



Designing a Fire Detection System Using Blynk Application Based ESP32 Microcontroller

Rossi Peter Simanjuntak^{1*}, Donna N. M. Sirait², Susi Diriyanti N³, Rahman Aulia⁴
 Politeknik Penerbangan Medan^{1,2,3,4}

*Correspondence:

rossipetersimanjuntak@gmail.com

ABSTRACT

Article info

Received: 23-6-2024
 Final Revision: 21-6-2024
 Accepted: 25-6-2024
 Available online: 30-6-2024

Keywords:

Sensor DS18B20; Sensor MQ13; Microcontroller ESP32; Blynk.

Fire is a disaster that often occurs in society, resulting the problem in material loss and can cause loss of life. In order to reduce **the problem** caused by fire incidents, early detection is needed to prevent fires from occurring. In this research, a fire detection system was built by detecting the temperature and smoke contained in a room. The **research objective** in the detection system uses a DS18B20 temperature sensor and an MQ135 smoke sensor connected to an ESP32 microcontroller connected to an LED layer. The fire detection system is monitored via a smartphone connected using the blynk application. This research uses the Prototype **Method**, this method involves creating an early version of software system used to demonstrate concepts, test designs, and find more potential problems and solutions. Methods in the fire detection system is conditioned to detect temperatures > 450C and/or detect smoke > 600 ppm. When the temperature sensor detects that the temperature is > 450C and/or the smoke sensor detects that the smoke is > 600 ppm, the buzzer will sound and a fire warning notification will be displayed on the smartphone in real time. From the experimental **results**, it was found that when the temperature sensor detected a temperature of 48.90C, the buzzer immediately sounded and a temperature of 48.90C appeared on the smartphone and a fire warning notification on the smartphone in real time. An experiment was also carried out to detect the presence of smoke, when the smoke sensor detected 669 ppm, a buzzer immediately sounded and appeared on the smartphone, 669 ppm smoke, and a fire warning notification on the smartphone in real time

Reccomended Citation:

APA Style

INTRODUCTION

Building is an infrastructure facility used as a place for people to carry out their daily activities to fulfill all their needs. Building is an infrastructure facility used as a place for people to carry out their daily activities to fulfill all their needs. According to the Government Regulation of the Republic of Indonesia Number 16 of 2021 and the Regulation of the Minister of Public Works and Housing Number 20 of 2021, there are several functions of buildings, namely "(1) residential function which is a building that primarily functions as a residence to meet human needs for housing, (2) business function which is a building that functions as a place for business activities including buying, selling, and renting, (3) social and cultural function which is a building primarily used for social and cultural activities including education services, health services, cultural activities, laboratories, and public services, (4) religious function which is a building primarily used to meet human needs as creatures of God as places of worship such as mosques,

churches, temples, shrines, and monasteries, (5) special function which is a building that, due to its function, has a high level of secrecy and security for national interests or because its operation can endanger the surrounding community and/or have a high risk of danger. Taking into account the functions of buildings, there are interactions that occur both among humans and between humans and important documents that support the activities carried out by humans.”

Fire is one of the tragedies that comes unpredictably, besides being unwanted by the community, it is also often uncontrollable once the fire becomes large (Siregar, T. S., Sertya, P. S., Gatot, E. P., & Maulana, M, I, 2021; Buser, 2023; Hopkins, 2024). Fire incidents pose significant danger and disruption to the lives and livelihoods of the community. Fires are categorized as one form of disaster. According to the National Disaster Management Agency (BNPB), a disaster is an event or series of events that threaten and disrupt the lives and livelihoods of the community caused by natural factors, non-natural factors, or human factors resulting in loss of life, environmental damage, property damage, and psychological impact (BNPB, 2012)

Fires are one of the disasters that often occur within communities and are highly detrimental. Based on fire data from 2014 to 2018, India is the country with the highest number of fires, with an average of 1.6 million fires annually, followed by several other countries such as the United States with 1,340,700 fires, Bangladesh with 18,266 fires, Russia with 142,576 fires, Japan with 39,864 fires, and Indonesia with 2,502 fires (N. Brushlinsky, M. Ahrens, S. Sokolove, and P. Wagner, 2020). One of the disasters that frequently occurs within communities is building fires, which can disrupt the environmental comfort of the community due to fire smoke affecting health (Roihan, A., Nina, R., and Stevanus, 2022). Additionally, fires also cause suffering and significant losses, especially for those affected by the disaster, including loss of property, physical disabilities, trauma, property damage, burning of important documents, loss of employment, and even loss of life (Saifullana, Joni, W, S., 2019). Building fires often occur when the building is empty. Fires frequently happen due to human negligence caused by several factors such as discarded cigarette butts, short-circuiting electrical connections, gas leaks, and delays in fire handling (Hermianto, K., Mohammad, A. Iqbal., dan Imam, A. R., 2022). Building owners usually are unaware of signs of a building fire until it has already grown large, resulting in significant material losses reaching billions of rupiah, in addition to casualties. Fires occur due to delayed information reaching the building owners, preventing them from anticipating the fire. One way to anticipate fires is by detecting temperature increases in rooms and detecting smoke, which serve as early fire detection measures.

Previous research has been conducted on a prototype to automatically detect early fires using LM35DZ temperature sensors and Photodiodes. It utilizes the ATmega8535 microcontroller as data manager and AVR SIP software (Narji, M., Arie, B. U., Dedi, S., and Tata, S., 2022). The IoT-based fire detection system using Arduino that can be monitored remotely. The system utilizes flame sensors, smoke sensors (MQ-7), temperature sensors (LM35), a buzzer, and a water pump. Warning alerts are delivered through phone calls and SMS via the SIM900 module, while danger alerts are sent to an Android application using the internet through the NodeMCU ESP8266 module (Siregar, T., Setya, P. S., Gatot, E. P., & Maulana, M. I., 2021; Mahzan, N. N., Enzai, N. M., Zin, N. M., & Noh, K. S. S. K. M., 2018; Adedeji, W. O., Adekoya, A. O., Olukayode, O., Oyewole, K. A., & Amosun, T. S., 2023). The fire detection system can detect early signs of fire caused by smoke, gas, and heat/temperature. The system will contact the fire department or the owner of the location via voice over IP phone call. This system is based on the ESP8266 microcontroller and also utilizes IFTTT (If This, Then That) from Google Assistant (Manfaluthy, M., Agung, P., Imam, N., 2022). The automatic fire detection and extinguishing device is based on the ATmega 2560 microcontroller utilizing three fire sensors to detect fire points with a light range ≥ 500 . If the range exceeds this threshold, the microcontroller commands the servo to the fire point angle and the relay directly activates the pump (Kurnia, M. A., 2022). Fire detection devices are designed with systems connected via the internet network and provide information if there are indications of fire to relevant parties. The system uses microcontrollers equipped with flame sensors, MQ-2, and MQ-7 (Zikrullah, A. P., Rima, T., & Iskandar, F., 2022).

Fire detection system is carried out using NodeMCU Esp 8266, fire sensors, smoke sensors, buzzer, GPS module, and Telegram application (Winarno, A., & Awang, J. M., 2023). Remote fire detection device utilizes an Android smartphone as an information medium and Arduino UNO board as an identification tool. The system used is Use Case Diagram modeling with UML and software including Android Studio, Arduino IDE, and Firebase Realtime Database. The device is equipped with thermal sensors, smoke sensors, temperature sensors that send information to be delivered as warning notifications

on Android smartphones (Pare, S., Tallulembang, T. M., & Jarot, B., 2022; Muheden, K., Erdem, E., & Vançin, S., 2016). By implementing a smoke detection system based on the Internet of Things using the MQ135 sensor, Arduino board, and NodeMCU. Then, a comparison of Quality of Service between two data communication protocols, Transmission Control Protocol and User Datagram Protocol, is conducted on the system. The Quality of Service parameters compared during the data transmission process are delay and data loss. For each protocol, simulations are conducted for 1 hour with data transmission every 5 seconds, 10 seconds, up to 1 minute. The results show that data loss with Transmission Control Protocol is lower than data loss with User Datagram Protocol, while delay with User Datagram Protocol is lower than delay with Transmission Control Protocol (Romony, P. V. B., Lanny, S., & Junaidy B. S., 2020).

ESP32 is a microcontroller introduced and developed by Espressif Systems, which is the successor to ESP8266 (Zaini, M., Safrudin, & Bachrudin, M., 2020). The ESP32 microcontroller features Wi-Fi and Bluetooth functionality, Bluetooth Low Energy (BLE) communication, independent time regulation, analog to digital and digital to analog converters (ADC and DAC), capacitive touch sensors, and Hall effect sensors. The ESP32 microcontroller comprises two 240 MHz cores, each with a 32-bit LX6 Tensilica Xtensa microcontroller (Cameron, N., 2023). The ESP32 microcontroller is compatible with the Arduino IDE and is well-suited for creating Internet of Things-based electronic projects. The ESP32 memory consists of 448 kB ROM, 520 kB SRAM, two 8 kB RTC memories, and a 4MB flash memory. The chip has 18 12-bit ADC pins, four SPI units, and two I2C units. The ESP32 has 12-bit ADC pins, meaning they range from 0 to 4095 (Widyatmika, I. P. A. W., N. P. A. W. Indrawati., I. W. W. A. Prastya., I. K. Darminta, I. G. N. Sangka., & A. A. N. G. Saptaka., 2021).

Previous research has been conducted by Abdullah, A. et al (2022), the results of the application of the fire alarm analysis in the main building are 0% with the inappropriate category, and the average in the study program building is 96.57% with a good category. The implementation of the detection system in the main building is 0% with the inappropriate category (no fire alarm and detection system installed), and the average in the study program building is 97.5% with a good category. By utilizing the MQ-2 sensor and ESP32 microcontroller, the cigarette smoke detection system is capable of detecting cigarette smoke by having the MQ-2 sensor detect the presence of smoke, including cigarette smoke, in the area in the form of an analog signal. This signal is then converted into digital code and interpreted by the ESP32 microcontroller. The ESP32 microcontroller then sends commands to the buzzer to respond according to the programmed instructions. The buzzer responds with an alarm sound, indicating the detection of cigarette smoke in the surrounding area (H. Erwin, Haris. D., Farid. B., Anan. S., Suliadi., & Aulia. Y., 2022).

DS18B20 is a digital temperature sensor that follows the 1-wire protocol and can measure temperatures from -55°C to $+125^{\circ}\text{C}$ (-67°F to $+257^{\circ}\text{F}$) with an accuracy of $\pm 5\%$. The data received from the single cable can be in 9-bit to 12-bit resolution. Because DS18B20 follows the 1-wire protocol, this sensor can be controlled via a single pin of the microcontroller. The 1-wire protocol is an advanced protocol, and each DS18B20 is equipped with a 64-bit serial number that helps control multiple sensors through one microcontroller pin (Fezari, M., & Ali, A. D., 2019; Liu, X., & Yan, K., 2023). DS18B20 has 3 pins, which are:

Pin #1 : Vcc (+5V)

Pin #2 : Data Pin (The cable used to read the temperature)

Pin #3 : GND (Ground)

The MQ-135 sensor is a type of gas sensor capable of detecting compounds such as Ammonia (NH₃), Nitrogen Dioxide (NO_x), alcohol, Benzene (C₆H₆), Carbon Dioxide (CO₂), Nitrous Oxide (NO_x), Hydrogen Sulfide (H₂S), smoke, and other harmful gases (Khyioon, Z., Intisar, S., Farah, N., 2020). The MQ-135 sensor shares similarities with other MQ series gas sensors in which it has both digital and analog output pins. When the MQ-135 sensor comes into contact with gas, the gas interacts with the sensitive layer on the sensor, causing its resistance value to change. This change in resistance can be measured analogously using a microcontroller or other electronic devices. The resistance value can be used to determine the concentration of the detected gas. If the gas concentration level exceeds the threshold in the air, the digital output pin becomes HIGH, while the analog output pin produces an analog voltage that can be used to predict the gas concentration level in the air (Rombang, A. R., Lukas, B. S., & Gunawan, D., 2022). The design of an automatic control system for air pollution detection in industries based on the PIC16F877A microcontroller will utilize the MQ-135 component as an integral part of the air quality control system. The MQ-135 sensor was chosen for its wide detection range, fast response, high sensitivity,

stability, and good resistance to industrial environments. By using the MQ-135 sensor, the system will be able to effectively detect pollutant gases. The PIC16F877A microcontroller will be responsible for data acquisition from the sensor and for adjusting necessary actions based on the detected values (R. Sequeira, et al., 2015). The MQ-135 sensor is used to detect CO, CO₂, and CH₄ gases. Additionally, a blower is used as a fan to clean the air by extracting air from the room (Widodo, et al., 2017).

Arduino IDE is an open-source software used for writing and compiling code into Arduino modules using the Java programming language, equipped with C and C++ language libraries (Arida, A. R., Bryan, A. S., & Kevin, S. L., 2020; Ismailov, A. S., & Jo'Rayev, Z. B., 2022). The menu in the Arduino IDE is divided into three sections: the Menu Bar, Text Editor, and Output Pane. LED (Light Emitting Diode) are small semiconductor devices with two terminals, namely the anode (+) and cathode (-). When forward bias current flows through them, LEDs can emit light because they consist of semiconductor chips that do not generate heat. Due to their small size, LEDs are highly suitable for use in electronic devices. When supplied with electrical voltage, LEDs produce light. The current flowing through LEDs is typically low, with a maximum limit of around 20 mA. If the current exceeds 20 mA, LEDs are at risk of damage. Therefore, in LED circuits, a resistor is often placed at the positive (+) terminal to regulate the current flow (Budi, A dan Fredy, S., 2017).

The LCD (Liquid Crystal Display) 16x2 is a data display module that uses liquid crystal as a medium to display text or graphics. This module can display 16 characters in 2 rows, totaling 32 characters. Typically, the 16x2 LCD requires 16 pins for its control, which can be quite inefficient in terms of pin usage. Therefore, a special driver is used to enable LCD control via the I2C (Inter-Integrated Circuit) module. With the I2C module, the 16x2 LCD only requires two pins for data transmission and two pins for voltage supply, reducing the required connections to the microcontroller to just four pins (Rombang, A, R., Lukas, B, S., dan Gunawan, D., 2022). Blynk is a platform that allows users to control and monitor hardware projects through their smartphones. It's used to control hardware remotely via internet or WiFi connections and is available for both Android and iOS devices. With a stable internet connection, we can control the system from anywhere through the Blynk app. The features available in the Blynk app are designed for ease of use and understanding, making the process of connecting hardware to Blynk quick and hassle-free. The practical, user-friendly, and swift usability of Blynk is the main reason why many choose it as their Internet of Things system. The Blynk design process consists of four stages. The first stage is "Create New Project," which allows for the creation of a new project. Next, there's the "Auth Token" step, which involves sending the Blynk authentication token via email for use in the program code. The final stage involves using the "Widget Box" with an easy-to-use interface. (Alifia, A., Syifauf, F., dan Endah, S., 2023).

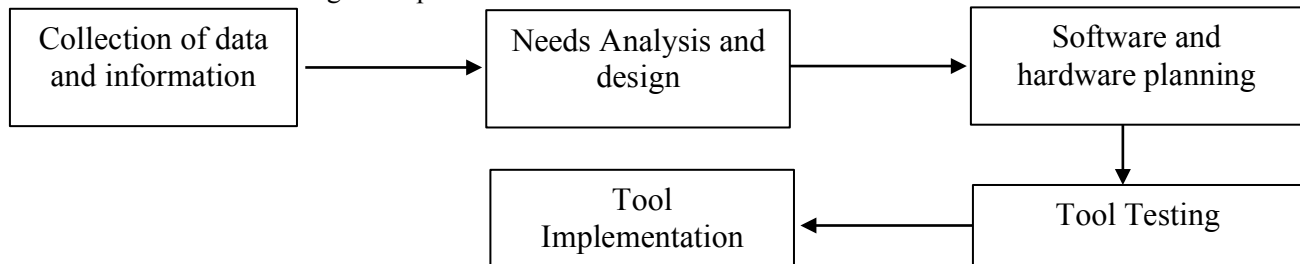
The fire detection system built and designed based on the Internet of Things with ESP32 microcontroller devices will perform temperature sensor DS18B20 readings and smoke sensor MQ135 placed in a room. The temperature sensor and smoke sensor are connected to a buzzer and LED that display the results from these sensors. They are also connected through a Blynk application accessible via an Android smartphone. Therefore, this design is based on the Internet of Things because the sensor reading values can exchange data and be accessed via an Android smartphone in real-time. The device find that the design tool can work well according to the desired conditions. When the DS18B20 sensor detects room temperature exceeding 45°C, the ESP32 will process it and display it on the LCD 16x2 screen. Additionally, the room temperature can be monitored remotely using a smartphone, where the sensor readings will be displayed on the smartphone screen, and danger notifications will be given if the sensor detects temperatures exceeding the set limit of 45°C. It was found that the designed tool can work well according to the desired conditions. When the MQ135 sensor detects smoke concentration in the room exceeding 600 ppm, the ESP32 will process it and display it on the LCD 16x2 screen. Additionally, the smoke concentration in the room can be monitored remotely using a smartphone, where the sensor readings will be displayed on the smartphone screen, and danger notifications will be given if the sensor detects smoke exceeding the set limit of 600 ppm. Notifications on the smartphone will be displayed when the DS18B20 sensor detects room temperature exceeding 45°C or the MQ135 sensor detects smoke concentration in the room exceeding 600 ppm, or both sensors detect the conditioned threshold.

METHOD

This research uses the Prototype Method, this method involves creating an early version of software system used to demonstrate concepts, test designs, and find more potential problems and

solutions. A system with a prototype model allows users to understand how the system works well. The primary goal of this method is to demonstrate concepts, test designs, and find more potential problems and solutions (Real, R., Snider, C., Goudswaard, M., & Hicks, B., 2021; Asri, S. A., Astawa, I. N. G. A., Sunaya, I. G. A. M., Yasa, K. A., Indrayana, I. N. E., & Setiawan, W., 2020). Here are some benefits of the prototype Method are allows developers to demonstrate the concepts they will develop, enabling users to understand how the system works well, enables developers to test designs and find potential issues that may arise, which can be addressed before the system is further develop, developers can find more effective and efficient solutions to the problems they face, enabling them to better understand their needs and expectations.

The following stages are required in creating a fire detection system using Android-based microcontrollers. These stages are presented below:



Picture 1. Work stage

The description of activities performed in accordance with the stages of work is data collection and analysis is conducted to gather the necessary information for the development of the fire detection system. This process involves a literature review aimed at exploring aspects related to relevant theories. The goal is to support effective planning and system design. Needs Analysis and Design are carried out to analyze the requirements needed to create the fire detection system. Additionally, the author creates a system circuit diagram to facilitate the overall system design. Software and Hardware Design planning is built by assembling hardware first, such as connecting sensors to microcontrollers, and then proceeding with the programming process. Tool Testing to ensure its compliance with expectations. This is done to promptly identify and address any deficiencies or failures. Tool Implementation is conducted to ensure that the fire detection system meets expectations. The system is tested to ensure that it can operate smoothly and according to the desired functions.

System Overview

The fire detection system using Android-based microcontrollers is created with the aim of reducing fire incidents and minimizing the losses they cause. The impact of fires is highly detrimental to humans, and such incidents can occur anytime and anywhere unpredictably. Therefore, it is crucial to always remain vigilant and alert to potential fire hazards.

The system to be developed will consist of the DS18B20 temperature sensor, MQ135 smoke sensor, GSM module sim8001 V2, ESP32 microcontroller, and a buzzer. This system will detect fires and send data to the ESP32 microcontroller. Additionally, the buzzer will activate, and notifications will be sent by sending text messages to the phone numbers stored in the program.

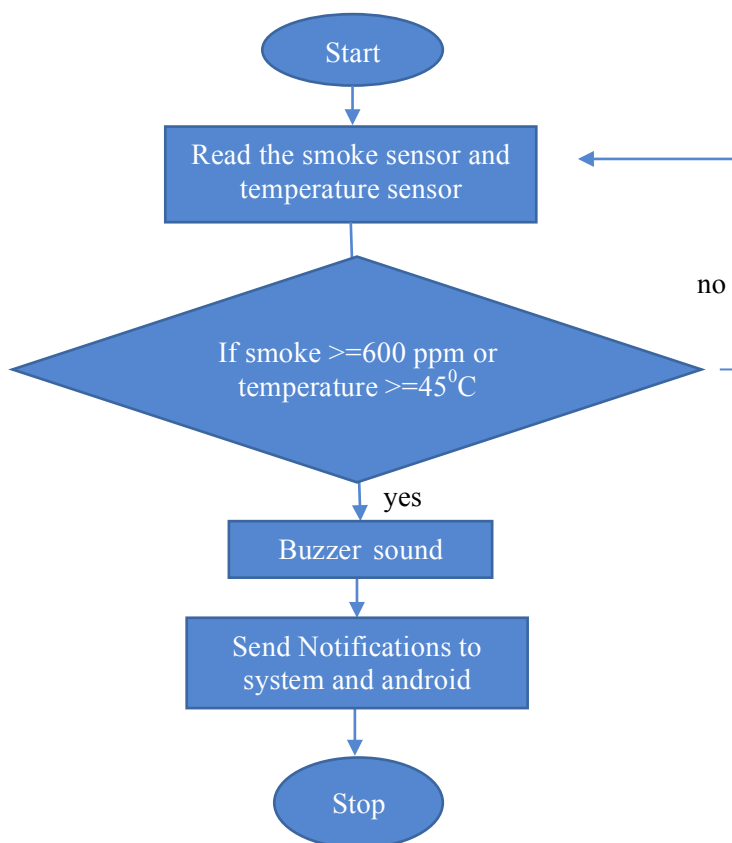
Requirements Analysis

Based on the analysis results, first we need to input requirements, the input requirements or inputs from the fire detection system using Android-based microcontrollers are temperature sensor data and smoke sensor data. Then output requirements or outputs from the fire detection system using Android-based microcontrollers are information on temperature sensor values and information on smoke sensor values. Hardware requirements of the fire detection system using Android-based microcontrollers are DS18B20 temperature sensor, MQ135 smoke sensor, GSM module sim8001 V2, ESP32 microcontroller, buzzer, 5V 2A power supply, 16x2 LCD, and Jumper cables. Last software requirements or software used in the fire detection system using Android-based microcontrollers are Arduino IDE and the Blynk application.

System Design

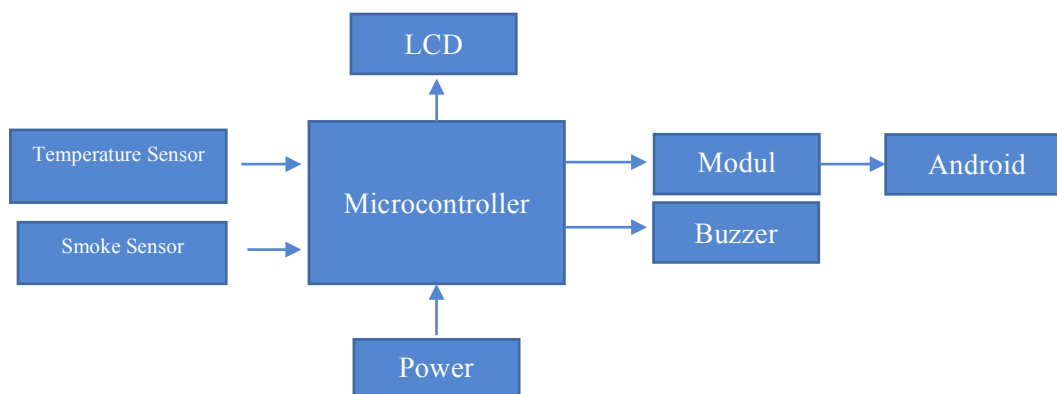
The components within the Internet of Things-based fire detection system using ESP32 microcontroller devices will read data from sensors placed in specific rooms. The sensor readings will be displayed through the Blynk application accessible via an Android smartphone. Therefore, this design is based on the Internet of Things concept because sensor data can be exchanged and accessed in real-time through an Android smartphone.

The flowchart below depicts the system workflow, where the values from the temperature and smoke sensors will be read by the ESP32 microcontroller in the form of analog signals. If smoke with a concentration greater than 600 ppm is detected by the MQ135 smoke sensor, the buzzer/alarm will activate, and notifications will be sent to the registered numbers in the system. Additionally, if the temperature detected by the DS18B20 sensor exceeds 45 degrees Celsius, the buzzer/alarm will also activate, and real-time phone calls to the registered numbers in the system will be initiated.



Picture 2. Flowchart System

The functions of each equipment are as follows: temperature sensor (detects the room temperature as input data for the microcontroller); smoke sensor (detects smoke in the room as input data for the microcontroller); power supply (converts voltage, regulates power, and provides power for the microcontroller); LCD: Displays output from the data processed by the microcontroller; microcontroller (processes data or signals received from the temperature sensor and smoke sensor); buzzer (functions as an alarm, activated under desired conditions); ESP32 module (WiFi module serving as an additional device for microcontrollers like Arduino to directly connect to WiFi and establish TCP/IP connections).



Picture 3. Equipment Block Diagram

The hardware component or hardware design of the fire detection system based on the Internet of Things (IoT) with ESP32 microcontroller, DS18B20 temperature sensor, MQ135 smoke sensor, buzzer, and LCD. Each sensor will be connected to the ESP32 microcontroller to exchange data. There is also power supply connected to all devices to power up the system.

Temperature Sensor Configuration

DS18B20 Temperature Sensor Configuration The DS18B20 temperature sensor functions to measure room temperature. The values generated by the temperature sensor will be read by the ESP32 microcontroller in digital signal format and then produce output in degrees Celsius. Figure 3.4 illustrates the process of configuring the temperature sensor with the ESP32 microcontroller. The temperature sensor has three output pins: VCC pin (5 volts), GND pin (ground), and out pin (data). The VCC pin serves as the positive power supply/power line from the voltage source, the GND pin serves as the negative power supply/power line from the energy source, and the out pin serves as the data pin for digital data transmission. Each pin will be connected to the microcontroller, where the VCC pin is connected to the VIN pin, the GND pin is connected to the GND pin, and the out pin is connected to the D32 and D33 pins.

Smoke Sensor Configuration The smoke sensor functions to detect the presence of smoke, where the content of smoke from fires is abundant, including carbon monoxide. The MQ135 smoke sensor is a type of chemical sensor sensitive to compounds such as NH3, Nox, alcohol, benzene, smoke (CO), CO2, and others. The sensor's output will be sent to the microcontroller for processing and presentation in notifications on an Android smartphone. The MQ135 smoke sensor has 4 pins: D0 pin (digital data), A0 pin (analog data), VCC pin (5 volts), and GND pin (ground). However, only three pins are used because the required data is in analog form, so the D0 pin is not used. The VCC pin serves as the positive power supply/power line from the voltage, the GND pin serves as the negative power supply/power line from the voltage, and the A0 pin serves as the analog data pin from the sensor. The three pins on the smoke sensor are connected to pins on the ESP32 microcontroller, where the VCC pin on the smoke sensor is connected to the VIN pin on the ESP32 microcontroller, the GND pin on the smoke sensor is connected to the GND pin on the ESP32 microcontroller, and the A0 pin on the smoke sensor is connected to the D26 pin on the ESP32 microcontroller.

Buzzer/Alarm Configuration The buzzer/alarm functions to notify in case of danger, damage, or unwanted events by providing audible warnings, enabling preventive measures to be taken. The buzzer/alarm has 2 pins: positive power pin and negative power pin. The positive power pin of the buzzer is connected to the D13 pin on the microcontroller, and the negative power pin of the buzzer is connected to the GND pin on the ESP32 microcontroller.

LCD Configuration The 16 x 2 LCD in the study is used as the output display. The LCD will display several characters according to the program and readings from the sensors. The pin configuration of the LCD is as shown in the table below:

Table 1. Pin Konfiguration LCD 16x2

LCD 16x2	Board ESP32
SCL (serial clock)	D21
SDA (serial data)	D22
VCC	VIN
Ground	GND

Designing Software for DS18B20, MQ135, and LCD 16x2 Sensors

After completing the hardware design, the next step is to design the software for the DS18B20, MQ135, and LCD 16x2 sensors as in the pseudocode below:

```
*/Program pengkondisian sensor/*
if(suhu1>45 || asap1>600) {
```

```

Blynk.logEvent("alarm1","Waspada Ruang A!");
digitalWrite(buzzer,HIGH); delay(200);
digitalWrite(buzzer,LOW); delay(200);
}
if(suhu2>45 || asap2>600) {
  Blynk.logEvent("alarm2","Waspada Ruang B!");
  digitalWrite(buzzer,HIGH); delay(200);
  digitalWrite(buzzer,LOW); delay(200);
}
else delay(500);
Serial.print("Suhu 1: "); Serial.print(suhu1,1); Serial.print("*C, ");
Serial.print("Asap 1: "); Serial.print(asap1); Serial.print("ppm, || ");
Serial.print("Suhu 2: "); Serial.print(suhu2,1); Serial.print("*C, ");
Serial.print("Asap 2: "); Serial.print(asap2); Serial.println("ppm");
}

```

Conditional statement for the sensors: if the temperature exceeds 45°C and the smoke concentration exceeds 600 ppm, the alarm will sound, and the LCD display will show the detected temperature and smoke concentration prominently.

WiFi Design

The first step is to initialize the libraries used. For the WiFi design, two libraries are used: WiFiClient.h and BlynkSimpleEsp32.h, which are used to connect to the Blynk application. After initialization, the Blynk authentication code is entered, along with the WiFi name and password, into the program to connect to the internet. If all goes well, Blynk will be connected.

RESULTS AND DISCUSSION

Hardware Testing Results After all circuit components were connected to form a unified system for use as a fire detection monitoring system, testing was conducted on the circuit. The aim was to ensure that all components used could function as expected. Testing covered both hardware and software aspects of the fire detection system. Testing was conducted by assessing the circuit's response to fire situations, such as igniting a flame to detect temperature through the temperature sensor and generating smoke to detect fires through the smoke sensor.

Results of DS18B20 Temperature Sensor, Buzzer, and LCD 16x2 Testing was conducted to evaluate the sensitivity of the DS18B20 sensor to air temperature in the environment. Experiments were carried out using a lighter gas. The DS18B20 sensor was used as input to detect potential fires based on the temperature around the room. The data detected by the sensor was then processed by the ESP32 to read temperature information. The sensor readings were displayed on the LCD 16x2 screen. The buzzer would activate if the temperature exceeded the threshold set, which was 45°C. If the temperature did not exceed the set threshold, the buzzer would remain inactive.

Table 2. Sensor Testing DS18B20

No	Testing (Temperature)	Buzzer	Location	Display on LCD
1	Electric lighter	inactive	Room A	33,5 ⁰ C
2	Electric lighter	inactive	Room B	39,4 ⁰ C
3	Electric lighter	active	Room A	63 ⁰ C
4	Electric lighter	active	Room B	48,9 ⁰ C

Test Results for MQ-135 Temperature Sensor, Buzzer and 16x2 LCD

Testing was conducted to assess the sensitivity of the MQ135 sensor to gases in the air surrounding the room. Experiments were carried out using smoke from burning paper. The MQ135 sensor was used as

input to detect potential fires based on the smoke concentration around the room. The data detected by the sensor was then processed by the ESP32 to read information about the gases in the air. The sensor readings were displayed on the LCD 16x2 screen. The buzzer would activate if the gas concentration exceeded the predefined threshold, which was 600 ppm. If the gas concentration did not exceed the predefined threshold, the buzzer would remain inactiv

Table 3 Sensor Testing MQ135

No	Testing (Temperature)	Buzzer	Location	Display on LCD
1	Paper burning smoke	inactive	Room A	153 ppm
2	Paper burning smoke	inactive	Room B	67 ppm
3	Paper burning smoke	active	Room A	1000 ppm
4	Paper burning smoke	active	Room B	669 ppm

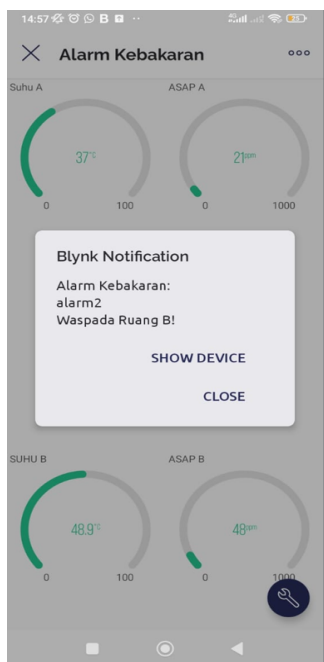
Overall Tool Testing

After conducting tests on the DS18B20 sensor, buzzer, and LCD 16x2 by measuring the temperature around the room, the test results showed good performance. Next, testing was conducted on the MQ135 sensor, buzzer, and LCD 16x2 by detecting smoke around the room, and the results were as expected. Subsequently, further testing aimed to assess the sensitivity of the DS18B20 and MQ135 sensors configured with the buzzer, LCD 16x2, and Blynk application. The data sent by the sensors would be displayed on the smartphone.

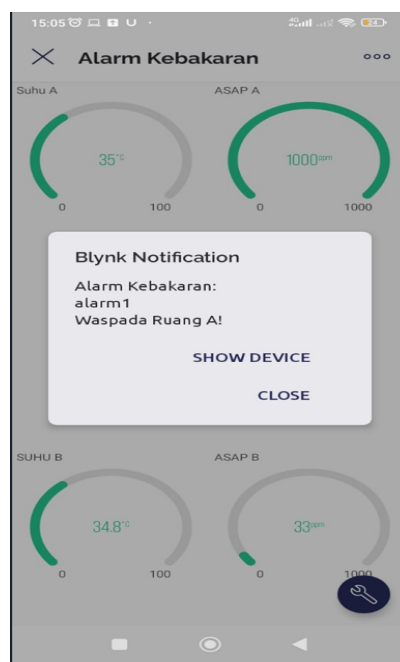
Testing was conducted using an electric lighter and smoke from burning paper, as shown in the table below:

Table 4 Overall Tool Testing

No	Testing	Room A		Room B		Buzzer	Blynk
		Temp (°C)	Smoke (ppm)	Temp (°C)	Smoke (ppm)		
1	Electric lighter (Room B)	37	21	48.9	48	Alarm2 (Active)	Alert Room B
2	Paper burning smoke (Room A)	35	1000	34.8	33	Alarm1 (Active)	Alert Room A



(a)



(b)

Picture 6(a). Smartphone Notification indications of a fire in Room B
Picture 6(b). Smartphone Notification indications of a fire in Room A

Jani, A. R. R., Handayani, D., & Noeman, A. (2021) conducted research on a fire detection system using a microcontroller that was not based on wireless fidelity (WiFi). In contrast, this study utilizes a microcontroller-based fire detection system connected to WiFi, making it more effective and efficient in detecting fires. The use of WiFi technology allows this system to send real-time alerts to users through mobile devices or computers. Additionally, the system can be integrated with smart home networks, enabling remote control and monitoring. Another advantage of the WiFi-based system is its ability to access data and perform centralized analysis, which enhances response to fire incidents and aids in faster decision-making. This research demonstrates a significant improvement in the speed and accuracy of fire detection compared to systems that do not use WiFi connectivity.

CONCLUSION

Based on the conducted tests, the author concluded the testing was conducted in Room A and Room B. When the temperature in Room A is detected to exceed 45°C or smoke concentration exceeds 600 ppm, alarm1 placed in Room A will be activated, and a notification on the smartphone will indicate that alarm1 is active and alert for Room A. Similarly, when the temperature in Room B is detected to exceed 45°C or smoke concentration exceeds 600 ppm, alarm2 placed in Room B will be activated, and a notification on the smartphone will indicate that alarm2 is active and alert for Room B. When both Room A and Room B are detected with temperatures exceeding 45°C or smoke concentration exceeding 600 ppm, both alarm1 and alarm2 will be activated, and a notification will be displayed indicating that alarm1 is active and alert for Room A and alarm2 is active and alert for Room B. Suggestions that need to be considered from the results of this research to improve the tools capabilities and further development include a 16x2 LCD was used, where the display on the LCD alternates with a 10-second delay. It is recommended for future research to use a 20x4 LCD so that sensor readings can be displayed all at once without waiting for switching then adding several other sensors for detection to increase accuracy so that more specific and abundant data can be obtained.

REFERENCES

- Adedeji, W. O., Adekoya, A. O., Olukayode, O., Oyewole, K. A., & Amosun, T. S. (2023). Design and Simulation of Programmable Logic Controller Based Multi-Channel Fire Detection and Alarm System. *Adeleke University Journal of Engineering and Technology*, 6(1), 22-32
- Alifia, A., Syifaul, F., dan Endah, S. (2023). Sistem Pendeteksi Gas Amonia Menggunakan MQ-137 Pada Air Berbasis Internet of Things Dengan Aplikasi Blynk di Android. *Techné Jurnal Ilmiah Elektroteknika*, 22(2), 285-304
- Arida A Rosa, Bryan A Simon, dan Kevin, S, L,. (2020), "Sistem Pendeteksi Pencemaran Udara Portabel Menggunakan Sensor MQ-7 dan MQ-135," *Ultima Computing Jurnal Sistem Komputer* 12(1), 23-28
- Asri, S. A., Astawa, I. N. G. A., Sunaya, I. G. A. M., Yasa, K. A., Indrayana, I. N. E., & Setiawan, W. (2020, July). Implementation of Prototyping Method on Smart Village Application. *Journal of Physics: Conference Series* (Vol. 1569, No. 3, p. 032094). IOP Publishing
- Abdullah, A. ., Nugraha, W., & Kusno. (2022). Fire Hazard Prevention and Protection Through Active Protection System. *Airman: Jurnal Teknik dan Keselamatan Transportasi*, 5(2), 104–112. <https://doi.org/10.46509/ajtk.v5i2.293>
- Budi Artono and Fredy Susanto, (2017). LED Control System with Cayenne Framework for the Internet of Things (IoT), *Electronic and Automotive Engineering (JEECAE)*, 2(1), p. 97

- Buser, S.J. (2023). Mental Health Issues in the Fire Service. In: Bourke, M.L., Van Hasselt, V.B., Buser, S.J. (eds) *First Responder Mental Health*. Springer, Cham. https://doi.org/10.1007/978-3-031-38149-2_2
- Cameron. N. (2023). *ESP32 Microcontroller*. In book: *ESP32 Formats and Communication*. DOI: 10.1007/978-1-4842-9376-8_1
- Fezari, M., Ali, A, D., (2018). *Integrated Development Environment (IDE) for Arduino*. https://www.researchgate.net/publication/328615543_Integrated_Development_Environment_IDE_For_Arduino.
- Fezari, M dan Ali. A. D., (2019). Exploring One-Wire Temperature Sensor “DS18B20” with Microcontrollers. <https://www.researchgate.net/publication/330854061>
- Hermianto, K., Mohammad, A. Iqbal., dan Imam, A. R. (2022). Alat Pendeteksi Kebakaran Menggunakan SMS Dan Panggilan Telepon. *Jurnal Elektro Kontrol (ELKON)*, 2(2), 55-65. DOI: 10.24176/elkon.v2i2.8583
- Hopkins, J.R., McKenna, T.P. & Bennett, A.E. (2024). Fire Season and Time since Fire Determine Arbuscular Mycorrhizal Fungal Trait Responses to Fire. *Plant Soil* (2024). <https://doi.org/10.1007/s11104-024-06500-5>
- H. Erwin., Haris. D., Farid. B., Anan. S., Suliadi., dan Aulia. Y., (2022). Sistem Pendeteksi Asap Rokok Dengan Sensor MQ-2 Berbasis Mikrokontroler ESP32. *Jurnal Aplikasi IPTEK Indonesia*, 6(1), 15-20, DOI: 10.240336/4.16457
- Ismailov, A. S., & Jo‘Rayev, Z. B. (2022). Study of Arduino Microcontroller Board. *Science and Education*, 3(3), 172-179
- Jani, A. R. R., Handayani, D., & Noeman, A. (2021). Fire Detection System Using Android-Based Microcontroller. *PIKSEL: Penelitian Ilmu Komputer Sistem Embedded and Logic*, 9(2), 173-184.
- Khyioon, Z., Intisar, S., Farah, N., (2020). Capable of Gas Sensor MQ-135 to Monitor the Air Quality with Arduino uno. *International Journal of Engineering Research and Technology* 13(10):2955-2959
- Kurnia, M, A. (2022). Rancang Bangun Alat Pendeteksi Dan Pemadam Kebakaran Otomatis Berbasis Mikrokontroler Atmega2560. *Prosiding Sains Nasional dan Teknologi* 12(1),551. DOI: 10.36499/psnst.v12i1.7211
- Liu, X., & Yan, K. (2023, May). Design and implementation of Fire Alarm Module for Smart Home. *International Conference on Electronic Information Engineering and Data Processing (EIEDP 2023)* (Vol. 12700, pp. 889-894). SPIE
- Mahzan, N. N., Enzai, N. M., Zin, N. M., & Noh, K. S. S. K. M. (2018, June). Design of an Arduino-based home fire alarm system with GSM module. In *Journal of Physics: Conference Series* (Vol. 1019, No. 1, p. 012079). IOP Publishing
- Manfaluthy, M., Agung, P., Imam, N. (2022). Prototipe Sistem Pendeteksi Kebakaran Berbasis ESP8266 dan IFTTT. *TELKA - Telekomunikasi Elektronika Komputasi dan Kontrol* 8(1), 60-73. DOI: 10.15575/telka.v8n1.60-73
- Muheden, K., Erdem, E., & Vançin, S. (2016, November). Design and implementation of the Mobile Fire Alarm System using Wireless Sensor Networks. *2016 IEEE 17th International Symposium on Computational Intelligence and Informatics (CINTI)* (pp. 000243-000246). IEEE.

- N. Brushlinsky, M. Ahrens, S. Sokolov, and P. Wagner. (2020). World Fire Statistics, in *International Association of Fire and Rescue Service*, pp. 1–67
- Narji, M., Arie, B. U., Dedi, S., dan Tata, S. (2022). Perancangan Prototype Embedded System Alat Pendeteksi Dini Kebakaran Berbasis Mikrokontroler Atmega8535. *Jurnal Teknologi Informatika dan Komputer* 8(2),307-319. DOI: 10.37012/jtik.v8i2.1253
- Pare, S., Tallulembang, T. M., dan Jarot, B. (2022). Pemanfaatan Teknologi Android sebagai Media Alat Pendeteksi Kebakaran Berbasis Arduino. *Jurnal Instek (Informatika Sains Dan Teknologi)* 7(1),59-68. DOI: 10.24252/instek.v7i1.27688
- Peraturan Pemerintah Republik Indonesia Nomor 16 Tahun 2021 tentang Peraturan Pelaksanaan Undang-Undang Nomor 28 Tahun 2002 Tentang Bangunan Gedung. Tanggal 2 Februari 2021.
- Peraturan Menteri Pekerjaan Umum dan Perumahan Rakyat Republik Indonesia Nomor 20 Tahun 2021 tentang Bangunan Gedung Fungsi Khusus. Tanggal 31 Maret 2021
- Real, R., Snider, C., Goudswaard, M., & Hicks, B. (2021). Dimensions of Knowledge in Prototyping: a Review and Characterization of Prototyping Methods and their Contributions to Design Knowledge. *Proceedings of the Design Society, 1*, 1303-1312
- Roihan, A., Nina, R., dan Stevanus. (2022). Perancangan Peringatan Dini Kebakaran Berbasis ESP8266 Menggunakan Sensor Fusion. *Jurnal Ilmiah Teknik dan Ilmu Komputer*, Vol. 1 No. 4, 12 - 17- ISSN: 2828-5344 | DOI: 10.55123
- Rombang, A, R., Lukas, B, S., dan Gunawan, D., (2022). Perancangan Prototipe Alat Deteksi Asap Rokok dengan Sistem Purifier Menggunakan Sensor MQ-135 dan MQ-2. *Techné Jurnal Ilmiah Elektroteknika*, 21(1):131-144
- Romony, P, V, B., Lanny, S., dan Junaidy B, S. (2020). Perbandingan Quality of Service Protokol Komunikasi Data pada Sistem Deteksi Asap Rokok Berbasis Internet of Things. *Jurnal Ilmiah Realtech* 16(1)
- R. Sequeira et al. (2015). Automated Control System for Air Pollution Detection in Industries. *International Journal of Students Research in Technology & Management*, vol.3, no.5, pp. 355-357.
- Saifullana, Joni, W, S. (2019). Sistem Pendeteksi Kebakaran Rumah Terintegrasi Smartphone dan Aplikasi Online. *Journal of Electrical and Electronics (JERC)*. Vol. 6 No. 2. Februari 2019
- Siregar, T. S., Sertya, P. S., Gatot, E. P., & Maulana, M, I. (2021). Rancang Bangun Sistem Pendeteksi Kebakaran Berbasis Iot Menggunakan Arduino. *AME (Aplikasi Mekanika dan Energi) Jurnal Ilmiah Teknik Mesin* 7(2),59. DOI: 10.32832/ame.v7i2.5063
- S. Widodo et al.,(2017). Rancang Bangun Alat Monitoring Kadar Udara Bersih Dan Gas Berbahaya CO, CO2 dan CH4 di dalam Ruangan Berbasis Mikrokontroler. *Jurnal Pseudocode*, vol.4, no.2, pp. 105-119
- Widyatmika, I. P. A. W., N. P. A. W. Indrawati., I. W. W. A. Prastya., I. K. Darminta, I. G. N. Sangka., dan A. A. N. G. Saptaka., (2021). Perbandingan Kinerja Arduino Uno dan ESP32 terhadap Pengukuran Arus dan Tegangan. *Jurnal Otomasi Kontrol & Instrumentasi*, Vol 13 (1), 37-45. DOI: 10.5614/joki.2021.13.1.4
- Winarno, A., dan Awang, J, M. (2023). Desain Sistem Pendeteksi Kebakaran Hutan Dengan GPS dan Telegram. *Tesla Jurnal Teknik Elektro* 25(1):1-12. DOI: [10.24912/tesla.v25i1.22931](http://doi.org/10.24912/tesla.v25i1.22931)

Zaini, M., Safrudin, dan Bachrudin, M., (2020). Perancangan Sistem Monitoring Tegangan, Arus dan Frekuensi pada Pembangkit Listrik Tenaga Mikrohidro berbasis IoT, *Tesla: Jurnal Teknik Elektro*, 22(2)

Zikrullah, A. P., Rima, T., dan Iskandar, F., (2022). Prototipe Sistem Monitoring Pendeteksi Kebakaran Menggunakan Fitur Looping. *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 7(1):224-230. DOI: 10.29100/jipi.v7i1.2536