# DEVELOPMENT OF AN IOT-BASED FIRE FIGHTING ROBOT BY USING MICROCONTROLLER

RINOSHEN A/L M.GOPAL



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

# DEVELOPMENT OF AN IOT-BASED FIRE FIGHTING ROBOT BY USING MICROCONTROLLER


**RINOSHEN A/L M.GOPAL**


**This report is submitted in partial fulfilment of the requirements for the degree of Bachelor of Electronics Engineering Technology (Industrial Electronics) with Honours**


**Faculty of Electronics and Computer Technology and Engineering**
**Universiti Teknikal Malaysia Melaka**


**2025**

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI DAN KEJURUTERAAN ELEKTRONIK DAN KOMPUTER

**BORANG PENGESAHAN STATUS LAPORAN**
**PROJEK SARJANA MUDA II**

Tajuk Projek : Development of an Iot-based Fire Fighting Robot by using Microcontroller

Sesi Pengajian : 2024/2025

Saya RINOSHEN A/L M.GOPAL 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:

______________________ ______________________

( RINOSHEN A/L M.GOPAL) (COP DAN TANDATANGAN PENYELIA)

Alamat Tetap: ......................
......................
......................
......................
......................

Tarikh : 20 Januari 2025 Tarikh : 20 Januari 2025

*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 an IoT-based Fire Fighting Robot by Using Microcontroller" 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 | : | RINOSHEN A/L M.GOPAL |
| Date | : | 20 JAN 2025 |

# 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 Electronics Engineering Technology (Industrial Electronics) with Honours

Signature : ........................................................................

Supervisor Name : **TS.TG MOHD FAISAL BIN TENGKU WOOK**

Date : ........................................................................

Signature : ........................................................................

Co-Supervisor : ........................................................................

Name (if any)

Date : ........................................................................

# DEDICATION

*This thesis is dedicated to the people who have assisted me through my studie. Thank you*

*For making me see this journey through to the end*

## ABSTRACT

The proposed firefighting prototype robot will use a flame sensor and automatically detect the flame by itself. It also uses a gas sensor to find LPG (Liquefied petroleum gas) in the surrounding place. The prototype has an ultrasonic sensor to avoid obstacles and move forward. The robot will have a water tank that is attached to a water pump and a servo motor that controls the nozzle so it can extinguish the fire with the liquid in the tank. Through the Telegram app, will get notification if LPG (Liquefied petroleum gas) is detected .If have any emergency situation can make emergency stop or run by sending commands at telegram. The prototype will monitor real-time data, obstacle avoidance. One of the main problem statements is to make an autonomous firefighting robot prototype that can detect fires, get through obstacles, and act quickly. Maybe in some situations, like when there is a limited room or dangerous surroundings, traditional firefighting methods might not work because it will be too dangerous for a human to enter. This project aims to provide a solution to improve emergency response in places that are prone to fire by combining sensors, microcontrollers, and the Telegram app.

# *ABSTRAK*

Robot prototaip pemadam kebakaran yang dicadangkan akan menggunakan penderia nyalaan dan secara automatik mengesan nyalaan dengan sendirinya. Ia juga menggunakan sensor gas untuk mencari LPG (Liquefied petroleum gas) di kawasan sekitar. Prototaip mempunyai sensor ultrasonik untuk mengelakkan halangan dan bergerak ke hadapan. Robot itu akan mempunyai tangki air yang dipasang pada pam air dan motor servo yang mengawal muncung supaya ia boleh memadamkan api dengan cecair di dalam tangki. Melalui aplikasi Telegram, akan mendapat pemberitahuan jika LPG (Gas petroleum cecair) dikesan. Jika ada sebarang situasi kecemasan boleh berhenti atau jalankan kecemasan dengan menghantar komen di telegram. Prototaip akan memantau data masa nyata, mengelakkan halangan. Salah satu penyataan masalah utama ialah membuat prototaip robot pemadam kebakaran autonomi yang boleh mengesan kebakaran, melepasi halangan dan bertindak dengan cepat. Mungkin dalam beberapa situasi, seperti apabila terdapat bilik terhad atau persekitaran berbahaya, kaedah memadam kebakaran tradisional mungkin tidak berfungsi kerana terlalu berbahaya untuk dimasuki manusia. Projek ini bertujuan untuk menyediakan penyelesaian untuk meningkatkan tindak balas kecemasan di tempat yang terdedah kepada kebakaran dengan menggabungkan penderia, mikropengawal dan aplikasi Telegram.

## ACKNOWLEDGEMENTS

First and foremost, I would like to extend my heartfelt gratitude to my supervisor, TS.TG Mohd Faisal Bin Wook, for his invaluable mentorship, knowledge, and unwavering assistance during this research. His valuable feedback and perceptive guidance have been crucial to the successful completion of this thesis.

Not to mention, my appreciation to my classmates Sashvini A/P Maran and Sharveen A/L Kalieswaran for helping me through out the project. Their support and collaboration were essential for me to carry out my project successfully. I am deeply indebted to Universiti Teknikal Malaysia Melaka (UTeM) and my parents for the financial support through the research grant that allowed me to accomplish this project.

I want to express my gratitude to my parents and friends for their unconditional love, support and prayers always.

**TABLE OF CONTENTS**

# LIST OF TABLES

## LIST OF FIGURES

# LIST OF SYMBOLS

| | | |
|---|---|---|
| $\delta$ | - | Voltage angle |
| % | - | Percentage |
| ° | - | Degrees |
| °C | - | Degrees Celsius |
| Ω | - | Ohms |

## LIST OF ABBREVIATIONS

| | | |
|---|---|---|
| $V$ | - | Voltage |
| cm | - | Centimetres |
| s | - | Seconds |
| A | - | Amperes |
| kHz | - | Kilohertz |
| ml | - | Millilitres |
| ppm | - | Parts per million |
| kg cm | - | Kilogram centimetres |
| ms | - | Milliseconds |
| IoT | - | Internet of Things |
| ESP32 | - | Espressif Systems Microcontroller |
| LPG | - | Liquefied Petroleum Gas |
| PWM | - | Pulse Width Modulation |
| DC | - | Direct Current |
| UART | - | Universal Asynchronous Receiver/Transmitter |
| IDE | - | Integrated Development Environment |
| ADC | - | Analog to Digital Converter |
| DAC | - | Digital to Analog Converter |
| BLE | - | Bluetooth Low Energy |
| GND | - | Ground |
| Vcc | - | Voltage Common Collector |
| IR | - | Infrared |
| UV | - | Ultraviolet |
| Wi-Fi | - | Wireless Fidelity |
| GSM | - | Global System for Mobile Communications |
| LED | - | Light Emitting Diode |
| AI | - | Artificial Intelligence |
| ML | - | Machine Learning |
| PSM | - | Projek Sarjana Muda |
| PCB | - | Printed Circuit Board |

## LIST OF APPENDICES

# CHAPTER 1

# INTRODUCTION

## 1.1 Background

According to the Department of Statistics Malaysia, there has been a decline in the number of accidents in 2021. A total of 21,534 incidents were reported, which is 34.1% lower than the previous year. Additionally, there were 301 fatal injuries [13]. The subsequent research underscores the significance of cultivating novel concepts such as autonomous rescue robots, capable of functioning in perilous locations that are typically inaccessible or too hazardous for humans, hence augmenting the likelihood of survival. Using this kind of advanced technology can lower the chance of death, injuries, and property damage caused by fire accidents. This is one of the crucial reasons firefighting robots are essential to Malaysia's emergency response system to avoid significant losses.



## 1.2 Societal/Global Issue Through Fire Fighting Robot Project

Global fire problems are a primary worldwide concern since they pose substantial risks to individuals, local communities, and the environment. Developmentnge, urban development, industrial operations, and limited resources are contributing factors that heighten the risk of fires. Climatic change has resulted explicitly in conditions that are more favourable for wildfires, leading to the destruction of ecosystems and the worsening of climatic problems. The growth of residential areas is aided by urban sprawl, and using risky chemicals in factories raises concerns about fires that could lead to accidents.

In 2020, fire departments in 48 countries responded to about 70 million calls, with 4 million being about fires. As mentioned in reference [16], the events caused 20,700 deaths and nearly 70,000 injuries. These numbers show that we need better ways to handle fire situations immediately.

More and more people realise autonomous rescue robots are the best way to solve these problems. These robots can be operated in dangerous conditions that human firefighters cannot enter, such as areas with toxic gases or limited space, so this is one of the main reasons we need a firefighting robot. They are equipped with features for live monitoring of flexible movement and may be controlled from a distance. This enables them to address fire emergencies proactively, perhaps rescuing lives and avoiding economic losses.

Gettingg money on tools like firefighting robots and fire-resistant buildings and getting the community ready is recommended for better fire safety. A world plan is being made to make it safer and better to put out fires. This plan uses the Internet of Things (IoT) to create a firefighting robot to find fires, spot dangerous gases, and get around barriers. This fits with the idea that people worldwide must develop new ways to deal with the growing problems that fire causes and make them less harmful to communities.

## 1.3 Problem Statement

Today, there are more fires, often in dangerous places. It is very important to make an automated robot that can put out fires. These robots need to be able to find fires quickly, get around them, and put them out. Most of the time, the normal ways of extinguishing fires don't work in these tough scenarios. This shows how important it is to find new ways to fight fires that might change how often they happen. The goal of this project is to make a system that combines different kinds of sensors, microcontrollers, and the Telegram app so that it is easy to handle from a distance. With the help of new technologies, the project will combine old ways of fighting fires with the needs of today's fire situations. The most important part of this work is making a firefighting robot with cutting-edge sensors that can find flames, sense LPG gases, and keep an eye on the surroundings. The robot and its operators being able to talk to each other makes people more aware of their surroundings, helps them make quick choices, and makes sure that everyone works together in an emergency. By the end of this project, we hope to have a full answer that improves how we handle emergencies and lowers the risks that come with fires. The project's goal is to create a cutting-edge firefighter robot with a good design so that the prototype robot can be useful in an emergency. With the right technology and a well-thought-out structure, the test robot can work well in an emergency. The main goals of this project are to keep people safe, keep property secure, and make it easy to extinguish fires.

## 1.4 Project Objective

a) To improve fire safety measures and minimize human exposure to dangerous conditions.

b) To integrate and calibrate multiple sensors to perceive its environment for detection purposes.

c) To detect flames within indoor spaces, then target the flame by spraying liquid towards the fire with minimal water waste.

## 1.5 Scope of Project

The scope of this project are as follows:

a) Design and Integration of Hardware Components

b) Development of Software Algorithms

c) Creation of the Telgram App InterfacE

d) Testing and Validation

e) Comprehensive Documentation

f) Advancing Firefighting Technology

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 Introduction

This literature study looks at the current state of firefighting robotics, focusing on essential parts like sensor technologies, navigation algorithms, communication systems, safety protocols, and real-world applications. Our goal is to find patterns, gaps, and possible ways to make more progress in this critical area by looking at newly published books, case studies, and technological advances. Firefighting robots are essential for improving emergency reactions, lowering human risk, and increasing firefighting efficiency. This is true whether used in cities, factories, or disaster zones. This look at how changes have been made to firefighting robotics shows how we are constantly working to make firefighting safer and more effective.

## 2.2 Understanding Global/Current Issue in the building fire

Building fires are a big problem worldwide because they hurt people, property, and the environment very badly. Fire protection and control problems have worsened as cities have grown and building designs have become more complicated. Many people living in cities together in close quarters make tall buildings dangerous because fires can spread quickly, and getting people out of the building is hard. Many hazardous materials that make fires more likely are kept in industrial buildings. This is because climate change has made extreme weather events happen more often, which has led to more fires. Even though traditional ways of fighting fires work, they are usually not enough in complex and dangerous situations. As a result, there is a growing need for cutting-edge options like robots that fight fires. These robots can move around in hazardous places, get to areas that are hard to get to and send real-time info that can help firefighters do their jobs better. The growth of these robots shows promise for reducing the damage caused by building fires, which could save lives and keep costs down.

Fire-fighting robots are a great way to help solve this global problem because they are more efficient, put people at less risk, and allow faster response times, so they can lower the risk. Additionally, they can work nonstop in hazardous conditions, which makes firefighter efforts more effective overall.

## 2.3 History and Evaluation

Despite being around for a long time, firefighting robots are still being developed and improved [1]. Initially, they were mostly ruled from a distance and couldn't think for themselves [1]. Explosions from fires are a significant threat around the world and can kill a lot of people [1]. Urban areas often have trouble controlling fires because they can't do as much. The robot mainly works with an Android app on a smartphone and talks to other robots using Wi-Fi [1]. The system uses motion detection to find and identify items and obstacles. For example, an Arduino microcontroller and Internet of Things (IoT) technology allow the robot to send emergency warnings quickly, explore, and find fires accurately. Even if a sensor fails, this device can work independently and put out fires using its built-in camera because it is part of a sensor network [1]. Tests have been done on the robot to ensure it can work independently and send live video [1]. More and more fires are happening, which poses severe risks to people, property, and the environment [2]. Every year, about 358,500 domestic fires occur in the US, killing 2,620 civilians and damaging homes worth about $12 billion [2]. Even though traditional fire detection systems are essential, they have trouble putting out flames once they are found. To fix this problem, a new fire detector that uses the Internet of Things (IoT) technology has been created [2]. This device can find fires on its own, identify what is on fire, suggest the best fire extinguisher, and quickly send mobile alerts to users and rescue workers in seconds [2]. Developing an intelligent fire detection device that works with the Internet of Things (IoT) is a big step forward in fire detection technology. Fire detection systems have set off alarms in the past, but they could only detect smoke [3]. For example, if the system finds a grease fire in a kitchen, it tells the person in charge to use a Class K extinguisher [3]. Acting quickly is necessary to stop the fire from spreading as soon as possible and ensure that help arrives on time [3]. When evaluating current firefighting robots, things like speed, agility, and skills [4] are considered to see how well they work in different settings. The review focuses on how faster, more accurate, and better integrated these systems are with innovative technologies than older ones [7]. This writing tells how firefighting technology has changed, focusing on the change from manual methods to automatic systems [8]. This text talks about the history of firefighting robots, showing how they've changed from simple versions to more advanced ones [8]. The evaluation looks at how well and safely current firefighting robots work, pointing out significant improvements in safety, efficiency, and response time compared to old ways of doing things [8]. The review looks at how healthy night vision cameras improve robots' abilities to do their jobs, especially in security and surveillance settings [10].

### 2.3.1 Safety and Reliability

Dependability and safety go hand in hand. It's hard for fireman robots to do their jobs in places with high temperatures, thick smoke, and buildings that aren't stable [2]. It's very important to make sure they last. They mustn't lose their usefulness when they get hot [2]. The technology cuts down on reaction time and gives exact instructions on which fire extinguishers to use, which makes fighting fires safer and less dangerous [2]. The fact that the device can quickly and correctly tell people how to put out fires shows that user safety is the most important thing in a fire [3]. When something works well in a fire, you know it's reliable [3]. It is very important to keep firefighters and firefighting robots safe. To achieve this, fail-safe features, redundant systems in key areas, and the ability to work from away can be added [4]. Robots that can constantly work in dangerous environments while staying safe are made with rigorous testing and validation processes that make sure they are reliable [5].

#### 2.3.2 Technoligical compenents

The Arduino chip and the Internet of Things (IoT) are what this fire warning system is built on [2]. A thermal camera finds fires, and an RGB camera with a programme based on deep learning finds things that have already been seen [2]. The device tells a central computer about the fire over the Internet as soon as it senses one. The server then sends alerts to apps on phones [2]. As of this writing, neither smoke alarms nor smoke monitors can tell you what kind of fire extinguisher you need [2]. This work changes a lot of things in this area. The smart fire alarm system has a thermal camera that can find fires and an RGB camera with a deep-learning program that can figure out what it sees [3]. Using these technologies makes it possible to accurately identify and classify fires, which is essential for figuring out the best way to put them out [3]. For instance, the thermal camera can pick up on the unique thermal pattern that a fire gives off, while the RGB camera can precisely pinpoint the elements that are on fire [3]. The NodeMCU ESP8266 lets you join wirelessly and monitor the system from afar, which makes the whole thing work better and respond faster [7]. The sensors send out sound waves and figure out how long it takes for the waves to bounce back. This lets the robot figure out where obstacles are in its way and avoid them [9].

## 2.4 Design of implementation

Making a firefighter robot takes a lot of cautious thought. Engineers make hardware, put sensors in the right places, and write control programmes. The fact that prototypes were made and tried successfully shows that the design was well thought out and uses a number of different parts to work best. The design includes putting the device in place, making sure it can be seen clearly, and connecting it to power and the internet [3]. It's important to plan and think about things like the structure's size, weight, and power. Here, we see how important it is to have strong patterns that can last through rough times. Still, we need to talk in more depth about certain design methods or case studies [4]. Building the frame, choosing motors and wheels to make it move, and adding safety measures that can handle high temperatures [5]. The fire detection and suppression system [7] were made, which is talked about in the design and execution part. It's important to pick the right sensors, microcontrollers, and gearbox parts before adding them to the process [7]. It's important to think about the materials used, how the weight is spread, and how long it will last in rough conditions [8]. From coming up with the first ideas to building and testing prototypes [8], the whole design process is full of careful thought that goes into making systems that work well. To use this method, you need to choose where to put the ultrasonic sensors, wire them up, and program a microprocessor like Arduino to handle sensor data and drive the motors [9]. The steps for building the night vision camera robot are explained in the design and execution section. The text describes picking and adding different parts, like setting up the IoT modules [10], putting the night vision camera, and connecting it to the microcontroller.

### 2.4.1 Water Deployment Systems

Water is essential for putting out fires. Still, the efficiency of distributing water depends on other things. How the valve is made is very important for how well the water is distributed. The ways that rescue robots get water are very important [4]. However, it doesn't give a complete look at or criticise the different technologies or methods fireman robots use to spread water [4]. Water tanks, pipes, and specially designed nozzles are used to get water precisely to the source of the fire [6]. To make firefighting more effective, the amount and pressure of water must be optimised. That being said, these systems have few detailed technical accounts [8].

### 2.4.2 Navigation and Mobility

It's hard to get around in places that are difficult inside. Path planning algorithms help robots navigate complicated environments by finding the fastest paths and avoiding obstacles. Finding your way in complex and dangerous places takes a lot of work. It would help if you used many navigating tools, such as GPS, LIDAR, SLAM, and moving platforms like tracked or wheeled cars. Another part of the study examines finding and avoiding problems [4]. This section discusses the technologies that help rescue robots find their way and move around when things get tricky [5]. Each is best for a specific type of ground [5]. The paper talks about more advanced navigation technologies, such as ultrasonic sensors, infrared sensors, and methods for avoiding obstacles. The goal is to help robots move through rough territory and avoid things that might get in their way [6].

### 2.4.3 Communication Protocols

Communication is needed for robots to be controlled from a distance and for data to be sent between them. Communicating within the system is made possible by Wi-Fi and Bluetooth. This makes it easier to send important data to a central computer and send alerts to smartphone apps. Such a system makes sure that both users and rescue workers get quick information about the fire's characteristics and location [3]. Communication methods are very important for letting people work from afar and send real-time data [4]. However, it doesn't give a detailed look at or discussion of the exact protocols that firefighting robots use [4]. Using encrypted channels of communication to make sure that information is sent safely and reliably during important tasks [5].

## 2.5 The comparation of existing fire fighting robot

Table 2.1: The comparation of fire fighting robot from the source [11]

| Model | LUF60 | FFR-1 | FIREMOTE -4800 | MVF5 | JMXLT50 | SACI | ArchiBot | Thermite | FFM3000 |
|---|---|---|---|---|---|---|---|---|---|
| Size (L x W x H) | 2.3 x 1.35 x 2 m | 1.62 x 1.14 x 1.38 m | 1.4 x 0.7 x 1.14 m | 3.8 x 2.18 x 2.1 m | 2.44 x 1.44 x 1.56 m | 1.8 x 1.5 x 1.6 m | 1.4 x 0.8 x 0.65 m | 1.88 x 0.89 x 1.4 m | 1.5 x 1.0 x 1.3 m |
| Speed (km/h) | 6 | 4 | 4.5 | 12 | 12 | 12 | 20 | 20 | 2.36 |
| Power source | Diesel engine | Battery | Battery | Diesel engine | Diesel engine | Battery | NA | Diesel engine | Battery |
| Control | Radio control | Wireless control | Wifi | Remote control | Radio control | Radio control | NA | Radio control | Radio contro |
| Weight (kg) | 2000 | 940 | 450 | 9274 | 1500 | NA | 450 | 744 | 910 |
| Price (USD) | 200,000 | NA | NA | NA | 49,000 | NA | NA | 98,000 | 40,000 |
| Source | http://www. nngov. com/fire/ downloads/luf60_fire_a pparatus_article | NA | http://www. rylandresearch.co.uk/remotelyoperatedvehicles/fire mote-4800 | http://www.doking.hr/pr oducts/firefighting/mvf_5?product Page=technical | http://info. fire. hc360.com/2010/06/09083222194 7.shtml | http://v2.armtecbrasil.com/index.php?pr oduto=2 | NA | NA | NA |
| Origin | Austria | US | UK | Croatia | China | Brazil | KOREA | US | Malaysia |

## 2.6 The comparation of Selected Literature Review

Table 2.2: Literature Review Comparative Analysis

| Title and Author | Description |
|---|---|
| [1] Smart Sensor Network-Based Autonomous Fire Extinguish Robot Using IoT (2021)<br><br>D. Radha<br><br>M. Arun Kumar<br><br>Nagarjuna Telagam | The creation of an automated robot that can put out fires using IoT and smart sensor networks. It shows how the robot can find and put out fires without any help from a person.<br><br>Advantages:<br><br>1. Makes people safer by reducing their involvement in dangerous fire scenarios.<br>2. Makes it faster and easier to handle fire situations by automating tasks.<br>3. Adds IoT for tracking and controlling in real time<br><br>Disadvantages:<br><br>1. High initial cost and maintenance requirements.<br>2. Potential technical issues and malfunctions.<br>3. Dependence on stable network connectivity for optimal performance. |
| [2] A Smart Fire Detector IoT System with Extinguisher Class Recommendation Using Deep Learning (2023)<br><br>Tareq Khan | The writing talks about using decentralised IOTA-based systems to build digital trust in protecting remote driving apps in city settings. It talks about the problems and ways to fix them when adding safe, trust-based systems to vehicle networks.<br><br>Advantages:<br>1. Better security: Using IOTA's decentralised ledger technology makes data transfers safer when driving from afar by lowering the chance of tampering.<br>2. Reliable Data Exchange: The system makes sure that the data sent between cars and infrastructure is correct and reliable, which is very important for safe remote driving.<br>3. Scalability: IOTA's architecture allows for a network that can grow as more connecting devices and cars are added without slowing down the service. |

| | |
|---|---|
| | Disadvantages:<br><br>1. Hard to set up: Adding IOTA-based systems to current vehicle networks can be hard and require a lot of resources.<br>2. Maturity of the Technology: IOTA and other decentralised technologies are still developing, which could make them less stable and reliable in the long run.<br>3. Problems with regulations: Using these high-tech gadgets in cities might be hard because of rules and regulations, which could slow down their spread. |
| [3] Autonomous fire Protection Robot with notification (2024)<br><br>NOR AYUNI BINTI ABD MAJID | A smart security system with many sensors that includes a fire-fighting robot that works on its own. Three flame sensors on the robot help it find fires. It then moves towards the cause of the fire and sends alerts to connected phones via GSM. Also, it can put out fires on both the upper and lower sides when turned 45 degrees.<br>Advantages:<br><br>1. The robot has three flame monitors that are meant to find fires.<br>2. It can move towards the source of the fire and send GSM alerts to any phone that is linked.<br>3. It is designed so that the robot will stop before it hits the fire.<br>4. It can extinguish fires at 45 degrees for both upper and lower sides.<br>5. The robot's design includes a mechanism to clip the fire extinguisher.<br><br>Disadvantages:<br><br>1. Limited coverage area for example the robot's movement range may be restricted.<br>2. Dependency on GSM network: Connectivity issues could affect notifications. |
| [4] Fire Fighting Robot (2022)<br><br>Amol A. Bhosle<br><br>Shardul S. Bodhe<br><br>Sakshi U. Bodhe | The proposed firefighting robot uses deep learning to detect and classify fires, reducing risks to firefighters. Achieved 98.25% fire detection accuracy and 92% fire-type classification accuracy. |

| | |
|---|---|
| | Advantages:<br>1. Cost-effective<br>2. Autonomous<br>3. Maintenance: Easier to maintain due to its simplified design.<br><br>Disadvantages:<br><br>1. Limited working range so the robot's movement may be restricted.<br>2. Dependency on sensors so reliability depends on sensor accuracy. |
| [5] Autonomous Fire Protection Robot with Notification (2022) | The development of an intelligent multi-sensor-based security system.<br><br>Advantages:<br><br>1. The robot is designed to detect fires using 3 flame sensors.<br>2. It can move toward the fire source and send notifications via GSM to any connected phone.<br>3. The robot is programmed to stop before colliding with the flames.<br><br>Disadvantages:<br><br>1. Limited coverage area for the robot's movement range may be restricted. |
| [6] FIRE-FIGHTING-ROBOT mini project (2022)<br><br>Amol A. Bhosle<br>Shardul S. Bodhe<br>Sakshi U | Advantages:<br><br>1. Reduces risk to human firefighters and the robot takes over hazardous tasks, minimizing danger to firefighters.<br>2. Can access hard-to-reach areas so it can navigate narrow spaces or reach high floors where humans might struggle.<br><br>Disadvantages:<br><br>1. High cost of production: Firefighting robots can be expensive due to high-tech components.<br>2. Limited battery life: Their operational time may be restricted during prolonged fire incidents. |

| | |
|---|---|
| [7] The Development of Fire Detection and Automated Fire Extinguisher System by Using Arduino and NodeMCU ESP 8266 | The system uses Arduino and NodeMCU ESP 8266 to create an automated fire detection and extinguishing system. It employs convolutional neural networks (CNNs) for accurate fire detection and AlexNet architecture for human presence detection in fire zones.<br><br>Advantages:<br><br>1. Early Detection that utilizes machine learning for prompt fire detection.<br>2. Automated Responsethat automatically activates extinguishing mechanisms.<br>3. Human Safety: Detects human presence, potentially saving lives.<br><br>Disadvantages:<br><br>1. Complexity: May require technical expertise for setup and maintenance.<br>2. Dependence on Power: Needs a constant power supply to function. |
| [8] Automated Fire Extinguishing System with GSM Alarm 2014<br><br>Rafat Shams<br>Shafkat Hossain<br>Shaoni Priyom, | To stress how important fire safety is and how important it is to have fire detection and suppression devices that work well. The suggested system includes GSM alarm features to improve communication and alerting during fires..<br><br>Advantages:<br><br>1. Fast and quick response<br>2. GSM Notification<br>3. Multi-Method Approach<br><br>Disadvantages:<br><br>1. Limited Coverage: The system's effectiveness depends on its deployment area.<br>2. Connectivity Dependency: GSM notifications rely on network availability. |
| [9] An Autonomous Obstacle Avoidance Robot Using Ultrasonic Sensor (2020)<br><br>Abayomi O. Agbeyangi | An autonomous robot that uses ultrasonic sensors to navigate and avoid obstacles, enhancing mobility and safety<br><br>Advantages:<br><br>1. Autonomous Movement: The robot can navigate without constant human control.<br>2. Ultrasonic Sensor: Provides accurate obstacle detection and avoidance. |

<table>
<tr><td></td><td>3. Control Application: The Android app allows manual control when needed.</td></tr>
<tr><td>[10] Using Night Vision Camera Robot Based on Internet of Thing (2021)<br><br>Sarjana<br>Sholihin<br>Adewasti</td><td>An IoT-based night vision camera robot for security that makes monitoring more effective in a variety of settings.<br>Advantages:<br>1. Mini Size<br>2. Night Vision Camera<br><br>Disadvantages:<br><br>1. Dependency on Available Light<br>2. Inability to See Through Certain Conditions</td></tr>
</table>

## 2.7 Summary

Fire accidents pose severe risks to people, property, and the environment. No matter how well conventional firefighting methods work, they may be unable to handle cases where people can't get to them, the fire is in a small space, or the area is dangerous. Robotics and technology working together have recently become a possible way to solve these problems. With their advanced sensors, cognitive algorithms, and remote-control interfaces, firefighter robots can help with emergencies, keep people safer, and make fighting fires more efficient. This literature review examines the latest developments in firefighting robots, focusing on sensor technologies, navigation algorithms, communication protocols, safety measures, and real-world uses. We want to find patterns, areas that need work, and possible ways to make more progress in this critical area by looking at all the current studies, case studies, and technical progress. Firefighting robots are essential for keeping people safe and limiting the damage that fires can do, whether used in towns, factories, or disaster zones. Let's read the books to find out what progress has been made that makes firefighting safer and more effective.

# CHAPTER 3

# METHODOLOGY

## 3.1 Introduction

This chapter is about the study and methods used in the project. The main goal of this chapter is to give a short summary of how each part of the project is progressing. The details of the hardware and software parts, the different sensors and actuators used, and the programming logic used for both arduino uno and esp 32 will be included. This section discusses why you should choose certain tool types of sensors and how to test them. The chapter thoroughly describes the methodology used to create a dependable and quick-acting fire-fighting robot by explaining the integration of hardware and software.

## 3.2 Sustainability for the Project of Fire Fighting Robot



The autonomous firefighting robot project takes social, economic, and environmental sustainability into account. Because it can quickly find and put out fires, the robot meets the need for ecological fire control by stopping the spread of flames and limiting damage to the environment. The robot cuts down on carbon pollution and water waste by figuring out the best way to use water. The project aims to lower the cost of traditional firefighting methods by offering reasonably priced fire response options that speed up operations and protect important assets from fire-related damage. Because it is made up of separate parts, the robot can be made more useful and cost-effective by changing its size to fit different situations. In a social sense, robots make things safer by removing risks to people and working in dangerous conditions. By working in dangerous situations, robots make things safer for human firefighters, which is good from a social point of view. They also make groups stronger, especially those that are isolated or don't have a lot of resources.

In addition, the initiative makes it easier to create jobs in engineering and technology, which is good for the business and the growth of experienced workers. In conclusion, the autonomous firefighting robot project advances community resilience, economic efficiency, social welfare, and environmental stewardship. It also supports environmental goals and takes care of the needs of the fire service today.

## 3.3 Methodology

The methodology outlines how the fire-fighting robot prototype has changed over time. The prototype is designed to move freely to achieve its objectives. The prototype has many sensors and motors that allow it to work both automatically with emergency stop and run . The robot's flame sensor will be sensing around to see if there are any fires, and the gas sensor picks up LPG (liquefied petroleum gas), which means that fires could happen and it is not safe to enter. The ultrasonic sensor will scan for obstacle and update how far away they are so it can change its path and move forward. If the distance of the obstacle is below 30 cm, the direction of the prototype will change. The telegram bot is for notification of LPG if detected. When the prototype is in autonomous operation, it moves on its own while actively looking for possible fire hazards. The fire sensor turns on the water pump when it senses fire. The water pump takes liquid from a tank on the prototype, sprays it out of a nozzle, and controls the servo motor to move 30 degrees to 150 degree continuously so the fire goes out completely. This prototype's job is to show what the fire-fighting robot can do and serve as a model for future changes and enhancements. The robot can correctly detect and respond to fire hazards thanks to its sensors, actuators, and the telegram app. This makes it a valuable tool for fire emergency situations and reduce risk . If have any emergency by sending stop at telegram bot the robot can be stopped until the command run sent the robot will start to move.

### 3.4 Project Block Diagram

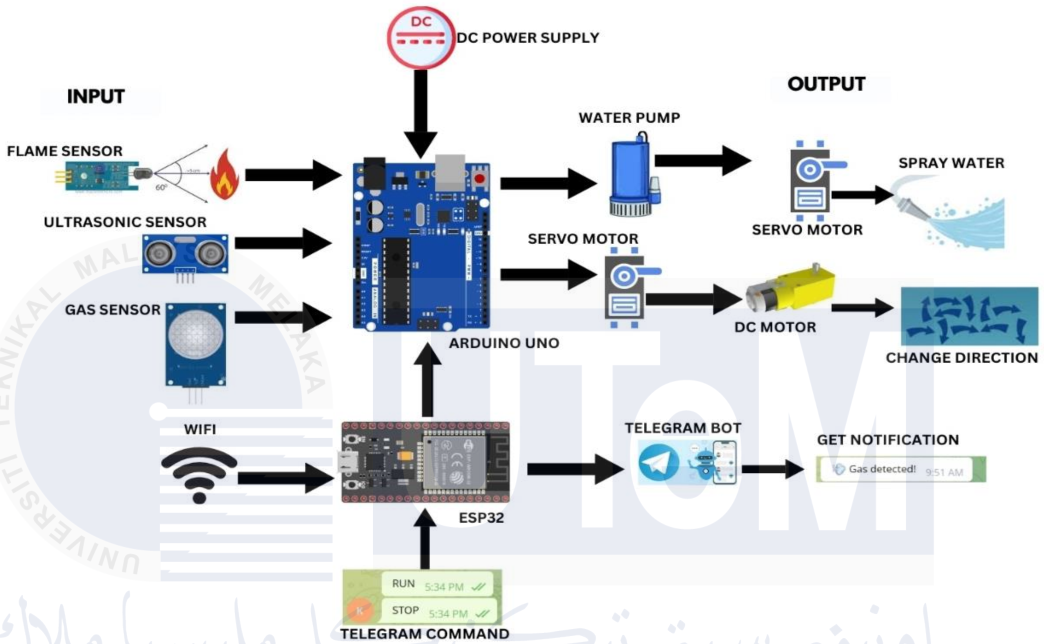



Figure 3.1: Block Diagram of the project



When the flame sensor detect a flame the water pump will activate and the servo start to oscilate. If ultrasonic sensor detect obstacle the servo will move the ultrasonic 0 degree to 180 degree to scan obstacle and the robot will move at the clear path. Each time detect gas the esp32 will sent notification through telegram bot.By sending Command in telegram the robot will stop or run.

## 3.5 Project Functionality Flow Chart

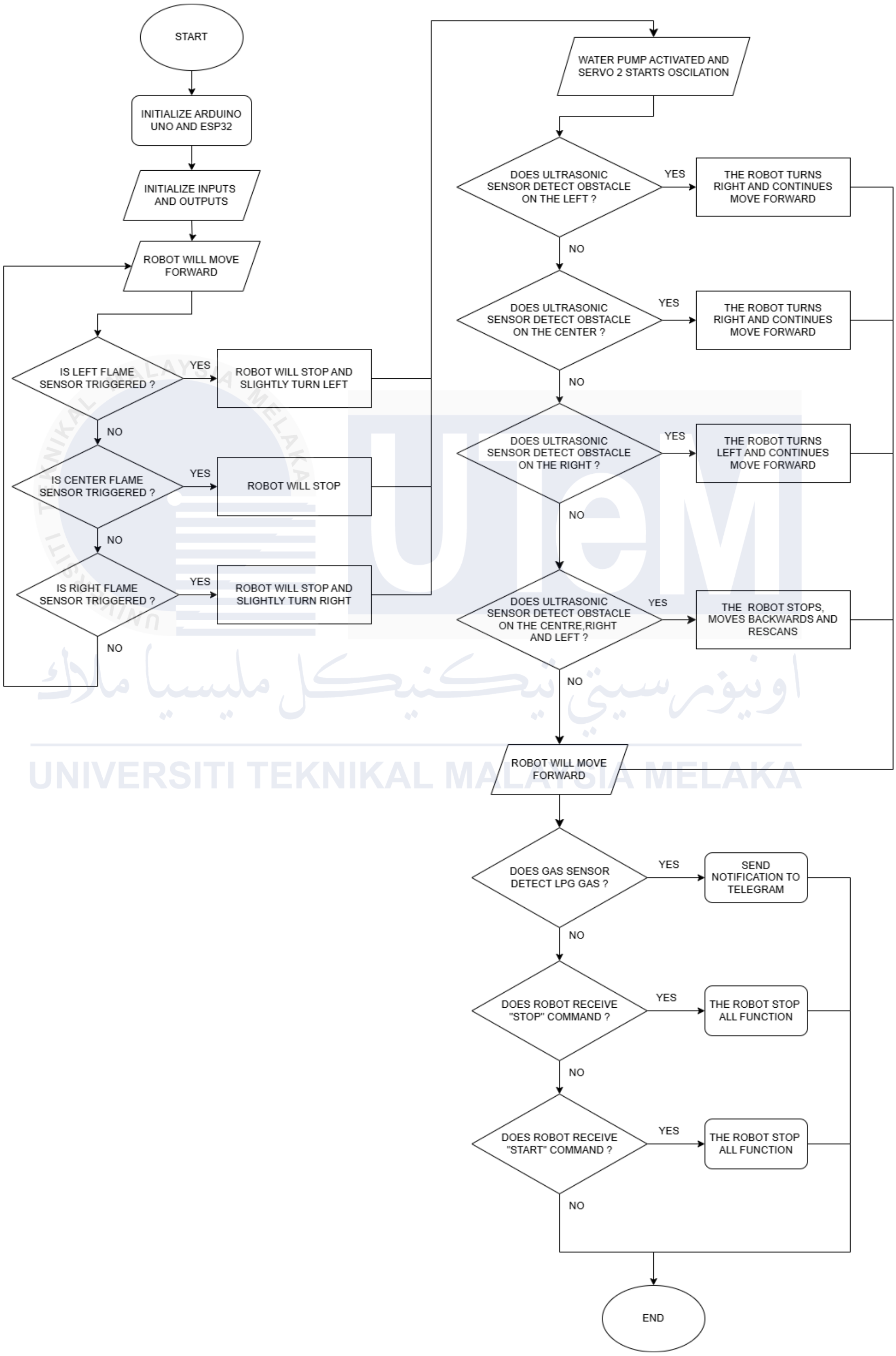



Figure 3.2: Flowchart of the project functionality

The flowchart shows the whole process of making the fire-fighting robot process. As shown in Figure 3.2 it shows how to set up the robot and what choices it makes so it can find fires on its own and put out those fires. Setting up the input and output parts and initialising the arduino uno and esp 32 are the first things that the system does. The robot uses a flame sensor to look for fires after looking for gases. Servo Motor 2 changes the angle of the water jet to make sure it is aimed properly when the flame sensor detects fire. The water pump then starts to release liquid from the tank.  Then ultrasonic sensor monitor is used to find things that might be in the way. A DC motor lets the robot change its path if it hits something. If no problems are found, the prototype will keep going in the direction to go after, the gas sensor checks to see if any LPG gases are present. If it finds any, it sends a message to the Telegram bot immediately. By sending Command in telegram the robot will stop or run. The robot can move around on its own, find possible threats, and stop them by following this plan. For fires, this makes sure that things are taken care of quickly and well.

### 3.6 Flame Detection and Extinguishing Algorithm Flowchart

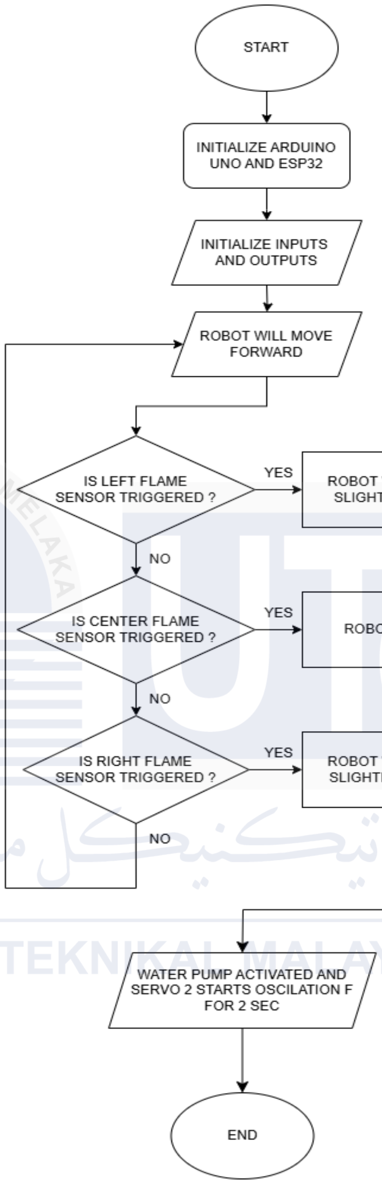





Figure 3.3 Flowchart of Flame Detection and Extinguishing

The flowchart shows the working algorithm of how the fire-fighting robot prototype will be able to detect the fire and extinguish it. As shown in Figure 3.3, the system starts by initialising the flame sensor, water pump and servo motor 2. If a flame is detected, the prototype robot will redirect and move towards the direction of flame, the water pump will activate, and servo motor 2 will activate to oscilate accordingly to extinguish the flame successfully.

### 3.6.1 Obstacle Detection and Navigation Algorithm Flowchart

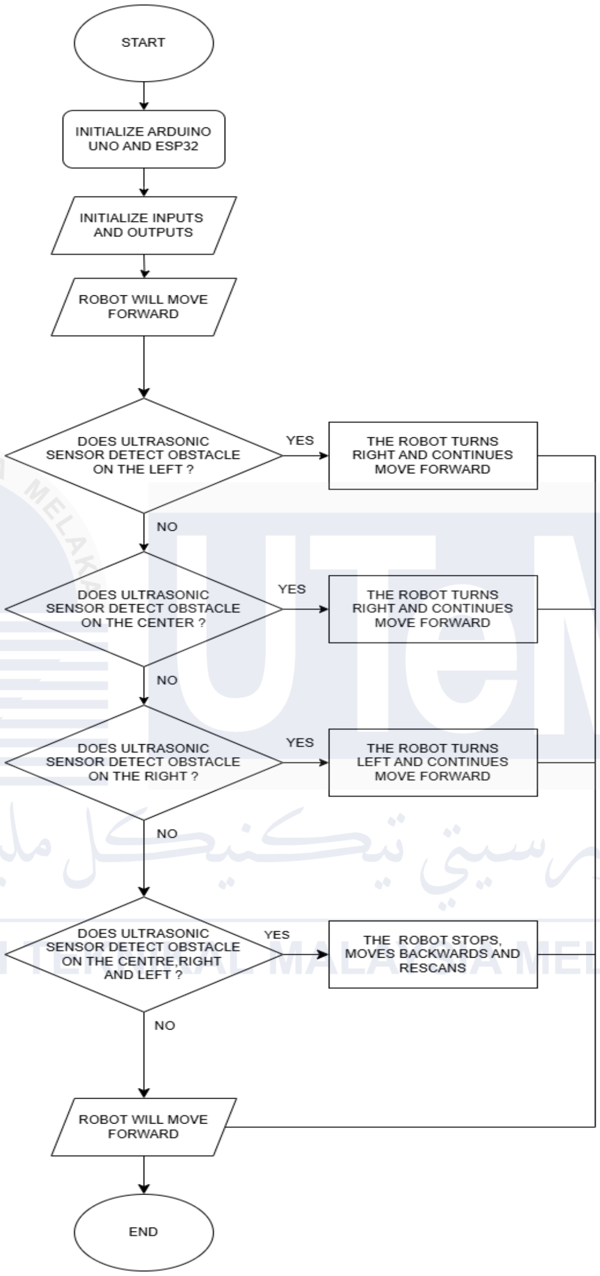



Figure 3.4 Flowchart of Obstacle Detection and Navigation

The flowchart shows the working algorithm of how the fire-fighting robot prototype will be able to automaticly move by itself while avoiding obstacle. As shown in Figure 3.4, the system starts by initialising the ultrasonic sensor and servo motor 1. If obstacle is detected, the robot will redirect the path and move accordingly to avoid and overcome the obstacle.

## 3.7 Software Imlementation

### 3.7.1 Arduino Software

Figure 3.5: Arduino Software

The Arduino Integrated Development Environment (IDE) is a free piece of software that lets you write code, compile it, and send it to Arduino microprocessor boards. Because of Arduino software (IDE) and boards (hardware), it is now easier to make electronics. With the help of other parts, this set lets you Acreate digital and interactive gadgets. We talked about Arduino boards in the last piece. We will learn what Arduino software (IDE) is and how to use it in this piece. The open-source Arduino software (IDE) is used to program Arduino boards. Arduino created an integrated development environment (IDE). Cc. Allow board users to write and share code. It also had a lot of tools and a set of examples of small projects. Arduino's integrated development environment (IDE) works with Windows, Linux, and Mac OS X, and it supports the computer languages C and C++.The Arduino software is simple to use, even for more experienced people. It can help you learn how to program gadgets and robots and make interactive prototypes.

### 3.7.2 Cirkit Designer IDE

Figure 3.6 Cirkit Designer IDE

Cirkit Designer is a multipurpose (IDE) for planning, designing, simulating, and prototyping electronic circuits. It integrates schematic capture, PCB layout, and circuit simulation tools that can be easily used. With features like vast component libraries, 3D visualisation, and microcontroller integration, it simplifies the design process from concept to implementation. Cirkit Designer allows users to simulate circuit, detect errors, and optimising designs to manufacturing, which makes it a useful tool for electronics design and development. This software also has most of the components and sensors.

### 3.7.3 Telegram App

Figure 3.7 Telegram App

Telegram Bot app in the firefighting robot project serves multiple key functions. It provides real-time notifications to alert users of detected flames or LPG gas, ensuring instant awareness of potential dangers. Users can monitor real-time data from the robot's sensors, including flame detection, gas levels, and obstacle avoidance, all remotely through the app. The bot also allows users to send control commands, such as starting or stopping operations or activating specific components like the water pump. Additionally, it automates emergency responses by sending critical alerts with situational details and logs key events for performance analysis, making it an essential tool for effective remote management and enhanced safety.Its a good app for Iot usages. Telegram is one of the apps usedin user friendly iot projects.

### 3.7.4 Solid Works Software

Figure 3.8: Solid Works Software

SolidWorks is a software that is used to draw a detailed 2d and 3d design and can merge parts also using that. SolidWorks is used a lot in many fields, like manufacturing, consumer goods, aerospace, and the automobile industry. It makes it easier to spread new ideas and speeds up the process of going from the planning phase to the manufacturing phase and making good use of the software.

## 3.8 Hardware Implementation

### 3.8.1 WIFI MODULE (ESP 32)

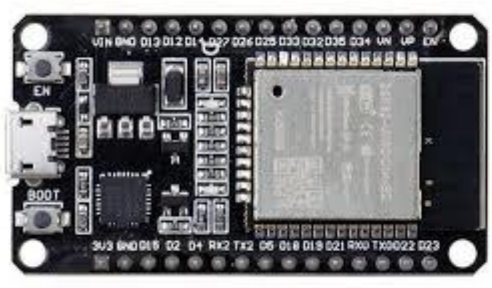


Figure 3.9: ESP 32

The ESP32 device is a powerful and affordable way to make IoT apps. A dual-core processor, built-in Wi-Fi and Bluetooth, a lot of general-purpose input/output (GPIO) pins, and low power usage are some of the things that ESP32 has. The ESP32 has a dual-core Tensilica (Santa Clara, CA, USA) Xtensa LX6 microprocessor, which gives it more processing power and makes it easier to do multiple things at once and complicated jobs quickly. The ESP32 has built-in Wi-Fi and Bluetooth ports that make it easier to connect to and talk to other networks or devices. It works with different Wi-Fi protocols, like 802.11 b/g/n, and lets you join via Bluetooth Classic and Bluetooth Low Energy (BLE). The ESP32 has a lot of GPIO pins that make it easy to connect and handle sensors and devices that are outside the board. These pins work with many devices, such as SPI, I2C, UART, and PWM. Because the ESP32 is made to use little power, it can be used to make IoT apps that use less power. It has sleep modes and power control features that help lower power use, so it can be used for projects that need to run on batteries or with limited power. ESP32 can be linked to screens, touchscreens, or LED indicators to make the interface easy for workers or staff to use.

Several development tools and languages can be used to program ESP32. C++ is the most popular programming language. The Arduino IDE or PlatformIO can be used to program in C++. The ESP-IDF (Espressif IoT programming Framework) also has a lot of libraries and tools that are designed to help with ESP32 programming.

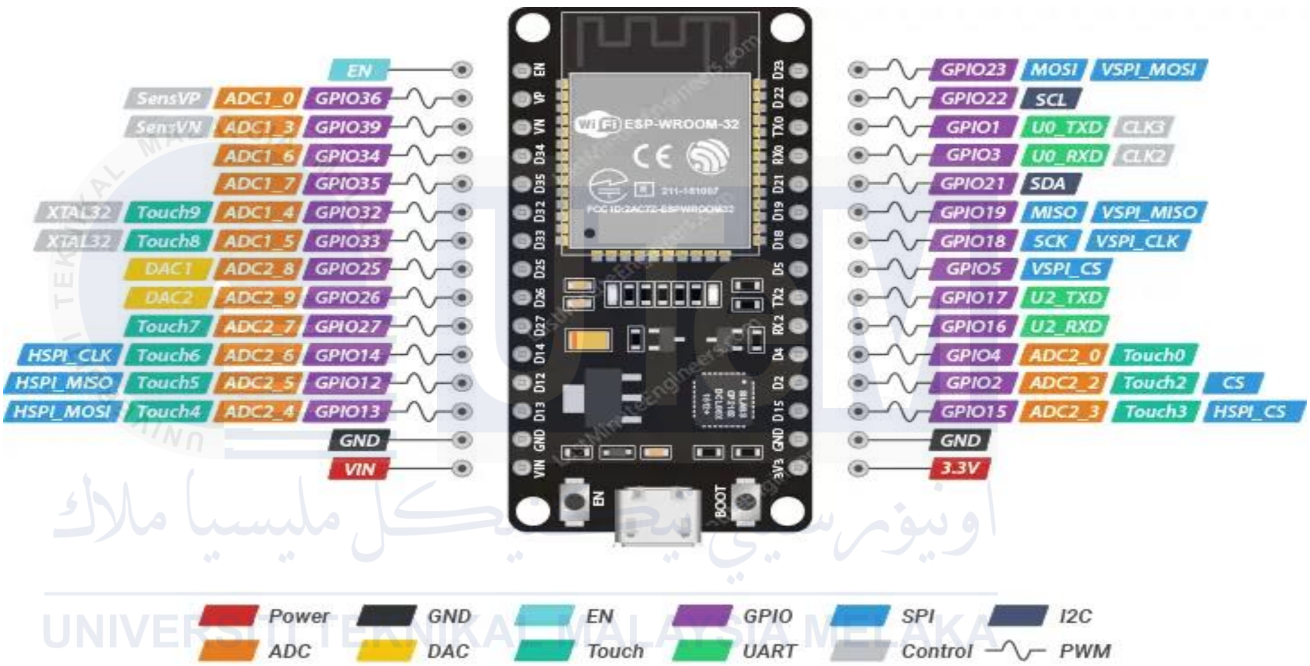



Figure 3.10: ESP 32 pin arrangments

There are a lot of different pins on the ESP32 microprocessor that let it do a lot of different things. As well as 34 GPIO pins for flexible digital input/output jobs, it has power pins (3V3 and GND) for a stable power supply. There are 18 ADC pins that can turn analogue signals into digital ones and 2 DAC pins that can turn digital ones into analogue ones. Most GPIO pins can handle PWM, which lets you do things like change the speed or brightness of an LED. There are 3 UART, multiple SPI, 2 I2C, and I2S for audio applications as communication ports. It also has 10 capacitive touch pins for touch-sensitive interfaces and special pins such as EN to reset the chip, BOOT to flash firmware, and RTC GPIOs for low-power activities while the chip is in deep sleep mode. The ESP32 can be used for a lot of different electrical projects because its pins can do different things.

### 3.8.2 Arduino Uno


Figure 3.11: Arduino Uno

The Arduino Uno is a popular microcontroller board based on the ATmega328P, designed for a wide range of electronic and robotics projects. It operates at 5V and has 14 digital I/O pins, six of which support PWM output, as well as six analog input pins for reading signals from sensors. The digital pins (0-13) can function as inputs to read signals or outputs to control components like LEDs or motors. The analog pins (A0-A5) convert analog signals from sensors into digital values using a 10-bit ADC. Special pins include RX and TX for serial communication, AREF for analog reference voltage, and a reset pin to restart the microcontroller. The board also features power pins, including 5V and 3.3V outputs to power components, VIN for external power input (7-12V), and multiple GND pins for grounding. With these versatile pins, the Arduino Uno is ideal for integrating sensors, actuators, and communication devices, making it a core component in robotics and automation projects.

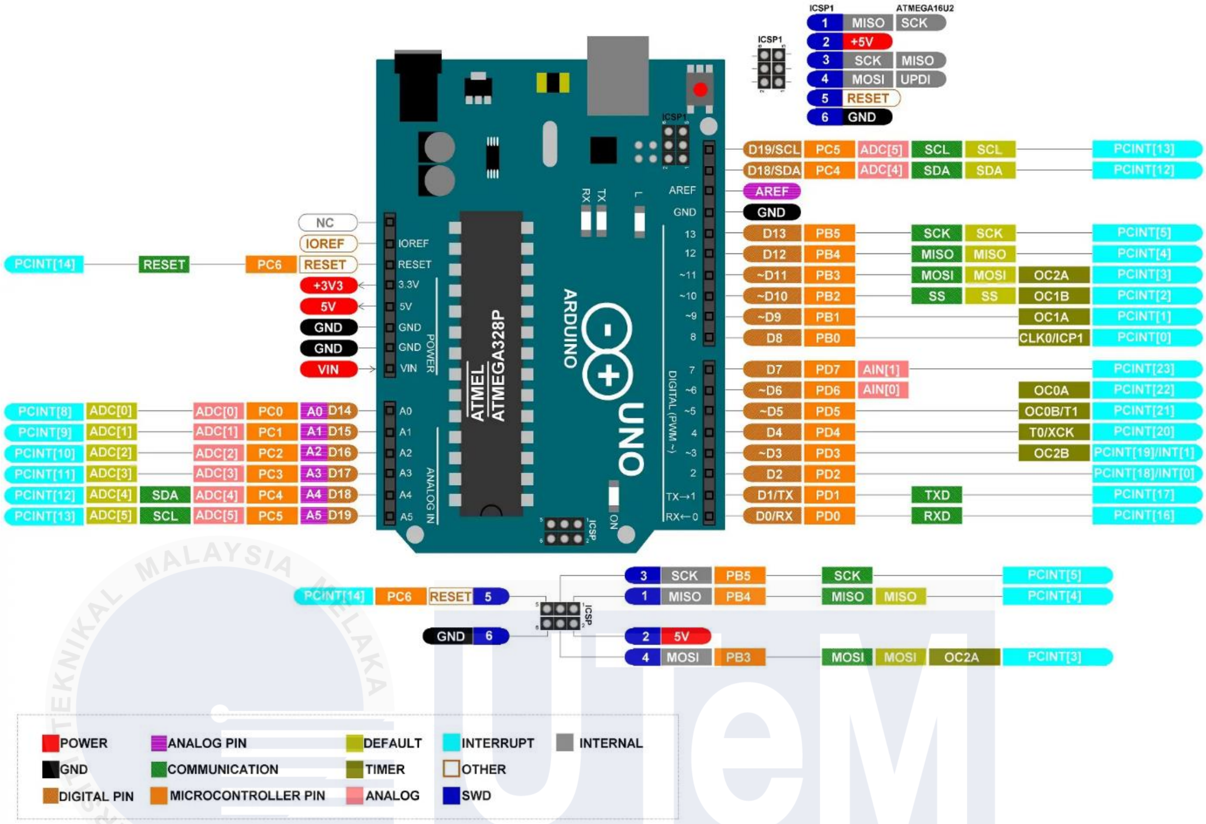



Figure 3.12: Arduino Uno pins configuration

The arduino uno board consist of 14 digital I/O pins and from tthat 6 pins will support pwm, 6 analog input pins with 10 bit resolution, and haven power pins for 3.3V, 5V and GND.It also have communication pins like UART (pins 0 and 1 for RX/TX), I2C (pins A4 for SDA and A5 for SCL), and SPI (pins 10–13). The Arduino Uno board can run in 5V and each I/O pin can handle up to 20mA, making the Arduino Uno suitable for various electronics and robotics projects.

### 3.8.3 Flame Sensor

Figure 3.13: Flame Sensor

Flame sensors primarily functions to allow early fire detection to improve safety. is the sensor detects a fire, hence it is typically followed by activating alarm or initiating fire suppression systems. These sensors have a wide viewing field ranging from 60 to 120 degrees. Additionally, flame sensors also have a rapid response time, usually less than a second with detection range is between 1 to 5 meters.  It is developed to indicate the presence of flame by detecting it with a specific wavelength consisting of ultraviolet (UV) and infrared (IR) light and operate at low voltages (3.3V to 5V).  After detecting fire, it processes signals to determine the presence of flame. In summary, the highlight of flame sensors include suitable for various enviroment energy efficiency high sensitivity, seamless integration into existing safety and automation systems, high response time.

### 3.8.4 Ultrasonic Sensor

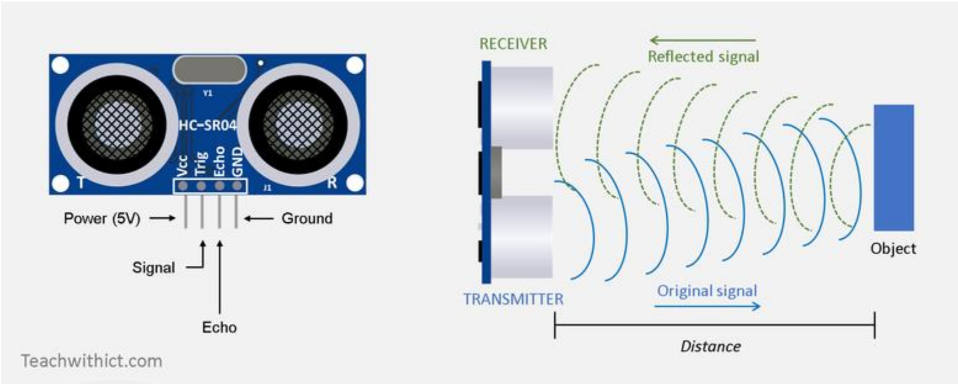



Figure 3.14: Ultrasonic Sensor

Ultrasonic sensors are flexible devices that use ultrasonic waves to find things and measure distance. They usually work at frequencies between 20 kHz and 40 kHz. It checks how long it takes for the echoes to come back. Ultrasonic sensors have an emitter that sends sound waves out and a receiver that picks up the waves that bounce back. These sensors can measure lengths from a few centimetres to several metres with great accuracy. Ultrasonic sensors are useful because they can measure without touching the object being measured, find a lot of different materials, be easily integrated into existing systems, and work well in a variety of situations. In industrial automation, robots, automotive systems, and other places, they are used to measure levels, find objects, and sense when something is close by. Ultrasonic sensors work well in a wide range of conditions because they can pick up on clear or shiny items and are not affected by light. Because of these qualities, ultrasonic sensors are necessary to improve accuracy, safety, and efficiency in many technological and commercial settings. In industrial automation, robots, automotive systems, and other places, they are used to measure levels, find objects, and sense when something is close by.

### 3.8.5 Gas Sensor


Figure 3.15: Gas Sensor

Gas sensors that detect LPG (liquefied petroleum gas) is a crucial sensor in identifying the concentration of combustible gas in the surrounding air. They determine the concentration via a chemical process that generates a signal in relation to the gas concentration is in the air. Tin dioxide (SnO2) semiconductor namely functions as a sensing element is employed by the sensor, changes resistance with the presences of gas. LPG (liquefied petroleum gas) is detected by the gas sensor.the sensor is highly sensitive and accurate due to its ability to detect LPG as low as a few parts per million (ppm). Hence with the usage of these gas sensor, the safety can be improved due to the sensor providing early warning signs. It is also very reliable with low power consumption and rapid response time. It can be widely used in commercial, residential and industrial settings, namely homes, kitchen and manufacturing plants.

### 3.8.6 Servo Motor

Figure 3.16: Servo Motor

The servo motor serves as an actuator that is commonly employed in tasks that demand accurate control of linear or angular orientation, speed, and acceleration. In this project, a ,micro servo motor is employed and it is made of a tiny DC motor, a gearbox, and a feedback control circuit. A small servo motor usually weighs around 9g and has an operating voltage ranging from 4.8V to 6V. Despite its compact size, the servo motors is capable of generating a high torque as much as 1.8 kg cm and is able to function with a speed of 0.1 seconds per 60 degrees. It can be vastly used in operations that needing a precise control of angular or linear position, velocity, and acceleration. These characteristics make it ideal for hobbyist projects, academic purposes, and lightweight automated systems in which space and weight limits are essential.



### 3.8.7 DC Motor

Figure 3.17: Dc Motor

A 12V DC motor is a type electrical equipment that transforms electrical energy into mechanical energy. It comprises of a rotor and stator can are well known fields such as automotive industries, home appliances, toys including electric traction. magnetic field is produces in the stator when the DC motor is turned on. By manipulating the supply voltage and the current, the speed and the torque of the motor can be adjusted. DC motors have plenty of advantages namely easy to control, compact design and high efficiency at a low cost. However, the drawbacks of this component, is that they may require additional component to ensure accurate control.

### 3.8.8 Water Pump

Figure 3.18: 5V Water Pump



A water pump is used to move, compress, or transfer water from a lower level to a higher level. A 5V water pump runs on 5V power supply and is commonly used in small-scale tasks namely aquariums, small automation and IoT projects. Hence one of the constraint of this component include limitations when it comes to larger projects.They are submersible or have an external design which makes installation easier. The pump functions by employing an electrical motor to drive an impeller, creating water flow. The highlights of utilizing a 5Vwater pump includes easy installation, low cost and low voltage power supply.

### 3.8.9 L298N DC Motor Driver Dual H Bridge

Figure 3.19: L298N DC Motor Driver

The L298N motor driver is a H-bridge that is mostly used to forward, reverse, and speed control for two DC motors or one stepper motor. It can handle up to 2A of continuous current per channel (3A peak) and runs on a motor supply voltage range of 5V to 35V. PWM signals pins are the enable pins (ENA and ENB) is mainly used to control speed. The input pins are (IN1, IN2, IN3, and IN4) are used to control direction. To manage back EMF from motors, the module has integrated protective diodes. It is perfect for robotics and automation projects because it is simple to combine with microcontrollers like Arduino and ESP32.

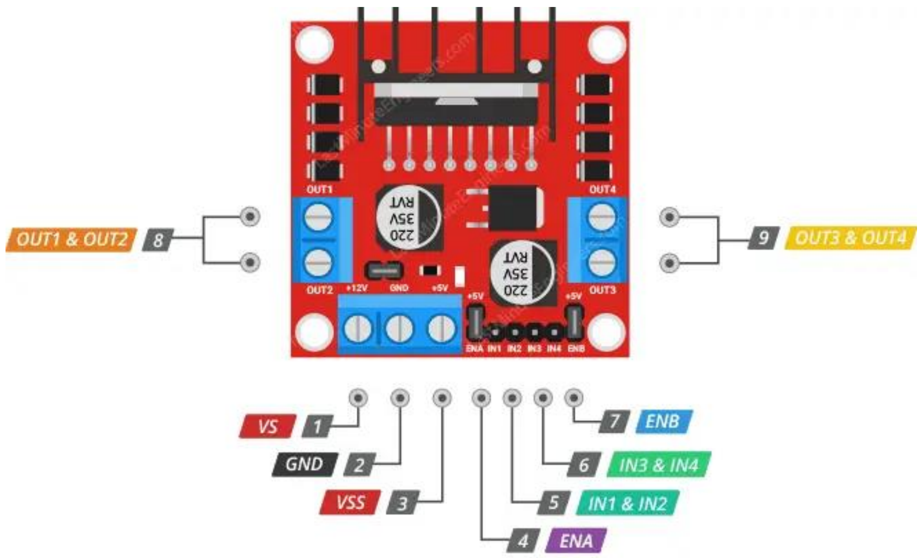



Figure 3.20: L298N motor driver pin configuration

The L298N motor driver have  Vcc for motor supply voltage, 5V to 35V, 5V for logic supply, for output if Vcc > 7V, and GND are the power pins in there. The control pins for dc motor are include IN1, IN2, and IN3, IN4, which to control the direction for motors A and B, and ENA and ENB, which enable and control the speed of motors A and B using PWM signals. The motors are directly connected to the motor output pins which is OUT1, OUT2 for motor A, and OUT3, OUT4 for motor B.

## 3.9 Experimental Setup

The aparatus of the development of an iot-based fire fighting robot by using microcontroller circuit design as shown in figure 3.2.1 ia a setup to conduct the experiment and the process to test each sensor with the system build which consist of:

1) Arduino Uno
2) Esp 32
3) L298N motor driver
4) Flame sensor
5) Ultrasonic sensor
6) Servo motor
7) Gas sensor
8) 5V Relay
9) Water pump
10) 1k and 2k Resistor
11) Voltage Regulator
12) Dual axis TT Gear motor
13) 7.4 V battery

### 3.9.1 Circuit Design

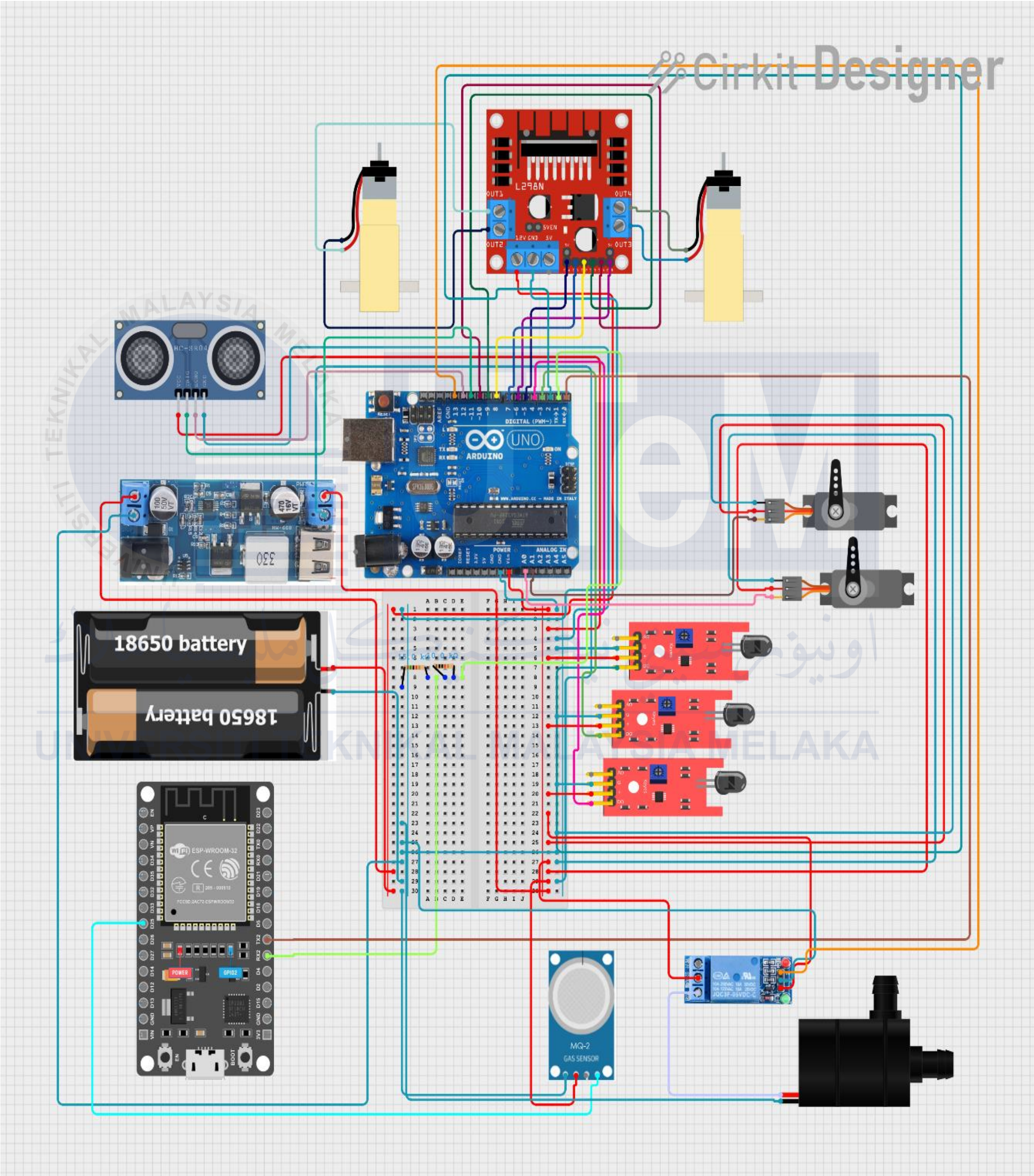



Figure 3.21: Circuit Design using Cirkit Designer IDE

### 3.9.2 Code Development and Algorithm Design

1) CODING FOR ARDUINO UNO

Brief explaination for codingis the code controls a fire-fighting robot equipped with sensors and actuators for detecting and extinguishing flames. The setup() function initializes all hardware components, including motor driver pins, flame sensors, a relay for activating a water pump, and two servos for ultrasonic scanning and flame extinguishing. The loop() continuously monitors flame sensors for fire detection and the ultrasonic sensor for obstacles, deciding the robot's behavior accordingly. Functions like handleLeftFlameDetection () and handleFlameDetection() control the robot’s movement toward detected flames and trigger the water pump while sweeping the extinguishing servo. The handleObstacle() function determines the safest path when an obstacle is detected using the ultrasonic sensor. Motors are controlled by rotateMotor(), adjusting speed and direction with PWM signals. Additionally, the RX/TX pins facilitate serial communication for data exchange between the Arduino and ESP32 for real-time coordination, enhancing the robot's functionality. Pins used in this coding:



| Pin | Function |
|---|---|
| A0 | Ultrasonic servo control |
| A1 | Flame-extinguishing servo control |
| 11 | Ultrasonic sensor trigger |
| 12 | Ultrasonic sensor echo |
| 2, 3, 4 | Flame sensors D0 (left, center, right) |
| 5 | Right motor speed control (ENA) |
| 7, 8 | Right motor direction control (IN1, IN2) |
| 6 | Left motor speed control (ENB) |

| | |
|---|---|
| 9, 10 | Left motor direction control (IN3, IN4) |
| 13 | Relay to activate water pump |
| RX | Serial data input for communication for ESP32 TX |
| TX | Serial data output for communication for ESP 32 RX |

2) CODING FOR ESP32

The code allows an ESP32 microcontroller to monitor flame and gas detection and send notifications via Telegram when either is detected. The Wi-Fi and Telegram Bot credentials are initialized to enable network communication. The UART pins RXD2 and TXD2 are used for serial communication between the ESP32 and Arduino. The setup() function initializes the UART communication, gas sensor pin, and Wi-Fi connection. Once connected, the ESP32 is ready to receive and process data. The gas sensor connected to the ESP32 is also continuously monitored, and if gas is detected, a notification is sent after a debounce delay of 2 seconds. The sendTelegramNotification() function ensures seamless interaction with Telegram, notifying the user promptly of any detected hazards. Pins use the this code are:

| Pin | Function |
|---|---|
| 16 (RXD2) | UART RX pin of ESP32, receives data from Arduino TX |
| 17 (TXD2) | UART TX pin of ESP32, sends data to Arduino RX |
| 25 | Gas sensor input pin |

### 3.9.3 Setting up a Telegram Bot

For the fire-fighting robot, a Telegram bot has been implemented to provide real-time notifications and updates. This bot is configured to alert the user whenever critical events, such as flame or gas detection, occur. It communicates wirelessly using the ESP32 microcontroller, which processes data from various sensors integrated into the robot. The bot sends concise, actionable alerts directly to the user's Telegram chat, ensuring timely awareness of any detected hazards. This feature enhances the robot's functionality by providing a seamless and efficient way for the user to monitor the robot's status and environmental conditions, even remotely. The Telegram bot serves as an essential communication interface, bridging the robot's on-field operations with user interaction.

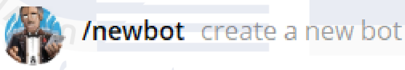



Figure 3.22: Telegram Bot Set up



Need to use telegram bot father to create new bot and name the bot. Once the bot is ready we will have the bot token with that we can open and use the bot. Next need to setup the chat id and the bot token at esp 32 coding to use iot projects

### 3.9.4 Tank 3D Modeling and Assembly

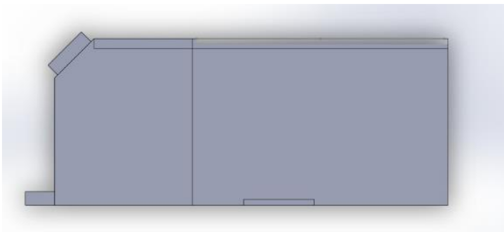
The 3D model shown represents the structural design of a tank base intended for a fire-fighting robot. The design includes a rectangular body with multiple circular cutouts, likely aimed at reducing material weight while maintaining structural integrity. The slanted front panel and open compartment suggest a space for mounting critical components such as sensors, cameras, or extinguishing mechanisms. The cylindrical top feature might serve as a mounting point for additional equipment like a hose or nozzle for dispersing water or extinguishing agents.

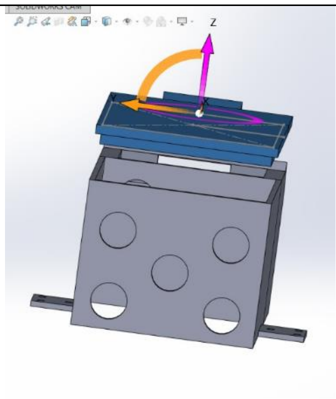
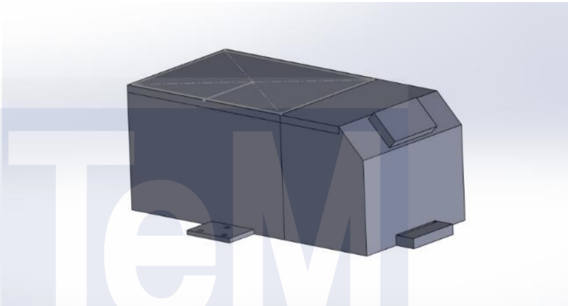
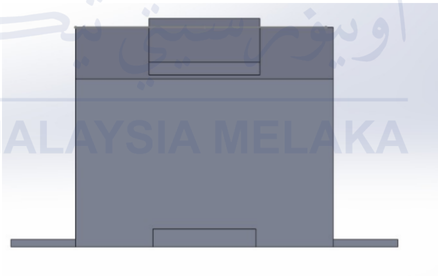
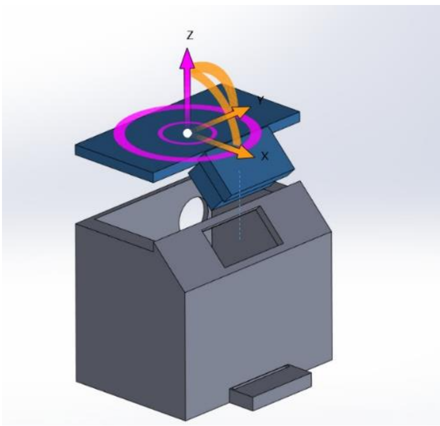
The flat base and extended sections provide a stable foundation, crucial for maintaining balance during movement on various surfaces. The compact design is functional for navigation in confined spaces, a key requirement for fire-fighting robots operating in challenging environments. The circular openings can also facilitate ventilation or act as cable-routing points for wiring.

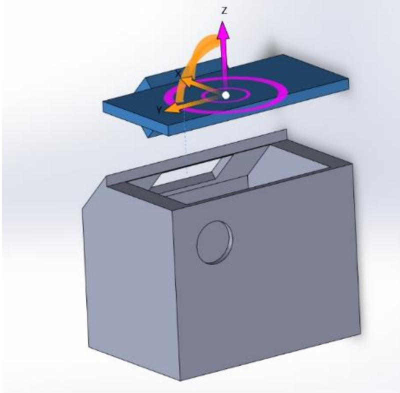
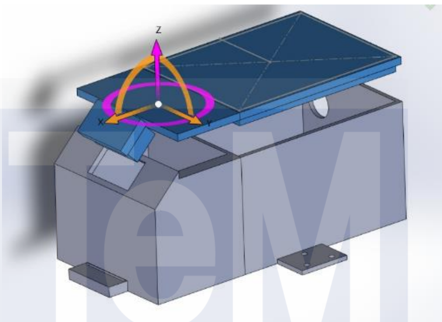
This design aligns with the robot's purpose by prioritizing stability, reduced weight, and space optimization for mounting essential components and sensors.



Table 3.1 SolidWorks 3d Drawing

| **DIMENSIONAL VIEW** | **RESULT** |
|---|---|
| Side view of the model |  |
| Front View of the model | |

| | |
|---|---|
| |  |
| Orthogonal View of the model |  |
| Back View of the model |  |
| Design of the water tank |  |
| Back view of tank | |

| | |
|---|---|
| |  |
| Top view of fully merged modal |  |



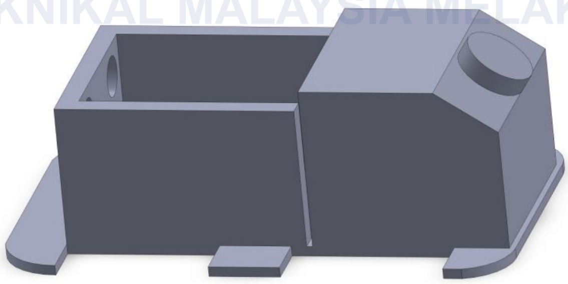


Figure 3.23: 3D Figure of the Prototype

## 3.10 Summary

This chapter presents the proposed methodology to develop the autonomous fire-fighting robot involved integrating hardware design, software development, and rigorous testing. Key hardware components included a arduino uno , esp 32, various sensors (ultrasonic, flame, gas), a servo motor, and a water pump. Software algorithms enabled real-time flame detection, obstacle avoidance, and environmental monitoring, with the telgram bot app providing a user-friendly interface for remote control and data visualization. Integration ensured seamless communication between hardware and software, while controlled testing validated the robot's capabilities in detecting fires, navigating obstacles, and deploying water accurately. Adjustments were made to enhance reliability and efficiency, resulting in a robust autonomous fire-fighting robot.

# CHAPTER 4

# RESULTS AND DISCUSSIONS

## 4.1 Introduction

This project chapter delves into the culmination of efforts and endeavours to develop an autonomous fire-fighting robot equipped with advanced sensors, microcontrollers, and remote-control functionalities. This chapter comprehensively overviews the results obtained through rigorous experimentation, testing, and validation processes. It elucidates the fire-fighting robot's efficacy in detecting flames, navigating obstacles, and deploying water with precision, thereby enhancing emergency response capabilities in fire-prone environments.



## 4.2 Results and Analysis

The development of this system shows that the prototype of an iot-based fire fighting robot uses a range of sensors and components to collect data and make the prototype work accordingly to the logic working flow. The sensors activate and operate under spesific logic working conditions.

### 4.2.1 System Prototype Model

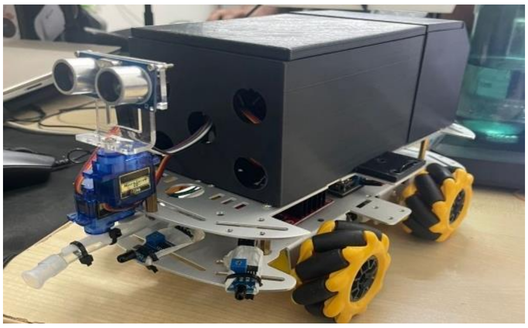


Figure 4.1 Hardware prototype model

Figure 4.1 shows the the sensors and the design of the prototype. The ultrasonic sensor is used to detect obstacles and help the prototype to avoid obstacles. The flame sensor is to detect flames and extinguish them. The prototype uses 3d printed tank and body to make sure its light and strong. The prototype uses 4 mecanum wheel with omni direction, with this function the prototype can easily move or rotate 360 degrees without sliping on tiles or stuck while moving and the motors are connected in tank system, the motor 1 and 2 connected at channel 1 which is right side of the prototype and motor 3 and 4 connected at channel 2 which is left side of the prototype. With this sytem the prototype can easily move around and can reach its goal.

### 4.2.2 Hardware and system design

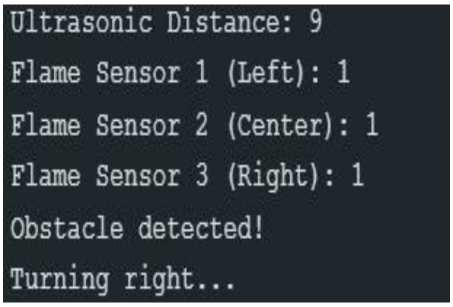
Figure 4.2: Circuit Design

The figure above shows the hardware wiring connection of the prototype arduino uno and esp 32. The wiring connection layout provides reliable power distribution with minimal interference. The setup also ensures safety during the operation.

### 4.2.3 Obstacle Avoidance Testing

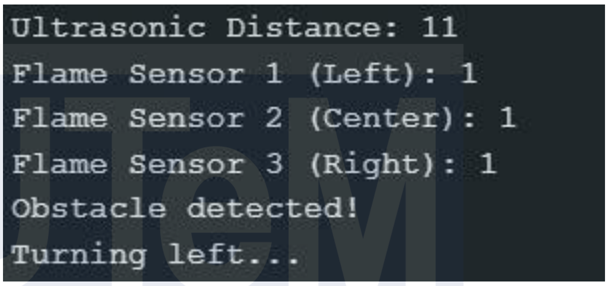


Table 4.1 Experiment and Result of Obstacle Avoidance

<table>
<tr><th>Obstacle Detection side</th><th>Path Chosen by Prototype</th><th>Motor 1 and 2 Movement</th><th>Motor 3 and 4 Movement</th></tr>
<tr><td>The obstacle is detected at centre only (9cm)</td><td>If the obstacle detect at centre only the prototype will turun right</td><td>The motor 1 and 2 which is connected at channel 1 will rototate CCW</td><td>The motor 3 and 4 which is connected at channel 1 will rototate CW</td></tr>
<tr><td colspan="4"> <br><br>The robot moves continuously and stops when an obstacle is detected at the centre. Then the robot will turn right to avoid the obstacle. The wheels on the right moves</td></tr>
</table>

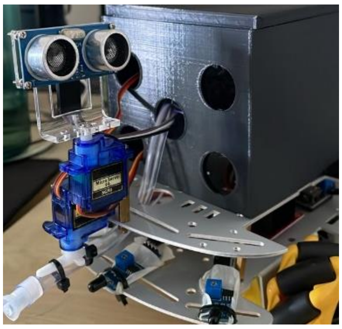
clockwise and the wheels on left moves counter clockwise.  Utilising this logic, the robot can succesfullly avoid the obstacle.The robot will detect and react the obstacle with a range of maximum 30 cm, as seen in the above figure the obstacle at 9 cm and the robot avoided the obstacle.

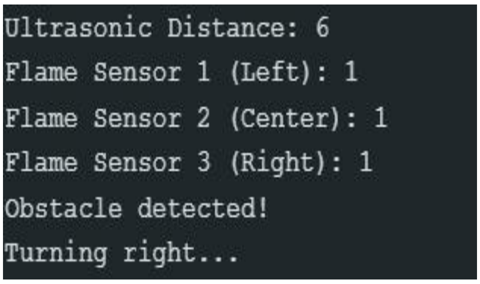
| Obstacle Detection side | Path Chosen by Prototype | Motor 1 and 2 Movement | Motor 3 and 4 Movement |
|---|---|---|---|
| The obstacle is detected at centre and right only (11cm) | If the obstacle detect at centre and right only the prototype will turun left | The motor 1 and 2 which is connected at channel 1 will rototate CW | The motor 3 and 4 which is connected at channel 1 will rototate CCW |




The robot moves continuously and stops when an obstacle is detected at the centre and right .Then the  robot will turn left to avoid the obstacle. The wheels on the right moves Counter clockwise and the wheels on left moves clockwise.  Utilising this logic, the robot can succesfullly avoid the obstacle.The robot will detect and react the obstacle with a range of maximum 30 cm, as seen in the above figure the obstacle at 11cm and the robot avoided the obstacle.

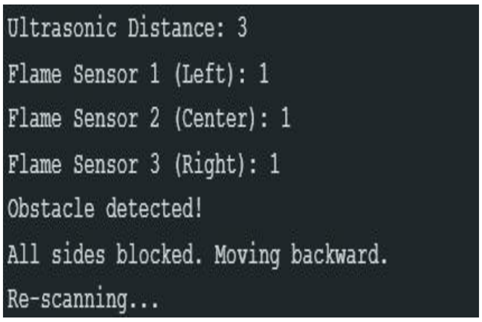
| Obstacle Detection side | Path Chosen by Prototype | Motor 1 and 2 Movement | Motor 3 and 4 Movement |
|---|---|---|---|
| The obstacle is detected at centre and left only (6cm) | If the obstacle detect at centre and left only the prototype will turun left | The motor 1 and 2 which is connected at channel 1 will rototate CCW | The motor 3 and 4 which is connected at channel 1 will rototate CW |






The robot moves continuously and stops when an obstacle is detected at the centre and left. Then the robot will turn right to avoid the obstacle. The wheels on the right moves Counter clockwise and the wheels on left moves clockwise. Utilising this logic, the robot can succesfullly avoid the obstacle.The robot will detect and react the obstacle with a range of maximum 30 cm, as seen in the above figure the obstacle at 6cm and the robot avoided the obstacle.

| **Obstacle Detection side** | **Path Chosen by Prototype** | **Motor 1 and 2 Movement** | **Motor 3 and 4 Movement** |
|---|---|---|---|
| The obstacle is detected at centre, right and left only (3cm) | If the obstacle detect at all 3 side the prototype will move backwards and rescan for clear path | The motor 1 and 2 which is connected at channel 1 will rototate CCW | The motor 3 and 4 which is connected at channel 1 will rototate CCW |

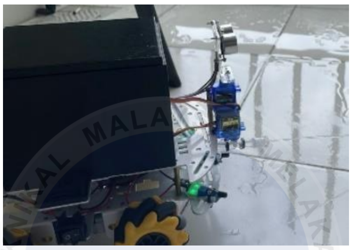



The robot moves continuously and stops when an obstacle is detected at all side .Then the robot will move backwards and rescan until got a clear path. The wheels on the right moves counter clockwise and the wheels on left moves counter clockwise. Utilising this logic, the robot can succesfullly avoid the obstacle.The robot will detect and react the obstacle with a range of maximum 30 cm, as seen in the above figure the obstacle at 3 cm and the robot avoided the obstacle.

### 4.2.4 Flame detection Testing

Table 4.2: Experiment and Result of Flame Detection

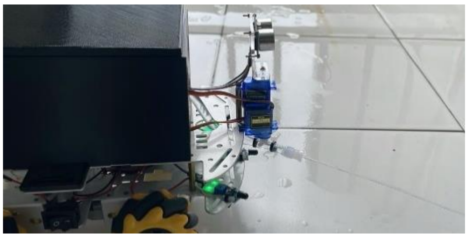
| Flame detection side | Motor 1 and 2 Movement | Motor 3 and 4 Movement |
|---|---|---|
| Flame detected at centre | No movement | No movement |



```
Ultrasonic Distance: 39
Flame Sensor 1 (Left): 1
Flame Sensor 2 (Center): 0
Flame Sensor 3 (Right): 1
Flame detected in the center!
Extinguishing flame...
No flame detected for 2 seconds, turning off pump and resetting servo.
```

When the flame is detected at the centre, the robot will immediately stop. Once it stop moving, the relay connected to water pump will be activated . The servo motor holding the hose will start to oscilate 30 degrees to 150 degrees. The relay and the servo will turn off after 2 seconds of no flame detected.

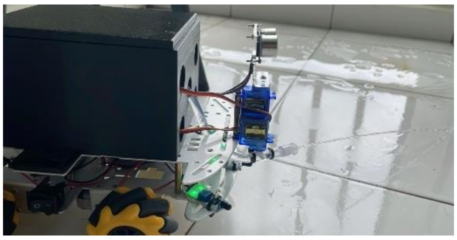
| Flame detection side | Motor 1 and 2 Movement | Motor 3 and 4 Movement |
|---|---|---|
| Right flame sensor activated | The motor 1 and 2 which is connected at channel 1 will rototate CCW | The motor 2 and 3 which is connected at channel 1 will rototate CW |

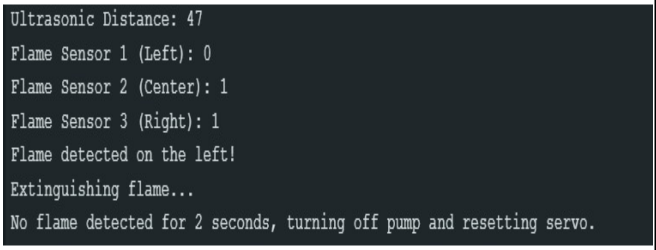


```
Ultrasonic Distance: 28
Flame Sensor 1 (Left): 1
Flame Sensor 2 (Center): 1
Flame Sensor 3 (Right): 0
Flame detected on the right!
Extinguishing flame...
No flame detected for 2 seconds, turning off pump and resetting servo.
```

When the flame is detected at right, the robot turn slightly to the right and stop. Once it stop moving, the relay connected to water pump will be activated . The servo motor holding the hose will start to oscilate 30 degrees to 150 degrees. The relay and the servo will turn off after 2 seconds of no flame detected.

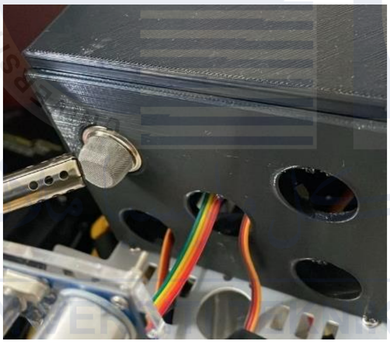
| Flame detection side | Motor 1 and 2 Movement | Motor 3 and 4 Movement |
|---|---|---|
| Left flame sensor activated | The motor 1 and 2 which is connected at channel 1 will rototate W | The motor 4 and 3 which is connected at channel 1 will rototate CCW |

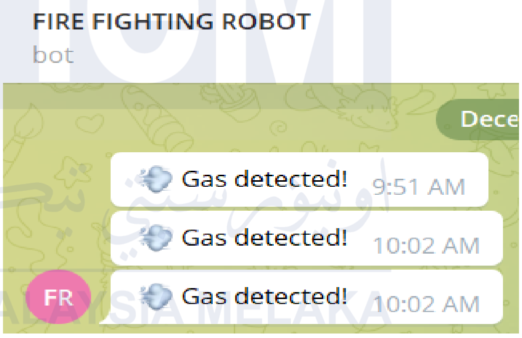




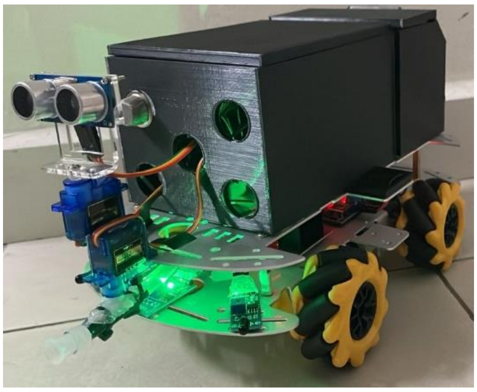

When the flame is detected at right, the robot turn slightly to the left and stop. Once it stop moving, the relay connected to water pump will be activated . The servo motor holding the hose will start to oscilate 30 degrees to 150 degrees. The relay and the servo will turn off after 2 seconds of no flame detected.

### 4.2.5 The Telegram Contol and notification






Eachtime the gas detected the Esp 32 will sent notification through telgram bot




`

The robot will stop moving once the command “STOP” sent and the robot wont move until the command “RUN” sent.This features is for emergency situation so the robot can be stoped and make run any time.

### 4.3 Result and Discussion

It is important to construct a comparison table to study the difference between this project and other existing or previous works. This makes interpreting ideas easier and allows clear comparison.

Table 4.3 Comparison of current project and existing project

| Feature | IoT-Based Firefighting Robot | Previous Projects |
|---|---|---|
| Sensors Used | Flame sensor, ultrasonic sensor, gas sensor | Flame sensors only or limited to infrared detection |
| Communication | IoT with Telegram notifications | Basic GSM or no remote communication |
| Mobility | Mecanum wheels for 360-degree movement | Traditional wheels or tracks with limited maneuverability |
| Power Source | Rechargable battery | Diesel engine or basic rechargeable batteries |
| Fire Extinguishing | Water pump with servo-controlled nozzle | Basic extinguishing without precise aiming mechanisms |
| Cost Effectiveness | Low-cost modular components | Higher cost due to specialized or proprietary designs |
| Applications | Indoor and small-scale environments | Limited to industrial or specific outdoor applications |
| Notifications | Real-time alerts via Telegram | Basic alerts (e.g., GSM SMS) or manual feedback systems |

There are also a few key points that can be highlighted to improve the project for future works. It should be noted that this discussion is important to understand the pros and cons of this project.

1. **Practical Applications**

The firefighting robot that uses IoT technology has many real-world uses. It can be utilized in small industrial environments, homes, and schools where rapid response is essential to halting the spread of fire. The robot's capacity to maneuver in narrow areas with Mecanum wheels makes it suitable for places such as warehouses, where traditional firefighting techniques might find it difficult to function properly.

Its connection with IoT and immediate alerts allows for remote observation, an essential capability for regions susceptible to dangerous gases. For instance, industrial plants handling flammable substances can greatly benefit from the robot's ability to detect LPG. Furthermore, the system's modular and compact structure renders it a cost-effective option for implementation in areas with limited resources, like rural fire stations or small business locations. Through the automation of fire detection and suppression, the robot diminishes dependence on human firefighters in hazardous settings, improving overall safety and effectiveness.

2. **Effectiveness of the Robot**

The firefighting robot is highly efficient due to its use of advanced sensors, modular components, and IoT connection. Its fire detection technology, with an water-spraying method, allows for quick and effective fire suppression while reducing water usage. The ultrasonic sensors provide reliable obstacle detection, enabling smooth navigation even in busy situations. Furthermore, Telegram's instant messaging ensures that users receive notifications in a short amount of time, allowing them to monitor actions from a distance and act as required.

## 3. Limitations and Challenges

Though efficient, the firefighting robot presents various drawbacks. Its dependence on battery power limits its operational time, which could be inadequate for extended firefighting situations. The capacity of the water tank is another limitation, restricting the robot's capacity to handle bigger fires. Furthermore, its efficacy is dependent on consistent IoT connectivity; interruptions in network access may impede real-time alerts and remote management.

The design of the robot is tailored for compact indoor settings, restricting its use in extensive industrial blazes or outdoor situations. Difficulties in severe environments, like intense heat or thick smoke, may also affect the sensor functionalities and movement. Subsequent versions ought to tackle these shortcomings by incorporating more advanced sensors, bigger power sources, and self-governing decision-making algorithms to enhance reliability and efficiency. The robot has a flame detection accuracy limitation and the the robot hard to have a precise navigation towards the flame. The excessive weight of the robot effectes the mobility and efficiency so it is hard to rotate accordingly. IN the coding parts have some logic issues, programming flaws affected the robot's performance and decision-making. The 3d printed tank falws and resulted in tank leaks.

## 4. Environmental and Economic Impacts

From an environmental viewpoint, the firefighting robot has a beneficial impact by minimizing water wastage thanks to its accurate extinguishing system. Its capacity to identify LPG leaks promptly also reduces the chances of serious environmental catastrophes. The integration of a rechargeable battery improves its environmental sustainability; however, its lifecycle effects could be improved by using recyclable materials.

From an economic perspective, the robot's inexpensive parts render it a viable option for extensive use, particularly in areas with limited resources. Automating fire detection and suppression decreases the dependence on human labor, which could result in lower operational expenses for firefighting organizations. Nonetheless, the costs associated with initial setup and ongoing maintenance might pose a challenge for smaller institutions.

## 4.4 Graphical Analysis

Table 4.4 Analysis of Graph of Distance vs Time Taken to Avoid Obstacle

| DISTANCE (cm) | TIME TAKEN TO AVOID OBSTACLE (s) |
|---|---|
| 10 | 4 |
| 15 | 3.8 |
| 20 | 3.5 |
| 25 | 3 |
| 30 | 2.9 |
| 35 | 2.5 |
| 40 | 2 |

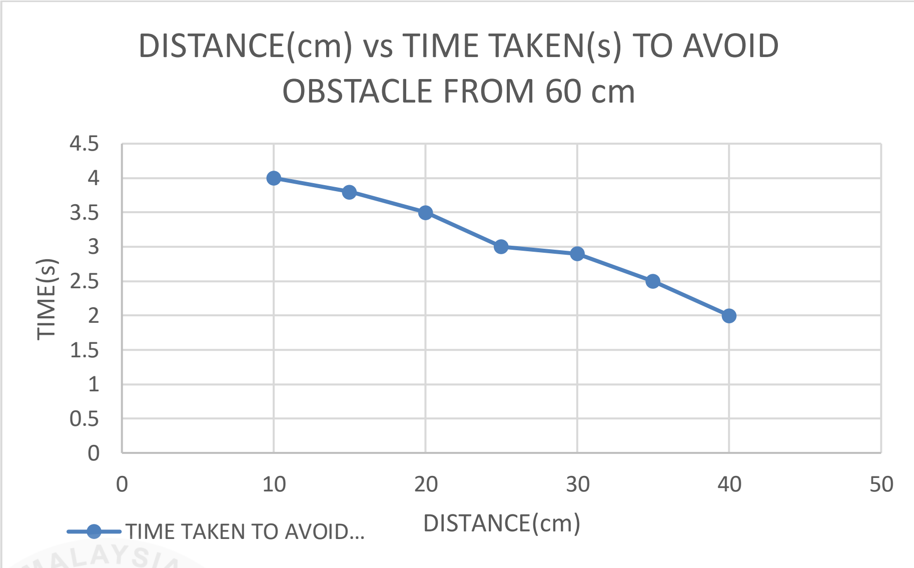



Figure 4.3 : Graph of Distance vs Time Taken to Avoid Obstacle

Based on the above graph, it can be observed that the firefighting robot's response time decreases as the obstacle distance increases. At shorter distances, the robot might need more time due to precise calculations for safe navigation and sensor delays. At longer distances, the robot has more reaction time, leading to quicker decisions. Variations could occur from sensor accuracy, motor response speed, or environmental factors.



Table 4.5 Analysis of Graph of Distance Set vs Distance Detect Obstacle

| **Distance Set (cm)** | **Distance Detect Obstacle (cm)** |
|---|---|
| 10 | 0 |
| 15 | 2 |
| 20 | 5 |
| 25 | 9 |
| 30 | 12 |
| 35 | 14 |
| 40 | 16 |

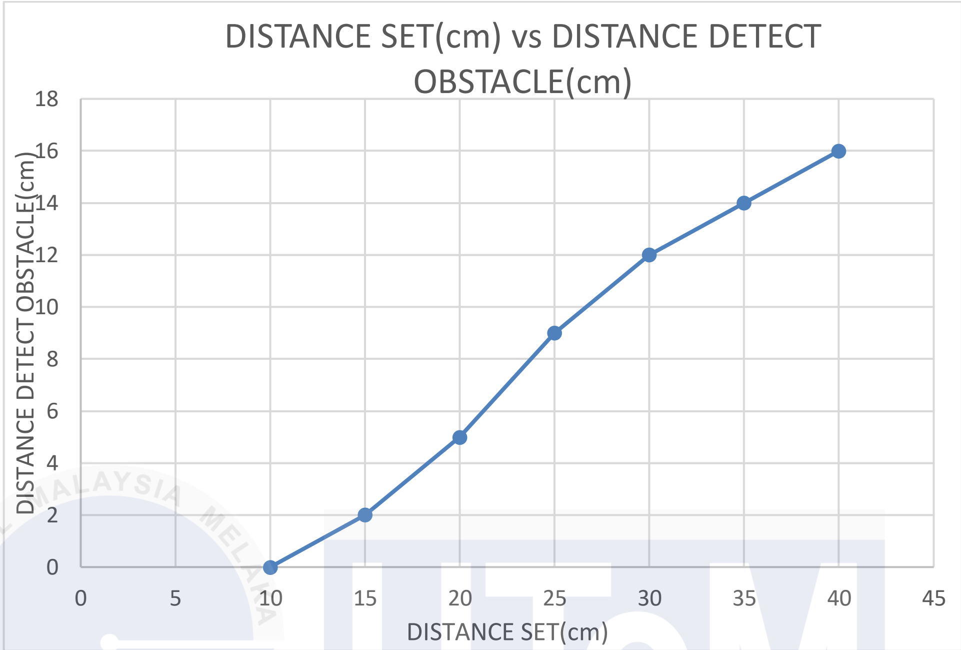



Figure 4.4 Graph of Distance Set vs Distance Detect Obstacle

The graph demonstrates the connection between the prescribed distance for flame detection and the distance that the sensor detects. As the established distance expands, the measured distance also elevates but stays steadily lower. This difference arises from sensor constraints, including decreased sensitivity at closer ranges, environmental influences like light disturbance or smoke, and possible calibration problems. Furthermore, the reduction of signal strength with distance impacts detection precision. These results highlight the importance of accurate sensor calibration and placement to improve the fire-fighting robot's capacity to identify and react to flames efficiently.

Table 4.6 Analysis of Graph Time Taken vs Amout Water Spray

| Time taken (s) | Amount water spray (ml) |
|---|---|
| 2 | 9 |
| 4 | 13 |
| 6 | 19 |
| 8 | 24 |
| 10 | 30 |

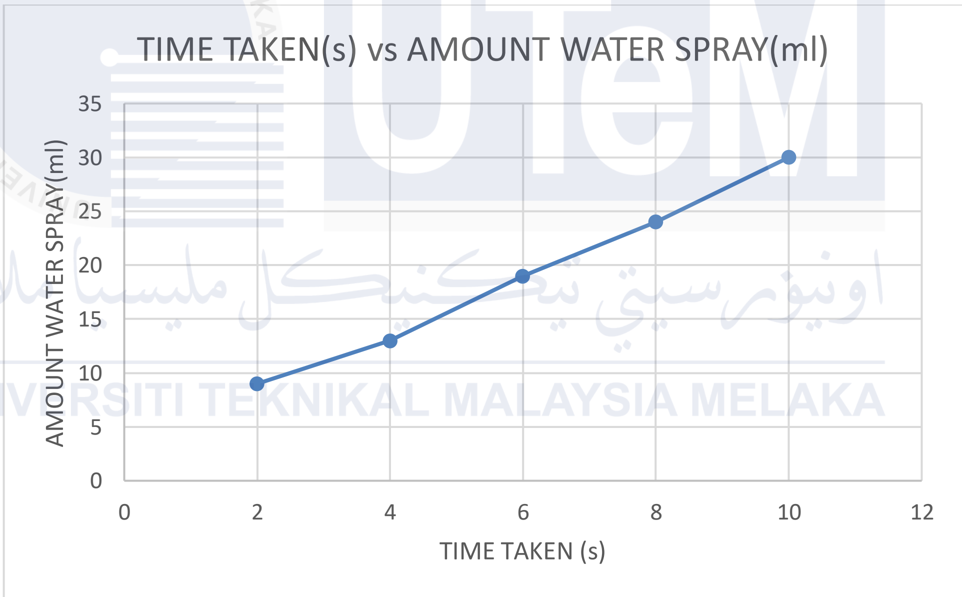



Figure 4.5 : Graph of Time Taken vs Amout Water Spray

The data indicates a gradual rise in the volume of spray released as the duration extends, demonstrating the reliable operation of the pump system. This pattern arises as the pump functions at a steady flow rate, signifying that the amount of water sprayed correlates directly with the duration it operates.

The small differences in the increase (for instance, 9 ml to 13 ml and 13 ml to 19 ml) may be attributed to slight changes in pump pressure or environmental influences such as water resistance in the hose. In general, the linear trend verifies the pump's effectiveness and dependability in supplying water, essential for allowing the fire-fighting robot to effectively put out fires within a certain timeframe.

## 4.5 Preliminary prototype design

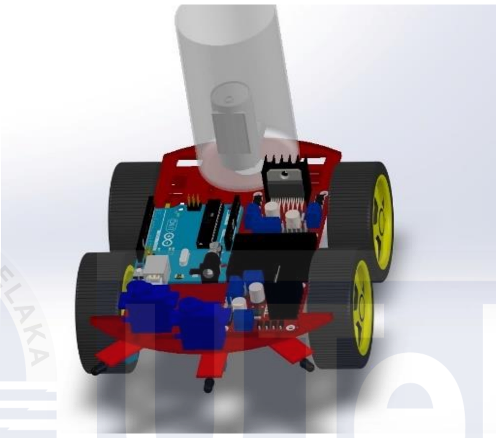


Figure 4.6: Fire Fighting robot prototype

Figure 4.6 shows the preliminary design concept of the prototype, which has been sketched and developed using SolidWorks. This initial design serves as a foundational blueprint, showcasing the structural layout and intended functionality of the prototype. By utilising SolidWorks, the design process has been streamlined, allowing for precise detailing, accurate dimensional representation, and the integration of various components essential for the prototype's operation.

## 4.6 Summary

In the conclusion of this chapter, we can state that the prototype can run itself. The flame sensor showed how good it was at finding flames. The ultrasonic sensor shows the range of detection. Adding emergencies through the Telegram app made it easier to keep an eye on things right away and improve emergency methods. Overall, the prototype's successful development is a big step forward in improving safety, and future versions of the robot could be even better and more creative.

# CHAPTER 5

# CONCLUSION AND RECOMMENDATIONS

## 5.1 Conclusion

This thesis paper talks about how to make a rescue robot that can connect to the Internet of Things (IoT) using a microcontroller. This is a big step towards making it easy to help people in fire emergencies. The robot was very good at finding fires, getting around barriers, and putting out fires by using different kinds of sensors, microcontrollers, and control through the Telegram app. The project's goals were met, which shows how important it is to use cutting-edge technology to put out current fires. The firefighting robot is a smart and useful way to lower the risk of fire in dangerous areas, thanks to its innovative systems and real-time tracking tools.

It also talks about how companies, universities, and the government can work together to make cutting-edge firefighting technologies better. Stakeholders can help fire emergency reactions go faster and last longer by getting people from different fields to share their knowledge and work together. It is best to keep improving and tweaking the firefighter robot's design and how it works so that it works better and can be used in more situations. It's also important for ongoing research and development to focus on solving new problems, like making sensors more accurate, making navigation systems work better, and adding extra features to make people more aware of their surroundings.

## 5.2 Future Works

In this modern world, technology is moving towards exciting new developments and ongoing growth in the future of firefighting robots. New improvements are coming that will try to get around problems and add new features that will make firefighting robots much better.

1. Expansion of Functionalities

To make the firefighting robot more capable, we can add a variety of sensors. For better obstacle avoidance, IR sensors or more ultrasonic sensors could be used, while an ESP32 camera would help detect flames more accurately. Adding gas filters to the sensors will make gas detection more precise and avoid false alarms. To handle tough environments, like high heat or thick smoke, the sensors should be upgraded. A bigger battery would keep the robot running longer, and LIDAR sensors could help it avoid obstacles more smoothly. A remote-control system could also be included, giving us manual control in emergencies.



2. AI and machine learning

Artificial intelligence and machine learning could really take the robot's abilities to the next level. With smarter algorithms, the robot could avoid obstacles more accurately and adapt to its surroundings. Machine learning could also make the robot better at detecting and moving toward flames, ensuring it acts quickly and efficiently. By making the robot capable of autonomous decisions, we could improve how it handles unpredictable situations.

3. Nozzle and Water Tank Connection

To make the robot lighter and easier to move, we could use lighter materials for the water tank and nozzle. At the same time, redesigning the tank to hold more water while preventing leaks would help the robot tackle fires for longer periods without needing a refill.

4. Heat-Resistant Materials

The robot needs to be able to handle extreme heat without breaking down. Using materials that can resist high temperatures would keep it sturdy and reliable, even in intense conditions.

5. Using compressed liquid carbon dioxide

Switching to a fire suppression system that uses compressed liquid carbon dioxide would make the robot more versatile. This type of system is great for putting out electrical fires and handling flames that water might not extinguish effectively.

6. Use tanker type Robot

Redesigning the robot to have a tanker-like structure would make it more stable and capable of carrying larger amounts of water or fire-suppressing materials. This would allow it to handle bigger fires or operate for longer without needing to reload.

In conclusion, upgrading the firefighting robot with smarter features and better materials will make it much more effective and dependable. Adding advanced sensors like LIDAR and ESP32 cameras will improve how it detects obstacles and flames, while machine learning can help it navigate and act more intelligently in different situations. Using lighter, heat-resistant materials and a tanker-style design will make the robot easier to move and more durable in tough conditions. Switching to compressed liquid carbon dioxide for fire suppression will also make it more versatile. With these upgrades, the robot will be better prepared to handle challenging firefighting tasks and become a valuable tool in emergencies.

# REFERENCES


[1] Radha, D., Arun Kumar, M., Telagam, N., & Sabarimuthu, M. (2021). Smart Sensor Network-Based Autonomous Fire Extinguish Robot Using IoT. International Journal of Online and Biomedical Engineering (iJOE), 17(01), pp. 101–110. https://doi.org/10.3991/ijoe.v17i01.19209

[2] Khan, T. (2023). A Smart Fire Detector IoT System with Extinguisher Class Recommendation Using Deep Learning. IoT, 4(4), 558–581. https://doi.org/10.3390/iot4040024I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[3] Binti, A., Majid, A., Tun, U., & Onn Malaysia, H. (n.d.). AUTONOMOUS FIRE PROTECTION ROBOT WITH NOTIFICATION. https://core.ac.uk/download/pdf/20488737.pdfR. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[4] IJRASET. (n.d.). Fire Fighting Robot. Www.ijraset.com. https://www.ijraset.com/research-paper/fire-fighting-robotM. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[5] Scholarworks@cwu, S., & Alhajri, M. (n.d.). Fire Fighter Robot Fire Fighter Robot. https://digitalcommons.cwu.edu/cgi/viewcontent.cgi?article=1135&context=undergradproj

[6] Development of a Compact Firefighting Robot for Hazardous Environments | PDF | Arduino | Electric Motor. (n.d.). Scribd. https://www.scribd.com/document/519209623/FIRE-FIGHTING-ROBOT-mini-project

[7] Latiff, A., & Mohammad, Z. (n.d.). THE DEVELOPMENT OF FIRE DETECTION AND AUTOMATED FIRE EXTINGUISHER SYSTEM BY USING ARDUINO AND NODEMCU ESP 8266. In Malaysian Journal of Industrial Technology. https://mitec.unikl.edu.my/mjit/13.%20Volume%205%20-%20Issue%205/2.%20THE%20DEVELOPMENT%20OF%20FIRE%20DETECTION%20AND%20AUTOMATED%20FIRE%20EXTINGUISHER%20SYSTEM%20BY%20USING%20ARDUINO%20AND%20NODEMCU%20ESP%208266.pdf

[8] RAFAT SHAMS, S. H. (2014, January). AUTOMATED FIRE EXTINGUISHING SYSTEM WITH GSM. BRAC University, Department of Electrical and Electronic Engineering. Dhaka: CORE. Retrieved from https://core.ac.uk/download/pdf/61804219.pdf.

[9] Agbeyangi, A., Alashiri, O., Odiete, J., & Adenekan, O. (2020). An Autonomous Obstacle Avoidance Robot Using Ultrasonic Sensor. Journal of Computer Science and Its Application, 27(1). https://doi.org/10.4314/jcsia.v27i1.15

[10] Sarjana, N., Sholihin, N., Adewasti, N., & Hesti, E. (2021). Using night vision camera robot based on internet of thing. Atlantis Highlights in Engineering/Atlantis Highlights in Engineering. https://doi.org/10.2991/ahe.k.210205.082

[11] Chee, F., Tan, Tan, C., Liew, S., Alkahari, M., Ranjit, S., Said, M., Chen, W., Rauterberg, G., & Sivakumar, D. (2013). Fire Fighting Mobile Robot: State of the Art and Recent Development. Australian Journal of Basic and Applied Sciences, 7(10), 220–230. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=1719ff4a3b06ccfa708752486b22f915178a58ac

[12] Engineer, B., May, 06, & 2021. (n.d.). Fire Safety Regulations: A Comparative Study of Malaysia and England and Wales. Www.buildingengineer.org.uk.

https://www.buildingengineer.org.uk/intelligence/fire-safety-regulations-comparative-study-malaysia-and-england-and-wales

[13] Department of Statistics Malaysia Official Portal. (n.d.). V1.Dosm.gov.my. https://v1.dosm.gov.my/v1/index.php?r=column/cthemeByCat&cat=492&bul_id=MkRoQ2IyZ0JkdElIZ0JidUhpaWxydz09&menu_id=WjJGK0Z5bTk1ZElVT09yUW1tRG41Zz09

[14] Malaysia, G. of. (n.d.). Fire and Rescue | data.gov.my. Data.gov.my. https://data.gov.my/dashboard/fire-and-rescue

[15] Saharul, M., & Rahim, A. (2015). The Current Trends and Challenging Situations of Fire Incident Statistics. Malaysian Journal of Forensic Sciences, 6(1), 63. http://forensics.org.my/mjofs/pdf/fssmVol.6No.1/Article%2009.pdf

[16] The CTIF World Fire Statistics Report № 27 now available for download. (2023, October 11). The CTIF World Fire Statistics Report № 27 Now Available for Download | CTIF - International Association of Fire Services for Safer Citizens through Skilled Firefighters. https://ctif.org/news/ctif-world-fire-statistics-report-no-27-now-available-download

[17] Jindal, K., Wang, A., Thakur, D., Zhou, A., Spurny, V., Walter, V., ... & Loianno, G. (2021). Design and Deployment of an Autonomous Unmanned Ground Vehicle for Urban Firefighting Scenarios. *arXiv preprint arXiv:2107.03582*.

[18] Jadhav, S. S., Bhosale, S. S., Shelar, V. G., Patil, P. B., & Jamadar, S. S. (2024). Fire Fighter Robot Using IoT and Mobile Application. *International Journal of Research in Science & Engineering*, 4(6), 49–57.

[19] Fan, J. (2025, January 11). Nvidia Scientist Says AI Robots Will Fight Wildfires Someday. *Barron's*.

[20] Diputación Foral de Bizkaia. (2024, October 18). Los bomberos de Bizkaia adquieren el robot Colossus, similar al que ayudó en el incendio de Notre Dame. *Cadena SER*.

[21] Sharma, A., Verma, R., Gupta, S., & Bhatia, S. K. (2014). Android Phone Controlled Robot Using Bluetooth. *International Journal of Electronic and Electrical Engineering*, 7(5), 443–448.

[22] Aung, P. W., & Win, W. Y. (2014). Remote Controlled Fire Fighting Robot. *International Journal of Scientific Engineering and Technology Research*, 3(24).

[23] John, A. J., Ashik, K., Vishnu, A., Fahmi, P., & Henna, P. (2016). Automatic Fire Extinguishing Robotic Vehicle. *International Journal of Scientific & Engineering Research*, 7(4).

[24] Malik, R. (2012). Fire Fighting Robot: An Approach. *Indian Streams Research Journal*, 2(II), 1–4.

[25] Kosasih, K., Sartika, E. M., Hasugian, M. J., & Muliady, M. (2010). The Intelligent Fire Fighting Tank Robot. *Journal of Robotics and Control*, 1(1), 1–6.



[26] Singh, H. P., Mahajan, A., Nagarajan, S., & Vashisht, A. (2015). Control of an Autonomous Industrial Fire Fighting Mobile Robot. *International Journal of Advanced Robotic Systems*, 12(1), 1–10.

[27] Tan, C. F., Singh, R. S., Kher, V. K., & Kong, H. F. (2013). Up-Scaling and Optimisation of Fire Fighting Ground Vehicle Track System. *Applied Mechanics and Materials*, 372, 3–7.

[28] Chattunyakit, S., Kondo, T., & Nilkhamhang, I. (2013). Development of Robotic Platform for Swarm Robots in Fire Detection Application. *International Journal of Advanced Robotic Systems*, 10(1), 1–10.

[29] Sonsale, P., Gawas, R., Pise, S., & Kaldate, A. (2014). Intelligent Fire Extinguisher System. *International Journal of Engineering Research and Applications*, 4(1), 1–5.

[30] Kumar, R., & Kumar, P. (2019). Internet of Robotic Things Based Autonomous Fire Fighting Mobile Robot. *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 495–499.

[31] Islam, M. S., Hossain, M. A., & Hossain, M. S. (2022). Design and Implementation of a Multiple Sensors Based Affordable Autonomous Firefighting Robot for Fire Management. *2022 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, 1–6. https://doi.org/10.1109/ICREST53941.2022.9714145

[32] Hossain, M. A., Islam, M. S., & Hossain, M. S. (2020). Design and Implementation of a Safety Fire Fighter (SAFF) Robot with Dual Controlling Mechanism. *2020 IEEE Region 10 Symposium (TENSYMP)*, 1–6. https://doi.org/10.1109/TENSYMP50017.2020.9230709

[33] Lee, J., Jung, S., & Kwon, T. (2013). Sensor Fusion Based Seek-and-Find Fire Algorithm for Intelligent Firefighting Robot. *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 1–6. https://doi.org/10.1109/SSRR.2013.6584304

[34] Kumar, S. S., Kumar, A. S., & Kumar, S. S. (2019). Design and Advancement of Firefighting Robot using Direction Control Model. *2019 International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*, 1–5. https://doi.org/10.1109/ICSTSN.2019.8862516

[35] Boston Consulting Group. (2019). *Advanced Robotics in the Factory of the Future*. Retrieved from https://www.bcg.com/publications/2019/advanced-robotics-factory-future

# APPENDICES

APPENDIX A Gantt Chart BDP 1

| No. | Task | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 | W14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | • Project proposal<br>• Project approval | | | | | | | | | | | | | | |
| 2 | • Discuss and improve project idea<br>• Writing project report chapter 1 | | | | | | | | | | | | | | |
| 3 | • Chapter 1 project report correction- scope and problem statement<br>• Make draft for project report chapter 2<br>• Research on methods of project | | | | | | | | | | | | | | |
| 4 | • Writing Project report chapter 2<br>• Presentation of work progress | | | | | | | | | | | | | | |
| 5 | • Research on hardware and software to be used | | | | | | | | | | | | | | |
| 6 | • Chapter 2 project report correction | | | | | | | | | | | | | | |
| 7 | • Writing Project report chapter 3 | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 8 | • PSM 1 report writing | | | | | | | | | | | | | | |
| 9 | • PSM 1 draft report submission to supervisor | | | | | | | | | | | | | | |
| 10 | • Presentation / Submission | | | | | | | | | | | | | | |

APPENDIX B Gantt Chart BDP 2

| No. | Task | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 | W14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | • BDP 2 briefing<br>• List down compenent to buy | | | | | | | | | | | | | | |
| 2 | • Do claim submission<br>• Test all the sensors and components | | | | | | | | | | | | | | |
| 3 | • Test hardware connection and coding<br>• Test the coding for unltasonic and flame sensor | | | | | | | | | | | | | | |
| 4 | • Adjust the coding<br>• Do connection with relay and servo motor | | | | | | | | | | | | | | |
| 5 | • Add telegram bot and test the coding with gas sensor | | | | | | | | | | | | | | |
| 6 | • Do system design and test the prototype with full working coding | | | | | | | | | | | | | | |
| 7 | • Draw design in Solid Works<br>• 3D print the model | | | | | | | | | | | | | | |
| 8 | • Methodology, result and discussion writing<br>• Draw flow chart and | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | presentation poster | | | | | | | | | | | | | | |
| 9 | • PSM 2 draft report submission to supervisor<br>• Mock presentation with supervisor | | | | | | | | | | | | | | |
| 10 | • PSM 2 final report and poster submission<br>• PSM 2 Panel presentation | | | | | | | | | | | | | | |
| 11 | • Psm 2 IIDEX2025 presentation | | | | | | | | | | | | | | |

APPENDIX C ESP 32 CODING

```
#include <WiFi.h>

#include <UniversalTelegramBot.h>

#include <WiFiClientSecure.h>


// Wi-Fi credentials

const char* ssid = "iPhone";

const char* password = "0112345678901";


// Telegram Bot credentials

const char* botToken =
"8161782950:AAGucwSm3pxV9B3ZagxWOu0n83K5YfuumE0";

const char* chatID = "759768472";


// UART pins for ESP32

#define RXD2 16 // ESP32 RX2 (connected to Arduino TX)

#define TXD2 17 // ESP32 TX2 (connected to Arduino RX)


// Gas sensor pin

#define GAS_SENSOR_PIN 25


WiFiClientSecure client;

UniversalTelegramBot bot(botToken, client);


// Gas detection state
```

```
int lastGasState = HIGH;
unsigned long gasLastNotification = 0;
bool robotRunning = true; // Tracks whether the robot is running

void setup() {
  Serial.begin(115200); // Debugging
  Serial2.begin(9600, SERIAL_8N1, RXD2, TXD2); // UART communication with Arduino

  // Initialize gas sensor pin
  pinMode(GAS_SENSOR_PIN, INPUT);

  // Connect to Wi-Fi
  Serial.print("Connecting to Wi-Fi");
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nConnected to Wi-Fi");
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());

  client.setInsecure();
  Serial.println("ESP32 Ready for Telegram Notifications");
```

```
}


void loop() {
  // Check for data from Arduino
  if (Serial2.available()) {
    String data = Serial2.readStringUntil('\n');
    Serial.println("Data received from Arduino: " + data);
    }
  }


  // Check for Telegram messages
  int numNewMessages = bot.getUpdates(bot.last_message_received + 1);
  while (numNewMessages) {
    handleTelegramMessages(numNewMessages);
    numNewMessages = bot.getUpdates(bot.last_message_received + 1);
  }


  // Gas sensor state
  int currentGasState = digitalRead(GAS_SENSOR_PIN);
  if (currentGasState == LOW && lastGasState == HIGH) {
    // Debounce gas notification
    if (millis() - gasLastNotification > 2000) {
      sendTelegramNotification("💨 Gas detected!");
      gasLastNotification = millis();
```

```
    }
   }
  lastGasState = currentGasState;


  delay(100); // Small delay for loop
}


void handleTelegramMessages(int numNewMessages) {
  for (int i = 0; i < numNewMessages; i++) {
    String chat_id = bot.messages[i].chat_id;
    String text = bot.messages[i].text;


    Serial.println("Telegram message received: " + text);


    if (chat_id == chatID) {
      if (text.equalsIgnoreCase("STOP")) {
        Serial.println("Stopping the robot...");
        robotRunning = false;
        Serial2.println("STOP"); // Notify Arduino to stop the robot
        sendTelegramNotification("🚫 Robot stopped.");
      } else if (text.equalsIgnoreCase("RUN")) {
        Serial.println("Resuming the robot...");
        robotRunning = true;
        Serial2.println("RUN"); // Notify Arduino to resume the robot
```

```
        sendTelegramNotification("▶️ Robot running.");
      }
    }
  }
}

void sendTelegramNotification(String message) {
  if (bot.sendMessage(chatID, message, "")) {
    Serial.println("✅ Telegram notification sent: " + message);
  } else {
    Serial.println("❌ Failed to send Telegram notification.");
  }
}
```

```
#include <Servo.h>
#include <NewPing.h>


// Pin definitions
#define ULTRASONIC_SERVO_PIN A0
#define ULTRASONIC_SENSOR_TRIG 11
#define ULTRASONIC_SENSOR_ECHO 12
#define MAX_DISTANCE 400
#define DISTANCE_TO_CHECK 30


// Motor pins (L298N Motor Driver)
int enableRightMotor = 5; // ENA
int rightMotorPin1 = 7;   // IN1
int rightMotorPin2 = 8;   // IN2


int enableLeftMotor = 6;  // ENB
int leftMotorPin1 = 9;    // IN3
int leftMotorPin2 = 10;   // IN4


// Flame sensor and relay pins
const int flameSensorPin1 = 2;   // Flame sensor 1 (left)
const int flameSensorPin2 = 3;   // Center flame sensor
const int flameSensorPin3 = 4;   // Flame sensor 3 (right)
const int relayPin = 13;         // Relay for water pump
```

```
// Flame-extinguishing servo pin
#define FLAME_SERVO_PIN A1

// Create objects for sensors and servos
NewPing ultrasonic(ULTRASONIC_SENSOR_TRIG, ULTRASONIC_SENSOR_ECHO, MAX_DISTANCE);
Servo ultrasonicServo;
Servo flameServo;

// Flags
bool extinguishing = false;
bool robotRunning = true; // Flag for robot's running state
unsigned long lastFlameDetectedTime = 0; // Track the last time flame was detected

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

  // Initialize motor pins
  pinMode(enableRightMotor, OUTPUT);
  pinMode(rightMotorPin1, OUTPUT);
  pinMode(rightMotorPin2, OUTPUT);
  pinMode(enableLeftMotor, OUTPUT);
  pinMode(leftMotorPin1, OUTPUT);
```

```
  pinMode(leftMotorPin2, OUTPUT);

  // Initialize flame sensors and relay
  pinMode(flameSensorPin1, INPUT);
  pinMode(flameSensorPin2, INPUT);
  pinMode(flameSensorPin3, INPUT);
  pinMode(relayPin, OUTPUT);

  // Attach servos
  ultrasonicServo.attach(ULTRASONIC_SERVO_PIN);
  flameServo.attach(FLAME_SERVO_PIN);

  // Initial positions
  ultrasonicServo.write(90); // Center
  flameServo.write(90);     // Neutral

  Serial.println("Robot Initialization Complete");
}

void loop() {
  // Check for commands from ESP32
  if (Serial.available()) {
    String command = Serial.readStringUntil('\n');
    command.trim(); // Remove extra spaces or newlines
```

```
    if (command == "STOP") {
      Serial.println("Robot stopped.");
      robotRunning = false;
      stopRobot(); // Stop all robot operations
    } else if (command == "RUN") {
      Serial.println("Robot running.");
      robotRunning = true;
    }
  }

  if (!robotRunning) {
    return; // If the robot is stopped, do nothing
  }

  unsigned long currentTime = millis();

  if (extinguishing) {
    // Turn off pump and servo if no flame detected for 2 seconds
    if (currentTime - lastFlameDetectedTime > 2000) {
      digitalWrite(relayPin, LOW); // Turn off pump
      flameServo.write(90);        // Reset servo to neutral position
      Serial.println("No flame detected for 2 seconds, turning off pump and resetting servo.");
      extinguishing = false;
    }
```

```
  return;
}

int distance = ultrasonic.ping_cm(); // Measure distance

// Flame sensor readings
int flameSignal1 = digitalRead(flameSensorPin1);
int flameSignal2 = digitalRead(flameSensorPin2);
int flameSignal3 = digitalRead(flameSensorPin3);

// Debugging Outputs
Serial.print("Ultrasonic Distance: ");
Serial.println(distance);
Serial.print("Flame Sensor 1 (Left): ");
Serial.println(flameSignal1);
Serial.print("Flame Sensor 2 (Center): ");
Serial.println(flameSignal2);
Serial.print("Flame Sensor 3 (Right): ");
Serial.println(flameSignal3);

// Flame Detection Logic
if (flameSignal1 == LOW) {
  Serial.println("Flame detected on the left!");
  handleFlameDetection(145, -145, 300); // Adjusted delay for left turn (300ms)
} else if (flameSignal2 == LOW) {
```

```
      Serial.println("Flame detected in the center!");
      handleFlameDetection(0, 0, 0); // Stop at the flame
    } else if (flameSignal3 == LOW) {
      Serial.println("Flame detected on the right!");
       handleFlameDetection(-145, 145, 300); // Adjusted delay for right turn
(300ms)
    }
    // Obstacle Avoidance
    else if (distance > 0 && distance < DISTANCE_TO_CHECK) {
      Serial.println("Obstacle detected!");
      handleObstacle();
    }
    // Default Behavior
    else {
      Serial.println("Moving Forward");
      rotateMotor(145, 145); // Increased speed for forward movement
    }
  }

  void stopRobot() {
    rotateMotor(0, 0); // Stop all motors
    digitalWrite(relayPin, LOW); // Turn off relay
    Serial.println("Robot is halted.");
  }
```

```
void handleFlameDetection(int rightMotorSpeed, int leftMotorSpeed, int turnDelay) {
  rotateMotor(rightMotorSpeed, leftMotorSpeed); // Move toward flame
  delay(turnDelay); // Fine-tuned delay for 90-degree turn
  rotateMotor(0, 0); // Stop at flame location

  extinguishing = true;
  digitalWrite(relayPin, HIGH); // Activate pump
  lastFlameDetectedTime = millis(); // Record the time flame was detected
  Serial.println("Extinguishing flame...");

  // Sweep flame-extinguishing servo
  while (millis() - lastFlameDetectedTime <= 2000) {
    for (int angle = 30; angle <= 150; angle += 2) {
      flameServo.write(angle);
      delay(10);
    }
    for (int angle = 150; angle >= 30; angle -= 2) {
      flameServo.write(angle);
      delay(10);
    }
  }
}

void handleObstacle() {
```

```
rotateMotor(0, 0); // Stop motors
delay(500);

// Scan left
ultrasonicServo.write(180);
delay(600);
int distanceLeft = ultrasonic.ping_cm();
if (distanceLeft == 0 || distanceLeft > MAX_DISTANCE) distanceLeft = MAX_DISTANCE;

// Scan right
ultrasonicServo.write(0);
delay(600);
int distanceRight = ultrasonic.ping_cm();
if (distanceRight == 0 || distanceRight > MAX_DISTANCE) distanceRight = MAX_DISTANCE;

// Scan center
ultrasonicServo.write(90);
delay(600);
int distanceCenter = ultrasonic.ping_cm();
if (distanceCenter == 0 || distanceCenter > MAX_DISTANCE) distanceCenter = MAX_DISTANCE;

// Decide direction
```

```
    if (distanceLeft <= DISTANCE_TO_CHECK && distanceRight <=
DISTANCE_TO_CHECK && distanceCenter <= DISTANCE_TO_CHECK) {
      Serial.println("All sides blocked. Moving backward.");
      rotateMotor(-145, -145); // Move backward
      delay(1000);          // Move back for 1 second
      Serial.println("Re-scanning...");
      return;               // Exit the function to allow re-scanning in the next loop
    } else if (distanceLeft > distanceRight) {
      Serial.println("Turning left...");
      rotateMotor(175, -175); // Left turn with equal speed at 175
      delay(600);          // Sharper turn delay
    } else {
      Serial.println("Turning right...");
      rotateMotor(-175, 175); // Right turn with equal speed at 175
      delay(600);          // Sharper turn delay
    }

  rotateMotor(145, 145); // Resume forward motion
}

void rotateMotor(int rightMotorSpeed, int leftMotorSpeed) {
  if (rightMotorSpeed < 0) {
    digitalWrite(rightMotorPin1, LOW);
    digitalWrite(rightMotorPin2, HIGH);
  } else if (rightMotorSpeed > 0) {
```

```
    digitalWrite(rightMotorPin1, HIGH);
    digitalWrite(rightMotorPin2, LOW);
  } else {
    digitalWrite(rightMotorPin1, LOW);
    digitalWrite(rightMotorPin2, LOW);
  }

  if (leftMotorSpeed < 0) {
    digitalWrite(leftMotorPin1, LOW);
    digitalWrite(leftMotorPin2, HIGH);
  } else if (leftMotorSpeed > 0) {
    digitalWrite(leftMotorPin1, HIGH);
    digitalWrite(leftMotorPin2, LOW);
  } else {
    digitalWrite(leftMotorPin1, LOW);
    digitalWrite(leftMotorPin2, LOW);
  }

  analogWrite(enableRightMotor, constrain(abs(rightMotorSpeed), 0, 255));
  analogWrite(enableLeftMotor, constrain(abs(leftMotorSpeed), 0, 255));
}
```

# 16% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

## Filtered from the Report

- Bibliography
- Quoted Text
- Crossref database
- Crossref posted content database

## Match Groups

**176** Not Cited or Quoted 16%
Matches with neither in-text citation nor quotation marks

**6** Missing Quotations 0%
Matches that are still very similar to source material

**0** Missing Citation 0%
Matches that have quotation marks, but no in-text citation

**0** Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

## Top Sources

10% Internet sources
2% Publications
14% Submitted works (Student Papers)

## Integrity Flags

**0 Integrity Flags for Review**

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

# 23% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

**Caution: Review required.**

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

## Detection Groups

**1** AI-generated only 21%
Likely AI-generated text from a large-language model.

**2** AI-generated text that was AI-paraphrased 3%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

**Disclaimer**
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

## Frequently Asked Questions

**How should I interpret Turnitin's AI writing percentage and false positives?**
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

**What does 'qualifying text' mean?**
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.