitalWrite(RELAY5_PIN, LOW);
}
void loop() {
  checkSensor(1, IR1_PIN, RELAY1_PIN);
  checkSensor(2, IR2_PIN, RELAY2_PIN);
  checkSensor(3, IR3_PIN, RELAY3_PIN);
  delay(200);
}
void checkSensor(int id, int sensorPin, int relayPin) {
  int state = digitalRead(sensorPin);
  if (state == HIGH && !sensorTriggered[id-1]) {
    sensorTriggered[id-1] = true;
    digitalWrite(relayPin, HIGH);
    digitalWrite(RELAY4_PIN, HIGH);
    digitalWrite(RELAY5_PIN, HIGH);
    if (!makeCall(ownerNumber)) {
      sendSMS(ownerNumber, "FIRE ALERT! Sensor " + String(id) + " triggered.");     }
  }
if (state == LOW && sensorTriggered[id-1]) {
    static unsigned long t = millis();
    if (millis() - t > 10000) {
      sensorTriggered[id-1] = false;
      digitalWrite(relayPin, LOW);
      if (!sensorTriggered[0] && !sensorTriggered[1] && !sensorTriggered[2]) {              digitalWrite(RELAY4_PIN, LOW);
        digitalWrite(RELAY5_PIN, LOW);       }
}
      t = millis();
    }
  }
}
bool makeCall(String number) {
  gsmSerial.print("ATD");
  gsmSerial.print(number);
  gsmSerial.println(";");
  unsigned long start = millis();   bool connected = false;
  while (millis() - start < 15000) {
if (gsmSerial.available()) {
      String r = gsmSerial.readString();
      if (r.indexOf("BUSY") > 0 || r.indexOf("NO CARRIER") > 0 || r.indexOf("NO DIALTONE") > 0)         return false;
      if (r.indexOf("OK") > 0 || r.indexOf("CONNECT") > 0) connected = true;
    }
  }
  gsmSerial.println("ATH");
  delay(1000);
  return connected;
}
void sendSMS(String number, String message) {
gsmSerial.println("AT+CMGF=1");
  delay(500);
  gsmSerial.print("AT+CMGS=\"");
gsmSerial.print(number);
  gsmSerial.println("\"");   delay(500);
  gsmSerial.print(message);   delay(500);
  gsmSerial.write(26);   delay(2000);
```