Faculty of Electronic & Computer Engineering And Technology

DEVELOPMENT OF A LORA-IOT BASED SOLAR POWERED FOREST FIRE DETECTION USING ARDUINO

MOHAMMAD IKHWAN HAZIQ BIN KHAIRUL ANUAR

Bachelor of Electronics Engineering Technology (Telecommunications) with Honours

2024

DEVELOPMENT OF A LORA-IOT BASED SOLAR POWERED FOREST FIRE DETECTION USING ARDUINO

MOHAMMAD IKHWAN HAZIQ BIN KHAIRUL ANUAR

A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Electronics Engineering Technology with Honours

Faculty of Electronic & Computer Engineering And Technlogy

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI KEJURUTERAAN ELEKTRIK DAN ELEKTRONIK

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : DEVELOPMENT OF A LORA-IOT BASED SOLAR POWERED FOREST FIRE USING ARDUINO

Sesi Pengajian :23/24 Semester 1

Saya Mohammad Ikhwan Haziq Bin Khairul Anuar mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (✓):

☐	SULIT*	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	TERHAD*	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
☑	TIDAK TERHAD	

Disahkan oleh:

(TANDATANGAN PENULIS)

Alamat Tetap: No 28 Jalan Makyong 4H/KU5 Bandar Bukit Raja, 41050 Klang, Selangor

Tarikh: 12/1/2024

(COP DAN TANDATANGAN PENYELIA)

NURUL HALIM BIN HASSIM
Pensyarah
Fakulti Teknologi Dan Kejuruteraan Elektronik Dan Komputer (FTKEK)
Universiti Teknikal Malaysia Melaka (UTeM)

Tarikh: 12/1/2024

*CATATAN: Jika laporan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that this project report entitled “DEVELOPMENT OF A LORA-IOT BASED SOLAR POWERED FOREST FIRE DETECTION USING ARDUINO” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :

Student Name : MOHAMMAD IKHWAN HAZIQ BIN KHAIRUL ANUAR

Date : 12/1/2024

APPROVAL

I approve that this Bachelor Degree Project 1 (PSM1) report entitled "DEVELOPMENT OF A LORA IOT BASED SOLAR POWER FOREST FIRE DETECTION USING ARDUINO" is sufficient for submission.

Signature :

Supervisor Name : NURULHALIM BIN HASSIM

Date : 12/1/2024

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Electrical Engineering Technology with Honours

Signature :

Supervisor Name : NURULHALIM BIN HASSIM

Date : 12/1/2024

DEDICATION

In expressing my profound gratitude, I extend heartfelt appreciation to my progenitors, siblings, and compatriots, whose unwavering support and unwritten encouragement have been the bedrock of my successful culmination of the final year project. Concurrently, I deem it fitting to dedicate this scholarly treatise to the venerable Nurulhalim Bin Hassim, my esteemed supervisor, whose sagacious counsel has been instrumental in navigating the intricate pathways leading to the triumph of my final year endeavor. My sincerest thanks resonate profoundly, the depth of my gratitude is immeasurable. I hold in high regard their inexorable sacrifices, boundless tolerance, and judicious considerations that have rendered this undertaking not only conceivable but triumphant. In attempting to articulate my sentiments adequately, I find myself bereft of words sufficient to encapsulate the extent of my appreciation for their steadfast allegiance, unwavering support, and unswerving belief in my capacity to actualize my aspirations.

ABSTRACT

The importance of forest fires in maintaining a tranquil environment and ensuring the sustainability of the environment for both people and wildlife. The Development of a Lora IoT Based Solar Powered Forest Fire Detection Using Arduino Uno is used to improve forest fire specifically for forest reserve. Forest reserve management and conservation initiatives are now seriously concerned about the potential of forest fires. The creation of a Lora Arduino-based Internet of Things (IoT)-based solar-powered forest fire detection that aims to lessen the harm caused by forest fires and help preserve natural resources. The system uses a mix of IoT sensors and Arduino Uno to keep track of the (DHT11) temperature, (DHT11) humidity, (FLAME) fire, (MQ2) smoke levels and voltage sensor within forest reserve. These sensors gather data, which is wirelessly relayed to a central control station where it is analysed and processed. When a fire is detected, the lora transmitter delivers the signal to lora receiver and nodemcu operates like the blynk app to obtain the alerts (notifications). The system immediately notifies the Emergency to the control station. This process an automated reaction to stop the fire from spreading. First off, it is a financially sensible solution that doesn't necessitate a sizable infrastructure investment. Second, it is a dependable and effective system that can identify fires of the earliest stages, enabling prompt intervention and fire containment. Finally, it is a green option that makes use of renewable energy sources, thus lowering the carbon footprint of forest management and conservation initiatives.

ABSTRAK

Kepentingan kebakaran hutan dalam mengekalkan persekitaran yang tenang dan memastikan kelestarian alam sekitar untuk manusia dan hidupan liar. Pembangunan Pengesanan Kebakaran Hutan Berkuasa Suria Berasaskan Lora IoT Menggunakan Arduino Uno digunakan untuk menambah baik kebakaran hutan khusus untuk hutan simpan. Inisiatif pengurusan dan pemuliharaan hutan simpan kini amat mengambil berat tentang potensi kebakaran hutan. Penciptaan pengesanan kebakaran hutan berasaskan Internet of Things (IoT) berasaskan Lora Arduino yang bertujuan untuk mengurangkan kemudaratan yang disebabkan oleh kebakaran hutan dan membantu memelihara sumber semula jadi. Sistem ini menggunakan gabungan penderia IoT dan Arduino Uno untuk menjejaki suhu (DHT11), kelembapan (DHT11), kebakaran (FLAME), (MQ2) paras asap dan penderia voltan dalam hutan simpan. Penderia ini mengumpulkan data, yang dihantar secara wayarles ke stesen kawalan pusat di mana ia dianalisis dan diproses. Apabila kebakaran dikesan, pemancar lora menghantar isyarat kepada penerima lora dan nodemcu beroperasi seperti aplikasi blynk untuk mendapatkan makluman (pemberitahuan). Sistem segera memberitahu Kecemasan kepada stesen kawalan. Proses ini merupakan tindak balas automatik untuk menghentikan api daripada merebak. Pertama sekali, ia adalah penyelesaian yang wajar dari segi kewangan yang tidak memerlukan pelaburan infrastruktur yang besar. Kedua, ia adalah sistem yang boleh dipercayai dan berkesan yang boleh mengenal pasti kebakaran pada peringkat terawal, membolehkan campur tangan segera dan pembendungan kebakaran. Akhirnya, ia adalah pilihan hijau yang menggunakan sumber tenaga boleh diperbaharui, sekali gus mengurangkan jejak karbon dalam inisiatif pengurusan dan pemuliharaan hutan.

ACKNOWLEDGEMENTS

In the name of the Most Merciful and Generous Allah

First and foremost, I want to express my gratitude and praise to Allah, the Almighty, who is also my Creator and Sustainer, for everything that has been provided for me since the beginning of my life. I express my gratitude to the Universiti Teknikal Malaysia Melaka (UTeM) for granting access to the research platform.

My sincere gratitude is extended to Nurulhalim Bin Hassim, my supervisor at Universiti Teknikal Malaysia Melaka (UTeM), for all of his assistance, counsel, and motivation. His unwavering patience in mentoring and imparting invaluable insights will always be cherished.

Finally, I would want to express my sincere gratitude to my family, who have supported me through all of my endeavours and have been a source of encouragement, tolerance, and understanding. I express my gratitude to my dear friends for their unwavering love, support, and prayers. Lastly, I would like to express my gratitude to everyone who helped, encouraged, and supported me while I started my education.

Table of Contents

CHAPTER 1

INTRODUCTION

1.1 Research Background

The Internet of Things (IoT) is a fast-expanding field that entails connecting real-world objects and machinery to the internet so that they may talk to and interact with one another. The detection of forest reserved fires are one potential use case for IoT technology. The environment, animals, and human life are all seriously threatened by forest fires, and it is essential to notice them early in order to stop them from spreading. Aside from that, solar power is another energy source that is becoming more and more popular in IoT devices because it is renewable and safe for the environment. Remote or off-grid areas, such as forests, can employ solar panels to power IoT devices.

Widely used for Internet of Things (IoT) projects is the open-source electronics platform Arduino. Due to its simplicity, adaptability, and affordability, it is well-liked among professionals, academics, and hobbyists. Several different IoT devices, including those for detecting and preventing forest fires, can be made using the Arduino platform. The usage of sensors to detect the presence of smoke and heat, as well as a microcontroller to analyse the sensor data and send an alert if a fire is detected, are required for the construction of a Lora IoT-based Forest fire monitoring using Arduino and solar power. All things considered, the development of a Lora Internet of Things (IoT)-based system utilising an Arduino and solar power for early forest fire monitoring could provide a workable, feasible option.

1.2 Problem Statement

Forest fires are a serious environmental risk that can seriously harm animals, natural resources, and human life. Traditional fire detection techniques are frequently ineffective and may not send timely alerts to firefighters, delaying their arrival and increasing the danger of damage.

Next, Lack of early detection of A major problem in managing forest fires is the absence of early detection technologies. Forest fires are frequently not discovered until they have already spread across a considerable region, making suppression efforts more challenging and dangerous.

Climate change, when it comes to forest fires, climate change is a serious issue. The frequency and intensity of wildfires have grown due to rising temperatures and shifting weather patterns, making it progressively harder to prevent and contain them. Human factors of other things that contribute to forest fires include campfires, smoking, and fireworks. Human behaviour, such as disregarding safety precautions or interfering with firefighting operations, can also make it challenging for firefighters to put out flames.

1.3 Project Objective

The main aim of this project is to propose a systematic and effective methodology to estimate system involves integrating Lora IoT and Arduino technologies to provide forest fire detection. Specifically, the objectives are as follows:

a) To design forest reserved fire due to early detection using sensors, lora, solar powered and IoT by blynk app.

b) To simulate the sensors, lora, solar powered and IoT by blynk app on Proteus 8 professional and fritzing.

c) To develop and evaluate the hardware of sensors, lora, solar powered and IoT by blynk app.

1.4 Scope of Project

The scope of this project are as follows:

- Design and development: Using the Arduino microcontroller and other hardware components, the system would first be designed and developed. This may involve creating the circuitry and programming the microcontroller, as well as choosing the proper sensors, communication components, and power sources.
- Solar Power: Since the system would be installed in isolated forest regions, it would be crucial to develop a solar power system that can offer a dependable source of electricity. This could entail picking the right solar cells and batteries controllers to supply the system with the necessary power.

- Lora Transmitter: To transmit the data up to 5km of the 433Mhz frequency by the sensors and output. It utilizing Arduino uno.
- Lora Receiver: To receive the data up to 5km of the 433Mhz frequency by the sensors and output. It utilizing Arduino uno.
- IoT Connectivity: To link with a central monitoring station and provide real-time monitoring and warnings, the system may employ IoT connectivity. This could entail sending data from the sensors to the monitoring station using Wi-Fi.
- Sensors: For the system to identify forest fires, sensors such as temperature, smoke, and flame sensors would be required. To monitor environmental elements like humidity, smoke, temperature, and fire that may affect the chance of a fire starting, additional sensors could be employed.
- Data processing: The sensors collected data would need to be processed and looked through in order to spot probable fires and generate reports.
- Alert Generation: Once a probable fire has been identified, the system must produce alerts so that the appropriate authorities are informed and preventative measures can be taken. The use of buzzer, led, and blynk app.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Forest fires are becoming more frequent and more dangerous, endangering both human life and the health of natural ecosystems. In order to lessen the devastation caused by these fires, early discovery and quick action are essential. New prospects for effective forest fire detection have been brought about by recent technical breakthroughs, especially with regard to the Lora Internet of Things and renewable energy sources (IoT). This project aims to build an IoT-based solar power forest fire monitoring system in order to maximise the benefits of IoT connectivity, solar energy, and Arduino microcontrollers. Significant research has been done on systems for detecting forest fires, IoT-based monitoring options, using Arduino in environmental applications, and integrating solar power for remote monitoring, according to the literature study done for this project. In order to detect forest fires, the review emphasises the significance of temperature, humidity, and smoke levels. In order to provide wireless communication and data transfer in IoT-based systems, it also emphasises the advantages and versatility of the ESP8266 Wi-Fi module. Numerous studies have examined the integration of the ESP8266 module with Arduino and solar power systems, proving the integration's practicality and efficacy. [1]

2.2 Related Previous Project

2.2.1 Forest Fire Detection System (Sensing)

Temperature-Based Detection

Because it rises dramatically during a fire, temperature is a crucial measure for locating fire incidents. The use of infrared sensors is one of the many temperature-based detection techniques that have been suggested. These sensors gauge the temperature outside or the fire's thermal emissions. Researchers have looked into a variety of methods, including threshold-based algorithms, temperature gradient analysis, and pattern recognition algorithms, to identify unusual temperature increases brought on by fires. Early alerts and precise fire detection have both been made possible by the incorporation of temperature sensors into systems for detecting forest fires. [2]

Humidity-Based Detection

The amount of moisture in the air, or humidity, is a key factor in how a fire behaves and starts. Vegetation and combustible substances are more susceptible to igniting as humidity levels drop. The relative humidity levels in forested regions are monitored as part of humidity-based detection systems. Humidity fluctuations can be a sign of possible fire hazards when paired with other elements like temperature. To detect abnormal humidity (DHT11 Sensor) fluctuations and activate fire alarms or alerts, researchers have used a variety of techniques, including threshold-based algorithms and statistical analysis. The incorporation of humidity sensors into forest fire monitoring systems improves the accuracy of fire detection and aids in the identification of dangerously high fire hazard circumstances.[3]

Smoke-Based Detection

Fires produce smoke, which is an obvious and observable result. Methods for smoke-based detection (MQ2 Sensor) rely on the measurement and examination of smoke particles as well as the air's opacity. Smoke particle detection frequently makes use of optical sensors like photoelectric sensors and light-scattering sensors. These light-emitting sensors track changes in light intensity brought on by the presence of smoke particles. The distinction between typical atmospheric conditions and the presence of smoke is made using algorithms and methods like thresholding, pattern recognition, and machine learning. Even in the early stages of a fire, when flames may not be visible, smoke-based detection is extremely helpful in locating the fire. [4]

2.2.2 Reserved Forest Fire wi-fi monitoring

Wi-fi Module

The ESP8266 module delivers robust wireless connectivity to the Arduino uno that ensure seamless interaction between the conserved reserved forest fire detection system and the IoT platform or central server. By employing the module's built-in Wi-Fi capabilities, the system may send sensor data, such as temperature, humidity, and smoke levels, or after a fire, a message by telegraph, to a distant site for monitoring and analysis. The ESP8266 module has been effectively incorporated for wireless data transmission in a variety of Internet of Things applications, including remote sensing and environmental monitoring, according to numerous studies.[5]

Figure 2.2.2 shows the node MCU.

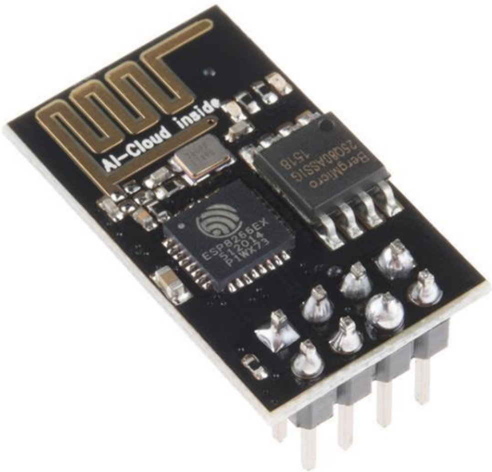


Figure 2.2.2: ESP8266 Wifi Module

2.2.3 Reserved Forest Fire detection for Arduino Uno

The ATmega328P microprocessor is the foundation of the open-source Arduino Uno microcontroller board. It offers a straightforward and user-friendly framework for programming and interacting with different electronic parts, such as sensors, actuators, and communication modules. Arduino Uno is the perfect option for creating IoT applications because of the availability of a huge selection of libraries, documentation, and community support.

Systems for detecting forest fires can be easily created by integrating a variety of sensors with an Arduino Uno, including temperature, humidity, and smoke sensors. Data collection and monitoring are made possible by the digital and analogue input/output ports on the board, which can be used to interface and connect these sensors. In order to avoid and detect fires, researchers have had success utilising the Arduino Uno to collect data from sensors and interpret it.

The Arduino Uno's computer capacity allows for real-time data processing and analysis. The board has the ability to execute algorithms to assess the sensor data collected from the forest environment. Algorithms have been created by researchers to identify smoke presence, sudden spikes in temperature, and drops in humidity, all indicators of potential fire outbreaks. The Arduino programming environment makes it simple to design and run these algorithms on the Arduino Uno device.

The forest fire protection system's actuators and other equipment are easier to control thanks to Arduino Uno. Researchers have been able to activate operations such as sounding alarms, or sending signals to distant devices for fire suppression or evacuation by using its digital output pins. Effective control tactics for preventing and containing forest fires are made possible by the Arduino Uno's capacity to link with many types of actuators.[6]

IoT connectivity can be achieved by connecting the Arduino Uno board with wireless communication modules, such as the ESP8266 Wi-Fi module. This integration allows the forest fire detection system to send sensor data, alarms, and notifications to a cloud-based platform or remote server for further monitoring and analysis. The Arduino Uno is ideally suited for smooth integration with the Internet of Things ecosystem because of its HTTP and MQTT connectivity.

Due to its low power consumption and effective power management features, Arduino Uno is a good candidate for solar power system integration. The forest fire detection can function independently in distant locations with little access to power sources by utilising solar energy. The Arduino Uno has been successfully integrated with solar power systems by researchers to guarantee the system's continuous operation and increase its sustainability. [7]

Figure 2.2.3 shows the Arduino Uno R3

Figure 2.2.3: Arduino Uno R3

2.2.4 IOT on Forest Fire

Real-time Monitoring and Data Acquisition

Due to its low power consumption and effective power management features, Arduino Uno is a good candidate for solar power system integration. The forest fire detection can function independently in distant locations with little access to power sources by utilising solar energy. The Arduino Uno has been successfully integrated with solar power systems by researchers to guarantee the system's continuous operation and increase its sustainability. [8]

Wireless Connectivity and Communication

Internet of Things-based Forest fire detection requires wireless connectivity to enable constant communication between sensors, control systems, and remote monitoring platforms. The central system or cloud-based platforms can receive sensor data transfers using wireless technologies like Wi-Fi, cellular networks, and LPWAN (Low Power Wide Area Network) protocols. Fast upgrades are guaranteed by this link, which also makes remote system control and monitoring possible.[4]

Cloud-based Data Management and Analysis

IoT-based systems for forest fire detection heavily rely on cloud computing. Scalable storage, data processing power, and cutting-edge analytics tools are all provided by cloud platforms. Data analysis, anomaly identification, and the forecasting of potential fire threats are all possible using sensor data acquired from the forest environment that has been safely saved and processed in the cloud. Additionally, data visualisation, reporting, and remote system access are made easier by cloud-based platforms.[9]

Integration with Artificial Intelligence and Machine Learning

The capabilities of forest fire detection are improved by the combination of IoT with artificial intelligence (AI) and machine learning (ML) techniques. AI algorithms can analyse large datasets from sensors and historical fire data to develop predictive models and identify patterns that indicate the risk of fire outbreaks. ML algorithms can continuously learn and adapt to changing environmental conditions, improving the accuracy of fire detection strategies.

Remote Monitoring and Alerting

Authorities, forest managers, and emergency responders may obtain real-time data and remotely monitor the fire situation thanks to IoT-based solutions for detecting forest fires. When a fire is detected or a dangerous environmental condition arises, alerts and notifications can be issued to the appropriate stakeholders by email, SMS, or push notifications. Remote monitoring and alerting enable prompt response actions and efficient firefighting effort coordination.

Solar Power Integration

Detecting forest fires autonomously and sustainably is made possible by the integration of solar power systems and Internet of Things (IoT) technology. Renewable energy generated by solar panels can power sensors, communication modules, and Internet of Things control systems. The system's dependability and self-sustainability are increased by the use of solar power, which guarantees constant operation in remote places with restricted access to the electrical grid. [10]

Figure 2.2.4 shows the IoT.

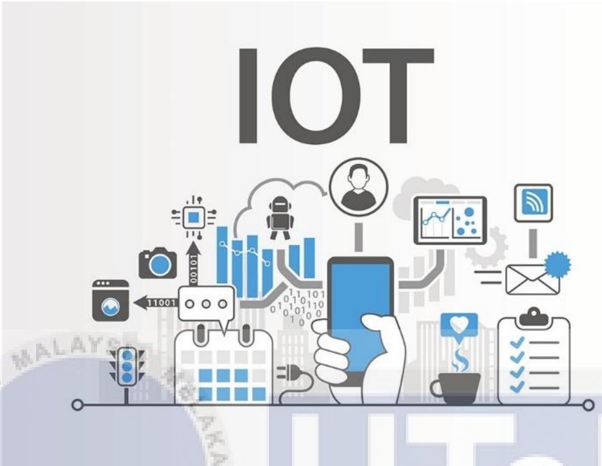


Figure 2.2.4: IOT on Forest Fire

2.2.5 Low Power and Solar Power Integration

<u>Energy Independence and Sustainability</u>

The forest fire monitoring benefits from solar power's energy independence and sustainability. The device may generate its own power, decreasing reliance on the electrical grid, by using solar panels to capture sunshine. This supports a sustainable approach to forest fire management and provides ongoing operation in remote places without access to traditional power sources.[11]

<u>Remote Deployment</u>

The remote deployment of the forest fire detection is made possible by solar electricity. Solar panels act as a dependable power source because forested areas do not have an electrical infrastructure. This eliminates the need for substantial electrical wiring or generator-based power supplies, allowing the system to be installed in a variety of settings, including remote forests.[11]

Autonomous Operation

The forest fire monitoring can operate autonomously thanks to solar electricity. The system's ability to store extra energy produced during the day and use it at night or during periods of low sunlight is made possible by the incorporation of energy storage equipment, such as batteries. This guarantees continuous operation and makes it possible to monitor the situation and take fire safety precautions even when it is dark outside.[12]

Scalability and Flexibility

Scalability and flexibility in system design are provided by solar power. To fulfil the system's rising energy needs or to increase its coverage area, further solar panels can be easily added. Based on the unique needs of the forested regions and the desired level of monitoring and protection, this scalability enables modification.[7]

Reduced Environmental Impact

The use of solar energy in the forest fire detection lessens the environmental impact. It provides a clean and renewable energy source, reducing the greenhouse gas emissions and air pollution caused by the production of conventional power. By using solar energy, the system adheres to ecological principles and helps to protect the forest ecology. [13]

Integration with IoT and Wireless Communication

The creation of the forest fire detection incorporates both solar power and IoT technologies. Temperature, humidity, smoke levels, and other environmental variables can all be measured by solar-powered Internet of Things devices that are also fitted with sensors and communication modules. Through wireless data transmission to a centralised monitoring system, these sensors provide real-time analysis, early fire detection, and prompt response.[6]

Reliability and Maintenance

In many applications, including the system for detecting and preventing forest fires, solar power systems have shown to be dependable. A dependable and effective power source is ensured by the proper design, sizing, and installation of solar panels and energy storage devices. To maximise their efficiency and lifespan, solar panels need to be maintained regularly, which includes cleaning and inspection.[10]

Cost-effectiveness

The operation of the forest fire detection can be done more affordably in the long run by using solar electricity. The system has low operational expenses since sunshine is abundant once the initial investment in solar panels and related equipment is made. Solar power is a desirable alternative because of its low cost, particularly for long-term installations in isolated forested locations. integrated TCP/IP protocol stack to give internet connectivity to the system. [7]

Figure 2.2.5 shows the function of solar-power.

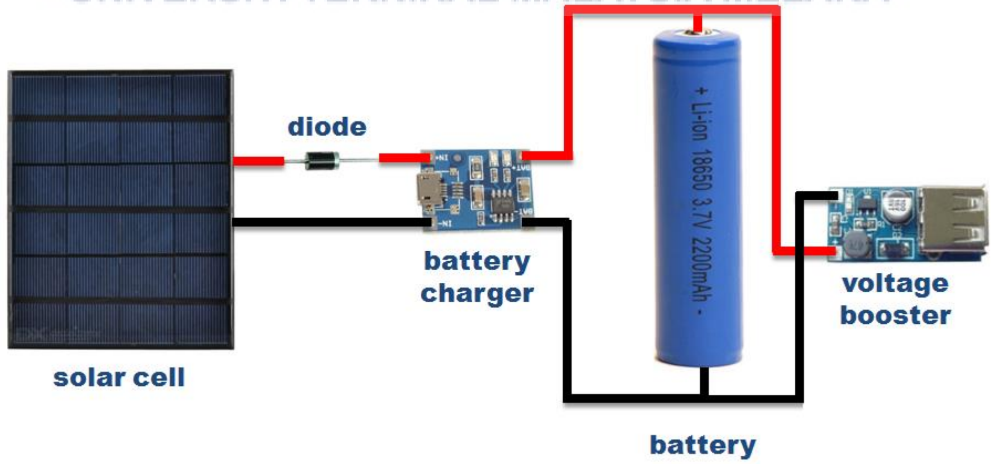


Figure 2.2.5: Solar Power Integration

2.2.6 Proposed Reserved Forest Fire architecture

Nodes for Lora Sensors and Data Collection To keep tabs on the forest environment, the architecture includes sensor nodes that are outfitted with a variety of sensors, including temperature, humidity, and smoke detectors[14]. Solar panels on these sensor nodes provide continuous functioning and lessen reliance on other power sources. Real- time data on environmental conditions is gathered by the sensor nodes, which is essential for preventing and detecting fires. [4]

Wireless Data Transmission and Communication Wireless transmission of the sensor nodes' collected data to a centralised control system allows for analysis and decision-making. Real-time and flawless data transmission are made possible by the usage of wireless communication technologies like Wi-Fi. Integrating solar power guarantees continuous communication, improving the system's dependability and effectiveness. [2]

Cloud-based Data Processing and Storage is the central control system makes use of cloud-based processing and storage resources. The gathered data is safely kept in the cloud, making it accessible and scalable. Real-time data analysis using machine learning algorithms and pattern recognition techniques is made possible by cloud-based processing, which helps to identify probable fire occurrences.

Advanced fire detection algorithms and early warning systems are incorporated into the proposed architecture to find patterns and abnormalities in the collected data that are related to fires. The technology can quickly identify and alarm possible fire emergencies by analysing elements including temperature rise, smoke levels, and ambient conditions[15]. Solar-powered early warning systems allow for prompt response and mitigation actions to lessen the effects of forest fires.[16]

Integration of Remote Monitoring and Control is the architecture enables remote monitoring and control of the system for detecting and preventing forest fires[17]. Through web-based or mobile interfaces, authorised staff can access the system and monitor sensor data, system status, and remotely control preventative measures[18], [19]. The use of solar power guarantees continuous operation and communication, enabling remote access even in locations with sparse infrastructure[20]

LoRa (Long Range) technology has grown in popularity, especially because of its applicability to low-power, low-data-rate, long-distance applications. LoRa is a proprietary spread spectrum modulation technology that was developed in response to the requirement for reliable communication over long distances over the Internet of Things (IoT). Its unique qualities, including as its exceptional ability to pass through barriers and buildings, make it ideal for outdoor and industrial applications, including environmental monitoring, smart cities, and smart agriculture.

The suggested architecture contains energy storage equipment, such as batteries, to store extra solar energy produced during daytime hours in order to ensure continuous operation. The system is powered by these energy storage systems at night or when there is little or no sunlight. In order to maximise the utilisation of stored energy and extend system autonomy, effective energy management strategies are used.

The suggested architecture contains energy storage equipment, such as batteries, to store extra solar energy produced during daytime hours in order to ensure continuous operation[23]. The system is powered by these energy storage systems at night or when there is little or no sunlight. In order to maximise the utilisation of stored energy and extend system autonomy, effective energy management strategies are used. Figure 2.2.6 shows the architecture of proposed reserved forest fire.

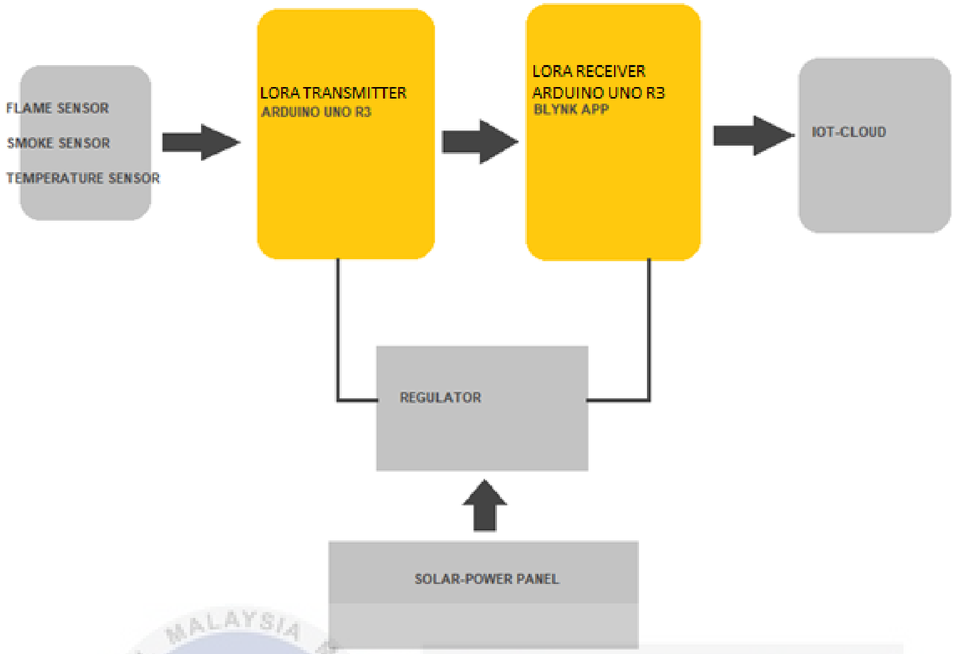


Figure 2.2.6: Proposed Reserved Forest Fire Architecture

2.3 Journal Comparison for Relevant Previous Research

Table 2.1 shows a comparison table for five distinct studies connected to this suggested project so that, after studying it, we may enhance or avoid the flaws.

The use of artificial intelligence (AI) and Internet of Things (IoT) technology for preventing forest fires is discussed in the article. To identify and stop forest fires, the authors suggest a system that integrates AI algorithms with IoT sensors. The system collects real-time data from numerous environmental sensors, including temperature and humidity sensors, analyses the data, and looks for potential fire threats using AI algorithms. The system intends to enhance forest fire prevention methods and lessen the damage caused by such disasters by using IoT-enabled devices and AI algorithms.[6]

Systems for producing electricity and renewable energy are the main topics of the article's next section. It talks about the various aspects of generating renewable energy, such as hydroelectric, wind, and solar power. The authors discuss the potential benefits and challenges of integrating renewable energy sources into the existing electrical grid.. They also look at new developments in renewable energy technologies and how they might be used to provide sustainable power. [10]

Following that, a presentation offered at the 2019 Renewable energy technologies and energy generation systems are the main topics of the International Conference on Advanced Computing & Communication Systems. It discusses advancements and challenges in the generation of harvesting which is re-new the energy and incorporating renewable energy sources into the electrical grid. The writers discuss several renewable energy sources, such as biomass, solar, and wind power, and they examine how these sources may affect the production of sustainable electricity. [24]

Lastly, "A Fire Detection System for Smart Home Based on IoT Data Analytics”. This article presents FireDS-IoT, an IoT data analytics-powered fire detection system designed for smart homes. The system uses Internet of Things (IoT) devices, for example, smoke detectors and temperature sensors, to collect data on fire-related properties in real-time. The system can identify fire occurrences in smart houses and initiate the proper responses, such as sounding alarms and notifying residents or emergency services, by analysing this data using IoT data analytics techniques. The authors outline FireDS-IoT's architecture and essential elements while highlighting the technology's potential to improve fire safety in settings such as smart homes. [3]

Table 2.1: Journal Comparison

No	System	Communication Interface	Controller	User Interface	Applications	Benefits
1	Wi-Fi based using Arduino Uno Microcontroller[24]	Wireless LAN	Hardware Interface Module	Remote Control	Forest sensors	Better scalability, flexibility, and aware.
2	Wi-Fi based using Arduino Uno Microcontroller [2]	Wireless LAN	Hardware Interface Module	Android Application	Robotic system	Autonomous and Scalable
3	Wi-Fi based using Arduino Uno Microcontroller [5]	Wireless LAN	Hardware Interface Module	Remote Control & Thingspeak	Collect Data	Better flexibility, Autonomous and Scalable
4	ArduinoUno Microcontrollerbased Wi-Fi system [4]	RF Module	Interface of Hardware Module	Web Based PC Application	Temperature, Gas, smoke and climate appliances	Better flexibility and Scalable, Remote access

5	Internet-of-Things Detection System. [12]	Wireless LAN and Bluetooth	Hardware Interface Module	Web Based PC Application and GPS	Intelligent Transportation, Smart cities and Tele-Medicine	Aware and Scalable, Remote access
6	Internet-of-Things Detection System. [3]	Wireless LAN	Hardware Interface Module	MongoDB	Collect Data	Autonomous and Scalable
7	This Project	Wireless	Hardware Interface Module	Android Application (blynk app)	Lora Forest sensors of Temperature, humidity, flame, smoke by Solar-powered.	Better flexibility and Scalable, Remote access, aware and sustainability.

2.4 Summary

The articles discuss a variety of subjects including IoT, renewable energy, artificial intelligence, and fire detection. While the second and third papers explore power generation systems and renewable energy technology, the first piece focuses on preventing forest fires with AI and IoT. The fourth article introduces FireDS-IoT, an IoT-based data analytics-based fire detection system for smart homes. Each article gives distinct insights into its subject matter and suggests potential solutions for dealing with problems and improving technology in that industry.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The exploring procedure is thoroughly explained in this section. It provides details on the research technique that was used as well as the justification for doing so. In addition, the part covers the many stages of the investigation, including information gathering and information investigation. Making flowcharts will therefore indicate a higher level of understanding. The materials for the task will be provided, along with instructions on how to build the circuit association. To develop a system of alarm from forest fire. The usage of programming tools like the Arduino IDE will generally be studied.

3.2 Project Workflow

A flowchart visual addresses the executive system undertaking that we are attempting to complete. The success of the upcoming project depends critically on the flowchart. There are numerous tactics and data that can be used to enhance a successful project's workflow. For instance, project stream data from high-efficiency projects might be improved, including analyses of various journals, research studies, and book publication studies.

3.2.1 Flowchart of Overall PSM

Figure 3.2.1 shows the flowchart of overall PSM by following the plans.

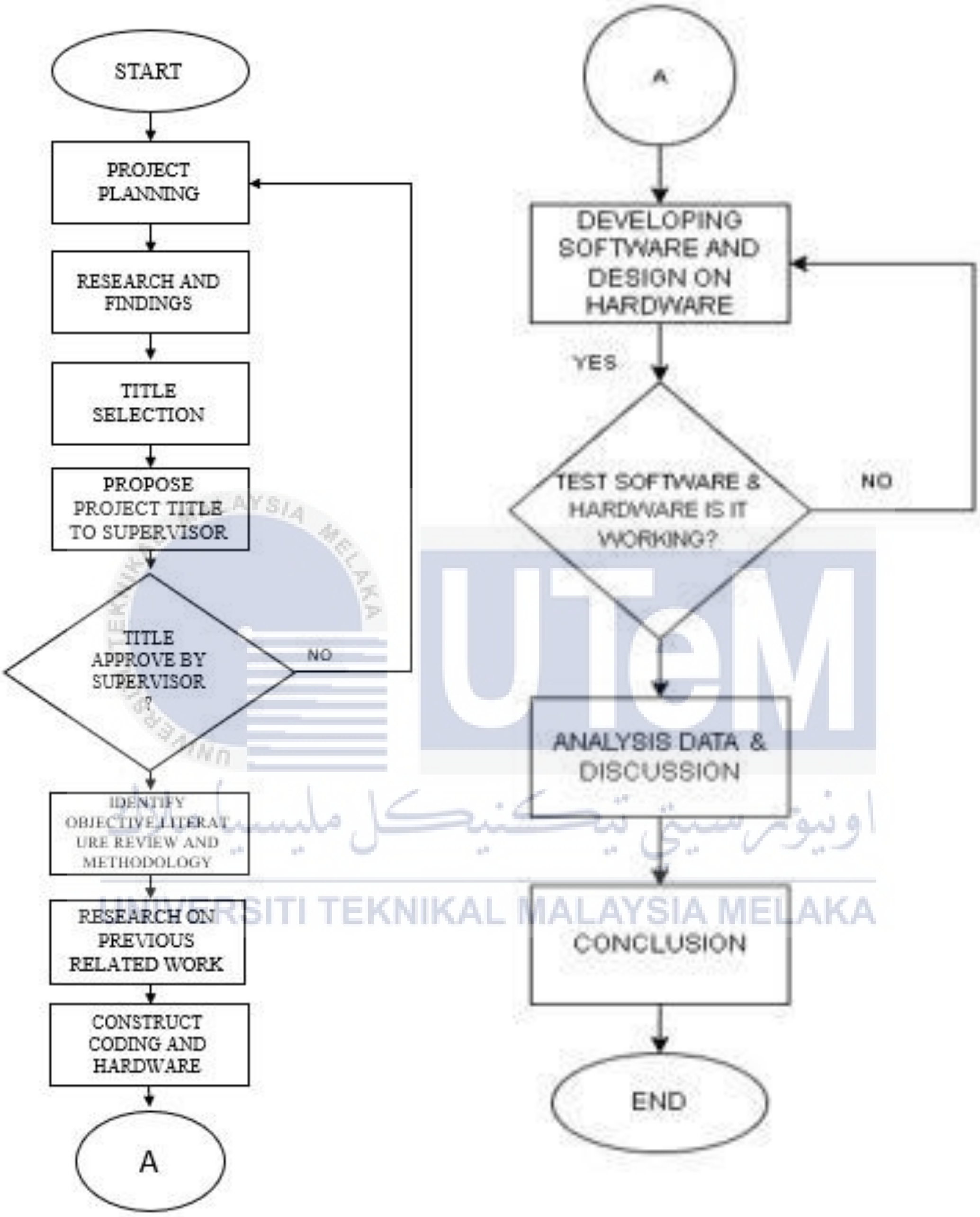


Figure 3.2.1: Flowchart of Overall PSM

3.3 Data Collection

Data collection is the process of obtaining all the information required to evaluate the project's efficacy. As a result, the literature review will serve as the project's guide for gathering data and information. A few examples of the vital information that will be stored on an Arduino Uno are sensors and volts. To obtain the required data, a system will be built using an Arduino Uno and all other components. To construct this project, all of the knowledge gathered from diverse sources was employed. Furthermore, studying previous initiatives could provide you a better grasp of the project, its objective, strategy, and outcomes, empowering you to make more wise decisions.

3.4 Design

Figure 3.4 shows the Diagram of System Design.

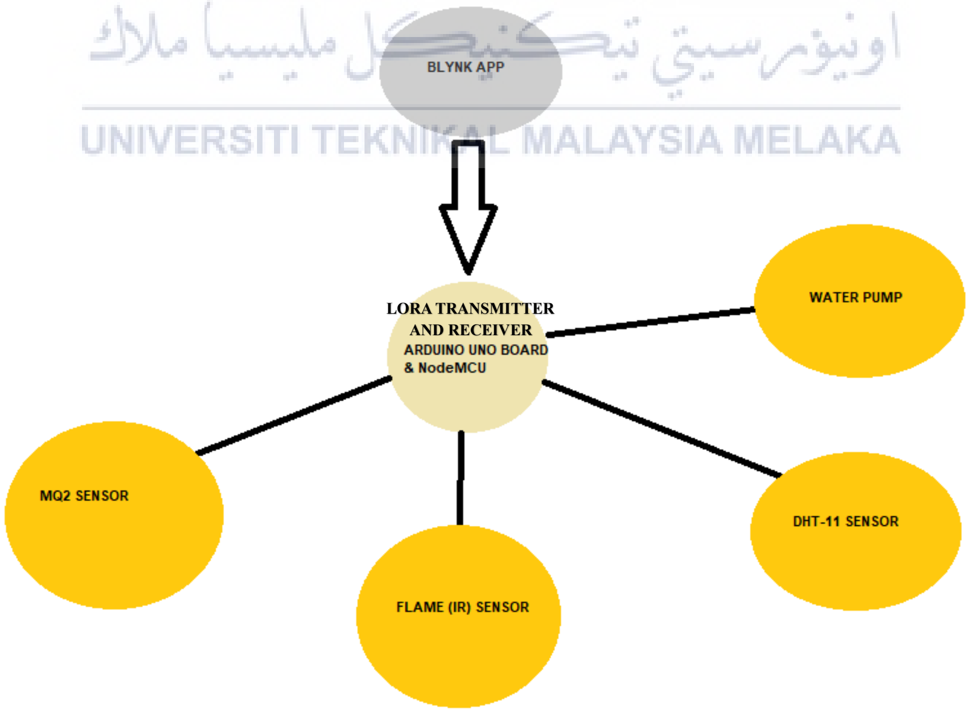


Figure 3.4: Diagram of System Design

This undertaking's framework breakdown is portrayed in Figure 3.4. In this undertaking, two techniques for correspondence are utilized: remote (through Blynk app) and NodeMCU. The ESP8266 is a Wi-fi module that permits the blynk app to the get information.

3.5 Hardware Specification

The execution of the entire framework is covered in this section, from the machines to the Arduino-Uno board. This framework is put together using an Arduino-Uno board, a wi-fi module (ESP8266), a transfer module, a DHT11 sensor, a Voltage sensor, an IR sensor (a flame sensor), a MQ2 sensor, a blynk app programme, and other hardware components.

3.5.1 Arduino Uno R3

An ATmega328 processor powers the Arduino Uno microcontroller board. The device comes with a 16 MHz clay resonator, six straightforward information sources, a USB port, a power connector, an ICSP header, and a reset button. Additionally, there are fourteen computerised yield/input pins, six of which can be utilised as PWM yields. This board is very simple to use and simply requires to connect it to a PC by using USB and power supply to it utilizing either a battery or an AC-to-DC adapter. The Arduino Uno varies from previous USB-to-chronic converters in that it uses the Atmega16U2 (or Atmega8U2 up to form R2) instead of the FTDI USB-to-chronic driver chip. The Arduino-Uno structure is shows in Figure 3.5.1.

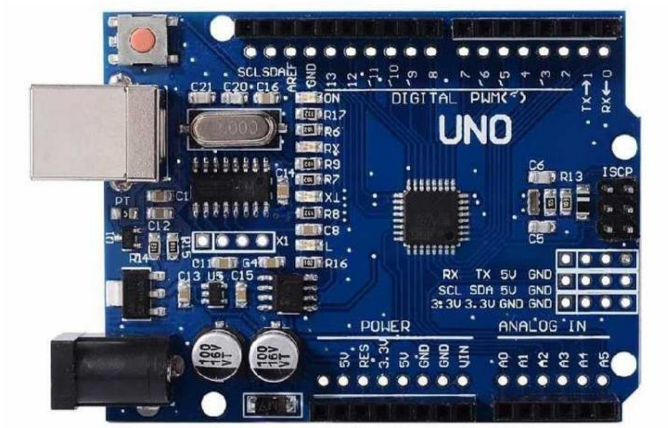


Figure 3.5.1: Arduino-Uno Structure

3.5.2 ESP8266 Wi-Fi Module

The NodeMCU development board has two Micro USB (Universal serial bus) connectors that enable direct connection to a desktop or laptop computer or other USB host devices, in addition to an embedded ESP8266 wi-fi module. 15 x 2 header pins are included for attaching it to a breadboard. It also includes a CP2102 USB to serial converter as an extra function. Node MCU is shows in Figure 3.5.2.

Figure 3.5.2: Structure of ESP8266 Wi-Fi Module (NodeMCU)

3.5.3 LORA

The process for creating an Arduino-based LoRa IoT-based solar-powered forest fire detection system entails creating a thorough system architecture that combines solar-powered components, LoRa modules, and forest fire detection sensors. Developing Arduino sketches for sensor data collection, LoRa communication, and power management techniques results in energy usage that is optimised by using solar energy and sleep modes. For dependable communication, a strong data transmission protocol is established, and LEDs are used in the implementation of an alarm system that allows for optional interaction with the Blynk app for remote monitoring and local feedback. Figure 3.5.3 shows Structure of Lora module.

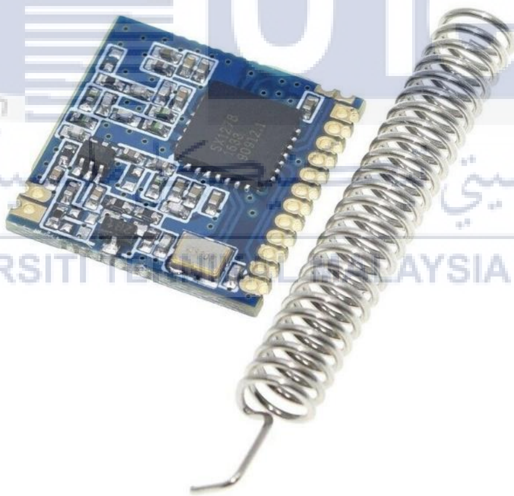


Figure 3.5.3: Lora module transmit and receiver

3.5.4 Flame Sensor

Based on the infrared sensor, the Fire or Flame Sensor Module can identify various light sources with wavelengths ranging from 760 to 1100 nanometers, in addition to fire flames. Its foundation is an NPN silicon phototransistor with great sensitivity and speed. The sensor has a detection angle of about 60 degrees and is very sensitive to the flame spectrum. The module's integrated op-amp (LM-393) is used to change the sensitivity level. The black epoxy makes it vulnerable to infrared rays as well. When a flame is detected, the signal LED illuminates (red light). The structure of the flame sensor (IR sensor) is depicted in Figure 3.5.4.

Figure 3.5.4: Structure of Flame Sensor (IR sensor)

3.5.5 Smoke Sensor (MQ2 sensor)

Thus, MQ2 gas sensors are electronic sensors used for sensing different gas concentrations present in the air like volatile organic compounds (VOCs), carbon monoxide, smoke etc. Figure 3.5.5 shows Structure of smoke sensor (MQ2 sensor).

Figure 3.5.5: Structure of smoke sensor (MQ2 sensor)

3.5.6 Temperature and Humidity Sensor

The DHT11 temperature and humidity sensor is a sensing device that uses a sophisticated calibration technique to provide a digital signal. Along with incorporating a digital data acquisition system, it uses a resistive-type humidity measurement system and a negative temperature coefficient (NTC) based temperature measuring system for sensing purposes. Therefore, when combined, the signal detecting and data collecting systems result in a very stable and long-lasting system that can be immediately connected to a high-performance microcontroller or CPU. As a result, DHT11 can offer a quicker reaction, outstanding quality, and the capacity to prevent interferences at a very reasonable price. Structure of Temperature and Humidity Sensor (DHT11 sensor)

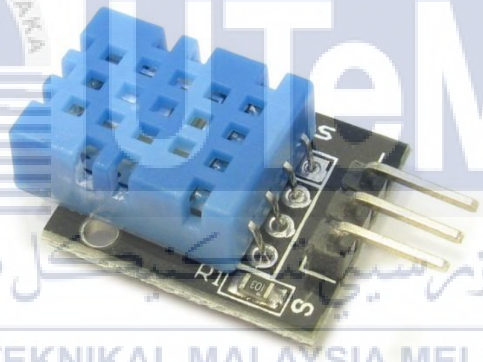


Figure 3.5.6: Structure of Temperature and Humidity Sensor (DHT11 sensor)

3.5.7 Voltage Sensor

A voltage divider circuit is commonly used in voltage sensors to divide the input voltage into smaller proportional voltages that an Arduino or microcontroller can measure. VCC, GND, and OUT are its three standard pins. To power the voltage sensor, the VCC pin is linked to a power source, typically 5V in the case of an Arduino. To provide a shared reference point, the GND pin is linked to the system's ground reference. To give the voltage measurement, the OUT pin is connected to one of the analogue input pins of the microcontroller or Arduino.

The voltage sensor internally transforms the analogue voltage into a corresponding digital value when voltage is applied to it. Often, an analog-to-digital converter (ADC) that is already integrated into the device is used to transform data. An digital value that indicates the voltage level is generated by the ADC after it samples the input voltage.

Once the voltage value is determined, it may be used to my project in a number of ways. The voltage sensor can be used, for instance, to track the voltage output of the solar panels in a system for solar-powered forest fire detection and suppression. A issue with the solar panels or charging system may be indicated if the voltage drops below a specified level, allowing you to take the necessary action, such as sending out notifications or turning on backup power sources. Figure 3.5.7 shows Structure of voltage sensor.

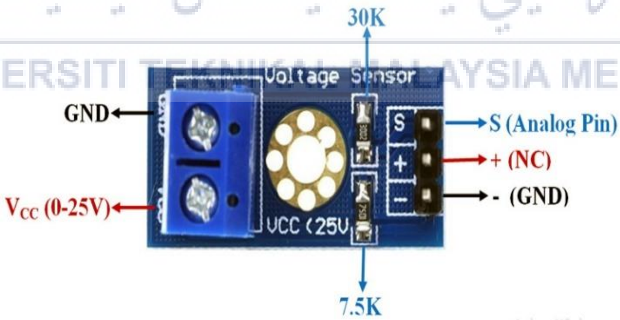


Figure 3.5.7: Structure of voltage sensor

3.5.8 Blynk App

The Blynk app, which offers a user-friendly interface for monitoring, controlling, and getting warnings remotely, is an essential component of any IoT-based solar power forest fire detection. Using widgets like Value Displays, Gauges, or Graphs, the Blynk software allows you to visualise real-time sensor data such as temperature, humidity, smoke, and voltage readings. You can configure the app to send you alerts and notifications that will send you an immediate notification in the event of important occurrences like excessive temperatures or smoke levels that could be fires. Additionally, Blynk gives you the ability to remotely control actuators, such as sprinkler systems or pumps, giving of the ability to take prompt action to reduce the chance of a fire. The ease of use and accessibility of my system are improved by the app's capacity to be accessed remotely and customised. This feature also enables you to monitor and control your solar-powered forest fire detection and at any time, from any location. Figure 3.5.7 shows Structure of blynk app works.

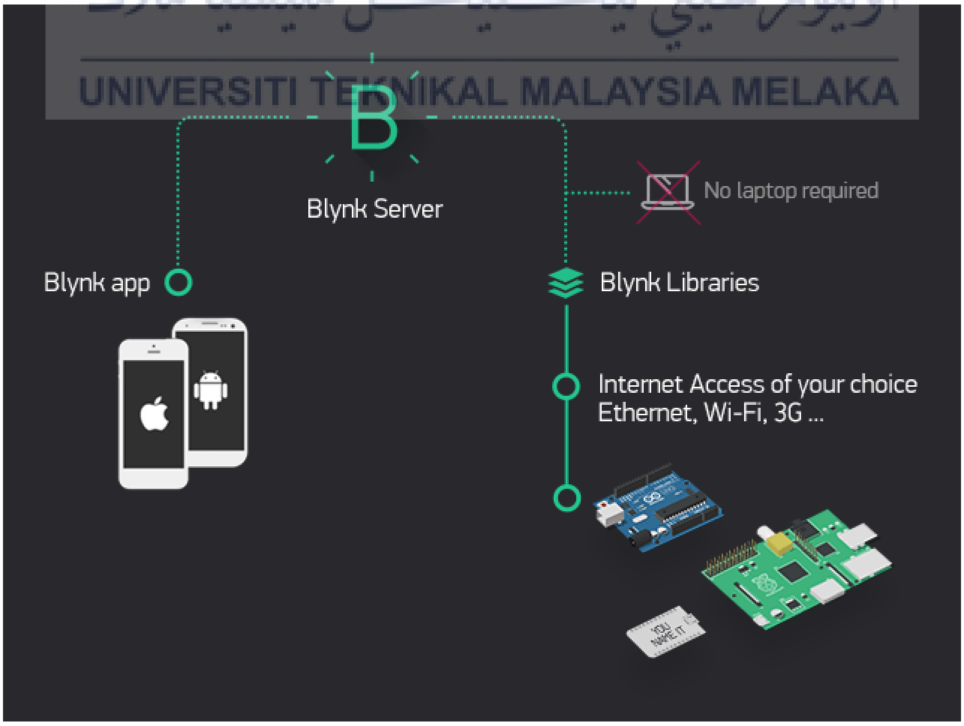


Figure 3.5.8: Structure of blynk app

3.6 Software Specification

3.6.1 Arduino IDE (Lora transmitter Arduino).

Figure 3.6.1 show the coding from Arduino IDE (Lora transmitter Arduino).

FOREST_FIRE_LORA_TRASMITER_ARDUINO | Arduino 1.8.14

File Edit Sketch Tools Help

FOREST_FIRE_LORA_TRASMITER_ARDUINO

```
#include <SPI.h>
#include <LoRa.h>
#include <LCD_I2C.h>
#include <SimpleDHT.h>

LCD_I2C lcd(0x27, 16, 2);
int pinDHT11 = 4;
SimpleDHT11 dht11(pinDHT11);
int counter = 0;
int smoke = A0;
int flame = 3;
int voltage = A1;
int value = 0;
float voltageValue;
float R1 = 47000.0;
float R2 = 33000.0;
String str;
int greenLed = 7;
int redLed = 6;

void setup() {
  Serial.begin(9600);

  pinMode(greenLed, OUTPUT);
  pinMode(redLed, OUTPUT);

  while (!Serial);

  Serial.println("LoRa Sender");

  if (!LoRa.begin(433E6)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
```

```
  lcd.begin(); // If you are using more I2C devices using the Wire library use lcd.begin(false)
  // this stop the library(LCD_I2C) from calling Wire.begin()
  lcd.backlight();

  pinMode(smoke, INPUT);
  pinMode(flame, INPUT);


}

void loop() {
  int VSmoke = analogRead(smoke);
  int VFlame = digitalRead(flame);
  byte temperature = 0;
  byte humidity = 0;
  value = analogRead(A1);
  voltageValue = value * (5.0 / 1024) * ((R1 + R2) / R2);
  Serial.print("Voltage =");
  Serial.println(voltageValue);

  dht11.read(&temperature, &humidity, NULL);
  Serial.print((int)temperature); Serial.print(" *C, ");
  Serial.print((int)humidity); Serial.println(" H");
  Serial.println(VSmoke);
  Serial.println(VFlame);

  // send packet
  LoRa.beginPacket();
  LoRa.print(str = ((String(",") + String(humidity) + String(",") + String(temperature) + String(",") + String(VSmoke) + String(",") + String(VFlame) + String(",")+ String(voltageValue)) + String(",")));
    LoRa.endPacket();
```

```
  lcd.setCursor(0, 0);
  lcd.print("Reserve Forest");
  lcd.setCursor(0, 1);
  lcd.print("Temperature: ");
  lcd.setCursor(13, 1);
  lcd.print(temperature);
  lcd.print("C");
  delay(1200);

  // put your main code here, to run repeatedly:
  if (VSmoke > 600 && VFlame == LOW) {

    lcd.setCursor(2, 0); lcd.print("Gas & Flame");
    lcd.setCursor(4, 1); lcd.print("DANGER FIRE");

    digitalWrite(greenLed, LOW);
    digitalWrite(redLed, HIGH);

    delay(12000);
  }
  else if (VSmoke > 600 && VFlame == HIGH) {

    lcd.setCursor(2, 0); lcd.print("   Gas");
    lcd.setCursor(4, 1); lcd.print("DANGER FIRE");

    digitalWrite(greenLed, LOW);
    digitalWrite(redLed, HIGH);

    delay(12000);

  }
```

```
  else if (VSmoke < 600 && VFlame == LOW) {
    lcd.setCursor(2, 0); lcd.print("   Flame");
    lcd.setCursor(4, 1); lcd.print("DANGER FIRE");

    digitalWrite(greenLed, LOW);
    digitalWrite(redLed, HIGH);

    delay(12000);
  }
  else if (VSmoke < 600 && VFlame == HIGH) {

    digitalWrite(greenLed, HIGH);
    digitalWrite(redLed, LOW);

  }

}
```

Figure 3.6.1: Arduino IDE (Lora transmitter Arduino).

3.6.2 Arduino IDE (Lora receiver Arduino).

Figure 3.6.2 show the coding from Arduino IDE (Lora receiver Arduino).

FOREST_FIRE_LORA_RECEIVER_ARDUINO | Arduino 1.8.14

File Edit Sketch Tools Help

FOREST_FIRE_LORA_RECEIVER_ARDUINO

```
#include <SPI.h>
#include <LoRa.h>

// or Software Serial on Uno, Nano...
#include <SoftwareSerial.h>
SoftwareSerial espSerial(4, 5); // RX, TX

String incoming = "";
String myString; // complete message from arduino, which consistors of snesors data
int Index1, Index2, Index3, Index4, Index5, Index6;
String firstValue, secondValue, thirdValue , fourValue, fiveValue;
String str;
int greenLed = 7;
int redLed = 6;
int buzzer = 3;

void setup() {
  Serial.begin(9600);
  espSerial.begin(9600);
  pinMode(greenLed, OUTPUT);
  pinMode(redLed, OUTPUT);
  pinMode(buzzer, OUTPUT);

  while (!Serial);

  Serial.println("LoRa Receiver");

  if (!LoRa.begin(433E6)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
}
```

```
void loop() {
  // try to parse packet
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    // received a packet

    // read packet
    if (LoRa.available() > 0) {
      incoming = (LoRa.readStringUntil('\n'));
      myString += incoming;
    }
    if (myString.length() > 0)
    {
      Index1 = myString.indexOf(',');
      Index2 = myString.indexOf(',', Index1 + 1);
      Index3 = myString.indexOf(',', Index2 + 1);
      Index4 = myString.indexOf(',', Index3 + 1);
      Index5 = myString.indexOf(',', Index4 + 1);
      Index6 = myString.indexOf(',', Index5 + 1);


      firstValue = myString.substring(Index1 + 1, Index2);
      secondValue = myString.substring(Index2 + 1, Index3);
      thirdValue = myString.substring(Index3 + 1, Index4);
      fourValue = myString.substring(Index4 + 1, Index5);
      fiveValue = myString.substring(Index5 + 1, Index6);
```

```
      Serial.print("  ");
      Serial.println(firstValue);//humidity
      Serial.print("  ");
      Serial.println(secondValue);//temperature
      Serial.print("  ");
      Serial.println(thirdValue);//smoke
      Serial.print("  ");
      Serial.println(fourValue);//flame
      Serial.print("  ");
      Serial.println(fiveValue);//voltage
      Serial.println("  ");

      str = ((String(",") + String(firstValue) + String(",") + String(secondValue) + String(",") + String(thirdValue) + String(",") + String(fourValue) + String(",") + String(fiveValue)) + String(","));
      espSerial.println(str);

      myString = "";
    }
```

```

  }
  if (thirdValue > "600" && fourValue == "0") {
    digitalWrite(greenLed, LOW);
    digitalWrite(redLed, HIGH);
    digitalWrite(buzzer, HIGH);
  }
  else if (thirdValue > "600" && fourValue == "1") {
    digitalWrite(greenLed, LOW);
    digitalWrite(redLed, HIGH);
    digitalWrite(buzzer, LOW);
  }
  else if (thirdValue < "600" && fourValue == "0") {
    digitalWrite(greenLed, LOW);
    digitalWrite(redLed, HIGH);
    digitalWrite(buzzer, HIGH);
  }
  else if (thirdValue < "600" && fourValue == "1") {
    digitalWrite(greenLed, HIGH);
    digitalWrite(redLed, LOW);
    digitalWrite(buzzer, LOW);
  }

}
```

Figure 3.6.2: Arduino IDE (Lora receiver Arduino).

3.6.3 Arduino IDE (Lora receiver Nodemcu).

Figure 3.6.3 show the coding from Arduino IDE (Lora receiver Nodemcu).

FOREST_FIRE_LORA_RECEIVER_NODEMCU | Arduino 1.8.14

File Edit Sketch Tools Help

FOREST_FIRE_LORA_RECEIVER_NODEMCU

```
/*************************************************************
  Blynk is a platform with iOS and Android apps to control
  ESP32, Arduino, Raspberry Pi and the likes over the Internet.
  You can easily build mobile and web interfaces for any
  projects by simply dragging and dropping widgets.

    Downloads, docs, tutorials: https://www.blynk.io
    Sketch generator:           https://examples.blynk.cc
    Blynk community:            https://community.blynk.cc
    Follow us:                  https://www.fb.com/blynkapp
                                https://twitter.com/blynk_app

  Blynk library is licensed under MIT license
  This example code is in public domain.

 *************************************************************
  This example runs directly on ESP8266 chip.

  NOTE: This requires ESP8266 support package:
    https://github.com/esp8266/Arduino

  Please be sure to select the right ESP8266 module
  in the Tools -> Board menu!

  Change WiFi ssid, pass, and Blynk auth token to run :)
  Feel free to apply it to any other example. It's simple!
 *************************************************************/

/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial
#include <SoftwareSerial.h>

```

```
/* Fill in information from Blynk Device Info here */
#define BLYNK_TEMPLATE_ID "TMPL6m31nGvOj"
#define BLYNK_TEMPLATE_NAME "forest fire"
#define BLYNK_AUTH_TOKEN "IpDA0VXk1fTp_5rkFW1TSS_gUt8x0sZc"


#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// Your WiFi credentials.
// Set password to "" for open networks.
//char ssid[] = "Yakyakyea";
//char pass[] = "0177377364";
char ssid[] = "MIH 2.4";
char pass[] = [redacted]";
//char ssid[] = "Ikhwanhzq";
//char pass[] = "haziq123";

SoftwareSerial mySerial(D5, D6);

String myString; // complete message from arduino, which consistors of snesors data
char rdata; // received charactors
int Index1, Index2, Index3, Index4, Index5, Index6, Index7;
String firstValue, secondValue, thirdValue , fourValue, fiveValue, sixValue;

void setup()
{
  // Debug console
  Serial.begin(9600);
  mySerial.begin(9600);
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
}

```

```
void loop()
{
  Blynk.run();
  Blynk.virtualWrite(V1, firstValue.toInt());
  Blynk.virtualWrite(V2, secondValue.toInt());
  Blynk.virtualWrite(V3, thirdValue.toInt());
  Blynk.virtualWrite(V4, fourValue.toInt());
  Blynk.virtualWrite(V5, fiveValue.toFloat());
  Blynk.virtualWrite(V6, sixValue.toFloat());

  while (mySerial.available() > 0)
  {
    delay(10);
    rdata = mySerial.read();
    myString += rdata;
  }
  if (myString.length() > 0)
  {
    Index1 = myString.indexOf(',');
    Index2 = myString.indexOf(',', Index1 + 1);
    Index3 = myString.indexOf(',', Index2 + 1);
    Index4 = myString.indexOf(',', Index3 + 1);
    Index5 = myString.indexOf(',', Index4 + 1);
    Index6 = myString.indexOf(',', Index5 + 1);
    Index7 = myString.indexOf(',', Index6 + 1);

    firstValue = myString.substring(Index1 + 1, Index2);
    secondValue = myString.substring(Index2 + 1, Index3);
    thirdValue = myString.substring(Index3 + 1, Index4);
    fourValue = myString.substring(Index4 + 1, Index5);
    fiveValue = myString.substring(Index5 + 1, Index6);
    sixValue = myString.substring(Index6 + 1, Index7);
```

```

    Serial.print(firstValue);
    Serial.print("  ");
    Serial.print(secondValue);
    Serial.print("  ");
    Serial.println(thirdValue);
    Serial.print("  ");
    Serial.println(fourValue);
    Serial.print("  ");
    Serial.println(fiveValue);
    Serial.print("  ");
    Serial.println(sixValue);



    myString = "";
  }
  //delay(1000);
}
```

Figure 3.6.3: Arduino IDE (Lora receiver Nodemcu).

3.6.4 Proteus 8 Professional

Proteus 8 Professional Software can be used to model, simulate, and draw electronic circuits, as shown in Figure 3.6.4. Using Proteus, we can also create two-dimensional circuit designs. With just a home computer and this engineering application, we may construct and simulate a wide range of electrical and electronic circuits. Moreover, planning circuits on Proteus takes less time than creating them. The probability of deformities, for example, free associations, which consume a large chunk of the day to recognize in an actual circuit, is limited with programming re-enactment. Figure 3.6.4 shows the Proteus 8 Professional Software.

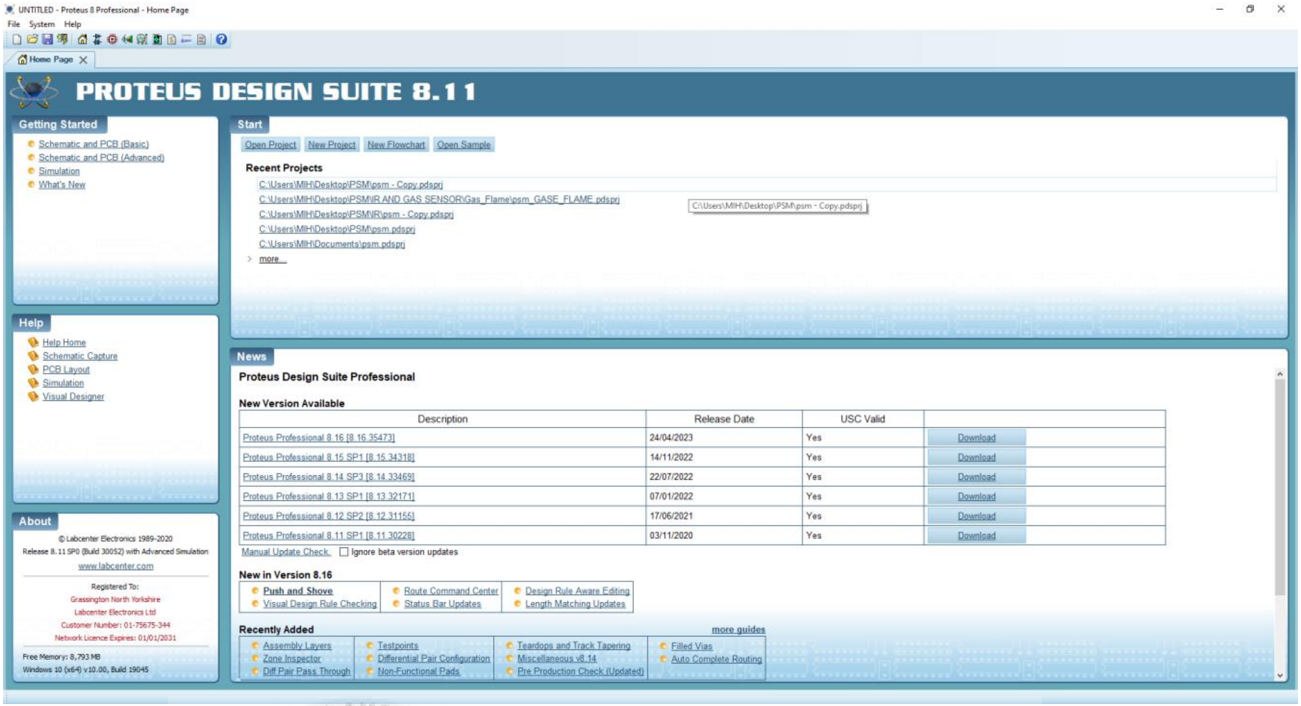


Figure 3.6.4: Proteus 8 Professional Software

3.7 Project Implementation

The project is to develop a solar-powered Internet of Things (IoT) forest fire warning system using Lora Arduino, Blynk software, and NodeMCU. To detect fires, the system makes use of sensors such the DHT-11 temperature and humidity sensor, the IR sensor, and the MQ2 gas sensor. The project begins with thorough planning and research, followed by hardware configuration and sensor integration. The NodeMCU and the Blynk app establish a wireless connection and begin collecting data continuously. Cloud connectivity enables data storage and analysis, and data analysis algorithms can anticipate future fire hazards. An alert system is put in place for prompt notification. Power management strategies are implemented using solar panels and batteries. The system undergoes extensive testing and is deployed in a controlled setting. the creation of a trustworthy and efficient system for identifying and averting forest fires as opposed to wildfires.

3.7.1 Flowchart of The System (Lora transmitter Arduino).

Figure 3.7.1 shows the flowchart of the system (Lora transmitter Arduino).

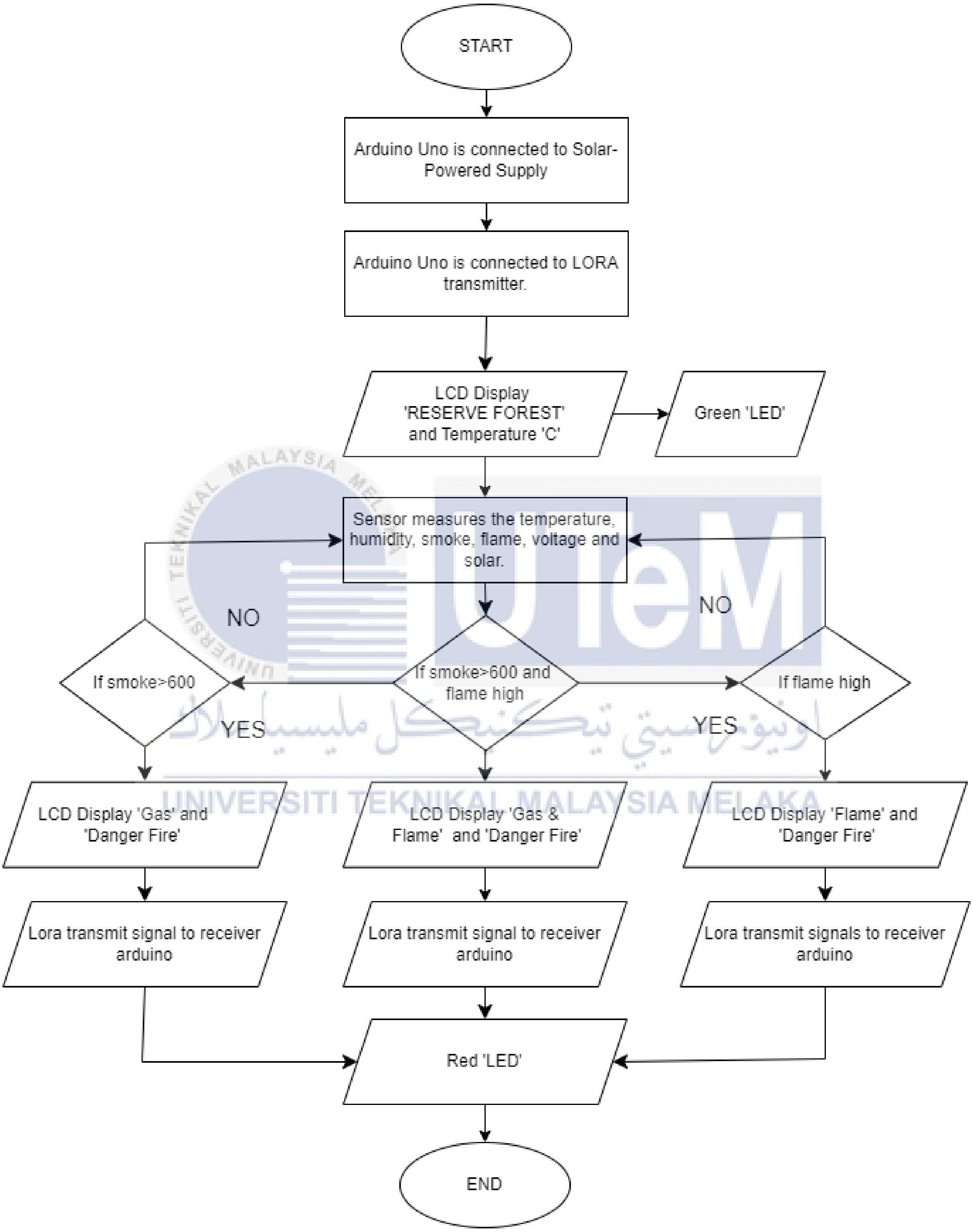


Figure 3.7.1: Flowchart of The System (Lora transmitter Arduino)

3.7.2 Flowchart of The System (Lora IoT receiver Arduino).

Figure 3.7.2 shows the flowchart of the system works (Lora IoT receiver Arduino)

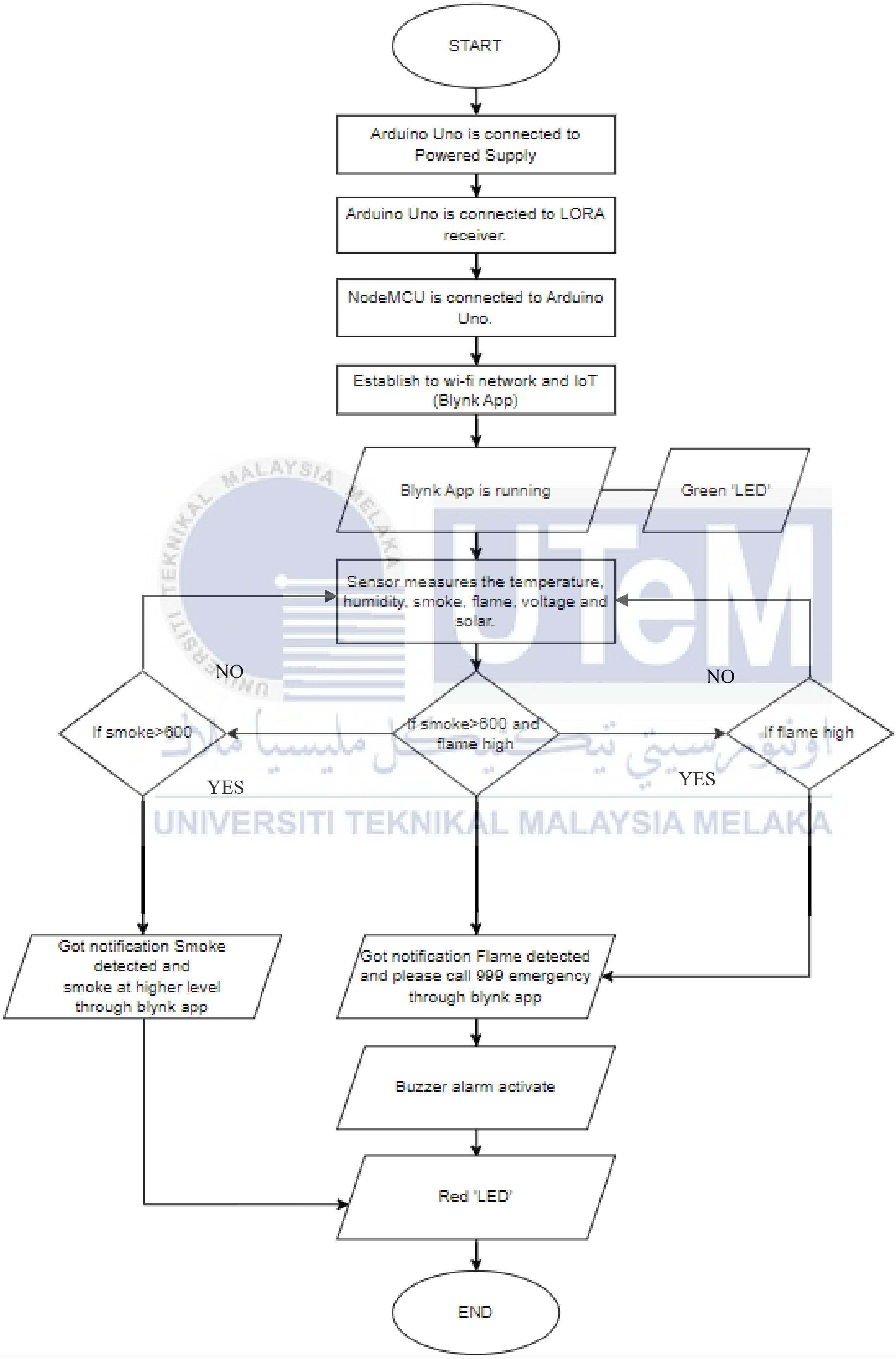


Figure 3.7.2: Flowchart of The System (Lora IoT receiver Arduino)

3.8 Summary

Eventually, the Development of a Lora IOT Based Solar Powered Forest Fire Detection Using Arduino can be developed if all the hardware and software are properly connected and satisfy the requirements. Errors in the project will be fixed in order to accomplish the goals. In Chapter 4, A detailed description of the methods for collecting and analysing data will be provided.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter outlines the outcomes of the project that was carried out. This chapter is critical in demonstrating whether all the project's objectives, as specified in Chapter 1, have been met.

4.2 Schematic design by fritzing

The Fritzing schematic design for a LoRa IoT-based solar-powered forest fire detection system using Arduino shows the visual representation of important components like the LoRa module (SX1278), Arduino Uno, temperature/humidity sensors, smoke and flame sensors, solar panels, rechargeable batteries, LEDs, optional NodeMCU for Blynk integration, and additional parts for power regulation and switches. Different wire colours are used for power, ground, and signal connections, and connections between components are clearly labelled. The schematic helps with documentation and future development by providing a visual representation of the electronic layout of the system.

4.2.1 Fritzing design (Lora transmitter Arduino).

Figure 4.2.1 shows the Schematic design (Lora transmitter Arduino).

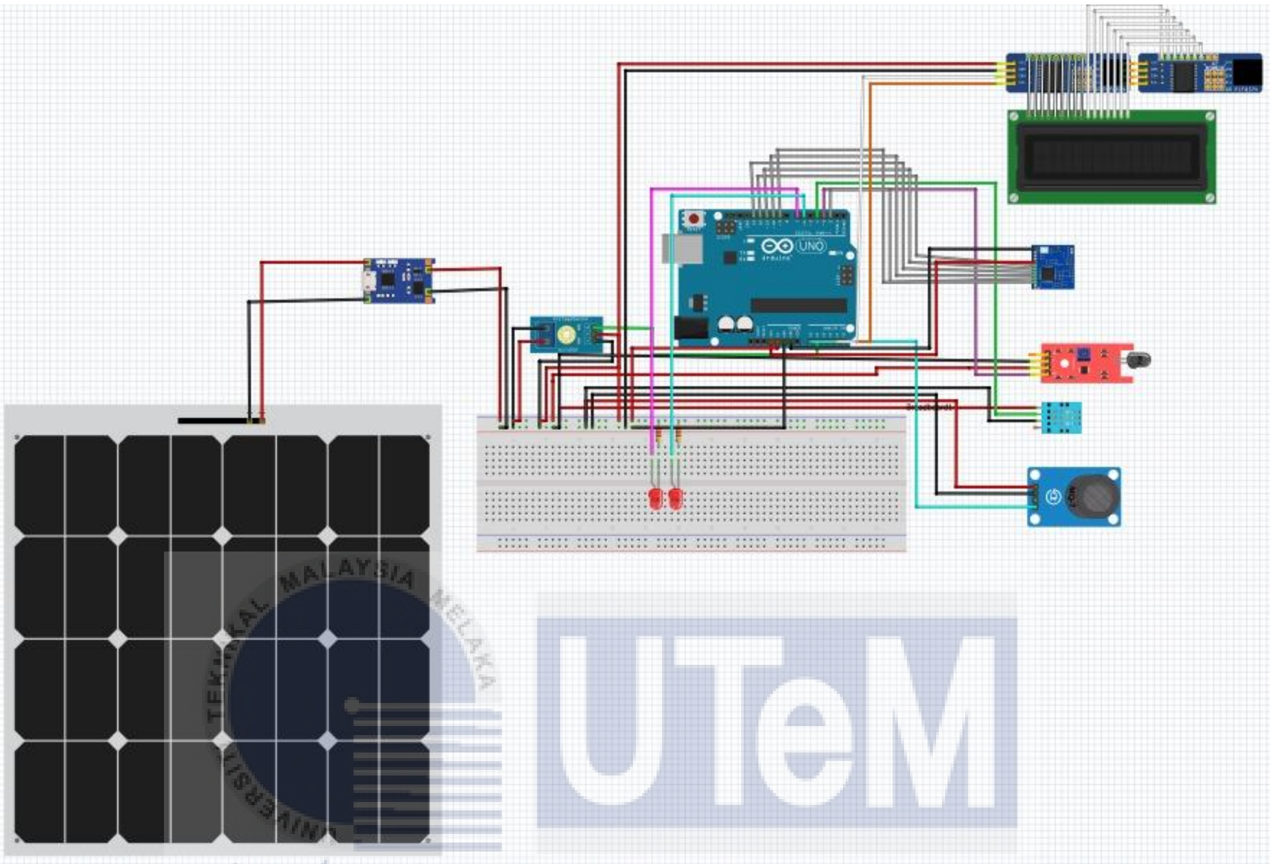


Figure 4.2.1: Fritzing design (Lora IoT receiver Arduino)

4.2.2 Fritzing design (Lora IoT receiver Arduino).

Figure 4.2.2 shows the Fritzing design (Lora IoT receiver Arduino).

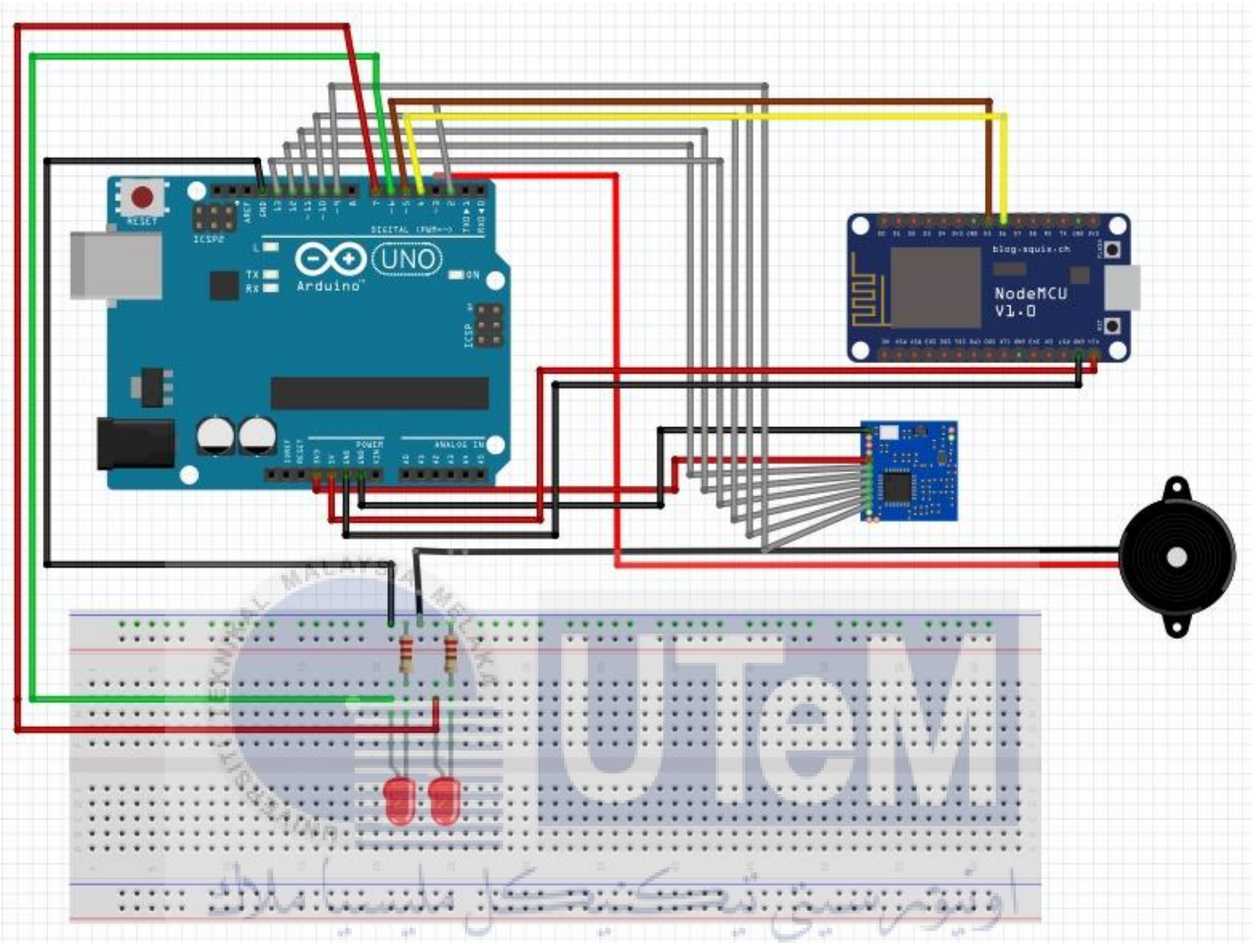


Figure 4.2.2: Fritzing design (Lora IoT receiver Arduino)

4.3 RESULTS

4.3.1 Proteus 8 circuit.

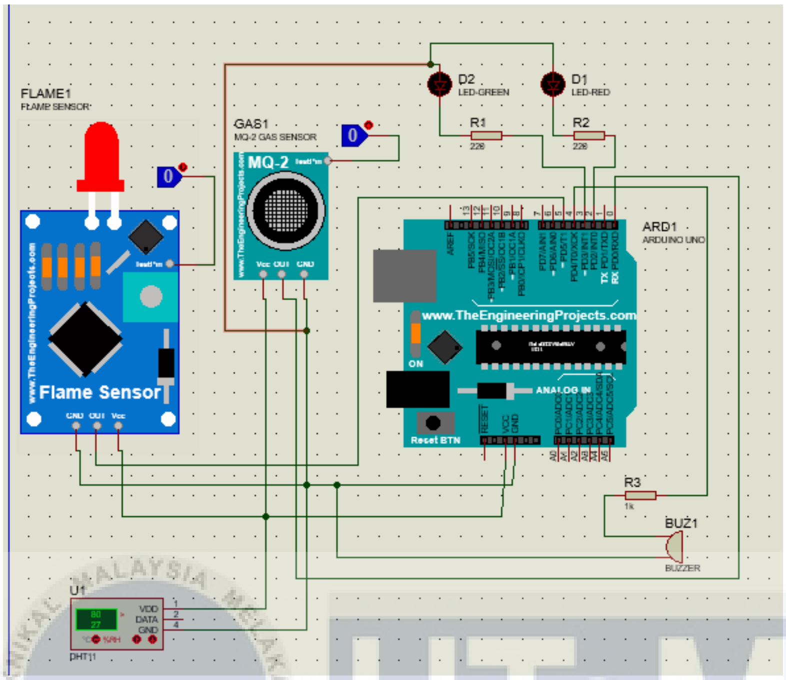


Figure 4.2: Reserved Forest Fire Sensor

The figure 4.2 shows the running circuit of reserved forest fire sensors. I could see the connection between all the devices and all the connected devices in ready state.

4.3.2 Structure of Results

Figure 4.3.2 shows the example of structure of results.

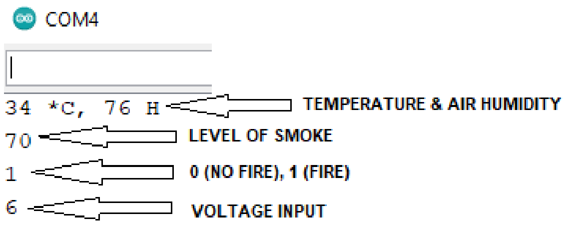


Figure 4.3.2: Example of Structure of Results

4.4 Normal situation

Figure 4.4.1 shows the output of normal situation

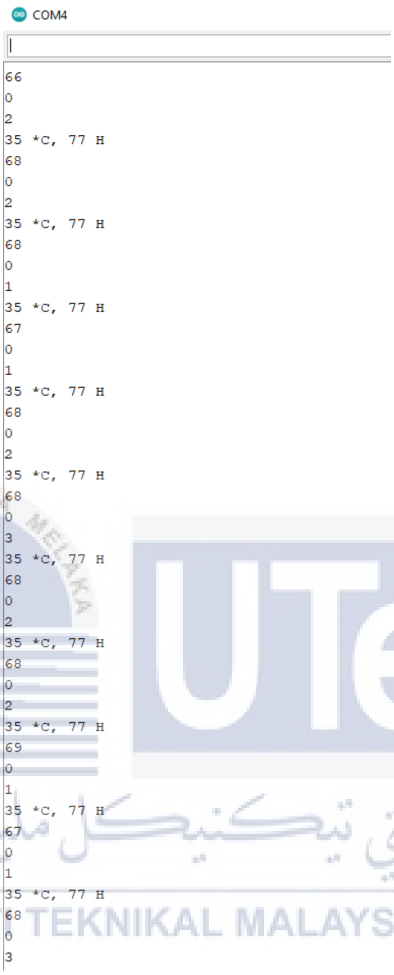


Figure 4.4.1: Arduino Output of Normal Situation

Figure 4.4.2 shows the hardware of normal situation works.

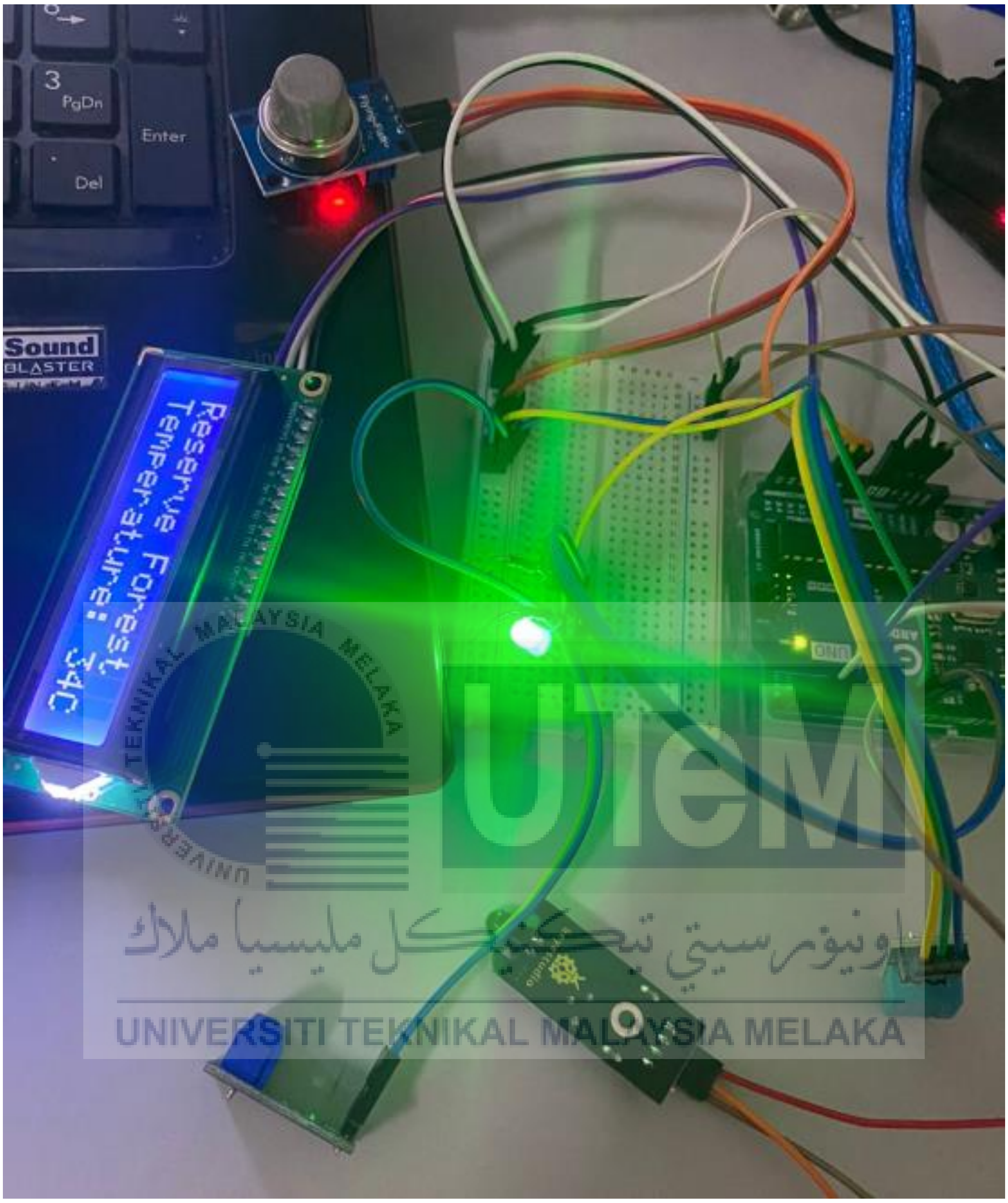


Figure 4.4.2: Hardware of Normal Situation

4.5 Smoke without Fire

Figure 4.5.1 shows the output of the situation Smoke without Fire.

```
COM4

35 *C, 73 H
101
0
5
35 *C, 73 H
101
0
5
35 *C, 73 H
100
0
6
35 *C, 73 H
101
0
6
35 *C, 73 H
99
0
5
35 *C, 73 H
98
0
5
35 *C, 73 H
99
0
4
35 *C, 73 H
96
0
5
35 *C, 73 H
95
0
5
35 *C, 73 H
94
0
```

Figure 4.5.1: Arduino Output of Smoke without Fire

Figure 4.5.2 shows hardware of the situation Smoke without Fire work.

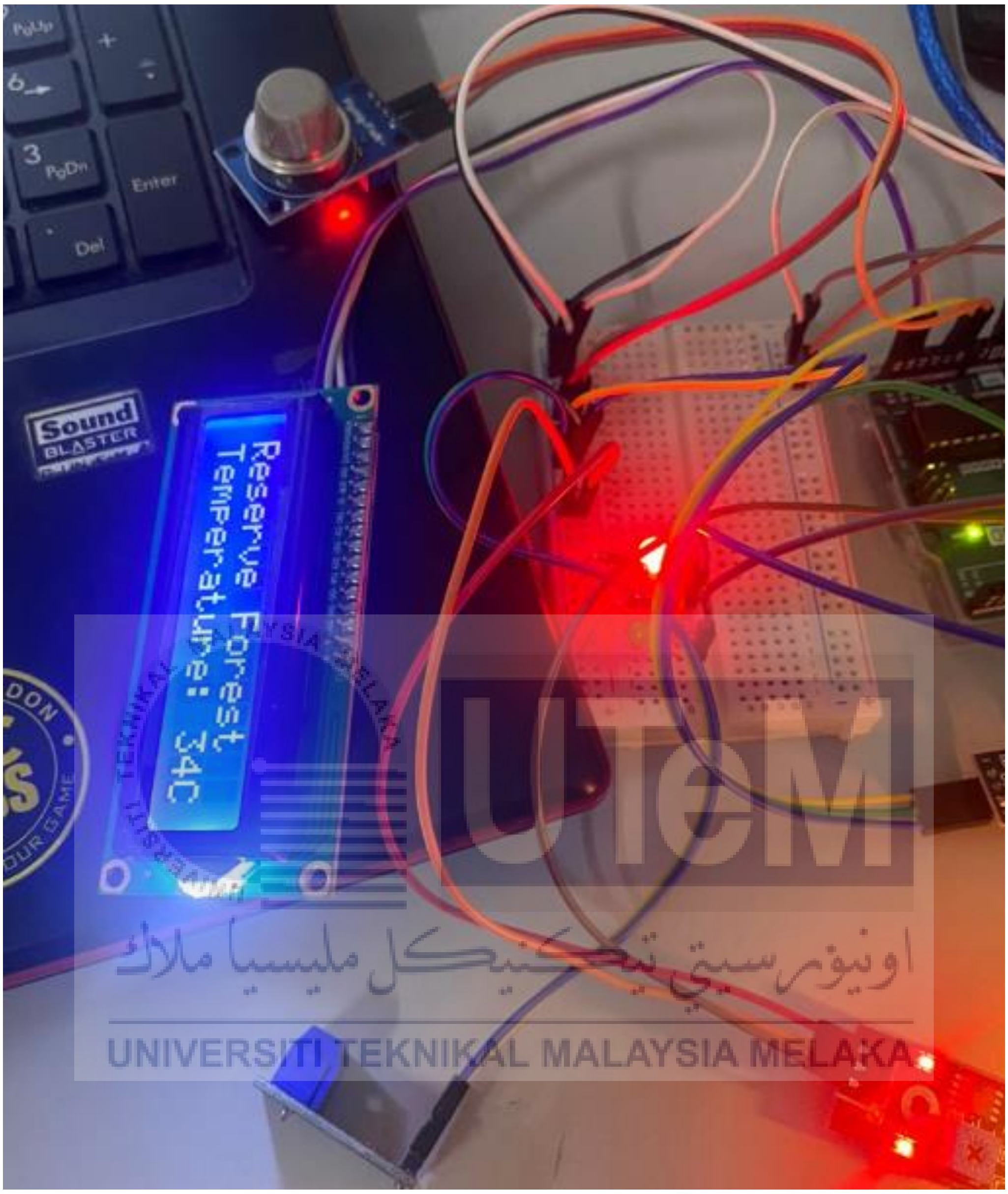


Figure 4.5.2: Hardware of Smoke without Fire

4.6 Fire without smoke

Figure 4.6.1 shows the output situation of Fire without Smoke.

```
COM4

34 *C, 76 H
70
1
6
34 *C, 76 H
70
1
5
34 *C, 76 H
69
1
7
34 *C, 76 H
69
1
6
34 *C, 76 H
70
1
7
34 *C, 77 H
70
1
5
34 *C, 77 H
70
1
6
34 *C, 77 H
69
1
7
34 *C, 77 H
70
1
7
```

Figure 4.6.1: Arduino Output of Fire without Smoke

Figure 4.6.2 shows hardware of the situation Fire without Smoke works.

Figure 4.6.2: Hardware of Fire Without Smoke

4.7 Fire within Smoke

Figure 4.7.1 shows the output situation of Fire within Smoke.

```
COM4

35 *C, 73 H
101
1
5
35 *C, 73 H
101
1
5
35 *C, 73 H
100
1
6
35 *C, 73 H
101
1
6
35 *C, 73 H
99
1
5
35 *C, 73 H
98
1
5
35 *C, 73 H
99
1
4
35 *C, 73 H
96
1
5
35 *C, 73 H
95
1
5
```

Figure 4.7.1: Arduino Output of Fire within Smoke

Figure 4.7.2 shows hardware of the situation Fire within Smoke.

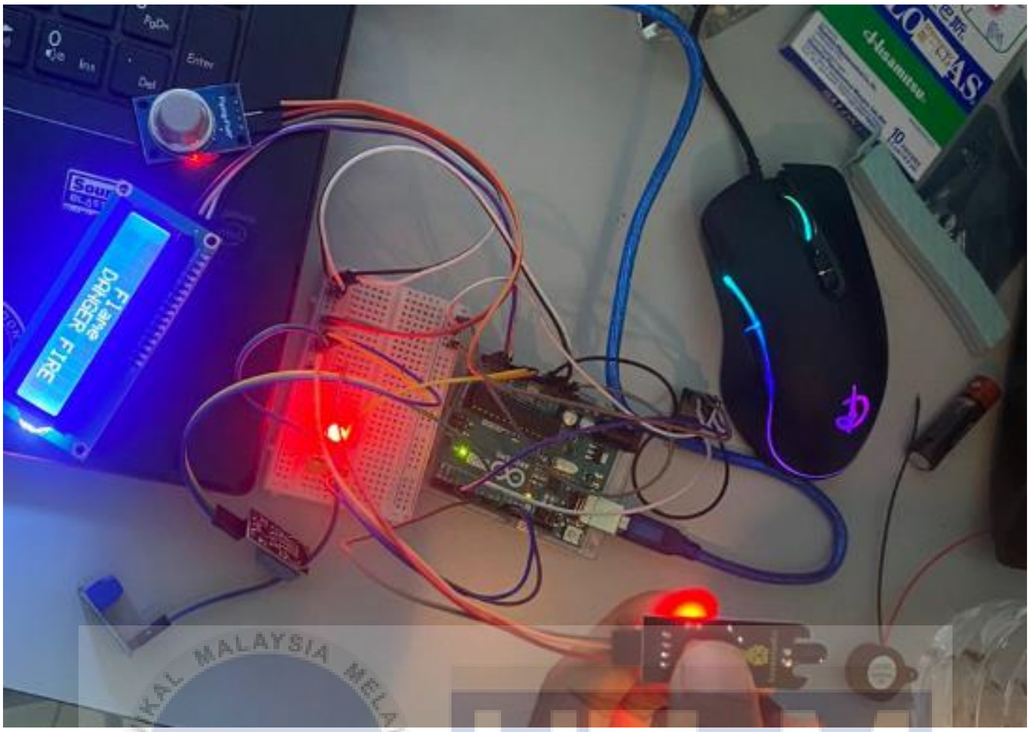


Figure 4.7.2: Hardware of Fire within Smoke

Table 4.8 provides results that illustrate how the system behaves in various smoke and fire detection scenarios. When everything is normal, the system is keeping sensing. A visible indicator of the high smoke level is provided by the red LED when smoke is detected in the absence of fire. The system provides alerts regarding both the high smoke level and the danger of fire when there is a fire present, whether or not there is smoke. It also turns on the buzzer, and activates a red LED.

Table 4.3: Situation of Results

SITUATION	Smoke Level	Flame Sensor	LED	Buzzer	Blynk app notifications
Normal Situation	Smoke < 80	Low	Green	No	No notification
Smoke without Fire	Smoke > 80	Low	Red	No	‘Smoke detected’ & ‘Smoke at higher level’
Fire without Smoke	Smoke < 80	High	Red	Yes	‘Flame detected’ & ‘Please call 999 emergency'
Fire with Smoke	Smoke > 80	High	Red	Yes	‘Flame detected’ & ‘Please call 999 emergency' and ‘Smoke detected’ & ‘Smoke at higher level’

4.8 Results and Analysis

The results that managed to get the distance of sending lora signals from lora transmitter Arduino to lora IoT receiver Arduino same as preliminary results but changed with the value of smoke level as following to the flowchart diagram.

4.8.1 Situation far from in a room 30 meters fire with smoke

Figure 4.8.1 shows hardware of the situation far from in a room 30 meters fire with smoke.


7:21
TELEGRAM
Mama
FaceTime Video
Speaker
Camera
Mute
Share
End

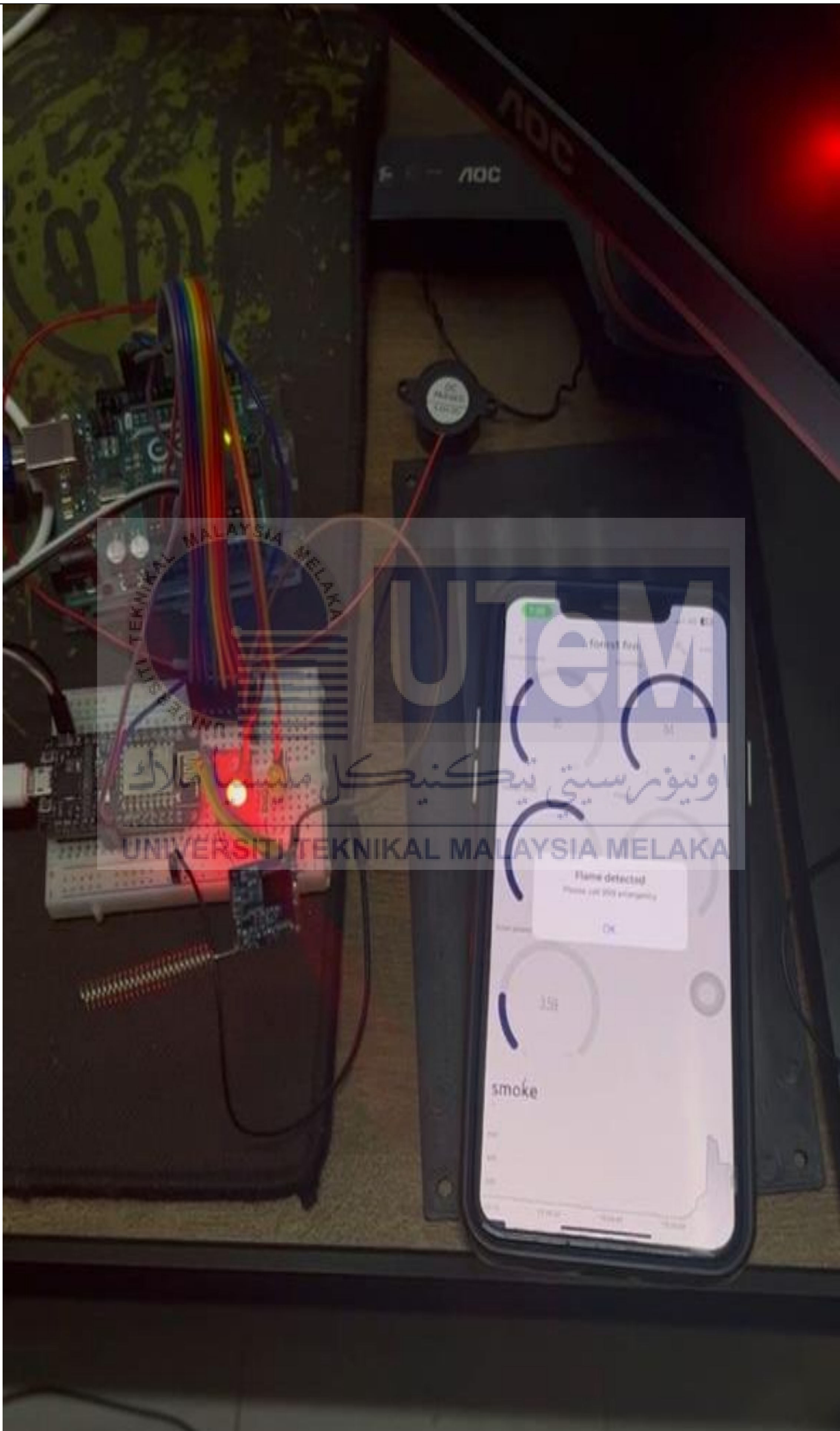
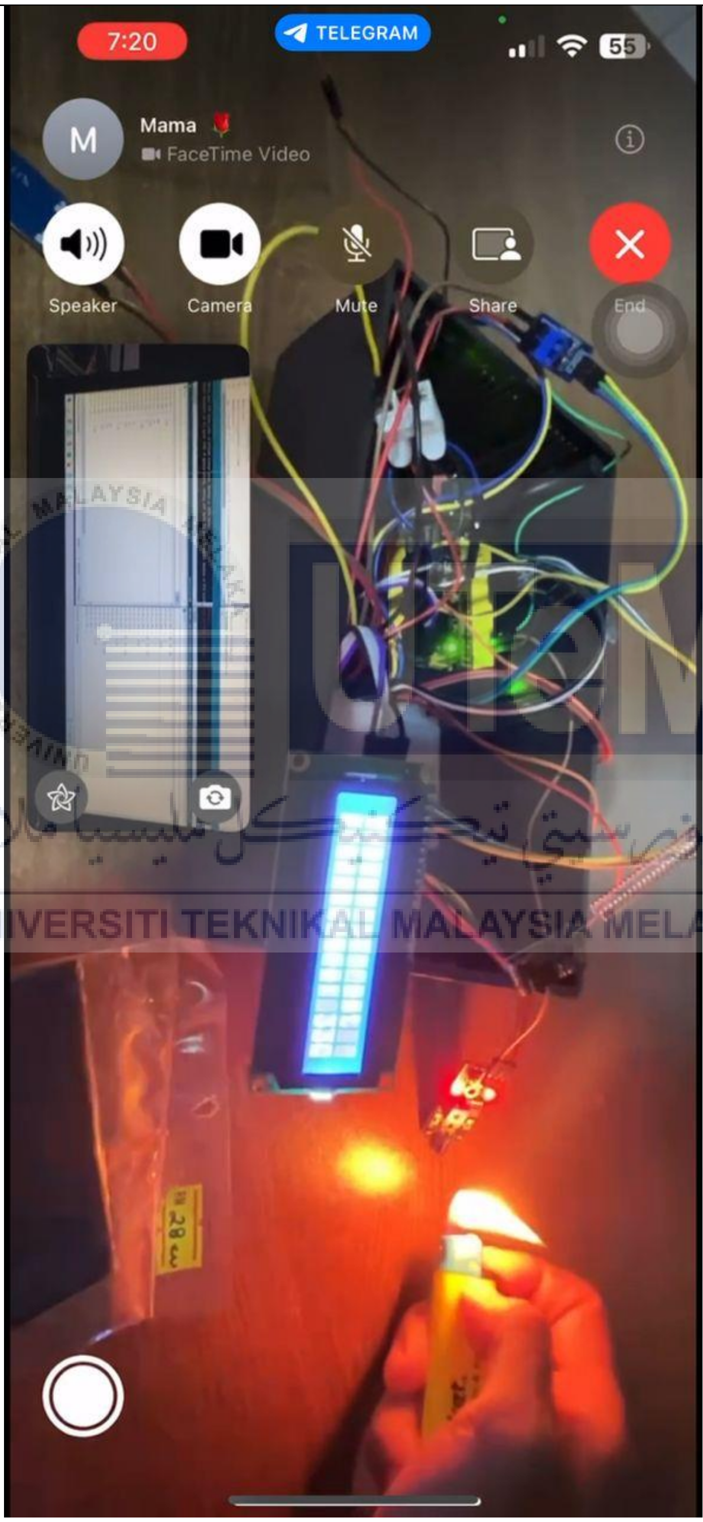

AOC
Flame detected
OK
smoke

Figure 4.8.1: hardware of the situation far from in a room 30 meters fire with smoke.

4.8.2 Situation far from in a room 30 meters fire without smoke

Figure 4.8.2 shows hardware of the situation far from in a room 30 meters fire without smoke.




TELEGRAM

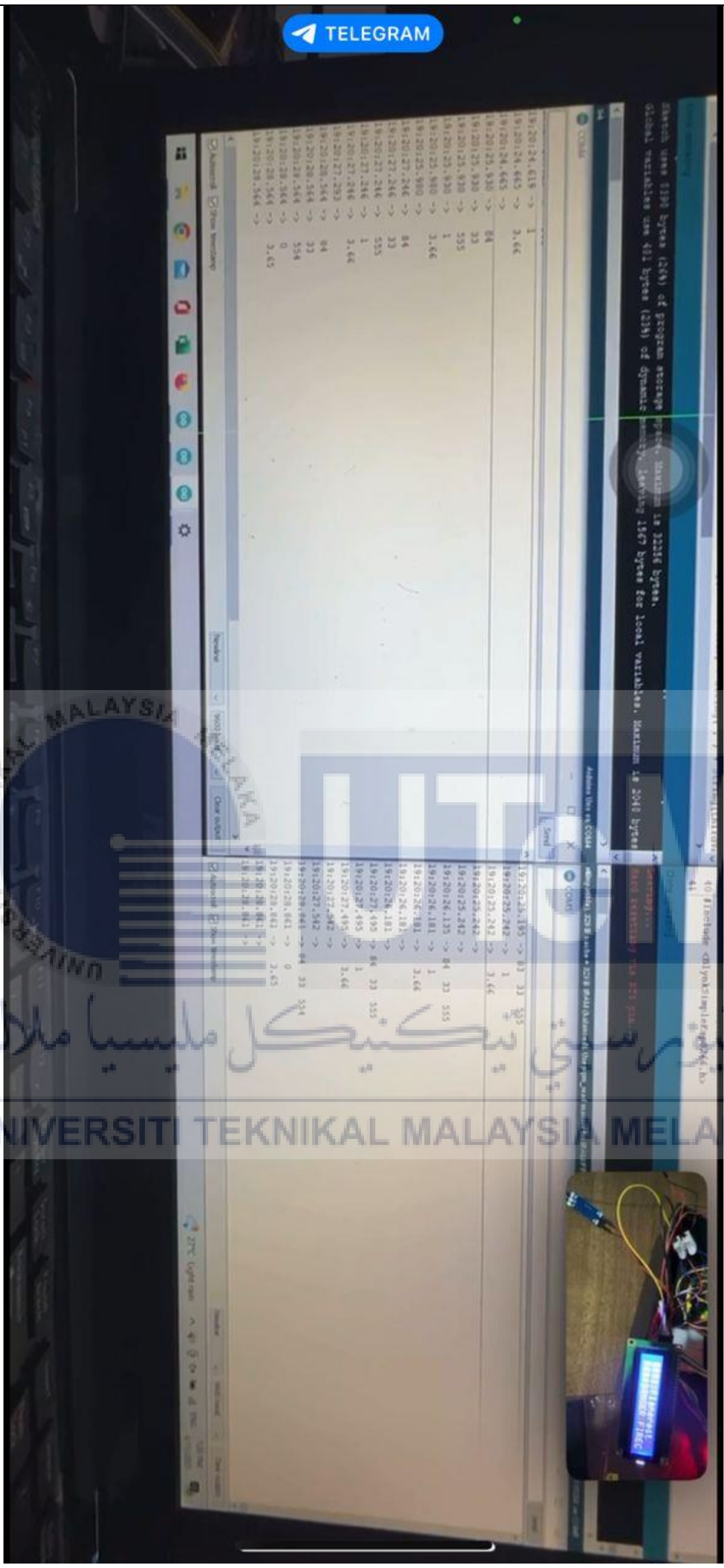

TELEGRAM

Figure 4.8.2: hardware of the situation far from in a room 30 meters fire without smoke.

4.8.3 Situation far from in a room 30 meters smoke without fire.

Figure 4.8.3 shows hardware of the situation far from in a room 30 meters smoke without fire

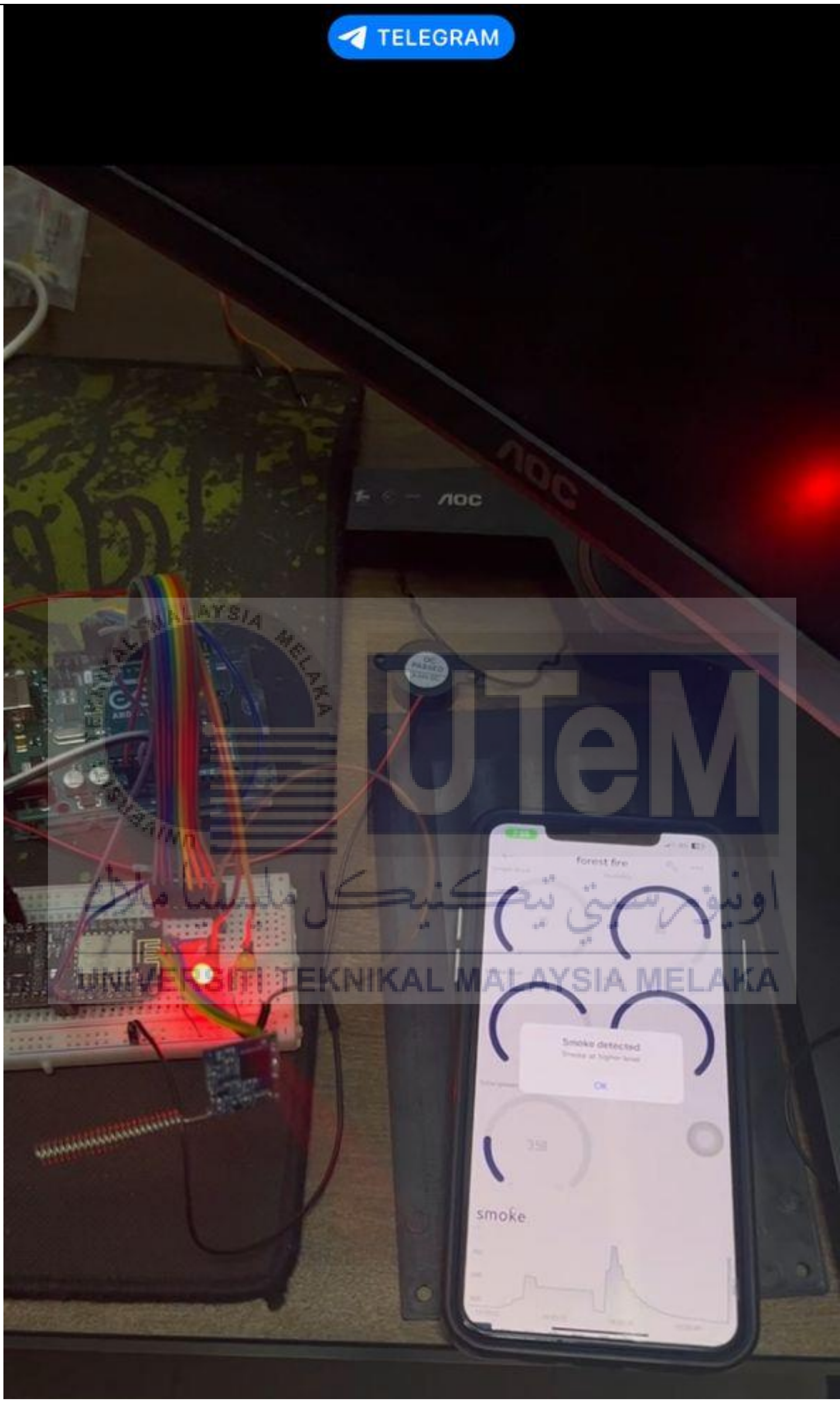


Figure 4.8.3: hardware of the situation far from in a room 30 meters smoke without fire.

Table 4.8 Situation of the distance Lora-Transmitter Arduino analysis provide This innovative forest fire detection system operates within a 40-meter range, employing a multi-sensory approach to assess the environment. When the smoke level is low (< 600), a green LED signals a calm 'Reserved Forest,' while a high smoke level prompts a red LED, triggering LoRa transmission for emergency alerts like 'Gas' or 'Danger Fire.' This dynamic response mechanism persists across different distances, underlining the system's adaptability and efficacy in proactively addressing potential forest fire threats.

Table 4.8 Situation of the distance Lora-Transmitter Arduino analysis

Distance	**Smoke Level**	**Flame Sensor**	**LED**	**Transmit to Lora Receiver**	**Lcd Display**
10 meter	Smoke < 600	Low	Green	Yes	‘Reserved Forest’ ‘Temperature: C’
10 meter	Smoke > 600	Low	Red	Yes	‘Gas’ ‘Danger Fire’
10 meter	Smoke < 600	High	Red	Yes	‘Fire’ ‘Danger Fire’
10 meter	Smoke > 600	High	Red	Yes	‘Gas & Fire’ ‘Danger Fire’
20 meter	Smoke < 600	Low	Green	Yes	‘Reserved Forest’ ‘Temperature: C’
20 meter	Smoke > 600	Low	Red	Yes	‘Gas’ ‘Danger Fire’

20 meter	Smoke < 600	High	Red	Yes	‘Fire’ ‘Danger Fire’
20 meter	Smoke > 600	High	Red	Yes	‘Gas & Fire’ ‘Danger Fire’
30 meter	Smoke < 600	Low	Green	Yes	‘Reserved Forest’ ‘Temperature: C’
30 meter	Smoke > 600	Low	Red	Yes	‘Gas’ ‘Danger Fire’
30 meter	Smoke < 600	High	Red	Yes	‘Fire’ ‘Danger Fire’
30 meter	Smoke > 600	High	Red	Yes	‘Gas & Fire’ ‘Danger Fire’
40 meter	Smoke < 600	Low	Green	Yes	‘Reserved Forest’ ‘Temperature: C’
40 meter	Smoke > 600	Low	Red	Yes	‘Gas’ ‘Danger Fire’
40 meter	Smoke < 600	High	Red	Yes	‘Fire’ ‘Danger Fire’
40 meter	Smoke > 600	High	Red	Yes	‘Gas & Fire’ ‘Danger Fire’

4.8.4 The graph of Distance Lora-Transmitter Arduino analysis.

Figure 4.8.4 shows the graphical representation provides a quick and clear understanding of how the forest fire detection system on distance of Lora transmitter arduino.

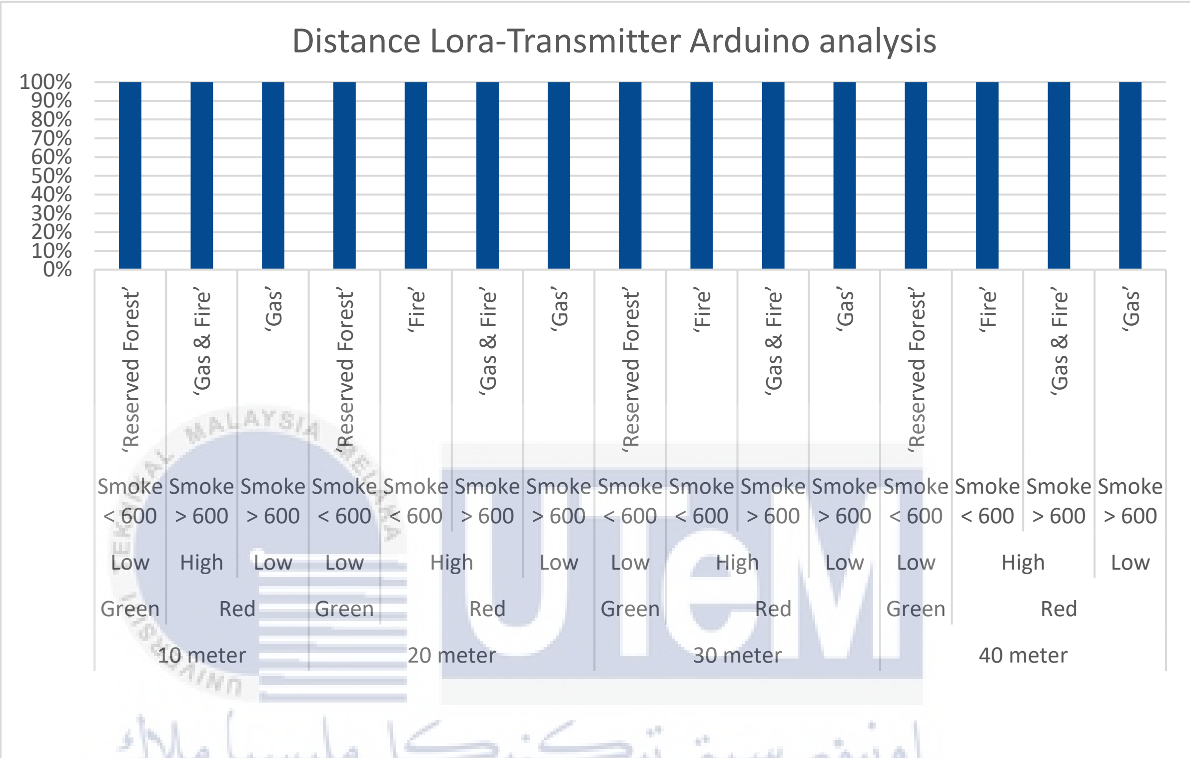


Figure 4.8.4: The graph of Distance Lora-Transmitter Arduino analysis

The situational analysis in Table 4.9 illustrates the LoRa-IoT Receiver Arduino's responsive behaviour across diverse distances in the context of forest fire detection. Operating at 10, 20, 30, and 40 meters, the system adeptly distinguishes between varying smoke levels, activates LEDs and the Buzzer based on Flame Sensor readings, and interfaces seamlessly with the Blynk app for nuanced notifications. At 10 meters, for instance, the system employs green and red LEDs to denote low and high smoke levels, respectively, with the Flame Sensor and Buzzer collaborating to identify flames. Blynk app notifications provide detailed alerts such as 'Smoke detected,' 'Smoke at higher level,' and urgent prompts to 'Please call 999 emergency' when flames are detected. This adaptability remains consistent across distances, exemplifying the system's robustness in conveying timely and precise information through both hardware and software interfaces, thereby enhancing its effectiveness in forest fire detection and response.

Table 4.9 Situation of the distance Lora-IoT Receiver Arduino analysis

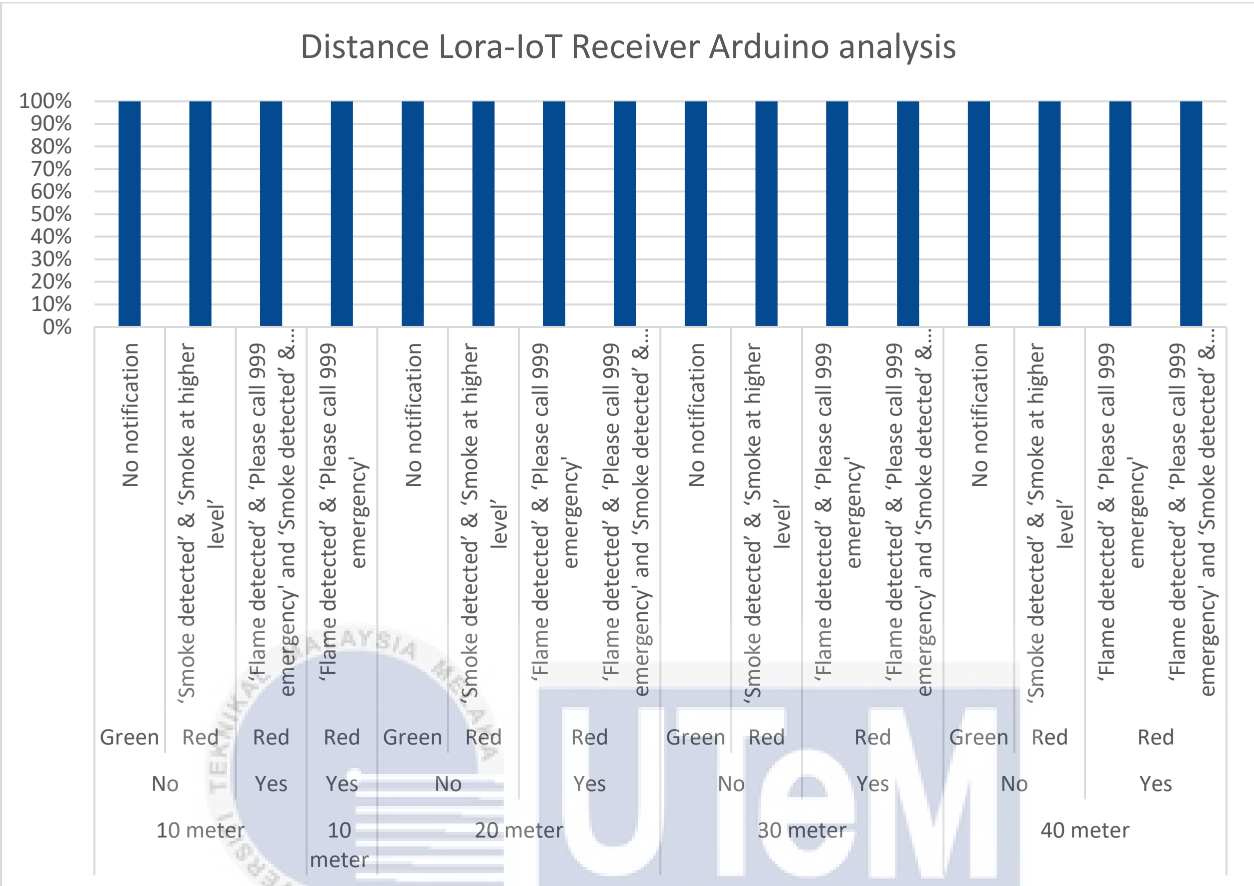
Distance	**Smoke Level**	**Flame Sensor**	**LED**	**Buzzer**	**Receive signals from Lora Transmitter**	**Blynk app notifications**
10 meter	Smoke < 600	Low	Green	No	Yes	No notification
10 meter	Smoke > 600	Low	Red	No	Yes	'Smoke detected' & 'Smoke at higher level'
10 meter	Smoke < 600	High	Red	Yes	Yes	'Flame detected' & 'Please call 999 emergency'
10 meter	Smoke > 600	High	Red	Yes	Yes	'Flame detected' & 'Please call 999 emergency' and 'Smoke detected' & 'Smoke at higher level'

20 meter	Smoke < 600	Low	Green	No	Yes	No notification
20 meter	Smoke > 600	Low	Red	No	Yes	'Smoke detected' & 'Smoke at higher level'
20 meter	Smoke < 600	High	Red	Yes	Yes	'Flame detected' & 'Please call 999 emergency'
20 meter	Smoke > 600	High	Red	Yes	Yes	'Flame detected' & 'Please call 999 emergency' and 'Smoke detected' & 'Smoke at higher level'
30 meter	Smoke < 600	Low	Green	No	Yes	No notification
30 meter	Smoke > 600	Low	Red	No	Yes	'Smoke detected' & 'Smoke at higher level'
30 meter	Smoke < 600	High	Red	Yes	Yes	'Flame detected' & 'Please call 999 emergency'
30 meter	Smoke > 600	High	Red	Yes	Yes	'Flame detected' & 'Please call 999 emergency' and

						‘Smoke detected’ & ‘Smoke at higher level’
40 meter	Smoke < 600	Low	Green	No	Yes	No notification
40 meter	Smoke > 600	Low	Red	No	Yes	‘Smoke detected’ & ‘Smoke at higher level’
40 meter	Smoke < 600	High	Red	Yes	Yes	‘Flame detected’ & ‘Please call 999 emergency'
40 meter	Smoke > 600	High	Red	Yes	Yes	‘Flame detected’ & ‘Please call 999 emergency' and ‘Smoke detected’ & ‘Smoke at higher level’

4.9.1 The graph of distance Lora-IoT Receiver Arduino analysis.

Figure 4.9.1 shows the forest fire detection system's responses at different distances (10m, 20m, 30m, 40m). It showcases variations in Buzzer activation, LED color, and Blynk app notifications. Notably, each distance category reflects diverse scenarios, highlighting the dynamic nature of the system's reactions to potential fire incidents. The consistent 'Count of Distance' for each scenario simplifies the representation, providing a concise overview of the forest fire detection system's behavior.



4.9.1: The graph of distance Lora-IoT Receiver Arduino analysis.

CHAPTER 5

CONCLUSION

5.1 Introduction

The LoRa-IoT based solar-powered forest fire detection system successfully utilizes two Arduino-based microcontrollers to create a robust and efficient solution. The transmitter node incorporates various sensors, including MQ2 for smoke detection, a flame sensor, DHT11 for temperature and humidity, and a voltage sensor for solar power monitoring. The LoRa SX1278 module facilitates long-range communication, and an LCD provides real-time information. On the receiver side, a NodeMCU (ESP8266) is employed, connected to the LoRa receiver. The system features LEDs and a buzzer for immediate local alerting, while the Blynk app integration provides remote monitoring and notifications. The use of solar power ensures sustainability and autonomy in remote forest areas. This project contributes to early forest fire detection, minimizing potential damage. The combination of LoRa for long-range communication, IoT for real-time monitoring, and solar power for sustainability makes this system a reliable and practical solution for environmental protection.

5.2 Recommendation

<u>Combining Algorithms for Machine Learning</u>

Consider incorporating machine learning algorithms into the forest fire detection system to improve its prediction capabilities. These algorithms could examine past information gathered by the sensors and enable the system to forecast future fire outbreaks by looking at patterns and trends. Machine learning models could help to more accurate and proactive fire detection, allowing for timely preventive measures and lowering false alarms.

Enhanced Energy Efficiency through Advanced Solar Technologies

Explore advanced solar technologies to optimize energy efficiency. This includes investigating more efficient solar panels, energy storage solutions, and power management algorithms. Improving the system's energy efficiency ensures sustained and reliable operation, especially in remote forest locations with varying sunlight conditions. Additionally, incorporating energy-saving techniques could extend the system's autonomy, reducing the need for frequent maintenance.

Implementation of Cloud Connectivity for Data Analytics

Integrate cloud connectivity to enable real-time data analytics and remote monitoring. By connecting the LoRa-IoT system to a cloud platform, data collected from the sensors can be analyzed comprehensively. This allows for a more in-depth understanding of environmental conditions, early detection of anomalies, and the possibility of implementing machine learning models on a larger scale. Cloud connectivity also facilitates remote system management and software updates, enhancing the system's scalability and adaptability.

REFERENCES:

[1] V. R. Karumanchi, S. H. Raju, S. Kavitha, V. Lakshmi Lalitha, and S. Vijaya Krishna, "Fully Smart fire detection and prevention in the authorized forests," in *Proceedings - International Conference on Artificial Intelligence and Smart Systems, ICAIS 2021*, Institute of Electrical and Electronics Engineers Inc., Mar. 2021, pp. 573–579. doi: 10.1109/ICAIS50930.2021.9395969.

[2] P. M. Jacob, J. Moni, R. B. Robins, M. E. Varghese, S. S. Babu, and V. K. Bose, "An Intelligent Fire Detection and Extinguishing Assistant System Using Internet of Things (IoT)," in *2022 International Conference on Decision Aid Sciences and Applications, DASA 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1057–1061. doi: 10.1109/DASA54658.2022.9765126.

[3] S. K. Bhoi *et al.*, "FireDS-IoT: A Fire Detection System for Smart Home Based on IoT Data Analytics," in *Proceedings - 2018 International Conference on Information Technology, ICIT 2018*, Institute of Electrical and Electronics Engineers Inc., Dec. 2018, pp. 161–165. doi: 10.1109/ICIT.2018.00042.

[4] J. Selle J, A. V. Krishna, M. S. Harish, and K. B. Reddy, "An IoT based Alert System with Gas Sensors in a WSN Framework for Evasion of Forest Fire," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering, ICACITE 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1516–1521. doi: 10.1109/ICACITE53722.2022.9823897.

[5] V. Kumari and F. A. Ali, "Early Detection of Forest Fire Using Internet of Things," in *Proceedings of 2022 1st IEEE International Conference on Industrial Electronics: Developments and Applications, ICIDeA 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 24–28. doi: 10.1109/ICIDeA53933.2022.9970130.

[6] G. Pettorru, M. Fadda, R. Girau, M. Sole, M. Anedda, and D. Giusto, "Using Artificial Intelligence and IoT Solution for Forest Fire Prevention," in *2023 International Conference on Computing, Networking and Communications, ICNC 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 414–418. doi: 10.1109/ICNC57223.2023.10074289.

[7] S. F. Hui, H. F. Ho, W. W. Chan, K. W. Chan, W. C. Lo, and K. W. E. Cheng, "Floating Solar Cell Power Generation, Power Flow Design and its Connection and Distribution."

[8] *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE, 2019.

[9] P. Raghavendra Reddy, P. Kalyanasundaram, and V. Suresh, "IOT based Novel speedy Detection of Forest fire using Sensors with improved accuracy by sensing Temperature and Atmospheric Carbon Dioxide Level using Node Microcontroller Unit in comparison with Arduino Microcontroller," in *14th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics, MACS 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/MACS56771.2022.10022408.

[10] *Power Generation System and Renewable Energy Technologies (PGSRET), 2015*. IEEE.

[11] *Power Generation System and Renewable Energy Technologies (PGSRET), 2015*. IEEE.

[12] K. Mehta, S. Sharma, and D. Mishra, "Internet-of-Things Enabled Forest Fire Detection System," in *Proceedings of the 5th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), I-SMAC 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 20–23. doi: 10.1109/I-SMAC52330.2021.9640900.

[13] Universitas Gadjah Mada. Department of Electrical Engineering and Information Technology, Institute of Electrical and Electronics Engineers. Indonesia Section, Institute of Electrical and Electronics Engineers, I. National Conference on Information Technology and Electrical Engineering (2014 : Yogyakarta, I. Regional Conference on Computer and Information Engineering (2014 : Yogyakarta, and I. Conference on Chief Information Officer (2014 : Yogyakarta, *Proceedings, 2014 6th International Conference on Information Technology and Electrical Engineering : "Leveraging Research and Technology through University-Industry Collaboration" : 7-8 October 2014, Eastparc Hotel, Yogyakarta, Indonesia*.

[14] Universitas Gadjah Mada. Department of Electrical Engineering and Information Technology, Institute of Electrical and Electronics Engineers. Indonesia Section, Institute of Electrical and Electronics Engineers, I. National Conference on Information Technology and Electrical Engineering (2014 : Yogyakarta, I. Regional Conference on Computer and Information Engineering (2014 : Yogyakarta, and I. Conference on Chief Information Officer (2014 : Yogyakarta, *Proceedings, 2014 6th International Conference on Information Technology and Electrical Engineering : "Leveraging Research and Technology through University-Industry Collaboration" : 7-8 October 2014, Eastparc Hotel, Yogyakarta, Indonesia*.

[15] R. G. do N. IEEE Symposium on Computers and Communications 2018 Natal *et al.*, *2018 IEEE Symposium on Computers and Communications (ISCC) 25-28 June 2018*.

[16] International Conference on Sciences and Techniques of Automatic Control and Computer Engineering 17. 2016 Sousse, IEEE Tunisia Section, International

Conference on Sciences and Techniques of Automatic Control and Computer Engineering 17 2016.12.19-21 Sousse, International Conference STA 17 2016.12.19-21 Sousse, and STA 17 2016.12.19-21 Sousse, *2016 17th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering, STA December 19-21, 2016, Sousse, Tunisia : proceedings*.

[17] Institute of Electrical and Electronics Engineers Indonesia Section, International Conference of Information and Communication Technology 3 2015.05.27-29 Nusa Dua, and ICoICT 3 2015.05.27-29 Nusa Dua, *2015 3rd International Conference of Information and Communication Technology (ICoICT) 27-29 May 2015, Nusa Dua*.

[18] S. Ishitha, S. Nagaraju, H. A. Mohan, M. Harshitha, G. R. Gowda, and N. Jeevan, "IoT based Anti-poaching and Fire Alarm System for Forest," in *2021 IEEE Mysore Sub Section International Conference, MysuruCon 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 711–715. doi: 10.1109/MysuruCon52639.2021.9641539.

[19] *2018 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT)*. IEEE, 2018.

[20] IEEE India International Conference on Power Electronics 7. 2016 Patiala, Institute of Electrical and Electronics Engineers, Thapar University Patiala, IEEE India International Conference on Power Electronics 7 2016.11.17-19 Patiala, India International Conference on Power Electronics 7 2016.11.17-19 Patiala, and IICPE 7 2016.11.17-19 Patiala, *7th IEEE India International Conference on Power Electronics IICPE 2016 : November 17-19, 2016*.

[21] Manipal University Jaipur. School of Computing and Information Technology and Institute of Electrical and Electronics Engineers, *2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT) : Manipal University, Jaipur, Sep. 28-29, 2019.*

[22] Institute of Electrical and Electronics Engineers., Institute of Electrical and Electronics Engineers. Bangladesh Section., and Bangladesh University of Engineering and Technology. Department of Electrical and Electronic Engineering., *Proceedings of ICECE 2012 : venue, Pan Pacific Sonargaon Dhaka, Dhaka-1215, Bangladesh, 20-22 December 2012*. IEEE, 2012.

[23] Institute of Electrical and Electronics Engineers. Region 6, Bay Area Council (Calif.), and Institute of Electrical and Electronics Engineers, *Proceedings of the Third IEEE Global Humanitarian Technology Conference : GHTC 2013 : October 20-23, 2013, San Jose/Silicon Valley, California USA*.

[24] Institute of Electrical and Electronics Engineers, *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*.

APPENDICES

PROJECT ACTIVITIES	STATUS	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14	WEEK 15	WEEK 16
BDP Briefing	E									M							S
	A																
Meeting with Supervisor	E									I							T
	A																
Distribution of project titles	E									D							U
	A																
PSM 1 Rubrics Explanation	E									B							D
	A																
Project planning	E									R							Y
	A																
Proposal preparation	E									E							
	A																
Abstract	E									A							W
	A																
Literature Review	E									K							E
	A																
Design Project	E																E
	A																
Flowchart	E									S							K
	A									E							
Construct the project	E									M							
	A																

Appendix A: Project Gantt Chart Bachelor Degree Project 1

E – Estimated, A – Actual-All project activities that have been estimated are in line with the actual timeline plan.

PROJECT ACTIVITIES	STATUS	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14	WEEK 15	WEEK 16
Draft Material List	E									M							S
	A																
Meeting with Supervisor	E									I							T
	A																
Test Hardware	E									D							U
	A																
Analyse Result	E									B							D
	A																
Complete Chapter 4:Result and Discussion	E									R							Y
	A																
Complete Chapter 5:Conclusion	E									E							
	A																
Submit draft report	E									A							W
	A																
Prepare Project Poster	E									K							E
	A																
Preparation for presen-tation	E																E
	A																
Presentation	E									S							K
	A									E							
Submit Final Report	E									M							
	A																

Appendix B: Project Gantt Chart Bachelor Degree Project 2

APPENDICES