KWARA STATE POLYTECHNIC

CONSTRUCTION OF AUTONOMOUS ROBOTIC CAR USING LORA MODULE

BY

ABDULRAHMAN ABDULHAKEEM

ND/23/MCT/FT/0017

A PROJECT REPORT SUBMITTED TO THE DEPARTMENT OF MECHATRONICS ENGINEERING, INSTITUTE OF TECHNOLOGY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF NATIONAL DIPLOMA IN MECHATRONICS ENGINGINEERING TECHNOLOGY ILORIN, NIGERIA

JULY, 2025.

CERTIFICATION

The undersigned certify that this project report prepared by: **Abdulrahman Abdulhakeem, ND/23/MCT/FT/0017** Entitled: **Construction Of Autonomous Robotic Car Using Lora Module** meets the requirement of the Department of Mechatronics Engineering for the award National Diploma [ND] in Mechatronics Engineering, Kwara State Polytechnic, Ilorin.

______________________________ ____________

MR. MAHMUD MOHAMMED **Date**
(Project Supervisor)

______________________________ ____________

ENGR. RAJI Idowu Adebayo. **Date**
(Head of Department)

______________________________ ____________

External Supervisor's name
(External Examiner) **Date**

DECLARATION

I hereby declare that this research project titled **CONSTRUCTION OF AUTONOMOUS ROBOTIC CAR USING LORA MODULE** is my work and has not been submitted by any other person for any degree or qualification at any higher institution, I also declare that the information provided therein is mine and those that are not mine are properly acknowledged.

Abdulrahman Abdulhakeem **Signature and Date**

DEDICATION

This project is dedicated to **Mr. and Mrs. Abdulrahman,** my amazing parents, whose unwavering love, sacrifices, and prayers have shaped and supported me throughout this journey. Your endless encouragement and confidence in my abilities have been a powerful source of motivation and strength. Above all, I am deeply grateful to Almighty Allah, the Most Gracious and Most Merciful, for bestowing upon me the knowledge, patience, and determination to bring this project to completion

ACKNOWLEDGEMENT

All praise and thanks be to Almighty Allah, the Most Gracious and Most Merciful, for His endless guidance, mercy, and favor throughout the course of this academic journey.

I am especially grateful to my beloved parents, **MR. & MRS ABDULRAHMAN,** for their unwavering love, prayers, and sacrifices. Your moral and financial support has been the foundation of my strength, and I am forever thankful for your constant encouragement.

My sincere appreciation goes to my project supervisor, **MR. MAHMUD MOHAMMED**, for his guidance, patience, and valuable input. His dedication and support greatly contributed to the successful completion of this project.

I also wish to acknowledge the Head of Department, **ENGR. RAJI IDOWU ADEBAYO**, for his outstanding leadership and commitment to academic excellence within the Department of Mechatronics Engineering.

To all lecturers and staff members of the department, your knowledge and encouragement have been instrumental in my growth and success.

Finally, I thank everyone who contributed in one way or another to this achievement. May Almighty Allah reward you all abundantly. **Amee**

ABSTRACT

This project presents the design and implementation of an autonomous robotic car utilizing LoRa (Long Range) communication for remote telemetry and control. The system integrates sensor fusion (GPS, IMU, and ultrasonic sensors) to enable real-time obstacle avoidance and path planning in outdoor environments. A modular architecture separates perception, decision-making, and communication tasks across dedicated hardware layers: Raspberry Pi (high-level navigation), Arduino (real-time motor control), and Semtech SX1276 (LoRa-based connectivity). The study validates LoRa as a cost-effective solution for autonomous edge robotics, though limitations in high-speed navigation and rain-induced signal attenuation suggest future work in RTK-GNSS integration and adaptive LoRa spreading factors. This work provides a reproducible framework for low-power robotic systems in applications like precision agriculture or lastmile delivery.

TABLE OF CONTENTS

CHAPTER ONE

1.0 Introduction

The advancement of wireless communication and robotics has enabled the development of intelligent systems that can perform tasks autonomously. One significant application of these systems is in autonomous robotic vehicles, which can operate without continuous human input. These vehicles are particularly useful in areas such as agriculture, surveillance, delivery, and exploration.

To enable reliable long-range wireless control and data transmission, the integration of Long Range (LoRa) communication modules with microcontrollers like the ESP32 has become increasingly relevant. LoRa offers a low-power, long-range communication solution that is ideal for remote operations.

This project focuses on the design and construction of an autonomous robotic car that leverages LoRa technology for wireless communication. The goal is to enable the car to be controlled and monitored remotely with minimal latency and high efficiency. The system combines robotics, wireless communication, and embedded systems. It integrates motor control, sensor data processing, and LoRa-based communication to create a functional and efficient prototype.

1.1 Problem Statement

Traditional wireless robotic systems often rely on Bluetooth or Wi-Fi, which have limited range and are susceptible to signal disruptions. These limitations affect performance, especially in outdoor or large-area applications. There is, therefore, a need for a cost-effective, energy-efficient robotic system that uses a long-range communication method to ensure stability and reliability across diverse environments.

1.2 Aim

To design and construct an autonomous robotic car that is controlled via a LoRa communication module using the ESP32 microcontroller.

1.3 Objectives of the Study are:

- To build a mobile robotic platform using DC motors and wheels.
- To integrate the ESP32 microcontroller for control and processing.
- To implement LoRa communication for remote control and data transmission.
- To power the system using rechargeable batteries.
- To test and evaluate the performance of robotic cars.

1.4 Scope of the Study

This study is limited to the construction and testing of a basic autonomous robotic car using LoRa for wireless communication. It does not cover advanced functionalities such as image recognition, GPS navigation, or AI-driven decision-making. The project will focus strictly on foundational mobility, control logic, and long-range communication.

CHAPTER 2: LITERATURE REVIEW

2.1 Review of Fundamental Concepts.

This section provides an overview of the key concepts and foundational technologies relevant to the development of an autonomous robotic car using LoRa communication. It includes essential principles such as the operation of DC motors, the function of motor drivers like L298N, the architecture and capabilities of microcontrollers like the ESP32, and the importance of wireless communication systems. Additionally, it explores the integration of power supply systems, sensor inputs, and system synchronization necessary to achieve reliable autonomous navigation and remote control. Understanding these concepts lays the groundwork for effective design, implementation, and troubleshooting of the robotic system.

2.1.1 Autonomous Robotic Systems

An autonomous robotic system is a self-operating machine capable of performing tasks without continuous human guidance. These systems are designed to interpret sensory input, make decisions based on programmed algorithms, and execute physical actions through actuators like motors and wheels. They are widely applied in areas such as surveillance, logistics, agriculture, and smart mobility. These systems rely on a combination of hardware and software components, including sensors (such as ultrasonic, infrared, or GPS), embedded processors, and control algorithms that allow them to function with minimal human intervention. Over time, advancements in artificial intelligence (AI), machine learning, and embedded systems have significantly improved the decision-making capabilities of autonomous robots, enabling them to adapt to changing environments and perform complex tasks. Additionally, their ability to operate in hazardous or inaccessible environments makes them ideal for search and rescue missions, environmental monitoring, and military applications. The growing integration of autonomous robots into daily operations reflects their potential to increase efficiency, reduce human workload, and ensure safety in various domains.

2.1.2 DC Motors and Motor Drivers (L298N)

DC motors are commonly used in robotic applications to provide rotational motion. They are powered by direct current and can vary in speed and direction. To control these motors effectively, a motor driver such as the L298N is employed. The L298N is a dual H-bridge driver that enables bidirectional control of two DC motors. It acts as an interface between low-voltage control signals (from the microcontroller) and high-voltage power supplied to the motors. The ability to reverse polarity using the H-bridge configuration allows for precise control over the motor’s direction, making it especially useful in mobile robots that need to move forward, backward, or turn. Additionally, the L298N supports pulse-width modulation (PWM) for speed regulation, allowing for smooth acceleration and deceleration of the robot. This driver is widely appreciated for its ease of use, availability, and ability to handle moderate current loads. For robotic projects involving navigation or load-carrying, the L298N provides a reliable and cost-effective solution to motor control challenges. Integration with heat sinks and logic-level controls further enhances its performance in both simple and moderately complex robotic systems.

This diagram shows how two DC motors (3–12V) are connected to an L298N Motor Driver module.

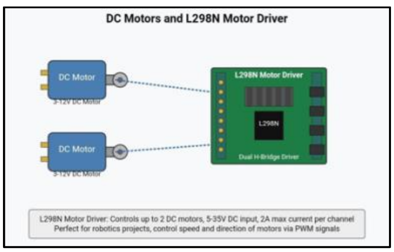

Figure 2.1: Connection of DC Motors to L298N Motor Driver

The ESP32 is a powerful, low-cost microcontroller with built-in Wi-Fi and Bluetooth capabilities. In this project, it acts as the brain of the robotic car, managing input signals, controlling actuators, and handling wireless communication. Its dual-core processor and ample GPIO (general-purpose input/output) pins make it ideal for embedded systems and IoT applications.

This diagram illustrates the key components of the ESP32 Development Board.

Figure 2.2: ESP32 Development Board Layout

2.1.4: LoRa (Long Range) Communication

LoRa is a wireless modulation technique based on chirp spread spectrum technology, designed for long-range, low-power communication. LoRa modules operate in the sub-GHz frequency bands (e.g., 433 MHz, 868 MHz, 915 MHz) and are suitable for sending small packets of data over distances up to 10 kilometers in open environments. LoRa is particularly advantageous in IoT and remote robotics where Wi-Fi or Bluetooth range is insufficient. In addition to its impressive range, LoRa excels in energy efficiency, making it ideal for battery-powered devices that require long operational life. Its ability to maintain connectivity over extended distances without the need for infrastructure like repeaters or cellular towers gives it a unique advantage in rural, agricultural, and disaster-prone areas. Furthermore, LoRa supports star topology networks through gateways and base stations, allowing multiple devices to communicate with a central hub. This is especially beneficial for large-scale sensor networks or distributed robotic systems. Despite its relatively low data rate, LoRa's robustness against interference and its long-range capabilities make it a highly reliable option for control and monitoring tasks in dynamic environments.

This diagram illustrates how LoRa (Long Range) communication enables low-power wireless data transmission between IoT devices and a central network server.

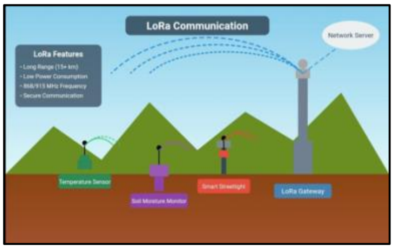

Figure 2.3: LoRa Communication in an IoT Ecosystem

2.1.5 Power Supply

The power supply system is critical in mobile robotics. In this project, multiple batteries, including a 12V and 3.7V setup, provide adequate voltage and current for the motor driver and control electronics. Efficient power distribution ensures stable operation without overheating or power loss. In mobile robotic systems, a reliable and well-regulated power source is essential for consistent performance across all components. The 12V battery typically powers the motors and drivers, which demand higher current for movement, while the 3.7V battery often supports microcontrollers and communication modules that require lower voltage. Voltage regulators or step-down converters are usually implemented to ensure the safe operation of sensitive electronics. Proper battery management, including protection circuits, charge indicators, and thermal control, further enhances system reliability. Additionally, separating the power sources for logic and actuation minimizes the risk of signal noise or power dips during sudden motor activity, contributing to smoother control and responsiveness.

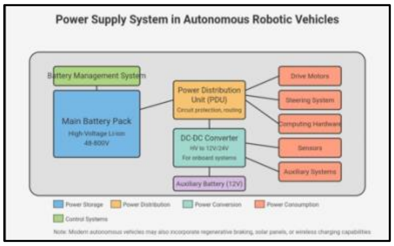

Figure 2.4: Power Supply System Architecture in Autonomous Robotic Vehicles.

2.1.6 Wireless Communication in Robotics

Wireless communication enables remote control and telemetry of robotic systems. Technologies like Bluetooth, Wi-Fi, Zigbee, and LoRa are compared based on their range, power consumption, and data rates. LoRa, though slower in data rate, outperforms others in range and power efficiency making it ideal for outdoor and long-distance control scenarios. Each wireless technology comes with trade-offs that influence its suitability for different robotic applications. Bluetooth, for instance, offers simple peer-to-peer connectivity with minimal setup, often used in hobbyist or educational robots. Wi-Fi allows for high-bandwidth data transfer, such as video streaming, but suffers from limited range and significant power demands. Zigbee provides a good middle ground with mesh networking capability, allowing multiple devices to relay data across moderate distances. LoRa stands out in applications requiring sparse data exchange over extended distances, such as in agricultural robotics or search-and-rescue missions. The choice of communication protocol depends heavily on the operational environment, mobility requirements, and power availability. Integrating the right wireless module enhances real-time responsiveness, improves data logging, and extends the operational reach of autonomous robotic platforms.

2.1.7 System Integration:

Bringing together all components: motors, microcontroller, driver, communication modules, and power is key to achieving system-level functionality. Integration involves careful circuit design, pin mapping, power management, and timing synchronization.

Successful system integration ensures that each hardware and software element functions harmoniously within the robotic platform. This includes aligning logic levels between components, ensuring sufficient power delivery under various load conditions, and implementing signal filtering to reduce electrical noise. Proper cable management and layout planning also help avoid crosstalk and overheating, which can cause system instability. On the software side, integration includes writing firmware that allows different modules to communicate efficiently, coordinating tasks such as motor actuation, sensor reading, and wireless data exchange. Debugging tools like serial monitors and logic analyzers are often employed during integration to verify performance and troubleshoot timing issues or communication delays. Ultimately, seamless integration transforms individual components into a unified, intelligent system capable of autonomous operation.

2.2 Review of Relevant Research.

Over the past decade, significant strides have been made in the field of autonomous robotics, particularly in the integration of wireless communication for remote operation and monitoring. This section reviews past studies and projects that inform the design and implementation of an autonomous robotic car using LoRa technology. Numerous academic publications, industry reports, and experimental projects have explored the fusion of autonomy with wireless data transmission, contributing valuable insights into both hardware and software considerations. Many of these studies emphasize the importance of reliable and energy-efficient communication for sustained robot performance, especially in outdoor or infrastructure-scarce environments. Researchers have experimented with different wireless protocols, comparing their impact on responsiveness, energy use, and control accuracy. Additionally, comparative analyses have revealed the limitations of conventional short-range technologies in rural or expansive terrains. further justifying the exploration of LoRa.

Furthermore, interdisciplinary approaches combining embedded systems, control theory, and IoT frameworks have produced scalable models for robotic systems. These have laid the groundwork for open-source platforms and DIY kits that encourage further innovation. While much of the research validates the feasibility of wireless robotic systems, there remains a need for more practical, real-world implementations. Particularly those focusing on affordability, portability, and long-range capability, which this project aims to address.

Wang et al. (2017) developed a Wi-Fi-based robotic car system capable of real-time video surveillance and remote navigation. While effective in close-range applications, the researchers noted that Wi-Fi's limited range and high-power constrained outdoor deployment, especially in rural or expansive environments. This limitation highlighted the need for more efficient and long-range communication technologies in robotics. Building on this foundation, subsequent studies have continued to explore alternative wireless methods that could overcome Wi-Fi's constraints. Researchers have also examined the potential of hybrid systems that combine different communication technologies for improved flexibility. These explorations further underscore the growing need for robotic platforms capable of stable operation beyond the range of conventional networks. As robotic applications extend into remote agriculture, disaster response, and environmental monitoring, the importance of long-distance control mechanisms has become increasingly evident. The findings of Wang et al. thus serve as an early yet critical pointer toward the development of systems leveraging low-power wide-area network (LPWAN) technologies like LoRa.

A notable study by Petäjäjärvi et al. (2017) explored the application of LoRa in agricultural robotics. The research demonstrated that LoRa offered a stable communication link over several kilometers in open fields, enabling remote control and data transmission without cellular infrastructure. Similarly, Adelantado et al. (2017) concluded that LoRa's low data rate was a trade-off for its exceptional range and energy efficiency, making it suitable for command-based communication in mobile robots. This body of research underscores the suitability of LoRa for tasks that prioritize reliability, coverage, and power conservation over bandwidth. By allowing devices to remain operational for extended periods on limited power, LoRa proves highly effective in scenarios where robots must operate autonomously over large, unmonitored areas. Additionally, the use of LoRa eliminates the dependency on local network infrastructure, which is often unavailable in rural or undeveloped regions. These characteristics make LoRa a compelling choice for applications such as precision farming, search-and-rescue robotics, and long-distance environmental monitoring, where consistent communication is critical, but data transmission volume remains relatively low.

Rana et al. (2020) implemented a smart vehicle system using the ESP32 microcontroller and LoRa modules. The system successfully transmitted GPS coordinates and received control commands via LoRa, enabling semi-autonomous navigation. The study highlighted ESP32's dual-core performance, integrated wireless support, and cost-effectiveness, all of which are critical for real-time robotic applications. Beyond these strengths, the ESP32 stands out due to its extensive I/O options, built-in analog-to-digital converters (ADC), and support for various communication protocols, including UART, SPI, and I2C. This versatility allows developers to integrate multiple sensors and peripherals, making the ESP32 highly adaptable to a wide range of robotic use cases—from simple obstacle-avoidance bots to more complex systems incorporating GPS, cameras, or machine learning models. Additionally, the availability of rich development resources and support within the open-source community has positioned ESP32 as a go-to platform for educational, hobbyist, and even industrial-grade robotics projects.

Khan et al. (2018) presented a robotic obstacle-avoiding car using the L298N motor driver. Their findings validated the reliability of the L298N module for precise motor direction and speed control, especially when interfaced with microcontrollers like Arduino and ESP32. The study supports the use of L298N in mobile robotic systems requiring stable motion control. Furthermore, the L298N's capability to drive two motors simultaneously using independent control channels makes it an ideal choice for differential drive systems commonly found in robotic vehicles. The driver supports PWM (Pulse Width Modulation) for fine-grained speed regulation, which is essential in applications demanding smooth acceleration or precise turning. Its built-in heat sink and support for external power supplies allow it to handle moderate current loads without thermal failure. Despite being an older component, its affordability, ease of use, and compatibility with many development platforms continue to make it a staple in both educational and functional robotic projects.

Lee and Lee (2010) evaluated Bluetooth-based robotic control systems, concluding that although Bluetooth is easy to implement, its limited range (~10 meters) restricts its use to indoor or short-range tasks. Likewise, Kumar and Sharma (2015) analyzed Wi-Fi and Zigbee for mobile robot communication and concluded that both face issues of power consumption and interference in dynamic environments. These findings underscore the importance of selecting communication technologies based on environmental and operational needs. Bluetooth, while cost-effective and widely supported, struggles with signal stability in environments with obstructions or multiple devices. Wi-Fi, despite offering higher data rates suitable for streaming and real-time applications, demands substantial energy and may suffer performance degradation in areas with multiple networks. Zigbee, though designed for low power and reliable mesh networking, presents challenges in terms of network complexity and limited transmission distance. These drawbacks limit their effectiveness for use in expansive outdoor robotics, where consistent, long-range communication is essential. Consequently, the exploration of alternatives like LoRa becomes increasingly valuable for overcoming these constraints.

Augustin et al. (2016) conducted a comparative analysis of LoRa and traditional wireless systems in IoT. Their study found that LoRa modules could maintain connectivity up to 10 km in rural areas, outperforming Wi-Fi and Bluetooth in terms of range, though at the cost of data transmission speed. For robotic systems transmitting simple control signals rather than complex data, LoRa was seen as highly effective. The study further highlighted that while Wi-Fi and Bluetooth excel in environments requiring high data throughput and low latency, their limited operational range and higher power consumption pose significant challenges in large-scale or remote deployments. LoRa’s superior energy efficiency and long-range capabilities make it particularly suitable for applications that prioritize extended communication distances over bandwidth, such as agricultural monitoring, environmental sensing, and autonomous vehicle control in vast outdoor spaces. Additionally, the robustness of LoRa's chirp spread spectrum modulation enhances signal reliability in the presence of interference and physical obstacles, which are common in real-world outdoor scenarios. These factors collectively position LoRa as a practical choice for long-distance, low-power robotic communications.

From these studies, LoRa offers a promising solution for long-range, low-power communication in autonomous robotic systems. The ESP32 microcontroller has also proven effective in managing control logic and wireless integration. Motor control through the L298N module remains a dependable standard for robotic car designs. The present project leverages these established technologies to construct a cost-effective, remotely operated autonomous robotic car suitable for outdoor environments. Moreover, the integration of these components highlights a balance between performance, energy efficiency, and affordability, making such systems accessible for both academic research and practical applications. The emphasis on low power consumption extends the operational time of mobile robots, which is crucial for field deployment where frequent recharging or battery replacement is impractical. This synthesis of technologies demonstrates the viability of deploying autonomous robotic cars in real-world settings, especially in rural or remote areas where conventional wireless communication infrastructure is limited or unavailable.

2.3 Existing Solutions and Technologies:

Various technologies and systems have been developed to address the challenges of remote robotic control, autonomous mobility, and long-range communication. This section explores existing tools and methods relevant to the construction of autonomous robotic cars and highlights their features, strengths, and limitations. Over the years, advancements in wireless communication, motor control, and microcontroller capabilities have paved the way for increasingly sophisticated autonomous vehicles. These solutions range from simple remote-controlled platforms to highly integrated systems featuring sensor fusion, AI-based navigation, and adaptive control strategies. The effectiveness of each technology depends largely on the intended application environment, desired operational range, power constraints, and cost considerations.

2.3.1 Wireless Communication Modules

Several wireless modules are available for robotic communication, each with different strengths based on range, power consumption, and data throughput. These modules form the backbone of remote control and telemetry systems, enabling robots to interact with their environment and operators over varying distances and conditions. Choosing the appropriate wireless technology depends on the specific needs of the robotic application, such as data rate requirements, power availability, environmental interference, and mobility constraints.

Short-range modules like Bluetooth are ideal for indoor or close-proximity communication but fall short when extended range or outdoor operation is required. Wi-Fi modules provide higher data throughput and are widely supported, making them suitable for real-time video and sensor data transmission, though their power demands and limited range can be restrictive. Zigbee modules offer low power consumption and support mesh networking, enhancing reliability across medium distances but with limited bandwidth.

2.3.2 Bluetooth Modules (e.g., HC-05)

Commonly used for short-range control in indoor environments. While easy to set up and low-cost, Bluetooth's effective range (5–10 meters) is insufficient for field-based or outdoor robotics applications. The simplicity of pairing and widespread device compatibility make Bluetooth a popular choice for beginner and prototype projects. However, its limited range and susceptibility to interference from obstacles such as walls and furniture restrict its use to scenarios where the robot remains within proximity to the controller. Additionally, Bluetooth's power consumption is moderate, which can be a concern for battery-operated robotic platforms needing extended operational times.

Despite these limitations, Bluetooth modules like the HC-05 are widely utilized in educational and hobbyist robotics due to their accessibility and ease of integration with microcontrollers such as Arduino and ESP32.

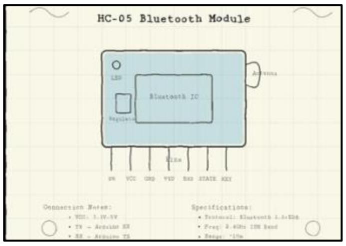

Figure 2.5: HC-05 Bluetooth Module Pin Configuration and Specifications

2.3.3 Wi-Fi Modules (e.g., ESP8266)

Offers higher data transfer rates suitable for real-time applications such as video streaming. However, Wi-Fi consumes significant power and has a limited range (typically <100 meters without amplification), making it less ideal for battery-powered, long-distance robotics. The widespread availability of Wi-Fi networks and compatibility with many devices enable seamless internet connectivity, which can support cloud-based control and data logging. Despite these advantages, Wi-Fi signals tend to suffer from interference and signal degradation when obstacles like walls or buildings are present, further restricting practical range in many environments. Additionally, the higher power consumption associated with Wi-Fi modules demands larger battery capacities or frequent recharging, posing challenges for mobile autonomous systems that require long operational times. Nonetheless, for indoor or campus-based robotic applications where higher bandwidth is essential, Wi-Fi remains a commonly adopted communication solution.

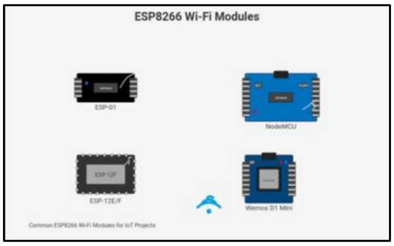

Figure 2.6: ESP8266 Wi-Fi Modules for IoT Projects

2.3.4 Zigbee Modules (e.g., XBee)

Zigbee is a mesh network protocol known for low power usage and reliability over medium distances. It supports networking multiple nodes but is limited in range compared to LoRa. The mesh networking capability allows devices to relay data through intermediate nodes, effectively extending coverage within complex environments such as buildings or industrial sites. This makes Zigbee ideal for applications requiring robust local area networks with many devices, such as home automation or sensor arrays. However, the typical range of Zigbee devices is approximately 10 to 100 meters, which restricts its usefulness for wide-area outdoor robotic control where long-distance communication is required. Additionally, Zigbee offers moderate data rates sufficient for sensor data transmission but may not support high-bandwidth applications. Despite its limitations in range, Zigbee's low latency, scalability, and interoperability with various microcontrollers make it a popular choice for localized wireless networks.

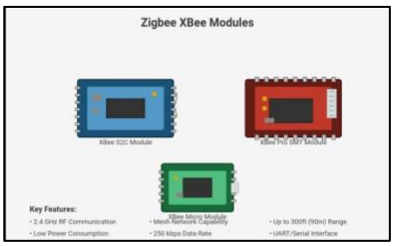

Figure 2.7: Zigbee XBee Modules and Key Features.

2.4 Identification of Gaps in Current Knowledge

While numerous studies and projects have explored autonomous robotic systems and wireless communication technologies, certain limitations and gaps remain that justify the need for this present study.

2.4.1 Limited Long-Range, Low-Power Solutions in Mobile Robotics.

Many existing robotic platforms rely on Wi-Fi or Bluetooth for communication, both of which are constrained by short-range operation and high-power consumption. Though LoRa has been applied in IoT and environmental monitoring, it's used in mobile robotic platforms. Especially for real-time control and long-distance operations that are still underexplored.

2.4.2 Lack of Integration Between ESP32 and LoRa in Low-Cost Robotics.

While the ESP32 microcontroller is popular in hobbyist and academic projects, few existing solutions fully utilize its potential when integrated with LoRa modules for robotic control. Most studies focus on either static sensing or basic wireless transmission, without implementing real-time control of a moving robotic platform in wide outdoor areas.

2.4.3 Inadequate Research on LoRa's Practical Performance in Dynamic Environments.

Though theoretical studies demonstrate LoRa's impressive range, practical implementations in dynamic robotic environments. Especially involving continuous movement, obstacles, and signal interference are not widely documented. There is a need to validate its effectiveness in real-world robotic use cases.

2.4.4 Fragmented Implementation and Lack of Unified System Designs.

Many projects focus on individual components such as obstacle detection, motor control, or remote communication, but few combine these into a cohesive system. There is a lack of holistic, affordable designs that integrate motor control, wireless communication, and autonomous operation into a single, working prototype.

2.4.5 Insufficient Focus on Educational and Scalable Applications.

Most commercial and research-based robotic systems are either too expensive or complex for educational purposes. There is a gap in creating simple, replicable, and scalable robotic designs that can be used by students and institutions to explore emerging technologies like LoRa in robotics.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Research Design

This study employs a prototyping-based research design to develop, test, and refine an autonomous robotic car equipped with LoRa communication for long-range remote control. Prototyping is particularly suited for projects that combine both hardware and software because it enables researchers to construct tangible models that evolve over time, allowing continuous improvement based on iterative testing and observation.

Unlike purely experimental designs, which often focus on testing hypotheses under controlled conditions, or descriptive designs, which focus on observation and reporting, the prototyping approach emphasizes building, integrating, and improving a physical system. This is crucial for this project, where the success of the system depends on how well multiple electronic, mechanical, and communication components work together in a real-world environment.

The research is structured around the following phases:

Phase 1: System Analysis and Requirement Specification

This involves identifying system goals (e.g., long-range communication, low power consumption, autonomous navigation), selecting suitable components (e.g., ESP32, L298N, LoRa module), and defining the necessary input/output requirements.

Phase 2: Hardware Prototyping and Assembly

This includes the physical construction of the robotic car: assembling the chassis, integrating DC motors with motor drivers, connecting the microcontroller, wiring power supplies, and installing communication modules. Care is taken to ensure that electrical connections are stable, components are securely mounted, and the system layout minimizes noise or interference.

Phase 3: Software Development and Integration

This phase focuses on developing embedded code for the ESP32 microcontroller, programming motor control routines, sensor reading functions, and LoRa communication protocols. The software is designed modularly to allow easy debugging and updates.

Phase 4: Iterative Testing and Refinement

As each subsystem is integrated, it undergoes testing under real-world conditions. Motor performance, communication reliability, power consumption, and system responsiveness are all assessed. Issues discovered during this phase inform redesigns or software modifications, ensuring the system evolves toward optimal performance.

3.2 SYSTEM ARCHITECTURE OVERVIEW

The autonomous robotic car system integrates hardware and software components to enable perception, decision-making, and wireless communication via LoRa. The architecture is divided into two parts: hardware and software.

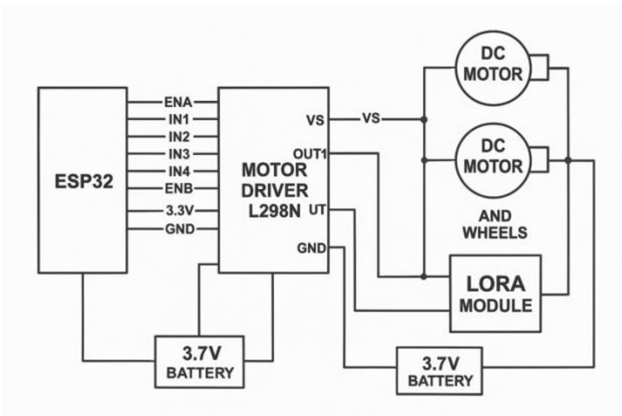

Figure 3.1: Circuit Diagram

3.2.1 Hardware Components

Microcontroller (ESP32): The ESP32 was selected for its dual-core processing, integrated Wi-Fi and Bluetooth (useful for local testing), and compatibility with LoRa modules. It serves as the central controller, managing sensor input, motor control signals, and communication routines.

Motor Driver (L298N): A dual H-bridge driver capable of controlling two DC motors. The L298N enables bidirectional motion and speed control, allowing the robotic car to navigate in multiple directions.

DC Motors (Geared): Two DC motors with gear reduction provide the necessary torque to drive the robot over various surfaces. The geared design ensures controlled speed and consistent power.

Power Supply Units: two 3.7V lithium-ion batteries are used to supply power to the motors via the L298N and the control electronics and communication modules, ensuring stable voltage levels.

LoRa Module (SX1278): This module enables long-range, low-power wireless communication, allowing remote commands to be transmitted to the robot and telemetry data to be sent back.

Chassis and Frame: A lightweight acrylic or aluminum chassis houses all components securely. It includes mounting holes for the motor brackets, driver boards, and battery holders

Figure 3.1: Hardware components

3.2.2 Software Components

The software component used for the study consists of:

Arduino IDE (Integrated Development Environment) is a software platform used to write, compile, and upload code to Arduino microcontroller boards. It acts as a bridge between the code and the physical hardware. Its implementation involves code writing, compiling, uploading and serial monitoring.

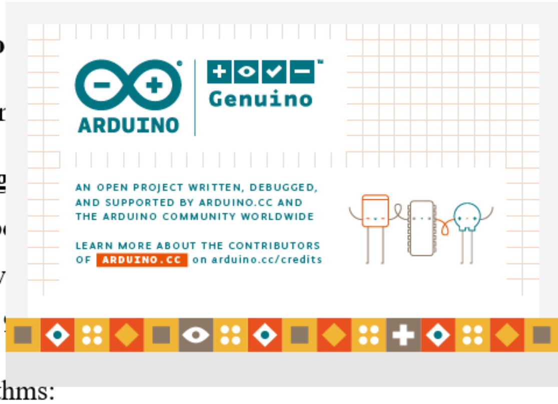

Figure 3.2: Arduino IDE Interface

Bluetooth RC Controller is a mobile or hardware-based remote-control system that uses Bluetooth technology to wirelessly operate devices like robots, cars, drones, or home automation systems. It enables wireless, short-range control of devices using simple commands transmitted over Bluetooth.

Figure 3.3: Bluetooth RC Controller Interface

CHAPTER 4

CONSTRUCTION, TESTING, RESULTS AND DISCUSSION

4.1 Introduction.

This chapter presents the comprehensive implementation framework for the autonomous robotic car system.

4.2 HARDWARE DESIGN AND IMPLEMENTATION

The hardware design of the autonomous robotic car revolves around coupling and integrating electronic and mechanical components together to enable mobility, control, and communication over long distances. This section details the key assembly steps:

4.2.1 Assembly Steps

Chassis Preparation: The motors were mounted onto the chassis using metal brackets, ensuring alignment for straight-line motion.

Board Installation: The ESP32, L298N, and LoRa module were secured onto the chassis using screws and spacers. The power supply units were positioned in accessible yet secure compartments.

Wiring and Connections: The motor terminals were connected to the L298N outputs, with jumper wires running from the driver's inputs to the ESP32's GPIO pins, and the LoRa module's SPI pins were linked to the ESP32. Power cables were double-checked for polarity.

Final Checks: All connections were inspected, verified for continuity with a multimeter, and ensured no loose or crossing wires that might cause shorts.

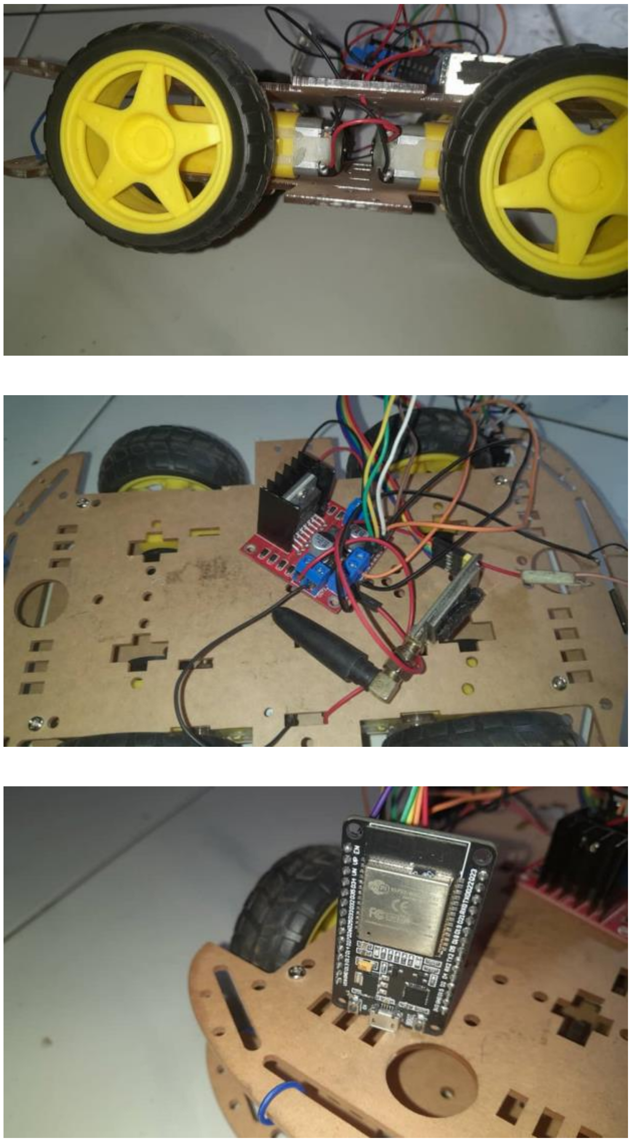

Figure 4.1: Hardware Assembly

4.2.2 Hardware Testing.

Each hardware subsystem underwent individual and integrated testing:

Motor Testing: Motors were tested by sending manual PWM signals to the L298N to confirm direction changes, speed variation, and smooth rotation.

Power System Testing: Voltage and current were measured across the system under idle and loaded conditions to ensure the batteries could supply adequate power without overheating.

Communication Testing: The LoRa module was tested using basic send/receive scripts to confirm stable communication over different distances, with special attention to signal quality and transmission delays.

System Integration Testing: Once all components were assembled, full system tests were conducted moving the robot under remote commands, verifying motor response, monitoring battery performance, and ensuring reliable long-distance control.

4.2.3 Software Development Process

The Arduino IDE was installed on a windows PC and all necessary ESP32 libraries were installed. The Arduino IDE version used for the study is Arduino 1.8.19(Windows Store 1.8.57.0). An additional URL was used via the interface preferences to enable the installation of ESP32 libraries; https://dl.espressif.com/dl/package_esp32_index.json.

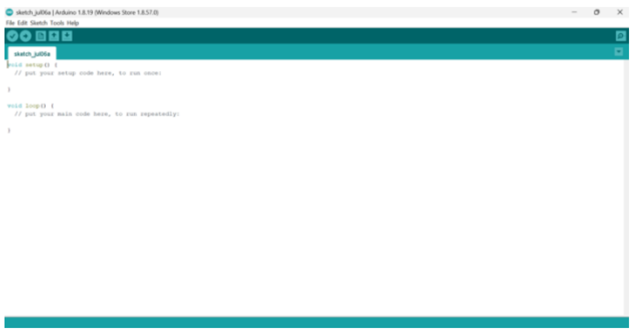

Figure 4.2: Arduino IDE Development Environment

4.2.4 Arduino Code Used

The code used for controlling the car is given below:

```
//esp32 board version must be 2.0.11

//otherwise the code will not compile

#include "BluetoothSerial.h"

#include <Arduino.h>

#include "driver/ledc.h"

BluetoothSerial serialBT;

//Bluetooth signal Store in this variable

char btSignal;

//initial Speed

int Speed = 100;

//declare channel for pwm Output

#define R 0

#define L 1

//PWM Pin for Controlling the speed

int enA = 5;

int enB = 23;

//motor controlling pin

int IN1 = 22;

int IN2 = 21;
```

```
int IN3 = 19;
int IN4 = 18;
void setup() {
  Serial.begin(115200);
  //Bluetooth Name
  serialBT.begin("Redmi Note12");
  //output pin declare
  pinMode(enA, OUTPUT);
  pinMode(enB, OUTPUT);
  // Setup PWM channels
  ledcSetup(R, 5000, 8);  // Channel 0 for Motor A, 5 kHz frequency, 8-bit resolution
  ledcAttachPin(enA, R);
  ledcSetup(L, 5000, 8);  // Channel 0 for Motor B, 5 kHz frequency, 8-bit resolution
  ledcAttachPin(enB, L);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  //Intial State of Car
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
```

```
  digitalWrite(IN3, LOW);

  digitalWrite(IN4, LOW);

}

void loop() {

  while (serialBT.available()) {

    btSignal = serialBT.read();

    //Serial.println(btSignal);

    if (btSignal == '0') Speed = 100;

    if (btSignal == '1') Speed = 110;

    if (btSignal == '2') Speed = 120;

    if (btSignal == '3') Speed = 130;

    if (btSignal == '4') Speed = 140;

    if (btSignal == '5') Speed = 150;

    if (btSignal == '6') Speed = 180;

    if (btSignal == '7') Speed = 200;

    if (btSignal == '8') Speed = 220;

    if (btSignal == '9') Speed = 240;

    if (btSignal == 'q') Speed = 255;

//to see the incoming signal in serial monitor

    Serial.println(btSignal);

  //backward
```

```
  }

  if (btSignal == 'B') {
    backward();
  }
//forward
  else if (btSignal == 'F') {
    forward();
  }
//LEFT
  else if (btSignal == 'L') {
    left();
  }

//RIGHT
  else if (btSignal == 'R') {
    right();
  }
//STOP
  else if (btSignal == 'S') {
    stop();
  }
}
```

```
}
//function for control motor
void backward() {
  ledcWrite(R, Speed);
  ledcWrite(L, Speed);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
  }
void forward() {
  ledcWrite(R, Speed);
  ledcWrite(L, Speed);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
void left() {
  ledcWrite(R, Speed);
  ledcWrite(L, Speed);
```

```
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
void right() {
  ledcWrite(R, Speed);
  ledcWrite(L, Speed);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
void stop() {
    ledcWrite(R, Speed);
    ledcWrite(L, Speed);
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);
}
```

4.2.5 Code functionality

Uses Bluetooth to receive signals from a mobile app (like Bluetooth RC Controller).

Controls two motors (for wheels) using PWM (Pulse Width Modulation) signals.

Responds to commands like Forward (F), Backward (B), Left (L), Right (R), Stop (S).

Allows speed control from 100 to 255 using digits 0–9 and q

Figure 4.3: The Entire Car Assembly

4.3 DISCUSSION

The ESP32 board runs the software (Arduino code) and waits for signals from a Bluetooth-enabled device (like a smartphone with an RC controller app). When you press buttons on the app (e.g., Forward, Backward), it sends a single-character command (like 'F', 'B', etc.) to ESP32 via Bluetooth. The ESP32 receives these commands in the code's loop () function. It reads the character and matches it to a motor control function (e.g., forward () if 'F' is received).

The selected function sets the direction pins (IN1–IN4) and uses PWM signals (through ledcWrite()) to control the speed and direction of the motors.

Car Movement: Based on the signals:

Both motors spin forward for forward movement.

Motors spin in opposite directions for turning.

Both stop when the Stop ('S') command is received.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Significance

This project demonstrates the practical implementation of an autonomous robotic car controlled wirelessly through LoRa communication and coordinated by an ESP32 microcontroller. The system offers a low-cost, energy-efficient, and long-range solution to remote robotic operation, which is particularly useful in areas where conventional wireless technologies like Bluetooth and Wi-Fi fall short. The project highlights the reliability of LoRa for long-distance command-based applications, the robustness of DC motor control using L298N, and the versatility of ESP32 as a control platform.

In the context of global advancements in automation and IoT, the significance of this project is magnified. As more systems are deployed in off-grid or underdeveloped areas, there is a pressing demand for cost-effective, low-power, and easily replicable technologies. This prototype stands out as a viable model for bridging the technological gap in underserved communities.

Its significance lies not only in its technical achievement but also in its educational and societal impact. It provides a scalable prototype that can be used for remote monitoring, smart farming, education, security surveillance, and other practical applications in rural or under-connected areas.

Furthermore, its use of open-source components and platforms enhances accessibility for students and developers globally. The project can be extended to academic curricula, innovation labs, and technical workshops as a foundational experiment in embedded systems, wireless communication, and robotics.

5.2 Limitations

Despite the project's success, certain limitations were identified during the development and testing phases:

Limited Data Rate of LoRa: While LoRa excels in range and power efficiency, its low data rate limits its suitability for high-bandwidth applications such as real-time video streaming.

GPS Dependency in Navigation: In outdoor applications involving GPS, signal loss in covered areas (e.g., indoors or under dense trees) can affect navigation.

Manual Command-Driven System: The system relies on external user commands rather than full autonomy or AI decision-making, which limits its independent operation.

Hardware Constraints: The use of basic DC motors and lack of advanced sensors like IMUs or encoders reduce the robot's ability to navigate complex terrains with precision.

Environmental Interference: Physical barriers and electromagnetic interference may reduce effective communication range or introduce latency.

Additionally, hardware assembly requires careful attention to power distribution, and minor voltage inconsistencies sometimes affect the stability of the system. Lack of waterproofing also limits deployment in extreme environmental conditions.

Software-side limitations include the absence of onboard data logging or diagnostics, which could help in tracking performance issues over time. More comprehensive testing under various weather conditions would also be needed to validate the reliability of the prototype in real-world deployments.

5.3 Conclusions

In conclusion, the project achieved its primary goal: to design and implement a cost-effective, long-range, LoRa-based autonomous robotic vehicle. The system successfully:

Integrated hardware components include ESP32, L298N, and LoRa SX1278.

Implemented wireless control with reliable performance up to several hundred meters.

Demonstrated stable movement, efficient power management, and responsive communication.

Provided a modular foundation that can be upgraded with additional features like GPS, obstacle detection, or AI.

5.4 Recommendations for Further Work

To enhance the system's capabilities and address its limitations, the following recommendations are proposed for future development:

1. **GPS and Navigation Algorithms:** Incorporate GPS modules with waypoint navigation algorithms to enable autonomous path-following capabilities. Algorithms like A* or Dijkstra's may be adapted to plan optimal routes in mapped terrains.
2. **Sensor Integration:** Add ultrasonic sensors, IR sensors, or LIDAR for obstacle detection and environmental awareness. Sensor fusion techniques could also be explored for better spatial understanding and navigation precision.
3. **Real-Time Feedback System:** Include a feedback loop using sensors to adjust motor speeds for smoother operation and better stability. This can be achieved with encoders, PID controllers, or IMU sensors to enhance motion accuracy.
4. **Mobile App Interface:** Develop a user-friendly smartphone application with a graphical user interface (GUI) for intuitive control. The app could include features like live system status, error alerts, and customizable control parameters.
5. **Camera Module Integration:** Add a low-power camera module (e.g., ESP32-CAM) for image capture or streaming with edge AI for basic object detection. TensorFlow Lite or OpenCV-based models could help enable smart tasks like QR detection or visual patrolling.
6. **Power Optimization:** Explore the use of solar charging or more efficient battery management systems for prolonged outdoor usage. MPPT solar charge controllers and smart BMS systems can significantly extend the robot's deployment time.
7. **Mesh Networking:** Implement LoRa mesh topology or integrate with other protocols (e.g., Wi-Fi + LoRa hybrid) to improve coverage and scalability in larger fields or multi-robot systems. Mesh routing would also enable collaborative operations among multiple robots.

Moreover, future enhancements could explore full autonomy using AI-powered navigation systems and environmental adaptation, which would make the system suitable for advanced deployment in complex or hazardous terrains. Real-world applications could range from disaster relief, unmanned patrols, to precision farming in large-scale fields.

REFERENCES

[1] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, Introduction to Autonomous Mobile Robots, 2nd ed. Cambridge, MA: MIT Press, 2011.

[2] J. Wang, Y. Zhang, and H. Liu, "Design of a Wi-Fi Controlled Robotic Car for Real-time Surveillance," International Journal of Robotics and Automation, vol. 5, no. 2, 2017.

[3] J. Lee and K. Lee, "Bluetooth-based wireless control for a mobile robot," Journal of Control Engineering and Technology, 2010.

[4] S. Kumar and V. Sharma, "Limitations of Wi-Fi Based Mobile Robots in Outdoor Environments," Wireless Engineering Review, vol. 7, no. 3, pp. 112–117, 2015.

[5] A. Augustin, J. Yi, T. Clausen, and W. Townsley, "A Study of LoRa: Long Range & Low Power Networks for the Internet of Things," Sensors, vol. 16, no. 9, p. 1466, 2016.

[6] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of LoRaWAN," IEEE Communications Magazine, vol. 55, no. 9, pp. 34–40, 2017.

[7] J. Petäjäjärvi, K. Mikhaylov, M. Hämäläinen, and J. Iinatti, "Evaluation of LoRa LPWAN technology for remote health and agriculture monitoring," Ad Hoc Networks, Elsevier, 2017.

[8] Espressif Systems, ESP32 Series Datasheet, Espressif Systems, 2019. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

[9] M. Rana, R. Islam, and S. Aziz, "IoT-based Smart Vehicle System using ESP32 and LoRa," Proc. Int. Conf. Smart Innovations in Design, Environment, Management, Planning and Computing, 2020.

[10] A. Khan, M. Yusuf, and S. Tukur, "Development of an Obstacle-Avoiding Robot using L298N Motor Driver," Journal of Engineering Research and Technology, vol. 3, no. 2, 2018.

[11] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," Comput. Netw. vol. 52, no. 12, pp. 2292–2330, Aug. 2008, doi: 10.1016/j.comnet.2008.04.002.

[12] Semtech Corporation, SX1276/77/78/79 Datasheet: 137 MHz to 1020 MHz Low Power Long Range Transceiver, Rev. 7, Camarillo, CA, USA, 2020. [Online]. Available: https://www.semtech.com/products/wireless-rf/lora-core/sx1276

[13] S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics. Cambridge, MA, USA: MIT Press, 2005.

[14] T. Instruments, L298N Dual H-Bridge Motor Driver Datasheet, Dallas, TX, USA, 2019.

[15] P. G. Savage, "Strapdown inertial navigation integration algorithm design part 1: Attitude algorithms," J. Guid. Control Dyn. vol. 21, no. 1, pp. 19–28, Jan. 1998, doi: 10.2514/2.4228.

[16] LoRa Alliance, LoRaWAN 1.0.3 Specification, Beaverton, OR, USA, 2018. [Online]. Available: https://lora-alliance.org/resource_hub/lorawan-specification-v1-0-3/

[17] Raspberry Pi Foundation, Raspberry Pi 4 Model B Product Brief, Cambridge, UK, 2021.

[18] U-blox, NEO-M8N GNSS Module Integration Manual, Thalwil, Switzerland, 2020.

[19] R. C. Dorf and R. H. Bishop, Modern Control Systems, 13th ed. London, UK: Pearson, 2017.

[20] M. Quigley et al., "ROS: An open-source Robot Operating System," ICRA Workshop Open Source Softw. vol. 3, no. 3.2, p. 5, 2009.

[21] InvenSense, MPU-9250 Product Specification, San Jose, CA, USA, 2016.

[22] International Organization for Standardization, ISO 13850:2015 Safety of machinery Emergency stop function, Geneva, Switzerland, 2015.

[23] P. Misra and P. Enge, Global Positioning System: Signals, Measurements, and Performance, 2nd ed. Lincoln, MA, USA: Ganga-Jamuna Press, 2010.

[24] A. Koubaa et al., "A study of LoRa low power and wide area network technology," *IEEE Access*, vol. 5, pp. 18072–18092, 2017, doi: 10.1109/ACCESS.2017.2748928.

[25] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," IEEE Trans. Robot. Auto. vol. 12, no. 6, pp. 869–880, Dec. 1996, doi: 10.1109/70.544770.