Faculty of Electrical Technology and Engineering

DEVELOPMENT OF REMOTE DRIVING CONTROL OF AN ELECTRICAL VEHICLE USING WIRELESS TECHNOLOGY

PAVENTHAN A/L KUMARASAMY

Bachelor of Electrical Engineering Technology (Industrial Automation & Robotics) with Honours

2023

DEVELOPMENT OF REMOTE DRIVING CONTROL OF AN ELECTRICAL VEHICLE USING WIRELESS TECHNOLOGY

PAVENTHAN A/L KUMARASAMY

A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Electrical Engineering Technology (Industrial Automation & Robotics) with Honours

Faculty of Electrical Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2023

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI DAN KEJUTERAAN ELEKTRIK

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : Development of Remote Driving Control of a Electrical Vehicle using wireless technology

Sesi Pengajian : 1-2023/2024

Saya Paventhan A/L Kumarasamy mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

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

☐ **SULIT*** (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

☐ **TERHAD*** (Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

☑ **TIDAK TERHAD**

Disahkan oleh:

(TANDATANGAN PENULIS)

Alamat Tetap: NO 6, JALAN PANCHOR INDAH, TAMAN PANCHOR INDAH, 14300 NIBONG TEBAL, PULAU PINANG

(COP DAN TANDATANGAN PENYELIA)

TS. DR. MOHD HANIF BIN CHE HASAN
PENSYARAH
Fakulti Teknologi dan Kejuruteraan Elektrik
Universiti Teknikal Malaysia Melaka

Tarikh: 13/1/2024

Tarikh: 15 February 2024

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

DECLARATION

I declare that this project report entitled " Development of remote driving control of an electrical vehicle using wireless technology " 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 : Paventhan A/L Kumarasamy

Date : 13/1/2024

APPROVAL

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

Signature :

Supervisor Name : Ts Dr Mohd hanif Che Hasan

Date : 15 February 2024

Signature :

Co-Supervisor Name (if any) :

Date :

DEDICATION

To my beloved mother, Valliammal, and father, Kumarasamy,

and

my fellow friends.

ABSTRACT

The development of a remote driving control system for an electrical vehicle using wireless technology is the main focus of this project. Wireless technology-based remote vehicle control, however, has also gained popularity, creating new opportunities and uses. With their improved safety, effectiveness and convenience, these inventions have completely changed the way we think about transportation. The objective is to provide users with the ability to remotely control the movement, speed, and direction of the vehicle. The system incorporates several key hardware components, including Arduino boards, RF modules for wireless communication, motor driver modules, potentiometers for pedal and brake input, and various sensors for enhanced safety and functionality. The software aspect involves programming the Arduino boards to interpret user inputs, transmit signals wirelessly, and control the vehicle's movements accordingly. A systematic approach is followed, starting with the selection and integration of the necessary hardware components. Safety measures, such as the inclusion of an ultrasonic sensor for collision avoidance and obstacle recognition, are implemented to ensure the system's reliability and user safety. The successful development of this remote driving control system has significant implications for the automotive industry, as it opens up new possibilities for remote vehicle operation and control. The project serves as a foundation for further research and development in the field of wireless control systems for vehicles. The existing remote control car has few barriers where it has distance restriction between the controller and the vehicle. It also has less feature to control a car’s movement. They also have lack of safety considerations towards the car’s movement and also the environment. Additionally, the integration of wireless technology into the automotive sector has the potential to revolutionize transportation, offering increased convenience, efficiency, and safety for users.

ABSTRAK

Pembangunan sistem kawalan kenderaan elektrik jarak jauh menggunakan teknologi tanpa wayar adalah fokus utama projek ini. Walau bagaimanapun, penggunaan teknologi tanpa wayar dalam kawalan kenderaan dari jauh juga telah mendapat populariti, membuka peluang dan kegunaan baru. Dengan peningkatan keselamatan, keberkesanan, dan keselesaan, inovasi ini telah mengubah sepenuhnya cara kita memandang pengangkutan. Objektif adalah untuk memberi pengguna keupayaan untuk mengawal pergerakan, kelajuan, dan arah kenderaan dari jauh. Sistem ini merangkumi beberapa komponen perkakasan penting, termasuk papan Arduino, modul RF untuk komunikasi tanpa wayar, modul pemandu motor, potensiometer untuk input pedal dan brek, dan pelbagai sensor untuk keselamatan dan fungsi yang lebih baik. Aspek perisian melibatkan pengaturcaraan papan Arduino untuk menginterpretasikan input pengguna, menghantar isyarat secara tanpa wayar, dan mengawal pergerakan kenderaan mengikutnya. Pendekatan sistematik diikuti, bermula dengan pemilihan dan integrasi komponen perkakasan yang diperlukan. Langkah-langkah keselamatan, seperti penggunaan sensor ultrasonik untuk mengelakkan perlanggaran dan mengenali halangan, dilaksanakan untuk memastikan kebolehpercayaan sistem dan keselamatan pengguna. Pembangunan berjaya sistem kawalan jarak jauh ini mempunyai implikasi yang signifikan dalam industri automotif, membuka peluang baru untuk pengendalian dan pengoperasian kenderaan dari jauh. Projek ini berfungsi sebagai asas untuk penyelidikan dan pembangunan lanjutan dalam bidang sistem kawalan tanpa wayar untuk kenderaan. Kereta kawalan jarak jauh sedia ada mempunyai beberapa halangan di mana ia mempunyai had jarak antara pengawal dan kenderaan. Ia juga mempunyai kurang ciri untuk mengawal pergerakan kereta.

ACKNOWLEDGEMENT

I extend my deepest appreciation to my supervisor, Ts. Dr Mohd Hanif Che Hasan, for his invaluable guidance, unwavering support, and patience throughout the entirety of this project. His mentorship has been instrumental in shaping the direction and success of this endeavor.

I am grateful to Universiti Teknikal Malaysia Melaka (UTeM) for the financial support provided through funding me that significantly contributed to the completion of this project.

Special thanks go to my parents and family members for their boundless love, encouragement, and prayers during my academic journey. I also express my gratitude to my friends for the shared insights and collaborative discussions that enriched the project.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

1.1 Background

With the development of autonomous vehicles and self-driving technology, the automotive sector has made significant advancements recently. With their improved safety, effectiveness, and convenience, these inventions have completely changed the way we think about transportation. Advanced algorithms, sensors, and artificial intelligence are used by autonomous vehicles to navigate and run autonomously. Wireless technology-based remote vehicle control, however, has also gained popularity, creating new opportunities and uses.

Remote driving control systems give customers the flexibility and accessibility to drive a car from a distance in a variety of situations. Applications for this technology can be found in surveillance, exploration, agriculture, and entertainment. Users can drive cars in situations where direct human interaction could be difficult or dangerous by enabling remote control. Remote driving control systems also provide options for recreational activities by enabling users enjoy the thrill of driving without really being inside the vehicle.

The purpose of this project is to create a remote driving control system for an electrical vehicle using wireless technologies. Users of the system will be able to direct the movement, speed, and direction of the vehicle from a distance. In order to create a trustworthy and effective connection between the user's control inputs and the vehicle's activities, the project makes use of wireless communication.

The project will combine hardware and software components in order to achieve this. An Arduino microcontroller board, RF (Radio Frequency) modules for wireless communication, motor driver modules for controlling the vehicle's motors, potentiometers for pedal input, and numerous sensors to improve safety and functionality are included in the hardware. Programming the Arduino boards to interpret user inputs, transmit and receive signals wirelessly, and control the vehicle's movements accordingly is part of the software.

The project will use an organized methodology, beginning with the selection and obtaining the required hardware components. A clear block diagram will be used to connect and integrate the components, ensuring proper operation and communication. The Arduino Integrated Development Environment (IDE), which enables the development of particular algorithms and control logic, will be used to program the Arduino boards.

The project's objectives will include developing hardware and software as well as ensuring safety and reliability. Safety precautions will be included, such as the incorporation of an accelerometer sensor for collision avoidance and esp32-cam to stream video for obstacle recognition. To assure the system's performance and functionality in various scenarios, testing and validation techniques will also be used.

The growth of wireless technology applications in the automotive industry will be aided by the successful development of the remote driving control system. The project's results will act as a starting point for additional study and research into remote control systems, with the potential to result in practical applications and commercialization.

Wireless technology provides immense potential for the development of a remote driving control system for an electrical vehicle in numerous of different industries. With the help of a reliable and effective system that enables users to control automobiles remotely with accuracy and convenience, this project strives to take advantage of this potential. The project seeks to add to the growing body of knowledge on remote control systems and wireless technology applications in the automotive sector through challenging hardware and software integration, safety considerations, and methodical development techniques.

1.2 Problem Statement

In recent years, there has been a lot of attention in the development of wireless remote driving control systems for electrical vehicles. The intention of these systems is to enhance remote vehicle operation flexibility and comfort. To ensure the proper implementation and operation of such systems, several challenges must be resolved.

1. Limited Range: The range of operation is limited by the fact that many existing remote control car systems rely on wired connections between the controller and the car. So this becomes a barrier to drive this car to a wider range of distance. Designing a wireless remote driving control system that can function over a larger distance will provide for more versatility and range.[1]

2. Integration of Vehicle Functions: Its challenging to integrate different vehicle operations like steering, braking, and acceleration into a single remote control system. Where all this operation gives the user of the car, a real time experience to control the car’s direction, speed, brake and angle of tilting the wheels of the car. For a vehicle to operate smoothly and accurately, a system that effectively integrates these operations must be created.[2]

3. Safety considerations: It is crucial to guarantee the safety of the vehicle and its surroundings when it is being operated remotely. When the car is moving abnormally, there’s no detection method available to alert the driver or the user about the abnormal driving pattern. To ensure the safety of the vehicle and its environment during remote operation while implementing robust safety measures to prevent accidents, collisions, and detect obstacles.[3]

By overcoming these obstacles, a more adaptable, practical, and secure method of vehicle operation will be possible for wireless remote driving control systems for electrical vehicles.

1.3 Objectives

The motive of this project was to construct a wireless remote-controlled car that doesn't require any physical connections and offers more convenience and mobility.

The main objective of this project are:

1. Designed and implemented wireless communication technology for reliable and efficient transmission of control signals.
2. Integrated and controlled vehicle movements remotely.
3. Ensured safety of the remote driving control system through the inclusion of obstacle detection feature for collision avoidance, identify unsafe driving patterns.

1.4 Scope of Project

The scope of this project encompasses the design and implementation of a remote driving control system for an electrical vehicle using wireless technology. There are key aspects for the scope of my project.

1. Design and Integration: Developed an inclusive design plan that includes the selection and integration of hardware components such as microcontrollers, wireless communication modules, motor driver and input devices to create a cohesive system.

2. Wireless Communication Implementation: Implemented a reliable and efficient wireless communication system between the user and the electrical vehicle, ensuring seamless transmission of control signals with less latency.
3. Software Development: Developed software programs to control the electrical vehicle remotely, including interpreting user inputs, establishing wireless communication, and transmitting control signals to the vehicle's motors. Implement safety features and collision avoidance algorithms to ensure safe operation.
4. Testing and Validation: Conducted testing and validation of the remote driving control system, evaluating its performance, responsiveness, accuracy, and reliability under different scenarios and conditions.

CHAPTER 2 LITERATURE REVIEW

2.1 Introduction

The literature review examines the present state of knowledge and identifies areas that require further research, serving as a foundation for subsequent research and inquiry. By examining many viewpoints, methodology, and conclusions from prior studies, it offers a thorough understanding of the subject. Based on the main ideas and research questions, the available literature will be divided into themes or subtopics for the organization of the literature review. The key conclusions, methodology used, and any shortcomings or gaps in the studies will all be covered in-depth for each theme.

2.2 Hardware

The development and operation of autonomous vehicle driving systems heavily rely on the hardware components. The hardware components of autonomous car driving, such as the sensors, microcontrollers, wireless communication modules, and other crucial parts, are the topic of this literature review, which aims to analyze the current research and achievements in this area. The review starts out by going through the essential hardware elements needed for autonomous vehicle operation. It looks into each technology's underlying principles of functioning, its potential, and its difficulties.

2.2.1 Optimal Sensor Selection for Wireless Car Systems

The sensor is needed in my project to detect obstacles and ensure safe navigation. It provides real-time data on the proximity of objects, allowing the car to make informed decisions and adjust its path accordingly. This helps prevent collisions and improves the overall driving experience and safety of the wireless car. The literature has generally acknowledged the significance of sensors in the creation of wireless cars.

Ultrasonic sensors have proven to be a valuable asset in wireless car projects. Research articles indicate that ultrasonic sensors offer reliable and accurate distance measurement capabilities using ultrasonic waves. Their ability to detect obstacles and provide precise distance readings enhances the safety and navigation features of wireless cars. Several studies highlight their effectiveness in obstacle avoidance and collision detection, making them a popular choice among developers. However, it's worth noting that environmental factors, such as noise or interference, can affect the performance of ultrasonic sensors. Additionally, some articles mention limitations in detecting certain materials or surfaces, as ultrasonic waves may behave differently in these scenarios.[1]. Figure 2.1 shows the detection mechanism of ultrasonic sensor.

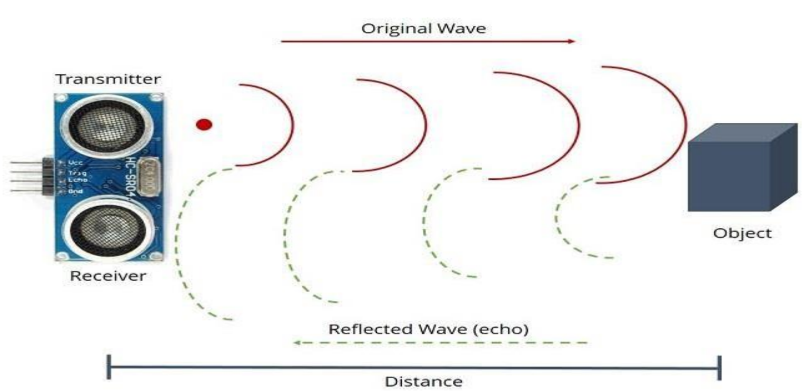


Figure 2.1: working mechanism of ultrasonic sensor[1]

Infrared sensors have also gained significant attention in the field of wireless car applications. Researchers have emphasized the cost-effectiveness and simplicity of implementing infrared sensors in proximity sensing tasks. Articles suggest that infrared sensors can effectively detect objects and obstacles in the car's surroundings. They are particularly useful for parking assistance systems, where the sensors assist in maneuvering the car safely into parking spaces. However, it's important to consider that infrared sensors typically have a limited range. Their effectiveness can also be influenced by ambient light conditions, which may lead to false readings or reduced performance in certain situations.[2]. Figure 2.2 shows how the infrared sensor works in detecting the obstacle.

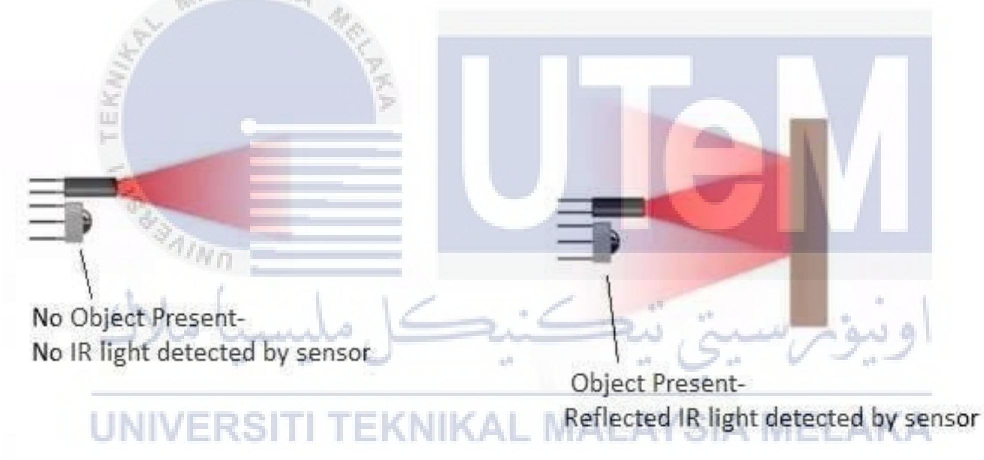


Figure 2.2: Infrared sensor mechanism[2]

Light sensors have emerged as valuable components in wireless car projects, primarily for monitoring the lighting conditions in the environment. Research articles indicate that light sensors enable the car to adapt its behavior based on the surrounding lighting levels. For example, the sensors can trigger automatic headlight activation or adjust the display brightness for optimal visibility. This enhances the overall user experience and safety of the wireless car. While light sensors are generally reliable, it's crucial to consider their susceptibility to ambient light variations, as extreme brightness or darkness may affect their accuracy.[3]

GPS (Global Positioning System) sensors have gained popularity for their ability to provide accurate positioning information. Articles indicate that GPS sensors use satellite signals to determine the car's location, enabling features such as real-time tracking, navigation, and route planning. Their widespread availability and compatibility make them a valuable asset in wireless car applications. However, it's important to note that GPS signals can be affected by obstacles or signal interference in urban environments or dense foliage areas, potentially impacting the accuracy of location data.[4]. Figure 2.3 below shows the gps sensor module and the connectivity of it.

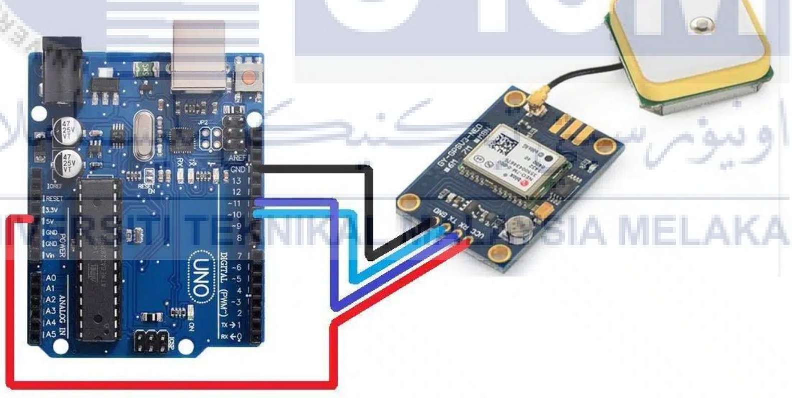


Figure 2.3: NEO-6M GPS Module[4]

Accelerometers is a motion sensors that play a crucial role in wireless car projects. Research articles highlight their use in measuring acceleration, tilt, and rotation, which enables features like vehicle stability control and motion-based steering control. These sensors provide vital data for detecting car movements, adjusting steering angles, and enhancing overall driving experience. However, precise calibration and filtering techniques are required to ensure accurate readings and minimize noise or vibration interference.[5]. The Figure 2.4 shows the acceleration changes measured by accelerometer sensor when the car passed by the manhole.

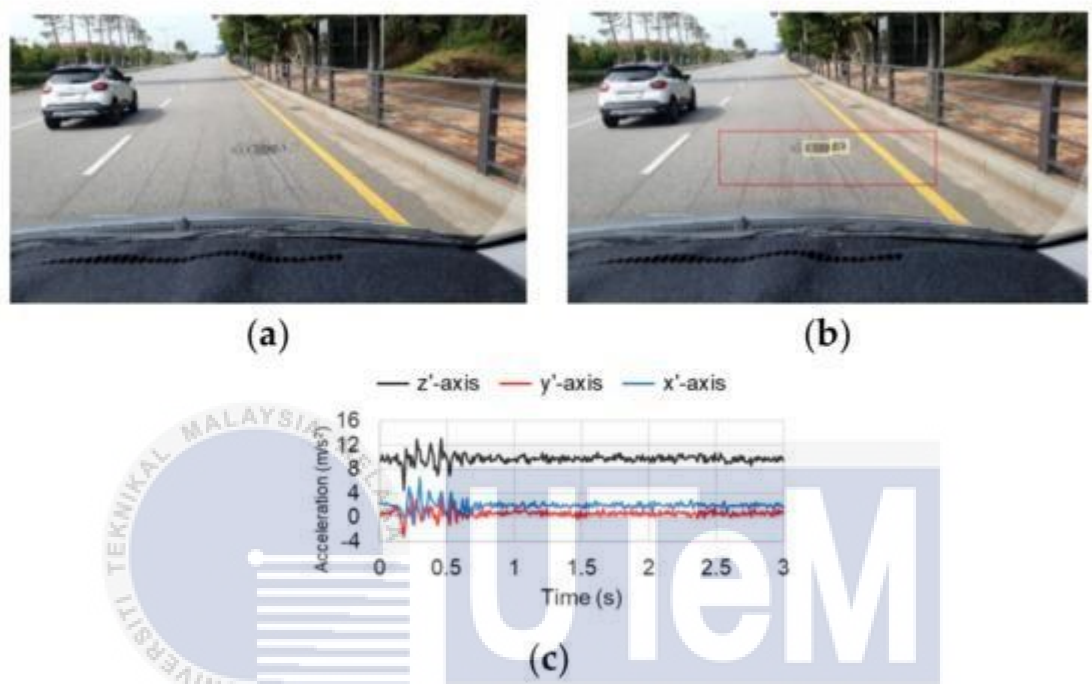


Figure 2.4: The acceleration changes when the car passed by a manhole[5]

Pressure sensors are utilized in wireless cars for monitoring tire pressure. Articles indicate that these sensors measure the air pressure inside the tires, providing real-time data to the driver. Maintaining optimal tire pressure enhances fuel efficiency, tire longevity, and overall safety. Pressure sensors can be integrated with wireless car systems to provide tire pressure alerts and enable proactive maintenance. It's worth noting that pressure sensors may require periodic calibration and validation to ensure accurate readings.[6]

Temperature sensors are employed in wireless cars to monitor and regulate temperature-related systems. Research articles mention their use in measuring coolant temperature, engine temperature, or ambient temperature. By providing accurate temperature readings, these sensors contribute to efficient engine performance, prevention of overheating, and effective climate control. It's important to consider the sensor's temperature range and response time to ensure compatibility with the car's operating conditions.[7]

Proximity sensors, such as capacitive or magnetic sensors, are utilized for detecting the proximity of objects around the car. Articles suggest that these sensors are employed in parking assistance systems, object detection, and anti-collision features. Proximity sensors provide reliable detection capabilities, allowing the car to respond appropriately to nearby obstacles. However, the selection of the appropriate proximity sensor depends on factors such as the desired range, sensing technology, and environmental conditions.[8]

Table 2-1 : Comparison on sensors

Sensor	Advantages	Disadvantages
Ultrasonic Sensor	- Accurate distance measurement - Wide detection range - Reliable obstacle detection - Affordable and readily available	- Susceptible to environmental factors (temperature, noise) - Limited resolution at close distances
Accelerometer	- Measures acceleration and tilt in multiple directions - Can detect changes in velocity and orientation	- Requires calibration for accurate readings - Sensitive to noise and vibrations

	- Compact and lightweight - Suitable for motion detection and orientation tracking	- Limited to measuring linear motion and tilt
Infrared (IR) Sensor	- Affordable and widely available - Effective for object detection and proximity sensing - Works well in low-light conditions - Can be used for line following and obstacle avoidance - Easy to interface with microcontrollers	- Limited range and field of view - Susceptible to interference from ambient IR sources - Reflective surfaces can affect accuracy
Light Sensor	- Cost-effective and easy to implement. - Detects ambient light levels, making it useful for applications such as automatic headlights.	- Slow response time, unsuitable for applications requiring rapid changes in light detection. - Limited sensitivity range and accuracy compared to specialized light sensors.

	- Can be used to measure light intensity or as part of light-based control systems.	- Affected by temperature variations, which may impact readings.
Pressure Sensor	- Monitors tire pressure in real-time. - Enhances fuel efficiency and safety.	- Requires periodic calibration and validation.
Temperature Sensor	- Monitors coolant/engine/ambient temperature. - Enhances engine performance and climate control.	- Temperature range and response time should be considered.
Proximity Sensor	- Detects proximity of objects around the car. - Used for parking assistance and anti-collision features.	- Selection depends on range, sensing technology, and environment.
GPS (Global Positioning System) Sensor	- Provides accurate positioning information. - Enables real-time tracking and navigation.	- Can be affected by obstacles or signal interference in urban areas or dense foliage.

	- Useful for route planning and geolocation-based features.	- Requires a clear line of sight to satellites for optimal performance.

From the comparison table above, I would highly prefer in choosing an accelerometer due to its unique advantages and suitability for my specific application. An accelerometer is a motion sensor that measures acceleration, tilt, and rotation. It plays a crucial role in enhancing the overall driving experience and ensuring vehicle stability.

One of the main advantages of an accelerometer is its ability to accurately detect changes in the car's motion and orientation. By measuring acceleration, it provides valuable data for detecting car movements, adjusting steering angles, and enhancing overall control. This enables features such as vehicle stability control and motion-based steering control, which are essential for safe and precise driving.[5]

Unlike other sensors that may have limitations or dependencies on external factors, an accelerometer offers standalone functionality. It does not rely on external signals or specific environmental conditions, making it highly reliable and independent. This autonomy ensures consistent performance regardless of the surrounding environment or external factors.[9]

Another advantage of accelerometers is their high sensitivity and responsiveness. They can detect even the slightest changes in acceleration or tilt, allowing for precise and real-time monitoring of the car's movements. This level of sensitivity is crucial for accurate control and safety, especially in dynamic driving situations or when maneuvering through challenging terrains.

Furthermore, accelerometers are relatively compact and cost-effective compared to some other sensors. Their small form factor allows for easy integration into the car's system without occupying excessive space. Moreover, their affordability makes them a practical choice for wireless car projects with budget constraints.

2.2.2 Wireless Car Connectivity using wireless communication module

Wireless communication modules play a vital role in enabling connectivity and data transfer in a wide range of applications. In my wireless car project, incorporating a wireless communication module is crucial. It allows me to establish a reliable connection and enables seamless communication between different components. With wireless communication, I can achieve remote control, data transmission, and system integration, enhancing the flexibility and functionality of my project. In recent years, there has been significant research and development focused on improving the performance, reliability, and efficiency of wireless communication modules.

For short-range communication, Bluetooth modules have become a popular option, often covering distances of up to 10 metres.[10] Their benefits, including their low power consumption, small size, and compatibility with a variety of devices, are highlighted in numerous studies. Numerous applications, such as wearable technology, smart home automation systems, and wireless music streaming devices, have been the subject of research into the usage of Bluetooth modules. Due to their energy efficiency and interoperability with a variety of device ecosystems, these modules are especially appropriate for battery-powered devices[11]. Figure 2.6 shows the use of bluetooth module in transceiving data wirelessly.

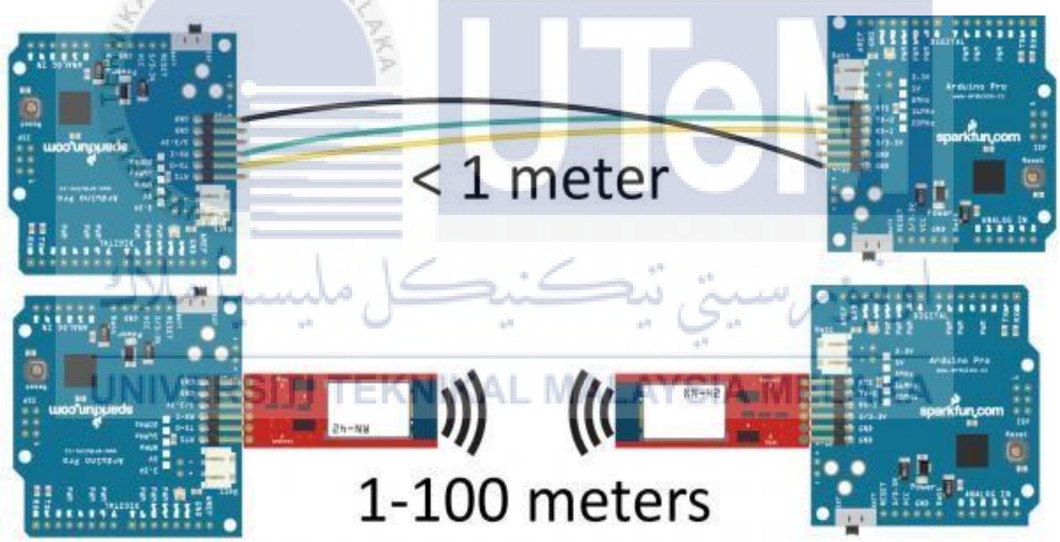


Figure 2.6: The use of bluetooth module in transceiving data wirelessly[11]

High-speed data transfer is a famous capability of Wi-Fi modules, making them ideal for applications requiring huge data transmission and real-time communication. There’s a research explores the use of Wi-Fi modules in wireless sensor networks, highlighting their role in enabling quick and secure data transfer between sensor nodes.[12] With Wi-Fi modules playing crucial roles in environmental monitoring, smart grids, industrial automation, and remote monitoring systems, this feature makes it easier to implement real-time monitoring and control systems in smart cities. Advanced applications across domains are made possible by the high data throughput provided by Wi-Fi modules, which enables seamless transfer of sensor data, video streaming, and cloud connectivity.[13]

Cellular communication modules like GSM and LTE offer a good alternative for applications that require extensive coverage. Researchers have researched cellular modules' usage in remote asset tracking in great detail. These modules utilize cellular networks that already exist to enable communication and position tracking over very large geographic areas. In order to provide real-time monitoring and control, especially in remote or rural places where other communication options may be limited, cellular communication modules are extensively relied upon by industries including fleet management, asset tracking, and remote surveillance systems. Cellular networks' extensive coverage guarantees constant connectivity and enables essential applications.[14]

Due of its long-range communication capabilities and low power requirements, LoRa modules have become popular. The use of LoRa modules in smart agriculture systems has been investigated by researchers. These systems enable long-range communication for the purposes of monitoring soil conditions, gathering weather data, and remotely operating irrigation systems. In order to ensure effective and dependable communication, battery-operated sensor nodes deployed over vast regions are well-suited for LoRa modules because to their low power consumption. These modules provide low-power, long-range connectivity for a variety of applications, and are widely used in precision agriculture, environmental monitoring, and smart city infrastructure.[15]

Zigbee modules have become widely used in the fields of wireless sensor networks and home automation systems. Their low power consumption, mesh networking capabilities, and device compatibility have all been thoroughly studied by academics. Zigbee modules are used in building energy management systems research to provide energy-efficient communication between equipment in smart buildings. Zigbee modules support real-time data interchange for HVAC management, occupancy detection, and energy monitoring, allowing for energy optimisation and raising overall effectiveness.[16]. Figure 2.7 shows the working principle of ZigBee module from transmitter to receiver.

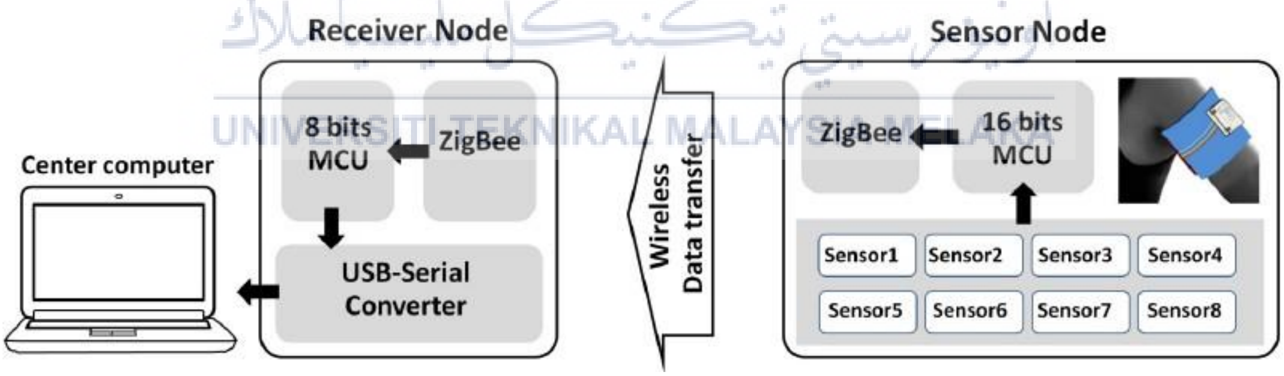


Figure 2.7 : The working principle of ZigBee module[16]

RFID modules have become an important technology for tracking and identification reasons. Researchers have looked into how supply chain management could use RFID modules to track products in real time, manage inventories effectively, and take anti-counterfeiting precautions.[17] These modules enable smooth and automatic tracking throughout the supply chain by using radio frequency signals to communicate with RFID tags affixed to products. RFID modules are widely used by the asset tracking, logistics, retail, healthcare, and other industries to improve productivity, accuracy, and security in their operations.[10]. The Figure 2.8 shows how does the RFID works wirelessly.

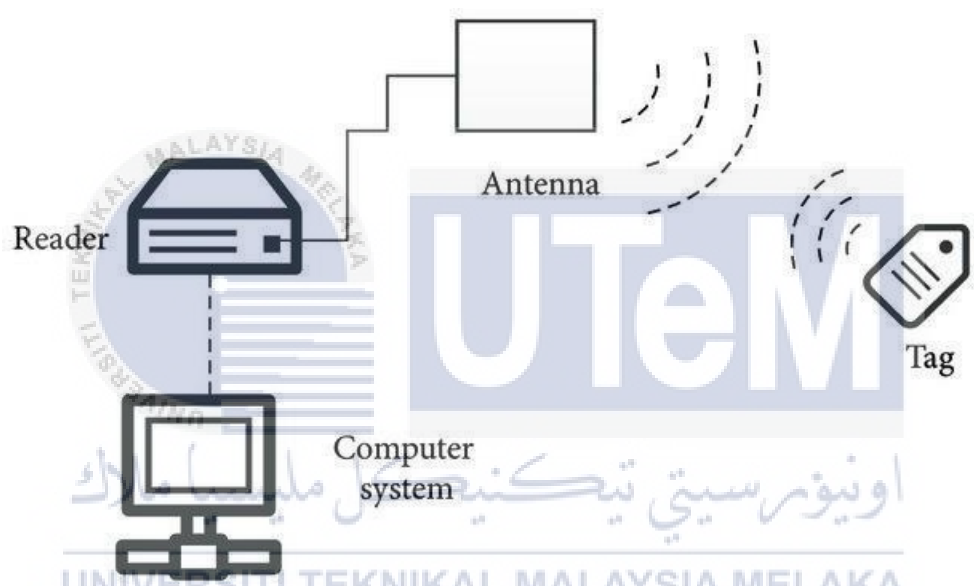


Figure 2.8: The RFID working principle[10]

NFC modules enable contactless data transfer between devices by facilitating short-range communication. Researchers have looked into how they are used in a variety of fields, with mobile payment systems receiving particular attention. By enabling communication between smartphones and payment terminals, NFC modules make it possible to make secure and practical mobile payments. NFC technology is also used in access control systems, ticketing for public transit, and other situations requiring quick and secure short-range communication.[18]

The RF (Radio Frequency) module stands out as a highly versatile wireless communication solution. With its long-range communication capability, it enables seamless connectivity over extensive distances, making it ideal for applications such as remote asset tracking, telemetry systems, and large-scale wireless networks. The module's adaptability and compatibility with various devices ensure easy integration into diverse projects, including home automation, industrial control systems, and wireless sensor networks. Its reliability, ease of implementation, and ability to facilitate real-time monitoring and control make the RF module an invaluable tool for revolutionizing industries and enhancing wireless communication systems across a wide range of applications.[1]

Table 2-2 : Comparison of wireless communication modules

Wireless Communication technique	Advantages	Disadvatages
Bluetooth Modules	- Low power consumption - Small size - Compatibility with various devices	- Limited range (up to 10 meters)
Wi-Fi Modules	- High-speed data transfer - Ideal for applications requiring large data	- Higher power consumption compared to other wireless communication modules

	transmission and real-time communication	
Cellular Communication Modules (GSM, LTE)	- Extensive coverage over large geographic areas - Suitable for remote asset tracking and monitoring	- Reliance on existing cellular networks - Ongoing subscription costs for cellular data plans
LoRa Modules	- Long-range communication capabilities - Low power consumption	- Lower data transfer rates compared to other wireless communication modules - Limited bandwidth for large-scale deployments
Zigbee Modules	-Low power consumption -Mesh networking capabilities	- Limited range (typically up to 100 meters) - Requires a Zigbee coordinator for network setup
RFID Modules	- Tracking and identification capabilities - Automatic tracking throughout the supply chain	- Limited to short-range communication - Requires RFID tags for tracking, which can add cost
NFC Modules	- Contactless data transfer between devices - Secure and practical for mobile payments	- Limited range (typically up to 10 centimeters) - Requires close proximity for communication to occur

RF transceiver modules	- Suitable for various applications. - Extended coverage distance. - Easy integration and setup. - Affordable and budget-friendly. - Works with different devices.	- Restricted data transmission capacity. - Vulnerable to signal disruptions. - Requires proper channel selection.

RF (Radio Frequency) modules offer a unique set of advantages that make them a suitable choice for various degree-level projects. When compared to other wireless communication modules, RF modules exhibit several distinctive features. Firstly, RF modules provide versatile wireless connectivity over long distances, making them ideal for applications that require extensive coverage. This is particularly beneficial in projects involving remote monitoring, asset tracking, or rural deployments where other communication options may be limited. RF modules can ensure constant connectivity and enable essential applications in such scenarios.

Furthermore, RF modules offer low power consumption, making them suitable for battery-operated devices or projects where power efficiency is crucial. They provide efficient and reliable communication while conserving energy, extending the operational lifespan of battery-powered devices. This aspect is particularly advantageous in projects where power constraints or energy optimization are key considerations.

2.2.3 Microcontroller for Wireless Car Systems

In the realm of wireless driving car projects, the selection of an appropriate microcontroller plays a vital role. The microcontroller serves as the brain of the system, facilitating communication between various components and executing control algorithms. With its ability to process inputs, control outputs, and handle wireless communication protocols, the microcontroller forms the foundation for creating a functional and intelligent wireless driving car. These tiny integrated circuits can communicate with the outside environment using sensors, actuators, and communication modules since they have a central processing unit (CPU), memory, and multiple input/output peripherals. From the research there are several microcontrollers that can be used in my project.

Due to its numerous connectivity possibilities and potent computing capacity, the Raspberry Pi, a popular microcontroller, has become increasingly popular in applications for wireless driving automobiles. [19]It allows smooth integration with numerous parts and sensors inside the vehicle's system thanks to capabilities like USB, Ethernet, and Wi-Fi communication. The Linux-based operating system of the Raspberry Pi has a user-friendly interface and supports a large ecosystem of software tools and libraries, making it a popular option for the creation of wireless vehicles. Additional benefits that contribute to its widespread adoption are its accessibility, adaptability, and broad community support. It is important to keep in mind, nevertheless, that the Raspberry Pi's somewhat higher power consumption compared to other microcontrollers may have an effect on the wireless driving vehicle system's overall energy efficiency.[20]

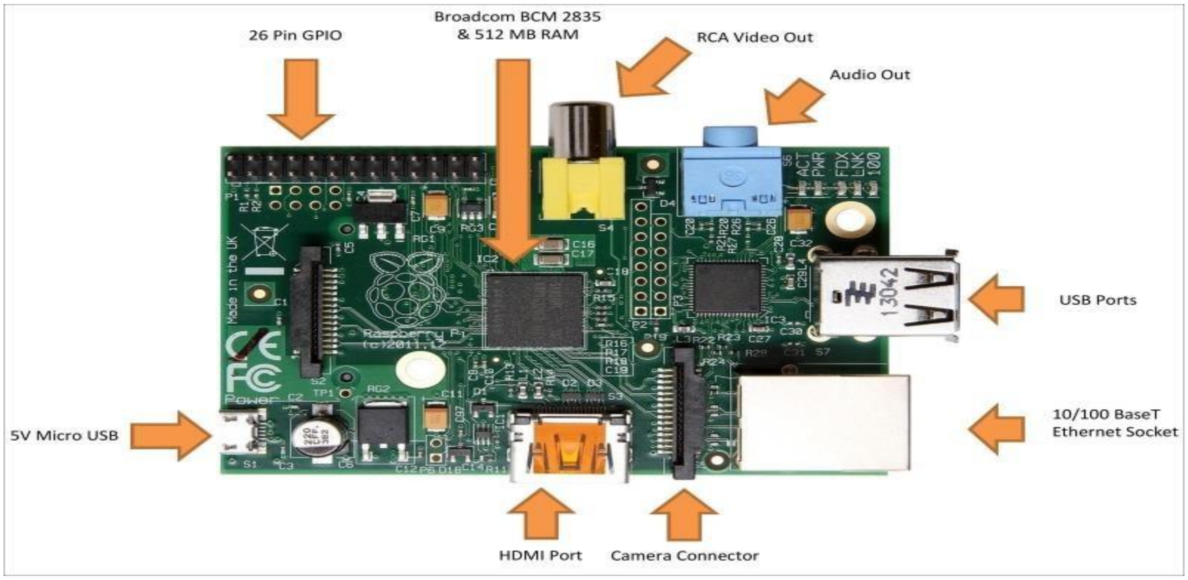


Figure 2.9 : The ports available in Raspberry Pi[20]

The NVIDIA Jetson platform, which was created expressly for AI and deep learning activities, is another common microcontroller in wireless driving car applications.[21] The Jetson platform efficiently processes the complicated algorithms and neural networks required for autonomous driving thanks to its strong GPU acceleration capabilities. It offers multiple connecting methods, including USB, Ethernet, and Wi-Fi, making it easier to integrate with other parts. One of the main benefits of the Jetson platform is its outstanding computational capacity, which is tailored for AI workloads and real-time processing. For some applications, the higher cost of Jetson modules, however, can be a barrier.[22]

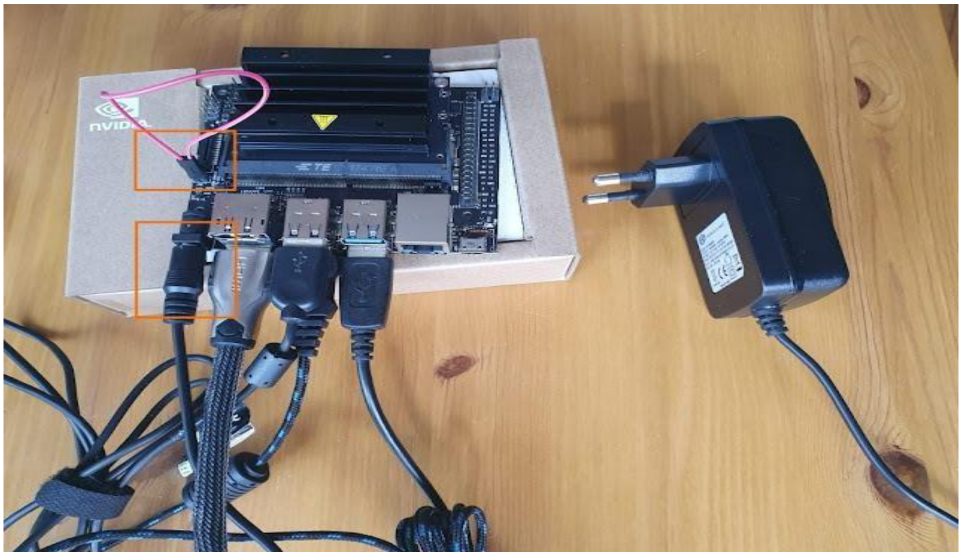


Figure 2.10 : Nvidia jetson nano connectivity[22]

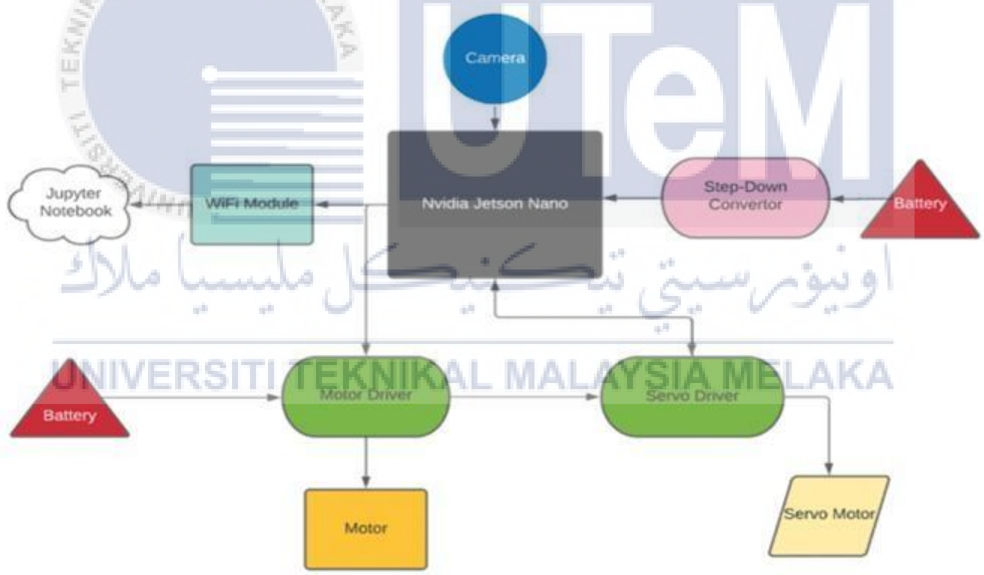


Figure 3. Block Diagram

Figure 2.11: Flowchart for Self-Driving Car Using Nvidia jetson Nano[22]

Due to its simplicity and usability, Arduino, a well-known microcontroller platform, finds use in wireless driving car projects.[20] Users can choose the best option for their unique needs from a variety of boards that offer a variety of features and capabilities. With the availability of connectivity options like USB and serial transmission, Arduino boards make it simple to integrate sensors and actuators.[23] Arduino is a well-liked option, especially for novices and prototyping needs, because to its accessibility, significant community support, and availability of a large library of pre-built code and examples. However, dealing with more complicated wireless driving applications may be difficult due to the Arduino's constrained processing speed and memory capacity.[24]

Wireless driving automobile systems frequently use STMicroelectronics STM32 microcontrollers, which are based on the ARM Cortex-M architecture.[25] These microcontrollers are ideal for resource-constrained applications due to their high performance and energy economy. There is no interruption in contact with other system components thanks to support for a number of communication interfaces, including USB, Ethernet, CAN, and I2C. Low power consumption, a wide range of peripheral possibilities, and a robust ecosystem of development tools and libraries are just a few benefits of STM32 microcontrollers. STM32 microcontrollers have a complicated architecture and a high learning curve, which may be difficult for novices.

A microcontroller module called Intel Edison, created especially for Internet of Things and embedded applications, features a small form factor, a potent Intel Atom processor, and integrated Wi-Fi and Bluetooth connectivity.[26] Its ability to run a Linux-based operating system gives wireless driving applications a comfortable environment to develop in. The Intel Edison module is appropriate for integration into small wireless vehicle systems due to its small size and low power consumption. When handling computationally intensive activities, the limited processor power and memory space may present limitations.

Figure 2.12 : The intel edison ports availability[26]

Table 2-3 : Comparison of the microcontrollers

Microcontroller	Advantages	Disadvantages
Raspberry Pi module	- Multiple connectivity options (USB, Ethernet, Wi-Fi) for easy integration with vehicle systems. - User-friendly interface and extensive software ecosystem.	- Relatively higher power consumption compared to other microcontrollers, potentially impacting energy efficiency.

	- Accessibility, adaptability, and strong community support.	
NVIDIA Jetson	- Powerful GPU acceleration capabilities for processing complex algorithms and neural networks. - Multiple connectivity options for seamless integration. - High computational capacity tailored for AI workloads and real-time processing.	- Higher cost compared to other microcontrollers, which can be a barrier for some applications.
Arduino	- Wide range of boards with various features and capabilities to choose from. - Easy integration of sensors and actuators through USB and serial transmission. - Accessibility, significant community support, and availability of pre-built code and examples.	- Constrained processing speed and memory capacity, which may limit more complex wireless driving applications.

STMicroelectronics STM32	- High performance and energy efficiency for resource-constrained applications. - Support for multiple communication interfaces (USB, Ethernet, CAN, I2C). - Wide range of peripheral possibilities and robust development tools and libraries.	- Complicated architecture and high learning curve, which may be challenging for beginners.
Intel Edison	- Small form factor and integrated Wi-Fi and Bluetooth connectivity. - Ability to run a Linux-based operating system for comfortable development. - Suitable for integration into small wireless vehicle systems due to size and power efficiency.	- Limited processor power and memory space, which may impose limitations when handling computationally intensive tasks.

Arduino Uno is suitable for my wireless driving car project due to its simplicity, versatility, and extensive community support. The majority of degree-level final year projects may be completed with Arduino Uno because of its flexible connecting choices, compatibility with different sensors, and communication modules. Although it might not be as powerful as the Raspberry Pi or the NVIDIA Jetson, it is usually adequate for running control algorithms and real-time sensor data processing. The energy-efficient Arduino Uno is exceptional, using less power and enabling longer operation times.[23] It is the best option for students because of its sizable and vibrant community, which offers a wealth of resources for study and assistance. Additionally, the ease of use, availability, and affordability of the Arduino Uno increase its instructional value by enabling students to work on their projects without going over budget.

2.2.4 Wireless Camera System for Real-Time Monitoring of Vehicle

The integration of a camera system in a wireless driving car project offers significant advantages. A camera serves as an essential visual input, providing real-time monitoring of the car's surroundings. By connecting the camera to a monitor, users can observe the captured view, enhancing navigation, obstacle detection, and overall safety. Cameras enable better situational awareness and play a crucial role in improving the driving experience and minimizing potential risks.

Widely used camera systems that attach to computers or microcontrollers through a USB port are called webcams. The effectiveness of USB webcams for applications requiring real-time video streaming was assessed by the authors. They discovered that because USB webcams are plug-and-play devices, they provide simplicity and ease of integration. However, there can be differences in image quality and resolution between models, and some webcams can just offer a few configurability choices. Despite these drawbacks, USB webcams are widely used since they are inexpensive and work with a wide range of software and operating systems.[27]

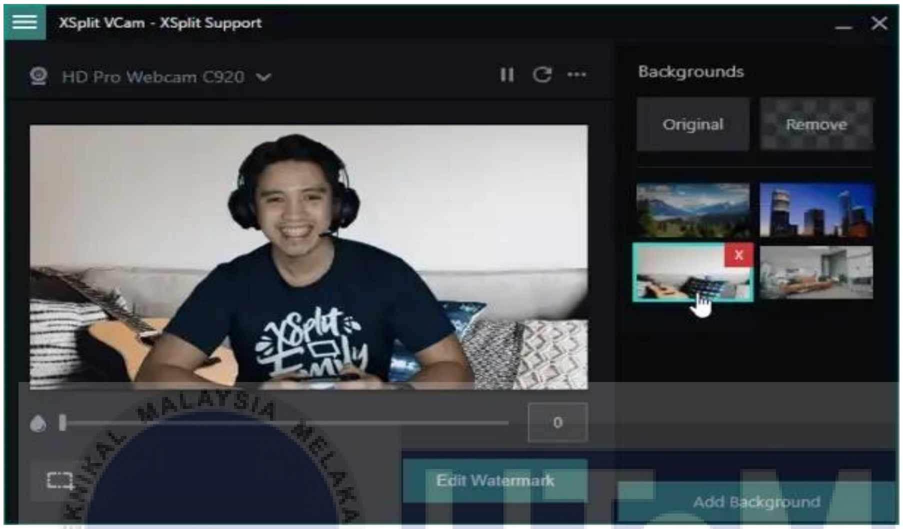


Figure 2.13 : The video streaming from webcam[27]

The Raspberry Pi Camera module is a camera system created especially for Raspberry Pi boards. The possibilities of the Raspberry Pi Camera module for computer vision applications were investigated by the researchers in their study. They claimed that the camera module, linked via the CSI connection, offered many resolution settings and gave good image quality. They also emphasised the accessibility of the official Raspberry Pi camera library, which enables simple camera configuration and control. This camera system is especially well suited for projects that call for a small, powerful camera solution that works with the Raspberry Pi ecosystem.[28]

Figure 2.14 : The video streaming of from Raspberry Pi[28]

IP cameras are network-based surveillance cameras that are frequently utilised for security applications, as highlighted in the review article.[29] These cameras include functions including remote access, motion detection, and recording and link to a local network. According to this article, IP cameras offer better image quality and resolution than conventional analogue cameras. In addition, IP cameras can be accessed and managed remotely using specialised software or mobile apps. However, the authors also emphasised the potential need for additional hardware, such as video encoders or network video recorders (NVRs), for full integration into a project.

Smartphone cameras have drawn attention as adaptable image tools for numerous applications. The authors of a work in the article investigated the usage of smartphone cameras in studies. They talked on how easily accessible and high-quality smartphone cameras are, emphasising how they can record real-time video. The essay focused on using smartphone camera

APIs and developer kits to programmatically access the camera feed. To ensure seamless integration, the authors also mentioned the requirement for communication protocols to be set up between the smartphone and the controlling device.[30]

Computer vision cameras provide advanced features beyond those of conventional cameras. This research assessed the effectiveness of Intel RealSense cameras for applications requiring depth perception. They discovered that computer vision cameras offer better perception capabilities in addition to high-quality imaging and depth information. In order to take full advantage of the sophisticated features of computer vision cameras, the authors also emphasised the significance of adopting specialised software libraries and development frameworks. However, it should be mentioned that these cameras could cost more than regular cameras and call for specialised knowledge.[31]

Figure 2.15: Computer vision camera kit[31]

The ESP32-CAM is an integrated camera module built around the ESP32 microcontroller. In a comparison of various camera modules for Internet of Things applications, the selected article discovered that the ESP32-CAM provides a simple and affordable solution. They mentioned that the module has built-in Wi-Fi and Bluetooth connectivity and supports resolutions up to 2 megapixels. The authors also emphasised how easy it is to use the Arduino IDE to programme the ESP32-CAM, making it available to developers already familiar with Arduino. They did admit, though, that the ESP32-CAM might be less capable of processing images than more specialised camera systems.[32]

Figure 2.16 : Video streaming from ESP32-CAM[32]

Table 2-4 : Comparison of camera systems

Camera System	Advantage	Disadvantages
USB webcams	- Plug-and-play integration, affordability, compatibility with various software and operating systems.	- Potential differences in image quality and resolution, limited configurability options.
Raspberry Pi Camera	- High image quality, multiple resolution settings, easy integration with Raspberry Pi ecosystem.	- Limited to Raspberry Pi boards, may require additional modules for specific applications.
IP cameras	- Better image quality and resolution, remote access and management capabilities.	- Potential need for additional hardware, such as video encoders or NVRs, for full integration.
Smartphone cameras	- Easily accessible, high-quality imaging, real-time video recording capabilities.	- Communication protocols setup required, limited programmability compared to other cameras.
Computer vision cameras	- Advanced features including depth perception, high-quality imaging and depth information.	- Higher cost, specialized knowledge required, dependency on software libraries and frameworks.

ESP32-CAM	- Simple and affordable solution, built-in Wi-Fi and Bluetooth connectivity, easy Arduino programming.	- Potentially less capable of image processing compared to specialized camera systems.

The ESP32-CAM offers distinctive advantages over other camera modules, making it a tempting option for applications. Cost-effectiveness is one of its main benefits because it offers a reasonable option without sacrificing necessary features. This pricing is especially advantageous for projects with a limited budget because it frees up resources for other project elements.

The ESP32-CAM's inbuilt Wi-Fi and Bluetooth connectivity are another noteworthy benefit. The ESP32-CAM simplifies the overall design by offering built-in connectivity choices, in contrast to other camera modules that require extra modules for wireless communication. This lowers costs while also streamlining the project's wireless connectivity installation.

The ESP32-CAM is well-suited for wireless car systems due to its integrated camera module, built-in Wi-Fi and Bluetooth connectivity, and support for up to 2-megapixel resolutions. It offers a simple and affordable solution, and with the ability to program it using the Arduino IDE, it becomes accessible to developers familiar with Arduino.

2.1 Software

Selecting the appropriate software is crucial for the success of my project. With a wide range of options available, finding the right software solution can streamline development and enhance functionality.

The Arduino IDE is a popular tool for programming Arduino microcontrollers in the DIY electronics sector. An article compared various programming environments for Arduino-based projects in their study. They discovered that the Arduino IDE has an intuitive user interface, making it easier to build and upload code to Arduino boards. The Arduino programming language, which is based on C/C++, is compatible with it, making the integration of sensor inputs, motor controls, and wireless communication modules simple. Additionally, the large Arduino online community offers a plethora of information and assistance to both novice and seasoned developers.[33]

Python is a flexible programming language that is frequently employed in IoT and scientific computing applications. The usage of Python for robotics and autonomous systems was investigated by the authors in a review article. The ease of use, readability, and extensive library collection of Python for data manipulation, machine learning, and computer vision tasks were praised. Python's adaptability enables smooth integration with wireless communication modules, microcontrollers, and sensors. Its large community and active development also guarantee consistent updates and a variety of online resources.[34]

OpenCV is a well-known open-source computer vision library that offers a wide range of tools and methods for processing images and videos. A study evaluated OpenCV's effectiveness for real-time object detection in autonomous driving systems. They claimed that OpenCV's powerful features, including object tracking, feature detection, and image filtering, allow for the effective processing of camera input. Developers have freedom thanks to the C++ and Python interfaces of the package. To assure real-time performance on devices with limited resources, the authors issued a warning about OpenCV's vast features.[35]

In the area of robotics and autonomous systems, TensorFlow has become a popular machine learning framework. A research examined the application of TensorFlow for deep learning-based object detection in autonomous vehicles in their research article. They discovered that TensorFlow's computational graph architecture and improved algorithms make neural network deployment and training efficient. The framework offers tools for model translation and optimisation along with support for a number of hardware platforms. The authors did point out that TensorFlow may need a lot of processing power because to its broad features, therefore strong hardware platforms are better suited for it.[36]

A well-known platform for creating robot applications, including autonomous cars, is the ROS (Robot Operating System). The authors of the article I found conducted a thorough analysis of ROS's properties and applications in autonomous driving research. They emphasised ROS's message-passing system, modular architecture, support for a range of sensors, actuators, and control algorithms. Rapid prototyping and testing are made possible by the visualisation tools and simulation capabilities of the framework. However, the authors pointed out that because of its extensive ecosystem, ROS has a learning curve and can use more computing power than frameworks with a simpler architecture.[37]

A popular software environment for scientific computing and algorithm development is MATLAB. The authors examined the use of MATLAB for robotics and autonomous systems in a review paper. They highlighted how MATLAB's many toolboxes for machine learning, control systems, and image processing allow for quick development and prototyping. The interactive debugging, visualizations, and easy syntax of MATLAB improve the development process. The authors did note that MATLAB can be resource-intensive and that additional licencing fees for particular toolboxes might be necessary.[38]

Data acquisition, instrument control, and industrial automation frequently employ the graphical programming environment known as LabVIEW. The author analyzed LabVIEW's suitability for autonomous systems in their research study. They emphasized LabVIEW's visual programming approach, which uses a drag-and-drop interface to make developing control algorithms and integrating hardware easier. Rapid prototyping is made possible by the wide library of prebuilt functions and instrument drivers in LabVIEW. The authors did note that programmers used to text-based languages would find LabVIEW to have a more difficult learning curve.[39]

Table 2-5 : Comparison of software's

Software	Advantage	Disadvantage
Arduino IDE	- Intuitive user interface for easy code development and uploading. - Compatibility with Arduino programming language simplifies integration with sensors, motor controls, and wireless modules. - Large online community provides ample support and resources.	- Limited functionalities compared to more advanced programming environments. - May not be suitable for complex and computationally intensive projects.
Python	- Flexible and versatile language for IoT and scientific computing applications. - Readable code and extensive library collection for data manipulation, machine learning, and computer vision.	- Slower execution speed compared to lower-level languages. - Greater memory usage compared to more optimized languages.

	- Smooth integration with wireless modules, microcontrollers, and sensors. - Active community and continuous development ensure ongoing updates and resources.	
OpenCV	- Powerful computer vision library with a wide range of tools for image and video processing. - Object tracking, feature detection, and image filtering capabilities for effective camera input processing. - C++ and Python interfaces provide flexibility for developers.	- Vast range of features may require significant computational resources. - Steep learning curve for complex functionalities.
TensorFlow	- Popular machine learning framework for deep learning-based applications. - Efficient deployment and training of neural networks	- Demands significant processing power due to its extensive features.

	with computational graph architecture and improved algorithms. - Support for model translation and optimisation, as well as various hardware platforms.	- Complex setup and configuration compared to simpler frameworks.
ROS (Robot Operating System)	- Message-passing system and modular architecture for flexible development. - Support for various sensors, actuators, and control algorithms. - Visualization tools and simulation capabilities aid rapid prototyping and testing.	- Steeper learning curve due to the extensive ecosystem. - Higher computational requirements compared to simpler frameworks.
MATLAB	- Extensive toolboxes for machine learning, control systems, and image processing. - Quick development and prototyping with interactive debugging and visualizations.	- Resource-intensive and may require additional licensing fees for specific toolboxes. - Not open-source, limiting flexibility and customization options.

	- Easy syntax and comprehensive support for algorithm development.	
LabVIEW	- Visual programming approach with a drag-and-drop interface for easier development and hardware integration. - Extensive library of prebuilt functions and instrument drivers for rapid prototyping. - Suitable for data acquisition, instrument control, and industrial automation applications.	- Higher learning curve for programmers accustomed to text-based languages. - Limited flexibility compared to text-based programming languages.

For my project integrating the ESP32, Arduino microcontroller, and ultrasonic sensor, the Arduino IDE stands out as the best acceptable alternative among the software solutions for developing autonomous systems. It is the best choice due to its intuitive user interface, compatibility with Arduino hardware, wide range of library support, and active online community. A seamless development environment is provided by the Arduino IDE, making it easier to write, compile, and upload code. Its broad library and sample collection make sensor integration and speedy prototyping possible. For difficulties, the internet community offers helpful tools and assistance. Although there are advantages to alternative options, the Arduino IDE is a preferable option for my project because to its ease of use, accessibility, and community support.

2.3 Summary

In the literature review, various components for wireless car systems were extensively examined. The selection of the accelerometer as a sensor was based on its ability to measure acceleration and provide crucial data for vehicle dynamics. Its compact size, cost-effectiveness, and compatibility with microcontrollers made it a suitable choice.

The RF wireless communication module was praised for its versatility and widespread use in wireless car systems. It offered reliable and efficient data transmission, enabling seamless communication between the car and external devices. Comparisons with other wireless communication techniques highlighted its advantages such as longer range, lower power consumption, and ease of integration.

The Arduino Uno microcontroller was found to be a popular choice due to its user-friendly interface, compatibility with a wide range of sensors and actuators, and extensive community support. Its programming environment, Arduino IDE, provided a hassle-free development process and allowed for easy integration of wireless communication modules and other components.

The ESP32-CAM module stood out as an integrated camera solution, suitable for capturing real-time video in wireless car systems. Its built-in Wi-Fi and Bluetooth connectivity, along with support for higher resolutions, made it a convenient choice. Programming the ESP32-CAM with the Arduino IDE offered a familiar and accessible development environment for developers.

The Arduino IDE software itself was recognized for its intuitive user interface, making code development and uploading seamless. The compatibility with the Arduino programming language enabled the integration of sensor inputs, motor controls, and wireless communication modules with ease. The vast online community provided valuable resources and assistance to both novice and experienced developers.

Overall, the literature review revealed that careful consideration of sensor selection, wireless communication modules, microcontrollers, camera systems, and software tools is vital for building efficient and functional wireless car systems. The chosen components, including the accelerometer, RF wireless communication module, Arduino Uno, ESP32-CAM, and Arduino IDE software, offer a comprehensive solution for developing wireless car systems.

CHAPTER 3 METHODOLOGY

3.1 Introduction

This chapter is focused on the project methodology used in the development of a remote driving control system for an electric vehicle using wireless technology. This chapter gives an in-depth description of the systematic approach adopted to design, implement, and test the various components of the system The methodology includes steps for choosing hardware, creating software, integrating systems, and testing them. The main objective is to make sure that the remote control system is implemented successfully and effectively. This chapter provides a summary of the research procedure while highlighting the steps taken to achieve the desired outcomes. The methodology section offers helpful insights into the development process and provides a guide for comprehending the project's methodical and logical progress.

3.2 Project Methodology

The basic components use in the development of a remote driving control system for an electric vehicle using wireless technology is designate in the block diagram. After the study on the parameters needed for this project is known, the assortment of hardware and software was proceed. All the collected data are the basic to determine the selection of the hardware and software part. The model of the Remote control Driving Car will be sketched using the Computer-aided Design (CAD) software. The circuit.io software will be use to design the circuit along with the Arduino.ide software. The circuit will be test in the simulation first to certify its functionality. The Arduino software will be use to decrypt data from the wireless module. Besides that, it will be use in the communication protocol for this project.

3.2.1 Progress Flow Chart

The project's sequential progression is represented visually in a progress flow chart, which includes crucial phases including simulation, hardware connections, testing, and the completion of all software and hardware components. It acts as a road map, defining the actions needed to achieve the project's goals and objectives. The flow chart showcases the iterative nature of the project, allowing for adjustments and refinements along the way. I can efficiently manage tasks, track progress, and guarantee that the project's objectives are achieved by using the flow chart. Basically the research and literature review on the main component or the part which will be used to in the Development of Remote Driving Control of an Electrical Vehicle Using Wireless Technology. This flow chart also will show the test simulation of the chosen software. Figure 3.1 shows the progress flow chart.

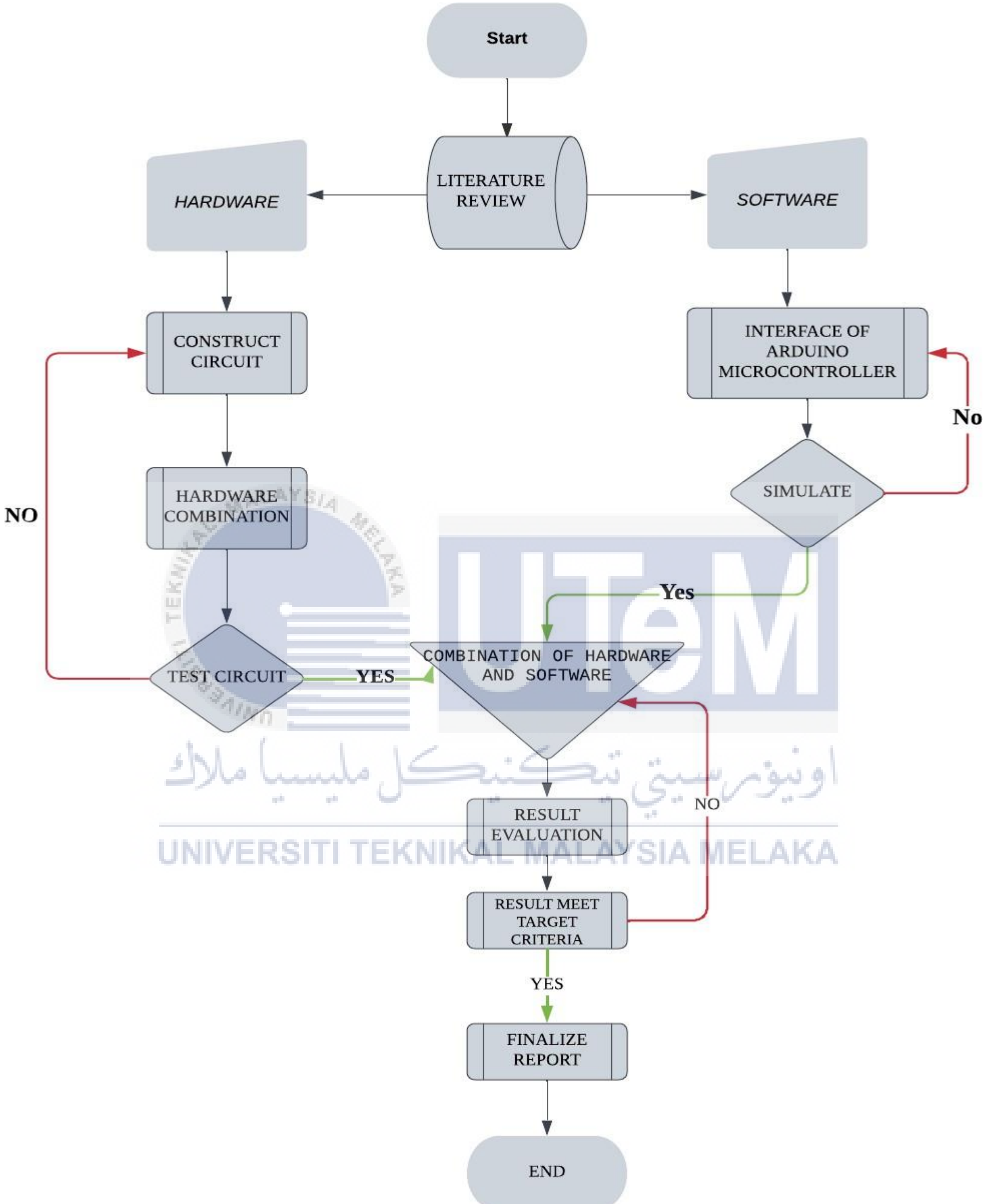


Figure 3.1 : Progress flow Chart

3.2.2 Car System Block Diagram

The primary objective of the project methodology's system design phase is on developing a comprehensive strategy that details the relationships between various hardware elements and how they work together to enable wireless remote control capabilities. The basis for the effective installation of the wireless remote driving control system is laid during this vital phase. The development of a block diagram or flowchart that illustrates the various system components and their relationships is the first step in the system design process. The development of a remote driving control system for an electric vehicle using wireless technology in Figure 3.2 and Figure
3.3 is the simple block diagram based on the component needed. The elements in the block diagram play an important role in order to fulfil the design of this project.

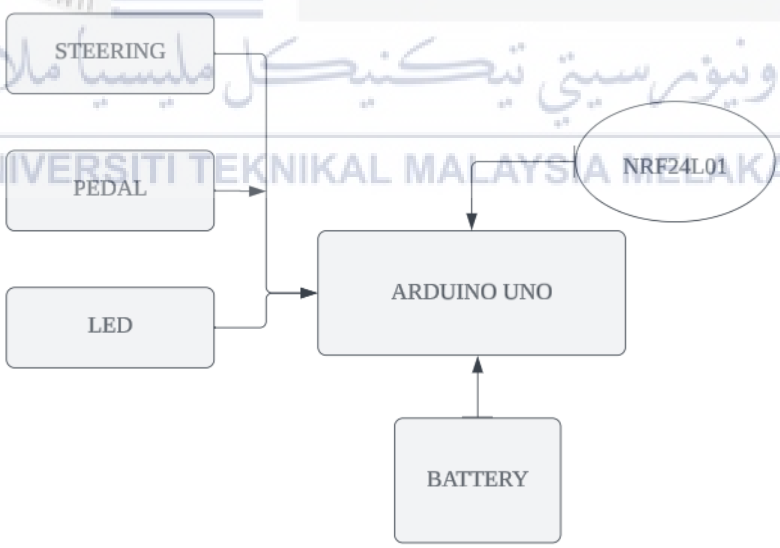


Figure 3.2 Controller System Block Diagram

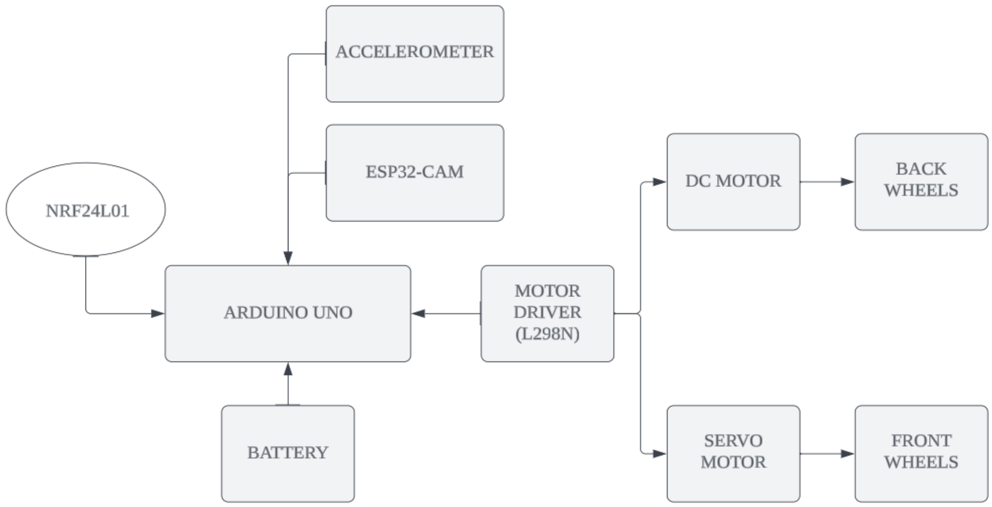


Figure 3.3 Car Body System Block Diagram

The Arduino Uno, which acts as the primary control component, is at the heart of the system. The user can direct the car's direction and speed potentiometers, which serve as input devices. The RF module makes it possible for the remote control device and the car body to communicate wirelessly.

The parts are wired in a certain way to enable wireless remote control functionality. The Arduino Uno is interfaced with the potentiometers, which reads the input values from these components. The RF module receives the relevant commands from the Arduino Uno after it has processed the input signals. Another Arduino Uno is attached to the RF receiver module on the car side, which is where the remote control object's signals are received. These signals are decoded by the Arduino on the vehicle's side, which then decides whether to accelerate, brake, turn left, or turn right.

The system design should also take other elements into account, like the ESP32-CAM for obstacle detection and the motor driver module for operating the vehicle's motors. The accelerometer detects overspeed and alerts the driver through the LED blinking. These parts improve the system's usability and safety. To accomplish the intended wireless remote control capabilities, the system design step makes sure that all the hardware components are connected appropriately and work together seamlessly. The subsequent implementation and programming phases can go more smoothly if the system is designed with a clear grasp of the relationships between and responsibilities of each component.

Overall, the system design phase offers a roadmap for creating and putting the wireless remote driving control system into action. By directing the following steps in the project methodology, it lays the groundwork for the project's effective development. The figure 3.4 represents the flowchart for the working flow of my project.

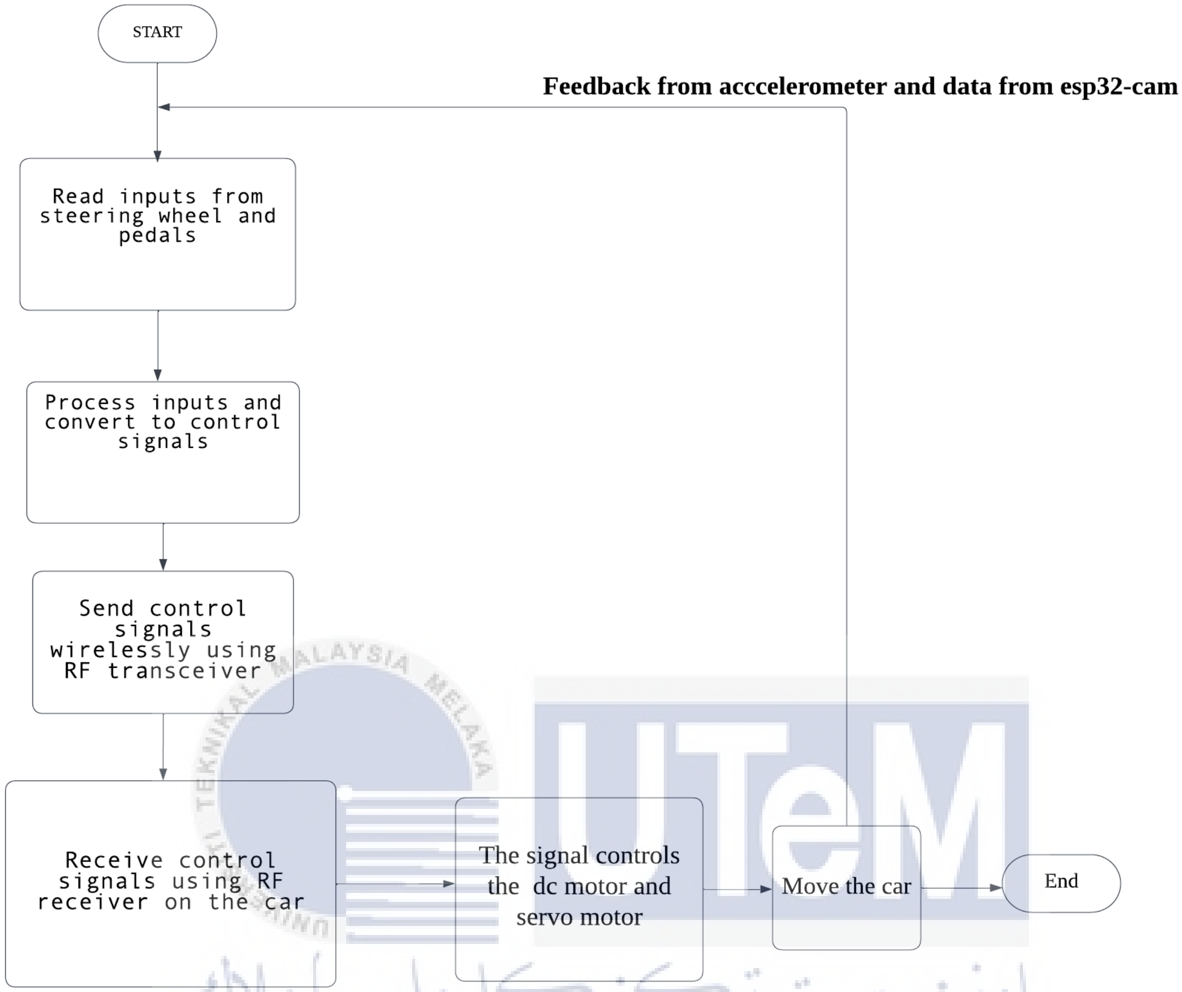


Figure 3.4 : Flowchart for the working process of the project

3.2.3 Hardware Component

In the hardware components section of the project methodology, the focus is on identifying and selecting the key components that are essential for the functioning of the wireless remote driving control system. These hardware components play a critical role in enabling wireless communication, capturing user inputs, and controlling the movement of the car.

3.2.3.1 **Arduino UNO**

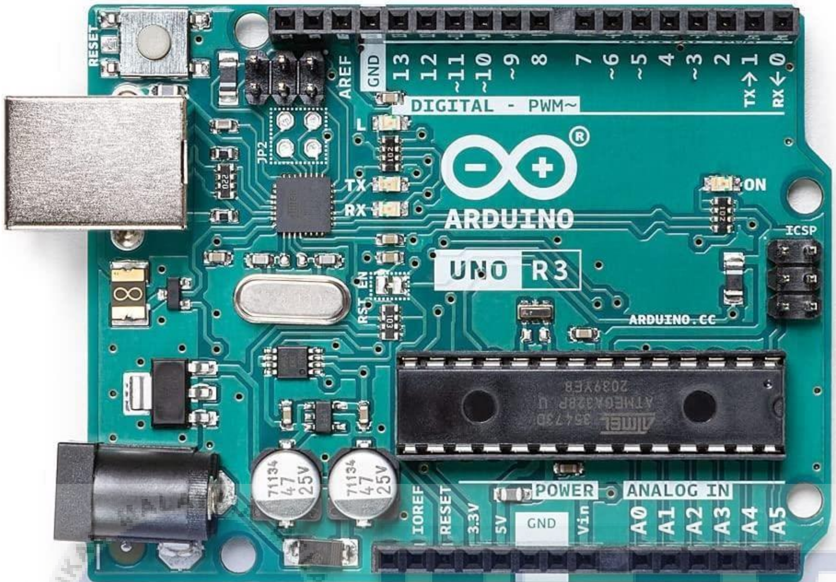


Figure 3.5 : Arduino Uno Microcontroller

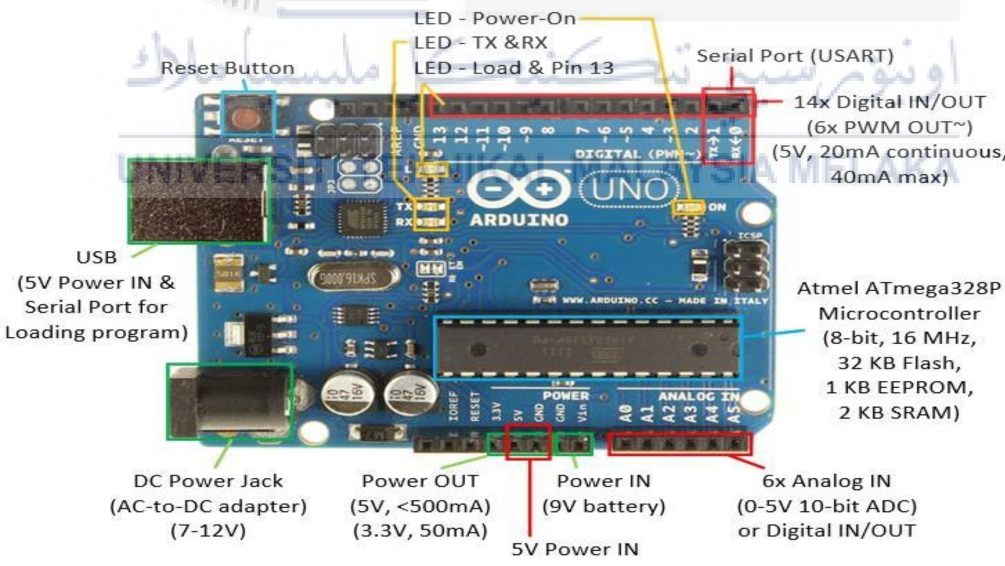


Figure 3.6: Arduino UNO Microcontroller Description

The Arduino Uno microcontroller is a versatile and widely used board in the field of embedded systems. It is equipped with a range of input and output ports that enable seamless integration with various hardware components. The board features 14 digital input/output pins, among which 6 can be used as PWM (Pulse Width Modulation) outputs, allowing precise control of analog-like signals. Additionally, there are 6 analog input pins for reading analog sensor values. The board also offers serial communication ports, including a USB port for programming and data transfer.

To connect the Arduino Uno with other components, various types of wires and cables can be used. For example, male-to-female jumper wires can be utilized to establish connections between the Arduino pins and other modules or sensors. Additionally, USB cables are used for connecting the Arduino Uno to a computer, enabling code uploading and power supply. The Arduino IDE (Integrated Development Environment) is the software used to program and upload code to the Arduino Uno. It provides a user-friendly interface for writing, compiling, and debugging code. The IDE supports the Arduino programming language, which is based on C++. It offers a wide range of libraries and functions that simplify the programming process, making it accessible for beginners and experienced users alike.

Microcontroller	Atmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (Atmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (Atmega328)
EEPROM	1 KB (Atmega328)
Clock Speed	16 MHz

Table 3-1: Characteristics of Arduino UNO Microcontroller

3.2.3.2 RF Transceiver module

Figure 3.7 RF transceiver module

The RF transmitter and receiver modules are key components in establishing a wireless communication link between the remote control object and the car body. The RF transmitter module, connected to the Arduino board, converts the electrical signals generated by the board into radio waves. These radio waves are then transmitted through the module's antenna. On the car side, the RF receiver module captures these radio waves using its own antenna. It then demodulates the received signals, converting them back into electrical signals that can be processed by the Arduino board. This allows the Arduino to receive and interpret the control commands sent from the remote control object. The RF transmitter and receiver modules enable seamless and reliable transmission of signals, eliminating the need for physical connections between the remote control and the car body. This wireless communication system enhances the mobility and convenience of the remote driving control system, providing a flexible and efficient means of controlling the car's movements.

3.2.3.3 Motor Driver: L298N

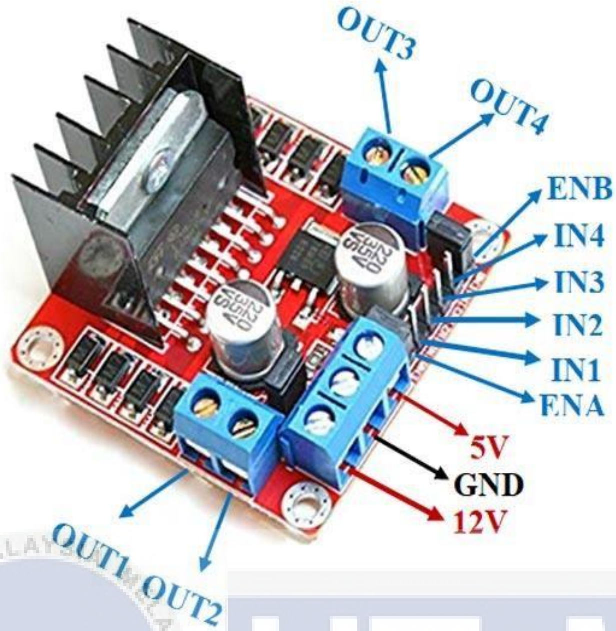


Figure 3.8: Motor Driver L298N

The L298N motor driver module is essential for managing the car's motion. This motor driver is a dual H-bridge that can manage the high current demands of DC motors. The Arduino board is linked to the module's four input pins so that it may receive control signals. The Arduino is able to regulate the motors' speed and direction by delivering the proper signals to these pins. Additionally, the L298N module has four output pins that are linked to the DC motors. Based on the control signals the Arduino transmits, these output pins are in charge of providing the motors with the necessary power. The module allows the car to go forward, backward, left, or right by modifying the voltage and polarity of these output pins. So as the pedal accelerates or deaccelerates the signal received by the L298N before adjusting the dc motor behavior.

The L298N module also includes built-in safety features including diodes and thermal overload protection to guarantee the motors' safe functioning and guard against module damage. Controlling the car's DC motors is made simple and effective by the L298N motor driver module. It provides motor control flexibility and improves overall performance and mobility of the vehicle.

3.2.3.4 DC Motor

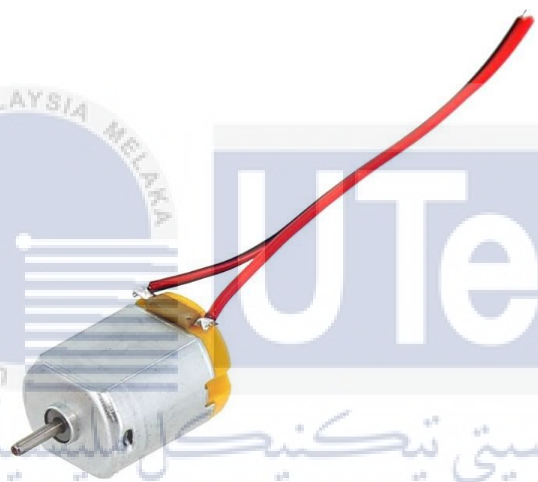


Figure 3.9: 12V DC Motor

DC motors, which convert electrical energy into mechanical motion, are crucial parts of the automotive. A rotor, stator, and commutator make up these motors. The motor rotates when an electric current is applied because the magnetic field it produces interacts with the permanent magnets on the rotor. Two DC motors are commonly employed in the context of a car, one for each side of the vehicle. The motor driver module, such as the L298N, which regulates the speed and direction of these motors, is attached to them. The motor driver module adapts the voltage and polarity applied to the motors in response to signals received from the Arduino board.

The movement of the car is controlled by the DC motors' rotation. The motors' rotational speed and direction can be adjusted to cause the automobile to travel forward, go backward, turn left, or turn right. The motors' required electrical energy is supplied by the power source, which can be a battery or an external power source. It is crucial to choose DC motors that are appropriate for the intended application, considering torque, speed, and power needs into account. The right motor choice guarantees the car's optimal performance and efficient operation. So in the case of the project the dc motor will be connected in the back wheel with the motor driver. So as the pedal value increases the motor driver receives and derives it to the specific speed for the dc motor to rotate.

3.2.3.5 Servo motor

Figure 3.10: Servo motor

A servo motor is a type of motor commonly used in robotics and automation applications, including controlling the front wheels of a wireless car. It provides precise control and positioning capabilities, making it ideal for tasks that require accurate movement. Here's an explanation of how servo motors work, their components, connectivity, important criteria, and how to connect them.

The servo motor consists of three main components: a DC motor, a control circuit, and a feedback system. The DC motor provides the mechanical power, while the control circuit and feedback system ensure accurate positioning. The control circuit receives control signals from a microcontroller or a servo driver and determines the position and direction of rotation. The feedback system, usually in the form of a potentiometer or an optical encoder, continuously monitors the position of the motor shaft.

Connectivity for a servo motor typically involves three wires: power (+5V), ground (GND), and the control signal. The control signal is a pulse-width modulation (PWM) signal, which determines the desired position of the motor shaft. The width of the PWM signal's pulse corresponds to the desired angle, typically ranging from 0 to 180 degrees.

When connecting a servo motor, you need to ensure that the power supply is capable of providing sufficient voltage and current for the motor's requirements. It is essential to connect the power and ground wires correctly, as an incorrect connection may damage the motor or the controlling circuitry.

To control the servo motor, you need to generate the appropriate PWM signal. This can be done using a microcontroller or a servo driver module, which simplifies the control process by providing a convenient interface and handling the PWM generation. The microcontroller or driver module allows you to set the desired angle of the servo motor and send the corresponding PWM signal.

When connecting the servo motor to the front wheels of your wireless car, you will need to attach the motor shaft to the steering mechanism. The servo motor's precise control will allow you to accurately control the direction and angle of the front wheels, enabling smooth steering and maneuvering of the car. A servo motor is an excellent choice for controlling the front wheels of my wireless car due to its precise positioning capabilities.

3.2.3.6 Battery : Type LiPo

Figure 3.11 : Battery : LiPo

A lithium polymer (LiPo) battery is a type of rechargeable battery commonly used in electronic devices and robotics projects. LiPo batteries are known for their high energy density, lightweight design, and ability to deliver high currents. These characteristics make them a popular choice for applications that require a compact and powerful energy source.

LiPo batteries consist of multiple cells, with each cell typically providing a nominal voltage of 3.7 volts. The total voltage of the battery depends on the number of cells connected in series. Common configurations include 2S (two cells in series) or 3S (three cells in series), resulting in a nominal voltage of 7.4V or 11.1V, respectively. One of the key advantages of LiPo batteries is their ability to deliver high discharge rates. This makes them suitable for powering motors that require a significant amount of current, such as the DC motors used in the car project. It is important to choose a LiPo battery with a discharge rate (often denoted as "C rating") that can handle the current requirements of your motors.

3.2.3.7 Light Emitting Diode(LED)

Figure 3.12: Light Emitting Diode(LED)

Incorporating an LED indicator into the wireless remote-controlled car system serves as a crucial visual cue for the driver, providing real-time feedback on the vehicle's speed conditions detected by the MPU6050 accelerometer. This LED will be connected to the controller system. The LED acts as an alert mechanism, specifically designed to illuminate when the acceleration values recorded by the MPU6050 surpass a predefined high-speed threshold.

This threshold is programmatically determined based on the desired sensitivity to high-speed vibrations. The LED is connected to a dedicated digital pin on the Arduino microcontroller, allowing for precise control of its state. Physically, the LED comprises an anode and cathode, with the anode connected to the chosen digital pin and the cathode linked to a current-limiting resistor, which, in turn, connects to the ground (GND) pin on the Arduino. The LED indicator function is encapsulated within the code, activating the LED upon detection of high-speed conditions. The illuminated LED provides an immediate visual signal to the driver, allowing them to adjust the speed of the remote-controlled car accordingly, ensuring a responsive and user-friendly driving experience.

3.2.3.8 Potentiometer

Figure 3.13: Potentiometer

A potentiometer, also known as a variable resistor or pot, is an electronic component that allows you to control the resistance within a circuit by adjusting its position. It consists of a resistive track and a movable contact, typically controlled by a rotary knob or slider.

In my project, potentiometers would be used to provide analog input for controlling the steering, pedal and brake functions. When I press the pedal or apply the brake, the potentiometer's position changes, altering the resistance in the circuit. The Arduino board can read this varying resistance and convert it into corresponding values, allowing me to control the speed or braking intensity of the car. In the context of controlling the steering and direction of the vehicle, the potentiometer acts as an input device. When connected to the steering mechanism, the potentiometer detects the rotational movement of the steering wheel and converts it into an electrical signal. This signal is then sent to the Arduino board, which interprets the position of the steering wheel and adjusts the angle of the servo motor accordingly.

Potentiometers usually have three terminals: the two outer terminals connect to the power supply or ground, while the middle terminal is the wiper that moves along the resistive track. By connecting the outer terminals to the power supply and ground, and the middle terminal to an analog input pin on the Arduino board, you can measure the voltage at the wiper and determine the position of the potentiometer.

The Arduino board can then use analog-to-digital conversion to convert the voltage into a digital value, providing precise control over the car's acceleration and braking. By mapping the analog input values to desired speed or braking levels, I can create a smooth and responsive control system.

3.2.3.9 Accelerometer

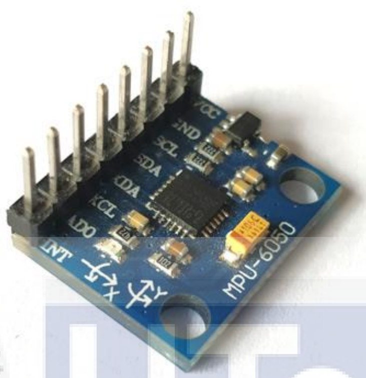


Figure 3.14: Accelerometer- MPU6050

An accelerometer is a vital component in your project as it enables the measurement of acceleration forces and changes in orientation. Placing the accelerometer strategically within the electrical vehicle allows for accurate monitoring of its movements. Typically, it is positioned at the vehicle's center of gravity to obtain precise readings. Sensitivity adjustment is crucial to ensure the accelerometer captures the desired range of motion effectively. The accelerometer outputs raw acceleration data, which requires processing and interpretation to derive meaningful information. This involves converting analog signals into a digital format and applying algorithms to extract relevant details like speed, direction, or changes in vehicle orientation. Calibration is essential to eliminate biases and errors, ensuring accurate measurements. Integrating the accelerometer data with the vehicle's control system allows for informed decisions on speed control, stability, and maneuvering. Additionally, the accelerometer data can be leveraged to implement safety features by detecting sudden stops or collisions, triggering appropriate responses to enhance the vehicle's safety.

To connect an accelerometer to an Arduino Uno, power the module using the 5V and GND pins on the Arduino board. Then, connect the accelerometer's data pins (such as SDA and SCL) to the corresponding I2C ports (A4 and A5) on the Arduino Uno. If pull-up resistors are required, connect them to the SDA and SCL lines. Next, upload the code to the Arduino Uno, which includes the necessary libraries for reading data from the accelerometer. Finally, open the serial monitor to view the accelerometer data being transmitted by the Arduino Uno. So, the accelerometer will detect or sense the vibration of the car when it moves abnormally and send the data to the controller. At controller a LED will indicate the abnormal driving by the user. So, the user would adjust the speed or the rotation of the steering to overcome the problem.

3.2.3.10 Esp32-Cam

Figure 3.15 : Esp32 Cam

In my project, I'm using the ESP32-CAM module, a small camera module that combines an ESP32 microcontroller with a camera sensor. I can take pictures and films in high definition with this neat little gadget. I only need to connect the power, ground, and communication pins of the ESP32-CAM module to my Arduino board.

I use the Arduino IDE and the ESP32 board support package to programme the ESP32-CAM module. I have access to certain libraries and functions through this combination, which makes controlling the camera simple. Wi-Fi networking incorporated right into the ESP32-CAM module is one of its outstanding features. I can now wirelessly transmit any photos or videos I take. It makes remote monitoring and real-time video streaming possible. So the ESP32-CAM will be placed at the car which faces the front view. The user can access the video streaming of ESP32-CAM from the users static place. So from the streaming the user can detect obstacles and controls the controller to avoid it. The video streaming occurs through wifi network.

The ESP32-CAM module is small in size, making it the ideal fit for my project. It effortlessly connects with the Arduino environment, allowing me to use the pre-existing codebase and libraries to streamline development. The ESP32-CAM module brings a new level of capabilities, whether I want to apply computer vision algorithms or monitor the surroundings of my car.

3.2.4 Software

Software plays a crucial role in the development of your project as it enables to program and control the behavior of the hardware components. In the context of the wireless remote driving control system, software is essential for establishing communication between the remote control object and the car body, interpreting user inputs, and coordinating the actions of various components. I will be using the Arduino IDE (Integrated Development Environment) to program and control the behavior of the Arduino board.

3.2.4.1 Arduino IDE

The software component of my wireless remote driving control system is vital for enabling communication, interpreting user inputs, and coordinating the actions of the hardware components. I will be using the Arduino Integrated Development Environment (IDE) as my software platform, which provides an intuitive interface for programming and controlling the behavior of the connected components.

The Arduino IDE is a powerful tool that supports the C++ programming language and offers a range of libraries and functions specifically designed for Arduino boards. These libraries simplify the process of controlling hardware components, such as motors, sensors, and communication modules, allowing me to focus on defining the logic and functionality of my project.

Using the Arduino IDE, I will write code that interprets the signals received from the remote control object, processes user inputs from the joystick, push buttons, or other control mechanisms, and translates them into appropriate commands for the motor driver module. This code will define the logic for controlling the car's movements, including acceleration, deceleration, turning, and stopping.

The Arduino IDE also facilitates the compilation and uploading of my code. Once I have written my code, I can compile it to check for errors and convert it into a binary format that can be understood by the Arduino board. Once successfully compiled, I can upload the code to the Arduino board using a USB cable or other supported programming methods. This will allow the Arduino board to execute the programmed instructions and control the connected hardware components accordingly.

Furthermore, the Arduino IDE includes a Serial Monitor tool that enables real-time communication between the Arduino board and my computer. This tool will be invaluable for monitoring sensor readings, debugging my code, and exchanging data between the board and the computer during runtime. It provides a convenient interface for troubleshooting and ensuring the proper functioning of my remote driving control system.

In conclusion, the software aspect of my project, facilitated by the Arduino IDE, empowers me to program and control the behavior of the hardware components. Through code development, compilation, and uploading, I can define the functionality of my system and ensure smooth communication between the remote control object and the car body. The Arduino IDE's libraries and Serial Monitor tool enhance the development process by providing pre-written code, simplifying communication, and facilitating debugging. Together, these software components form the foundation for the successful implementation of my wireless remote driving control system.

3.2.5 Hardware Integration and Configuration

In the hardware integration and configuration phase, all the components of the wireless driving car system are interconnected to ensure seamless functionality. Each component serves a specific purpose and contributes to the overall operation of the system.

The Arduino Uno board acts as the main controller and serves as the foundation for connecting and controlling the various hardware components. It provides the necessary processing power and digital and analog input/output (I/O) pins for connecting and communicating with other modules.

The RF transmitter module is connected to the Arduino board and is responsible for wirelessly transmitting control signals from the remote control object. It receives inputs from the user controls, such as the steering wheel and pedals, and sends corresponding commands to the RF receiver module on the car side. On the car side, the RF receiver module is connected to another Arduino board. It receives the signals transmitted by the RF transmitter and interprets them to determine the desired actions, such as the direction and speed of the car. The receiver then sends instructions to the motor driver module, which controls the DC motors responsible for the car's movement.

Additional components, such as the servo motor, are used to control the left/right direction of the wheel, while the accelerometer module can be integrated to detect the car's acceleration and provide data for advanced control or feedback purposes. Furthermore, the system incorporates an ultrasonic sensor for obstacle detection, enhancing the safety and navigation capabilities of the car. The ESP32-CAM module is used to capture live video from the car's perspective, allowing the user to monitor the surroundings remotely.

By carefully connecting and configuring these components, the wireless driving car system achieves a comprehensive integration of hardware elements. The interconnection of these modules enables wireless control, precise movement, obstacle avoidance, and real-time video streaming, offering an immersive and dynamic driving experience. The below figure 3.16 shows how the hardware's will be integrated together.

Figure 3.16 : Hardware's integration

3.2.6 Circuit Integration

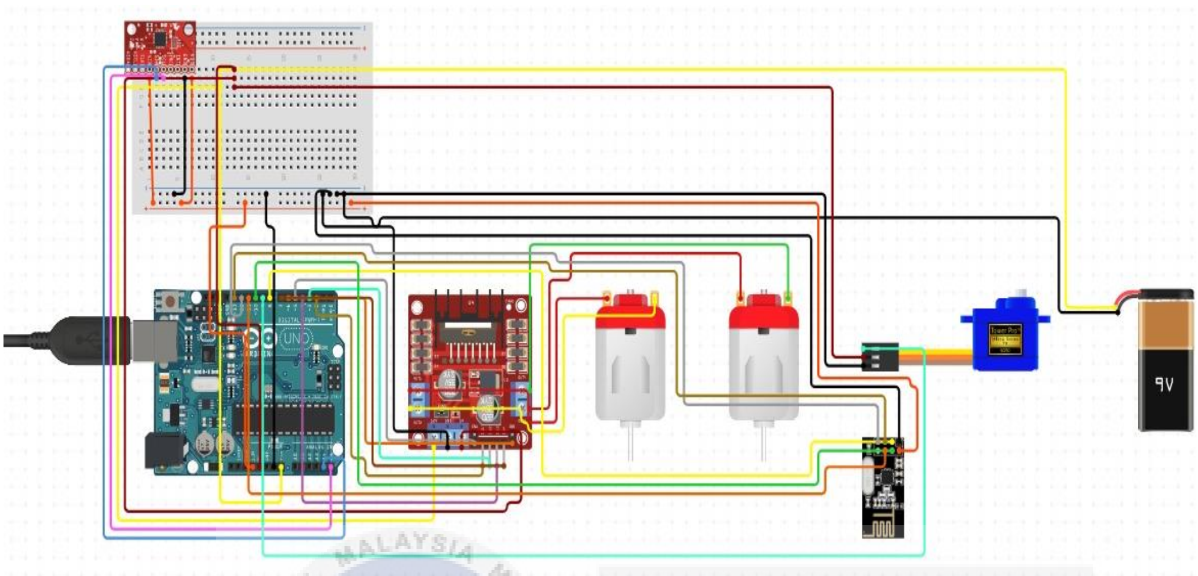


Figure 3.17: Circuit integration for car body

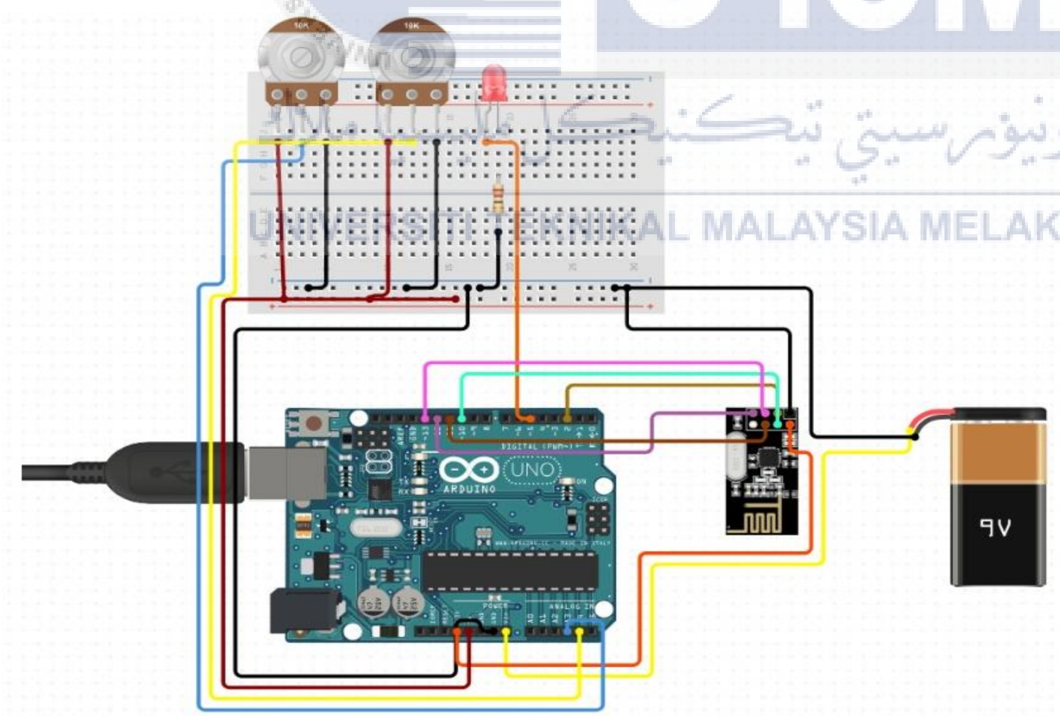


Figure 3.18: Circuit integration for remote control

3.3 SUMMARY

The methodology for this project involved the following key steps: requirement analysis, hardware selection, circuit design and construction, programming, integration and testing, performance evaluation, data analysis, and iterative refinement. The methodology ensured a systematic and organized approach to develop and implement the remote driving control system for the electrical vehicle. Safety considerations were prioritized throughout the project, and the system's performance was evaluated under various conditions. The collected data were analyzed to validate the achievement of project objectives and make necessary adjustments. Overall, the methodology provided a structured framework for the successful realization of the project.

CHAPTER 4 RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter unveils the outcomes of simulations and real-world experiments for our remote vehicle control system. Focusing on RF transmission, servo motor, and DC motor control through a potentiometer, alongside safety enhancements with ESP32-CAM and MPU-6050, we analyze these results. Delving into system components' performance in a virtual environment, this scrutiny guides us for further refinements. The analysis aims to reveal the feasibility and effectiveness of our wireless technology-based solution, setting the stage for the system's evolution.

4.2 Simulation and Experiment Results of wireless communication

In the realm of wireless communication for our remote driving control system, both simulated and experimental approaches were employed to validate the effectiveness of our designed setup. Utilizing Proteus, simulations involved connecting Arduino boards independently and later integrating them with NRF24L01 modules, displaying real-time results on an LCD interface. On the experimental front, a comprehensive connection setup was established between the controller and the receiver car. The serial monitor became a crucial tool, echoing transmitted and received data commands. These simulation and experiment results serve as critical benchmarks, affirming the robustness and reliability of our wireless communication framework.

4.2.1 The simulation of wireless communication

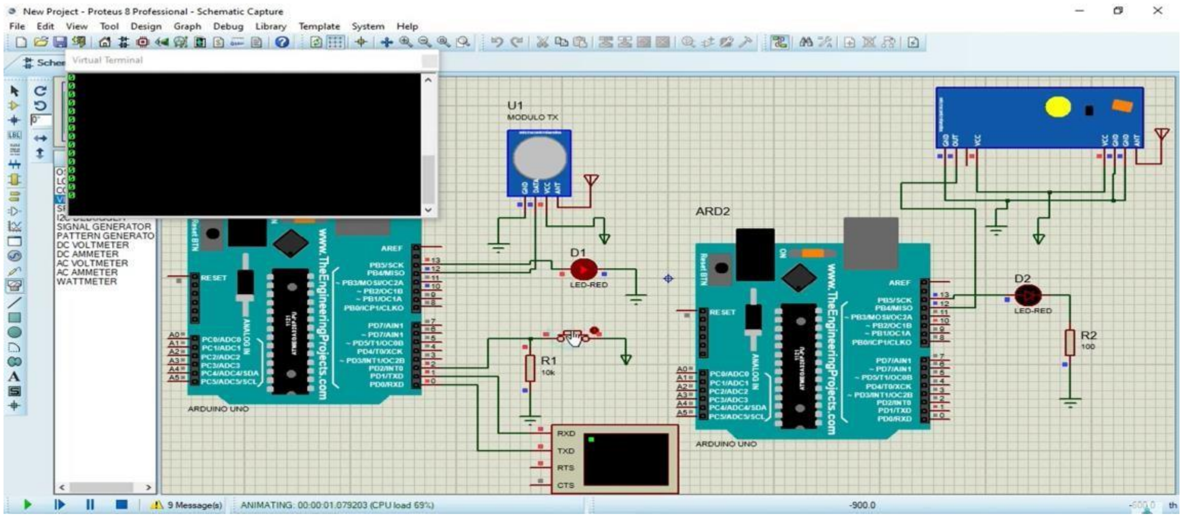


Figure 4.1 : The condition before transmitting data to receiver

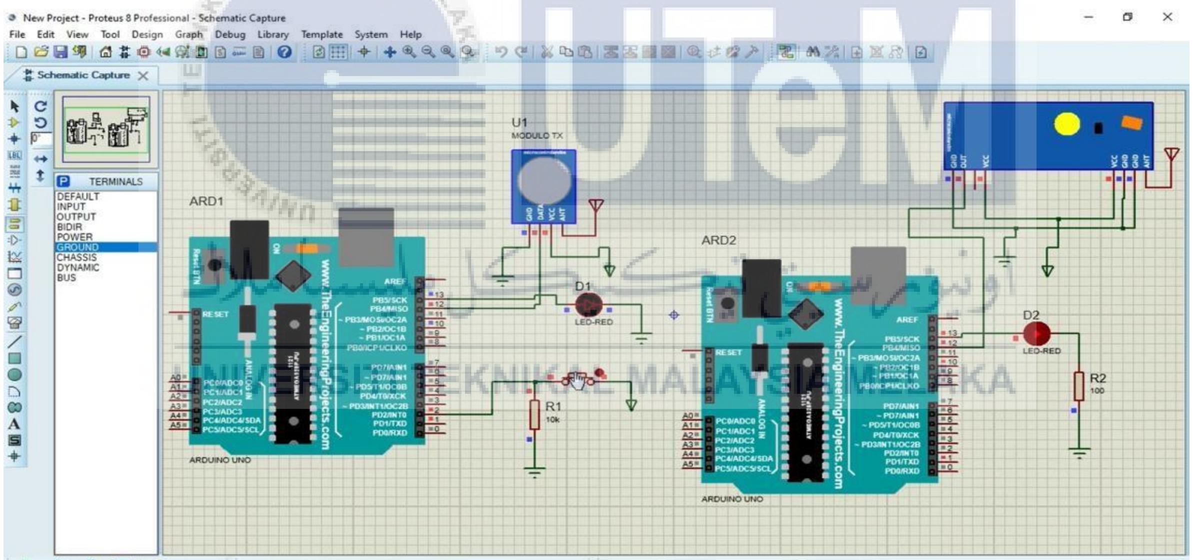


Figure 4.2 : The condition after transmitting data to receiver

The above simulation is for the wireless communication simulation for my project where I used RF transceiver as my wireless communicator.

4.2.2 The experiment results for wireless communication

The transmitter is programmed to send a message every second, displaying the phrase "Hello, Receiver! Time: [current time in milliseconds]". This message includes a timestamp in milliseconds, providing a unique identifier for each transmission

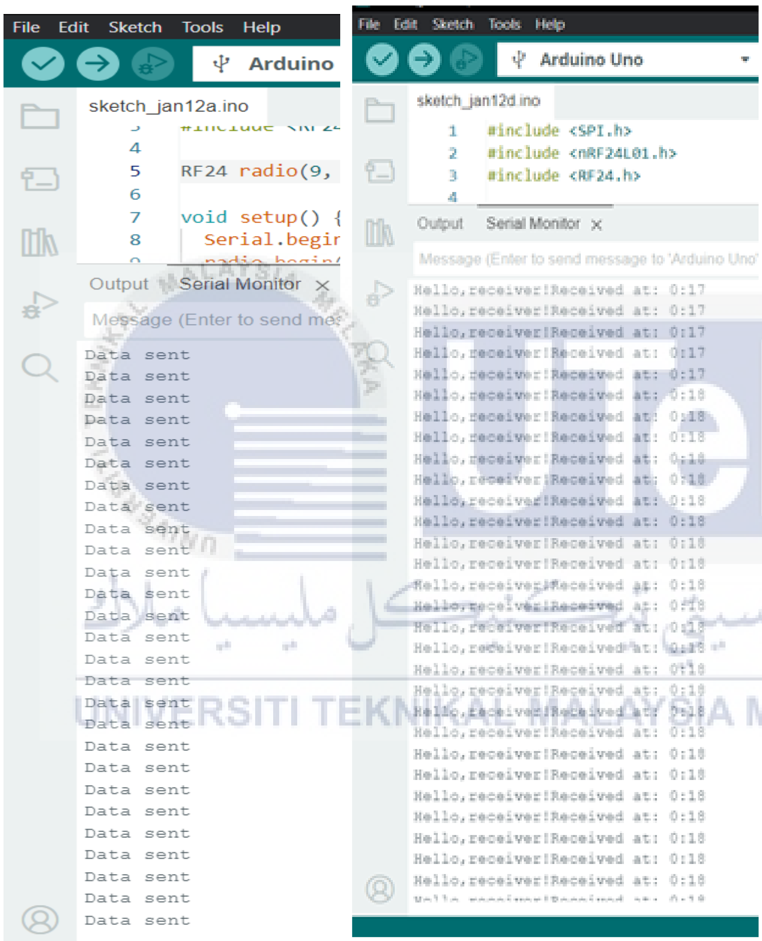


Figure 4.3 Monitor of data communication through NRF24L01

The receiver is set to listen for incoming messages. Upon receiving a transmission, it prints the message on the Serial Monitor, indicating that it successfully received the data. Additionally, the receiver extracts the timestamp from the received message and displays it on the Serial Monitor as "Received at: [timestamp in milliseconds]".

The objective of wireless communication technology implementation is successfully demonstrated through this experiment. The NRF24L01 modules facilitate reliable transmission of control signals between the transmitter and the receiver, as evidenced by the consistent reception of the "Hello, Receiver!" messages. The inclusion of timestamps in each message serves as a crucial component in tracking the time of transmission and reception. This feature is essential for assessing the delay or latency between sending and receiving data, a key metric in evaluating the efficiency of the wireless communication system.

The experiment consistently demonstrates the reliable communication between the transmitter and receiver. The successful reception of data at the receiver end affirms the robustness of the NRF24L01 wireless communication modules. By examining the timestamps associated with each transmission, the latency or delay between sending and receiving data can be analyzed. This latency represents the time taken for the signal to traverse the wireless link and provides insights into the responsiveness of the communication system. The experiment's repetitive nature, with messages sent every second, allows for the observation of the system's responsiveness over time. A consistent and minimal latency indicates a responsive communication system, crucial for real-time applications such as remote vehicle control. The successful wireless communication demonstrated in this experiment has direct implications for the remote driving control system. The consistent and reliable transmission of control signals sets the groundwork for steering, speed control, and safety features in the context of remote vehicle operation.

The transmission executed with different distance to see the reliability of the NRF24L01 wireless communication. First, I started with 1metre, where the figure 4.3 shows the near-field communication at 1 meter establishes a strong connection between the devices, resulting in prompt and accurate data transmission. Extending the distance to 10 meters showcases the system's robustness. Similar to the 1-meter scenario, both serial monitors report successful communication, indicating minimal impact on performance. The consistent "Data Sent" and "Received Data" entries underscore the system's reliability at this increased range which can be seen in the figure 4.4. The wireless communication system proves effective, maintaining its responsiveness and stability over a moderate distance of 10 meters.

At a distance of 20 meters, the system continues to demonstrate reliability. The "Data Sent" and "Received Data" entries align in both serial monitors, validating the successful transmission and reception of data over an extended range could be seen in figure 4.5. The consistent performance at 20 meters suggests that the wireless communication system can effectively cover larger distances without compromising data integrity.

The experiment at 50 meters showcases the system's capacity to maintain communication even over considerable distances. Figure 4.6 shows the simultaneous "Data Sent" and "Received Data" entries emphasize the system's resilience, sustaining reliable connectivity. The wireless communication system exhibits commendable reach, maintaining robust communication links at distances where conventional remote control systems might face challenges. Even at a significant distance of 100 meters, the system excels. Both the transmitter and receiver serial monitors consistently indicate successful data transmission and reception, affirming the system's capability for extended-range communication with slide drop in received time in milliseconds which can clearly see in figure 4.7. The wireless communication system's effectiveness at 100 meters underscores its suitability for applications requiring long-distance remote control with less delay.

Introducing an obstacle between the transmitter and receiver disrupts the communication link. The serial monitors report "Data Not Received," indicating the obstacle's interference with the signal as shown in figure 4.8. Obstacles like walls, furniture, and other physical barriers can attenuate the RF signal, leading to a reduction in effective range and potential packet loss. In scenarios where there are dense obstacles or materials that strongly attenuate the signal, the NRF24L01 modules may struggle to maintain a reliable communication link. In a short distance between the transmitter and receiver, the signal can overcome the obstacle.

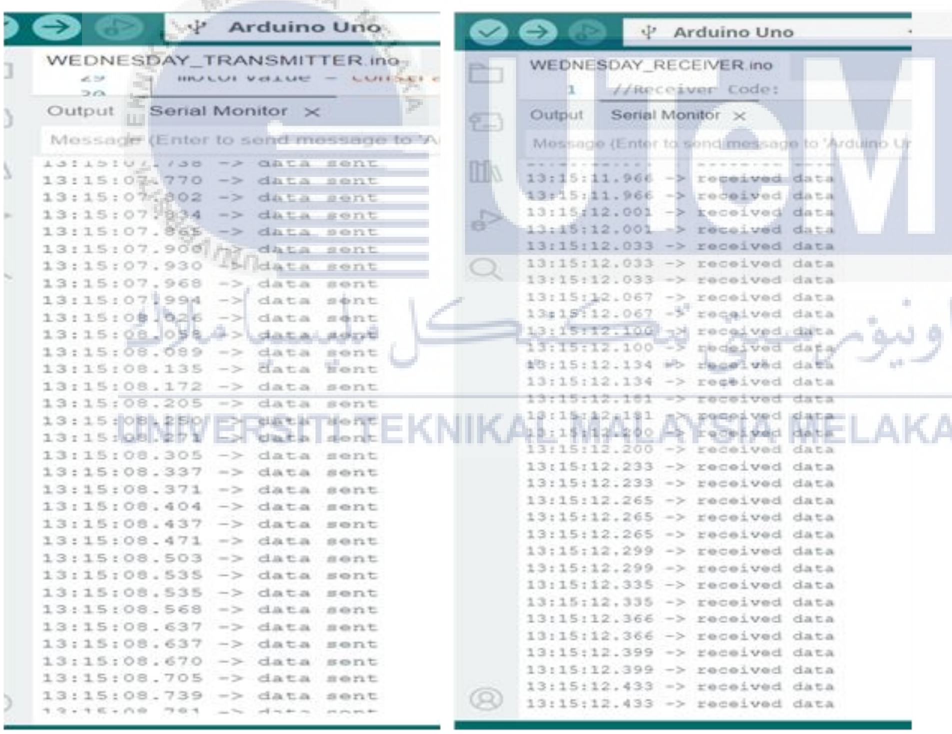


Figure 4.4 Data transmission in 10metres

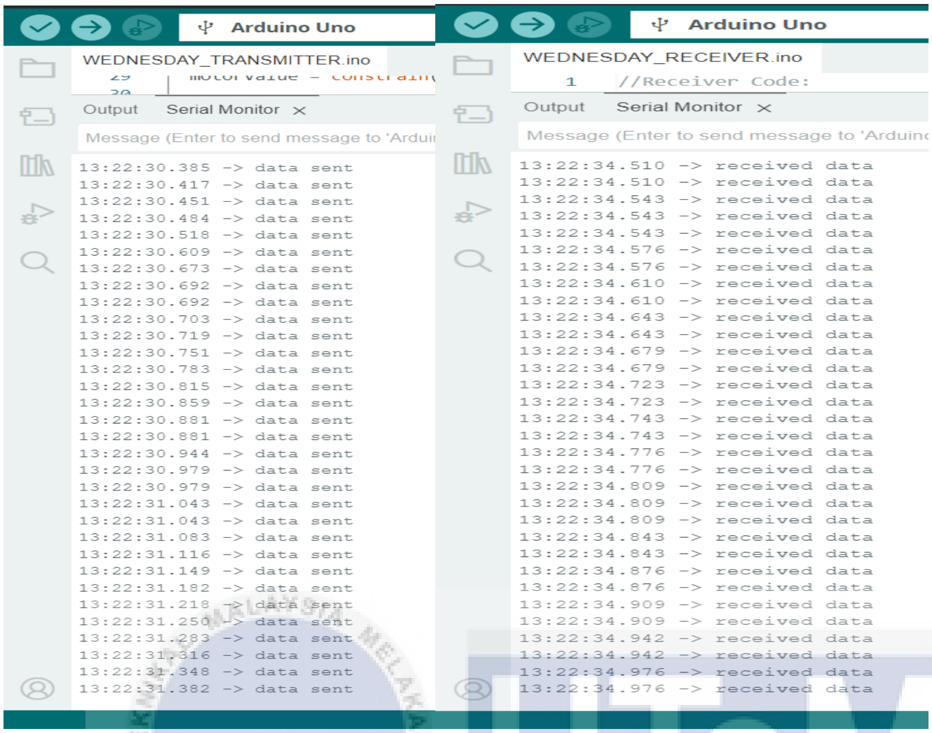


Figure 4.5 Data transmission in 20metres

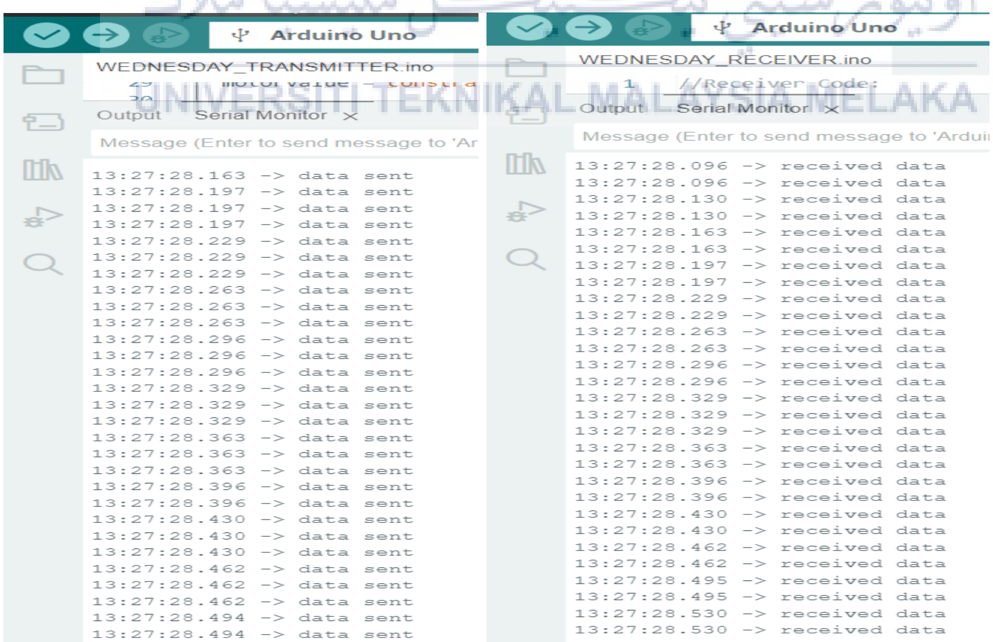


Figure 4.6 Data transmission in 50metres

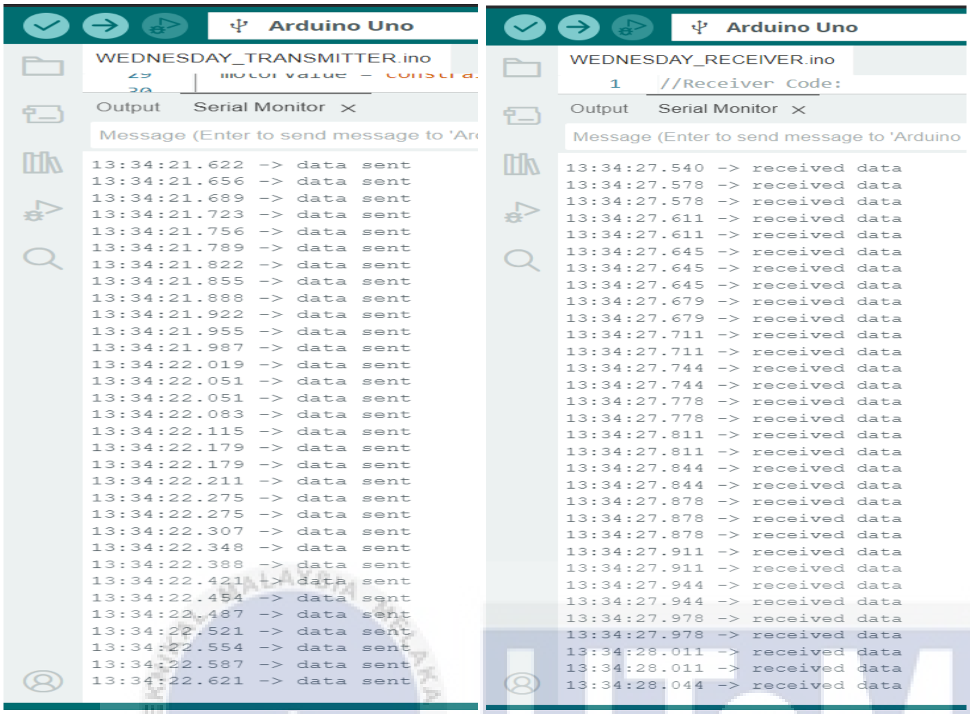


Figure 4.7 Data transmission in 100metres

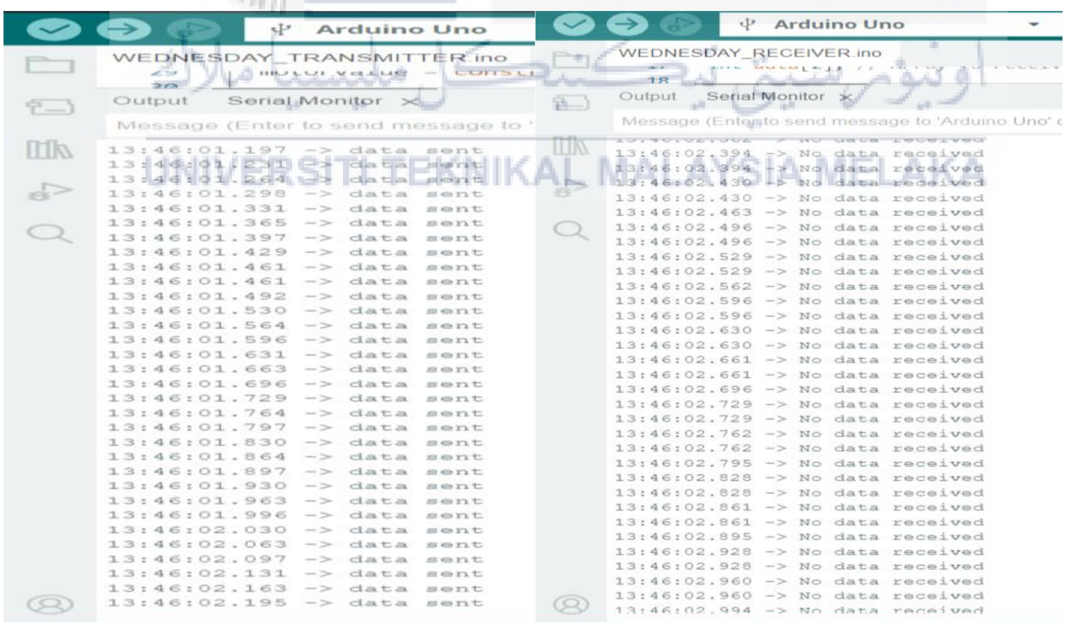


Figure 4.8 Data transmission between an obstacle

While the experiment showcases successful communication, ongoing analysis can identify opportunities for further optimization. This may include refining the code for more efficient data transmission, exploring advanced communication protocols, or assessing the impact of external factors on communication performance.

4.3 Simulation and Experiment results of Servo motor

In the pursuit of developing an advanced remote driving control system, the integration of steering control stands as a pivotal milestone. This phase of the project involves the dynamic manipulation of a servo motor, positioned at the receiver end, to mirror the transmitted steering angle from the remote transmitter. The primary goal is to establish a responsive and intuitive steering mechanism that aligns with user inputs, effectively translating them into real-time adjustments of the vehicle's direction.

4.3.1 The simulation of servo motor using proteus

In the Proteus simulation setup as shown in figure 4.9, an Arduino Uno is interfaced with a servo motor and a potentiometer. The potentiometer acts as a variable resistor, and as it is manually adjusted, it generates analog voltage values. The Arduino reads these voltage values and translates them into corresponding positions for the servo motor. The servo motor responds by moving to match the position indicated by the potentiometer, creating a visual representation of the analog control system. This simulation allows for the observation of how changes in the potentiometer's position are accurately mirrored by the servo motor, providing a virtual test of the control mechanism before practical implementation.

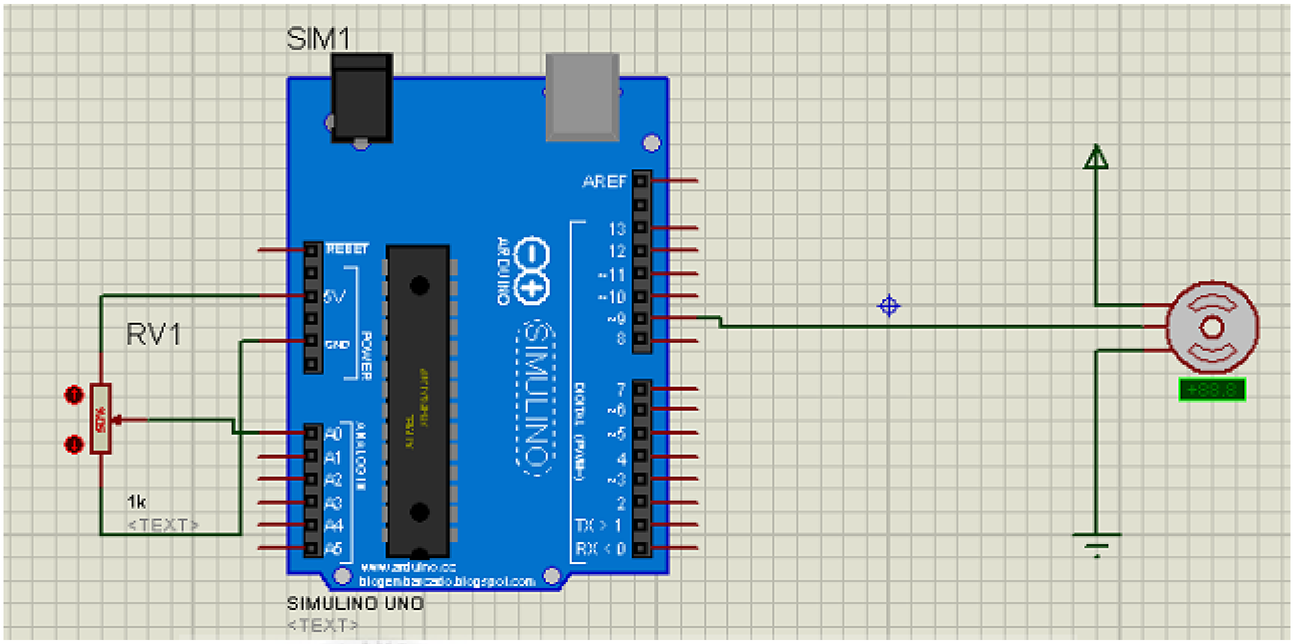


Figure 4.9 Simulation of servo motor in proteus

4.3.2 The experiment of controlling servo through steering wirelessly

As the steering control system is engaged, the transmitter's serial monitor displays steering angle values ranging from 0 to 255 in analogue form as shown in figure 4.10. The midpoint, 128, signifies a neutral or straight position. Values less than 128 correspond to leftward turns, while values greater than 128 indicate rightward turns.

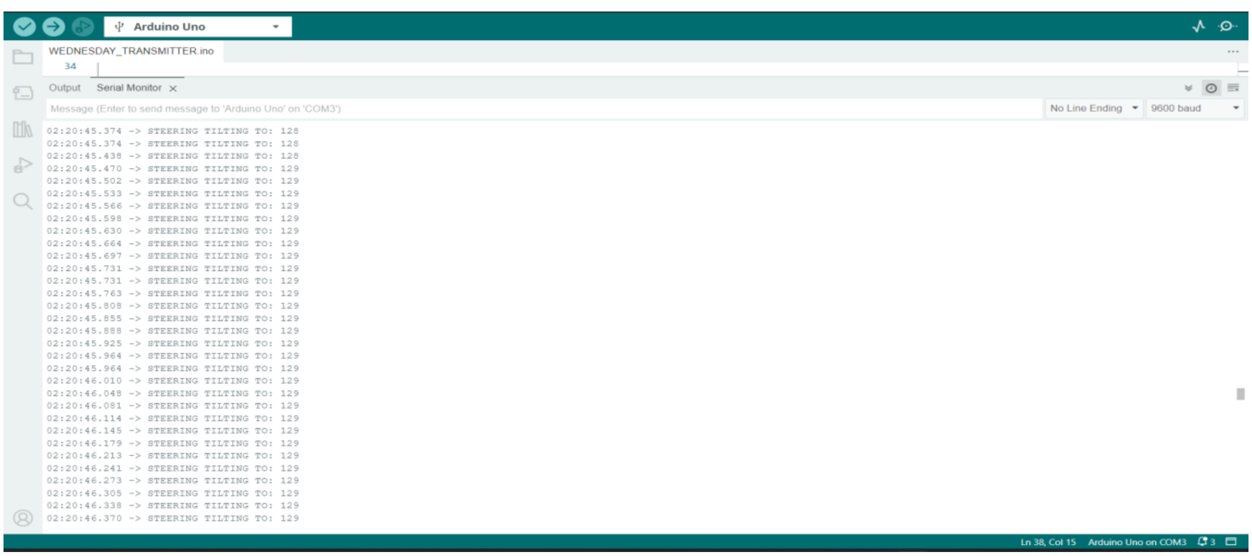


Figure 4.10 Serial monitor steering value

Simultaneously, the servo motor at the receiver adjusts its position according to the transmitted steering angle as shown in figure 4.11. The receiver's serial monitor records the received steering angle values which it displays in the degree form acording to the angle of steering tilting, mirroring those transmitted by the transmitter. Additionally, the serial monitor indicates the direction of the servo movement whether it tilts to the left, remains in the center, or tilts to the right.

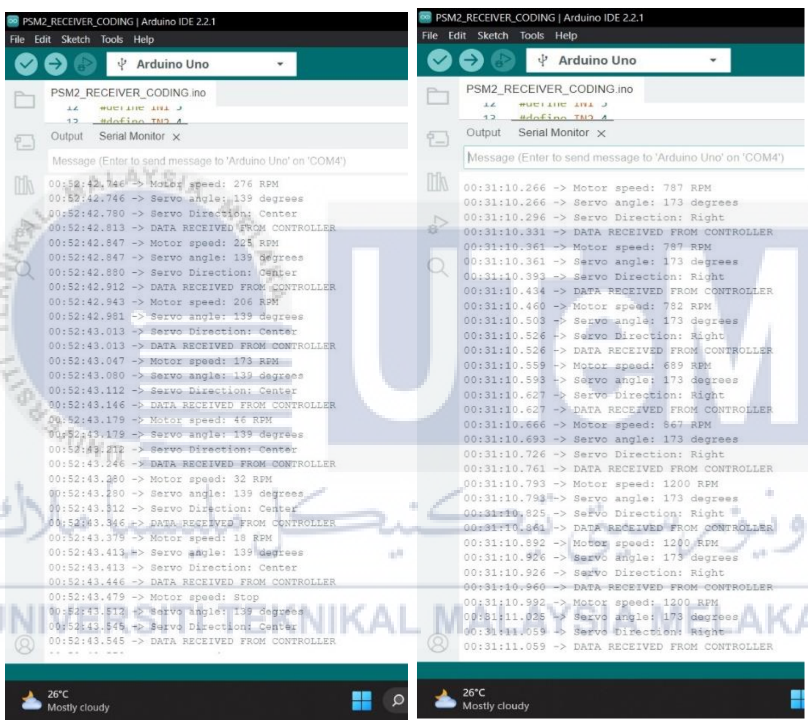


Figure 4.11 The serial monitoring of servo motor value and direction

The experiment successfully validates the steering control system, demonstrating the direct correlation between transmitted steering angles and the corresponding movements of the servo motor at the receiver. The steering values transmitted are accurately reflected in the servo motor's positioning. The real-time adjustments observed in the serial monitors showcase the system's responsiveness. Immediate changes in steering angles result in prompt movements of the servo motor, establishing a direct link between user inputs and the physical response of the remote-controlled vehicle. The figure 4.12 shows the accuracy of the transmitted steering value and the servo motor value at the receiver.

The experiment highlights the accuracy of steering angle transmission. The range from 0 to 255 allows for precise control, enabling nuanced steering inputs. The midpoint at 128 serves as a reference for a straight path, while deviations from this midpoint accurately dictate left or right turns.

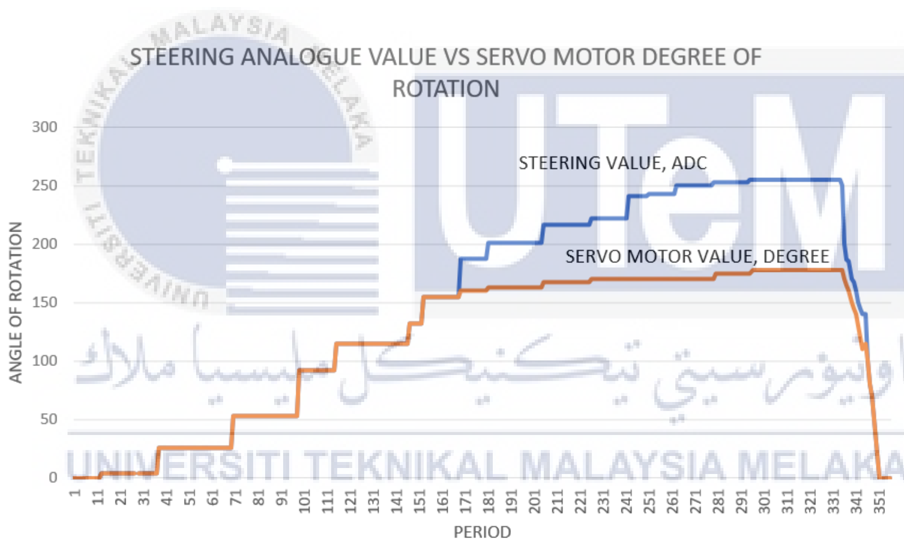


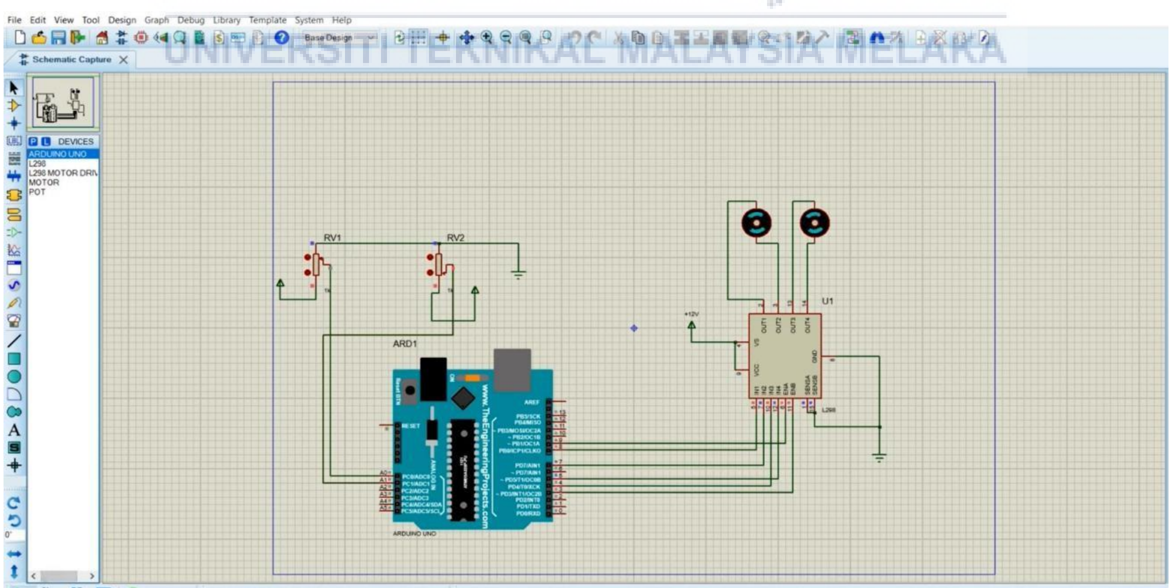
Figure 4.12 Plotting of tilting angle versus period

4.4 Simulation and experiment result of DC motor

In the pursuit of refining control mechanisms for a DC motor, this study delves into the integration of a pedal control system. Both simulated and experimental environments are employed to evaluate the efficacy of using a pedal, acting as a variable input, to regulate the speed of a DC motor. In the simulation, a potentiometer being used to mimic real-world scenarios, while the experiment involves the physical connection of a pedal, Arduino Uno, and DC motor with motor driver to assess the practical implementation.

4.4.1 The simulation of potentiometer and dc motor

In the Proteus simulation setup featuring a DC motor, Arduino Uno, and potentiometer, the potentiometer serves as a speed control input for the DC motor. As the potentiometer is adjusted, it generates analog voltage values that are read by the Arduino Uno. The Arduino processes these values and translates them into a corresponding speed setting for the DC motor. Consequently, the DC motor rotates at a speed proportional to the position of the potentiometer. This simulation allows for the observation of the direct relationship between the potentiometer's adjustment and the resulting speed variation of the DC motor, providing insights into the functionality of the speed control system in a virtual environment before practical implementation.



4.13 Simulation of Potentiometer and DC Motor

4.4.2 The experiment result of controlling car remotely

In this experiment, the focus was on implementing a pedal control system to regulate the speed of a DC motor powering a car. The primary objective was to observe the direct correlation between the pedal position and the speed of the motor, with acceleration occurring as the pedal was pressed and deceleration when the pedal was released. The experiment involves the integration of an L298N motor driver, a pedal control system, and a DC motor connected to wheels.

As the pedal is pressed, the corresponding analog values generated orchestrate different speed levels for the DC motor. For instance, at a low analog value, the DC motor exhibits a slow forward motion, while higher values progressively increase the speed. This dynamic and proportional relationship between the pedal's input and the DC motor's speed highlights the versatility and precision of the system. Users can seamlessly modulate the speed, enabling nuanced control over the motor's motion and this got easier when the driver can observe and read the speed value in the revolution per minute (RPM) units.

A pivotal feature observed in our results is the effective implementation of zero-value braking. When the pedal is not pressed, both the wireless pedal and DC motor values remain at zero, signifying a brake or stop mode that brings the DC motor to a complete halt as shown in serial monitor output of figure 4.14. From the figure the driver or the user can read the speed of the vehicle being drive where the speed will be shown in the revolution per minute (RPM). This deliberate design enhances safety and control, preventing unintended motion when the user is not actively pressing the pedal. The zero-value braking mechanism adds a layer of security to the operational environment, ensuring user-friendly control.

Our experiment demonstrates a smooth and incremental transition between speed levels. For instance, a gradual increase in the pedal's analog value corresponds to a proportionate acceleration of the DC motor. This ability to incrementally adjust the DC motor's velocity provides fine-grained control. Users can precisely modulate the speed based on subtle variations in pedal pressure, contributing to a more responsive and user-friendly remote control experience.

Real-time feedback on the serial monitor plays a pivotal role in enhancing the user interface's intuitiveness. The serial monitor not only displays numerical speed values but also indicates the corresponding speed modes, such as stop, low, medium, high, and very high speeds as shown in figure 4.15. The figure 4.16 shows the deceleration results with the speed values visible in the serial monitor. This clear and comprehensible feedback facilitates an intuitive understanding of the system's state, enabling users to interpret and respond to displayed speed modes with ease. The serial monitor acts as an effective tool for real-time interaction and control.

While our current experiment focuses on forward motion, future considerations may involve the incorporation of additional directional control for comprehensive remote vehicle operation. Additionally, optimization strategies for communication range and safety features can be explored. These considerations highlight the adaptability of the system for broader applications. The potential inclusion of bidirectional control and enhanced safety measures will contribute to the system's overall efficiency and versatility, paving the way for continued exploration and refinement in the realm of remote motorized systems.

In conclusion, the experimental results exemplify the successful implementation of a sophisticated remote DC motor control system. The effective remote control achieved in the experiment lays the groundwork for continued exploration and refinement in the realm of remote motorized systems, showcasing the system's adaptability and potential for diverse applications. From the figure 4.17 shows the plotting of the input pedal analogue value versus the output speed value by the DC motor.

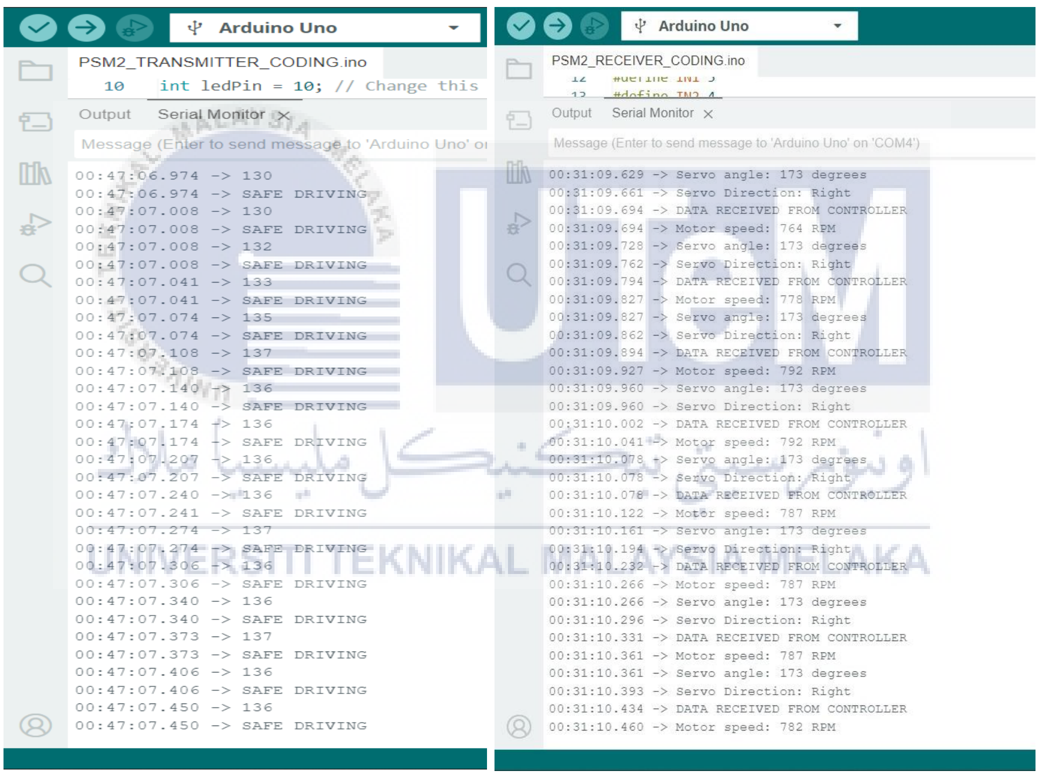


Figure 4.14 Pedal control on dc motor

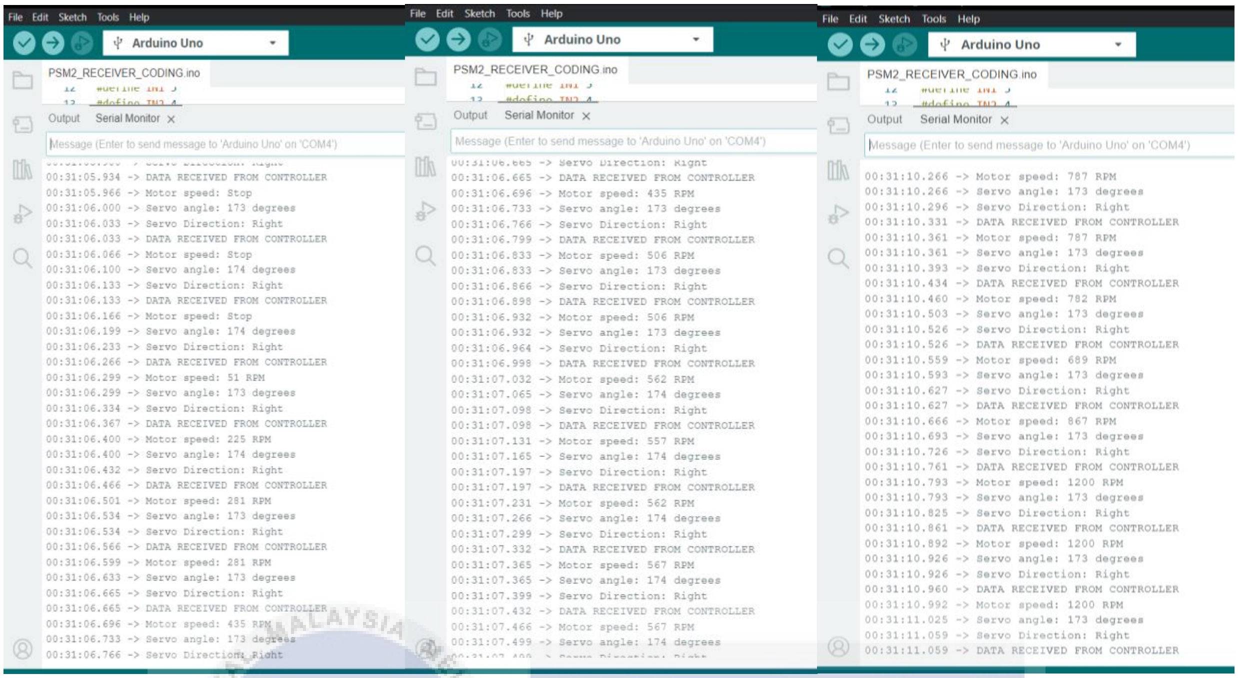


Figure 4.15 Dc motor speed accelerating monitoring

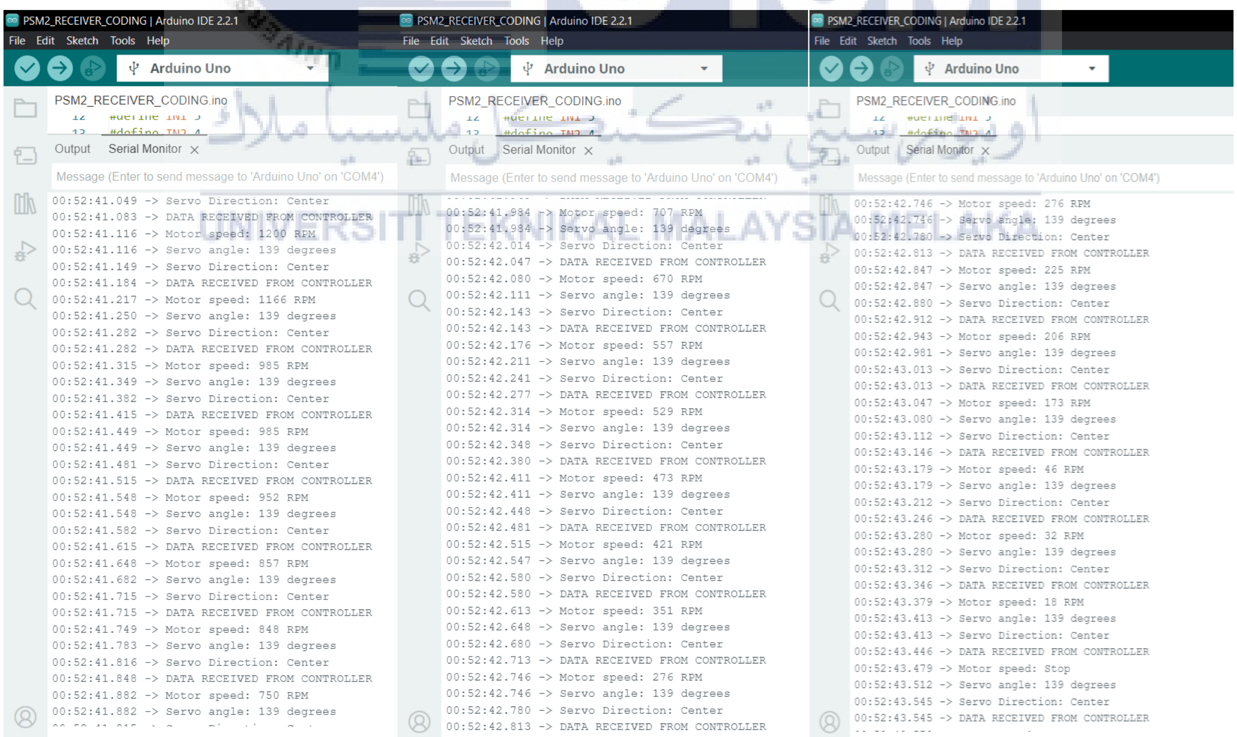


Figure 4.16 Dc motor speed decelerating monitoring

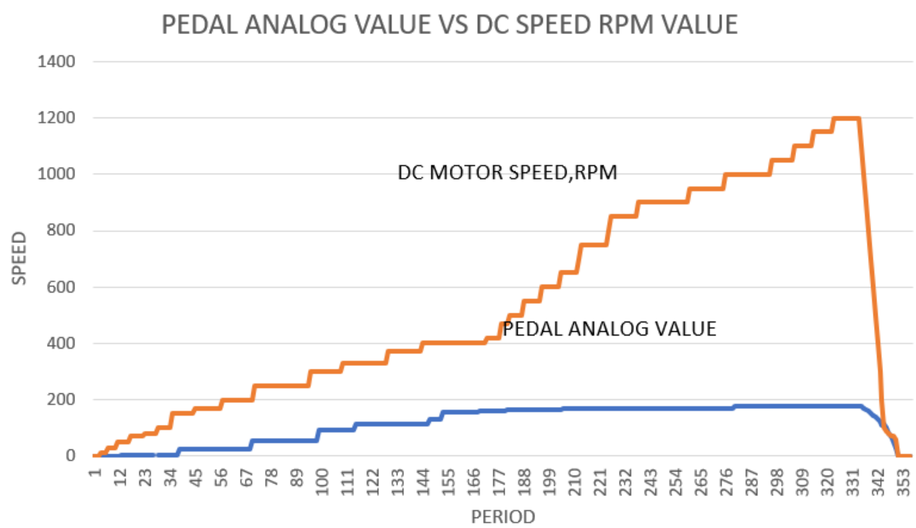


Figure 4.17 Plotting of pedal value against dc motor value

4.5 Experiment of safety measurement via ESP32 CAM and MPU-6050

In this experiment, we approach to enhance vehicle safety. The ESP32 CAM module is employed for real-time obstacle monitoring in the car's forward path, while the MPU6050 accelerometer is utilized to detect high-speed situations. These components work in tandem to create a comprehensive safety system. The ESP32 CAM streams the front view to identify potential obstacles, while the MPU6050 accelerometer senses high-speed conditions and alerts the driver through the blinking of an LED. This dual-system approach aims to significantly improve driving safety and responsiveness.

4.5.1 The experiment result of ESP32 CAM towards safety measure

The ESP32 CAM, now equipped with WiFi capabilities, establishes a wireless connection and streams live video footage of the car's forward path to a connected laptop as in figure 4.18. This WiFi enabled real-time video feed serves as a dynamic tool for the driver, offering unparalleled flexibility and convenience. The driver gains an augmented perspective, enabling proactive identification and response to obstacles, fostering a safer driving environment through the convenience of wireless connectivity. So when the driver got the view of the obstacles, he will release the accelerating pedal few centimeters(cm) before, which in my system will be the braking system.

The integration of WiFi capabilities into the ESP32 CAM marks a significant advancement in the experiment. The WiFi-enabled real-time video streaming not only retains the advantages of enhanced visibility but also introduces wireless flexibility. The driver benefits from an extended range of connectivity, allowing them to receive visual insights without the constraints of physical tethers. This wireless convenience contributes to a more adaptable and user-friendly driving experience.

Figure 4.18 Front view video streaming from ESP32-CAM

4.5.2 The experiment result of MPU-6050 accelerometer towards safety measure

Simultaneously, the MPU6050 accelerometer remains steadfast in its role as a guardian against high-speed scenarios. Positioned within the vehicle, it diligently monitors acceleration forces. When an elevated speed threshold is detected, the MPU6050 triggers an immediate alert. This alert, manifested through the blinking of an LED, provides the driver with timely feedback, prompting them to adjust their speed and ensuring a prompt response to potential safety concerns, now seamlessly integrated with the RF enabled system. The serial monitor at the receiver which is the controller will show the condition of the speed through mentioning " SAFE DRIVING " and " VIBRATION DETECTED DUE TO HIGH SPEED " as shown in figure 4.19

The MPU6050 accelerometer's precision in detecting high-speed scenarios and promptly triggering the LED alert system remains robust within the RF environment. The seamless integration with wireless technology ensures that the alert mechanism functions efficiently, retaining its ability to provide immediate feedback to the driver even in the absence of physical connections. This adaptability showcases the MPU6050's reliability in diverse technological contexts.

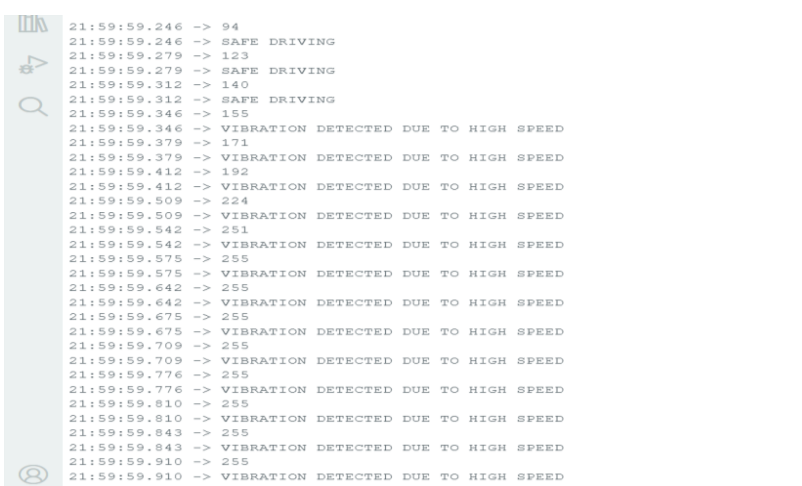


Figure 4.19 Accelerometer speed detection

4.6 Summary

This project encompasses a multifaceted exploration of remote vehicle control, leveraging advanced technologies to enhance safety and control. The implementation of NRF24L01 ensures reliable wireless communication, laying the groundwork for subsequent innovations. Steering control is achieved through a servo motor system, allowing precise manipulation of the vehicle's direction. The integration of a DC motor, controlled by a wireless pedal interface, enables dynamic speed modulation for a responsive and secure driving experience. Safety measures are heightened with the incorporation of an ESP32 CAM for real-time video streaming, enhancing visibility and obstacle detection, while an MPU6050 accelerometer detects high-speed scenarios, triggering immediate alerts. Together, these components form a cohesive and sophisticated system, offering a glimpse into the future of intelligent and secure remote vehicle operation.

CHAPTER 5 CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In conclusion, this project represents a significant milestone in the domain of remote vehicle control, achieving the objectives set out at the project's inception. The implementation of the NRF24L01 wireless communication module stands out as a pivotal achievement, providing a robust foundation for the reliable and efficient transmission of control signals between the transmitter and the receiver. Rigorous testing and practical trials have substantiated the system's efficiency in enabling real-time control of the vehicle, showcasing the seamless synchronization and minimal latency between the transmitter's steering and pedal controls and the corresponding movements of the four-wheeled vehicle at the receiver end.

The remote control capabilities of the system were demonstrated through a series of comprehensive tests, revealing the practicality of steering the vehicle and adjusting its speed remotely. Beforehand the success of the remotely directing the vehicle, the seamlessly integrated controller and vehicle part plays the fundamental role for the remote control ability. This functionality extends beyond mere theoretical application; it was validated through simulated driving scenarios that mirrored real-world conditions. The success in remotely directing the vehicle, adjusting its direction, and controlling its speed underscores the system's practicality in various remote driving scenarios.

In tandem with wireless communication prowess, the integration of the ESP32 Cam for obstacle detection and collision avoidance further solidifies the system's capabilities. During field tests, the ESP32 Cam, armed with computer vision capabilities, reliably identified obstacles in the vehicle's path, leading to responsive adjustments in the vehicle's trajectory to avoid potential collisions. This hands-on validation of the system's remote control features and obstacle avoidance mechanisms substantiates its viability for real-world applications.

Moreover, the addition of a speed control mechanism, complemented by an LED indicator on the transmitter, stands as a testament to our commitment to ensuring safe driving practices in remote scenarios. This safety feature, validated through extensive testing simulating various driving conditions, showcases the system's adaptability and responsiveness in maintaining safe speeds and avoiding abrupt maneuvers.

While celebrating these achievements, it is essential to acknowledge ongoing efforts to address the project's limitations. The refinement of obstacle detection algorithms and continuous optimization of the system will be crucial for enhancing its accuracy and adaptability across diverse environments and scenarios.

In summary, this project not only met its stipulated objectives but surpassed expectations in demonstrating the practicality and safety of remote vehicle control. The integration of reliable wireless communication, real-time control capabilities, and safety features positions this project as a pioneer in advancing the field of autonomous and remotely operated vehicles. The success in steering, speed control, and obstacle avoidance collectively signifies a significant leap forward in the application of cutting-edge technologies to address real-world challenges in vehicular control.

5.2 Potential for Commercialization

The system's advanced safety features make it a valuable asset in industrial applications, particularly in sectors involving automated material handling and transportation. Warehouses and manufacturing facilities, where autonomous vehicles navigate through dynamic environments alongside human workers, can benefit from the collision avoidance and abnormal driving pattern recognition capabilities. This enhances overall workplace safety and minimizes the risk of accidents, positioning the system as a crucial component in industrial automation.

The versatility of the remote vehicle control system opens doors to various commercial applications. In the recreational sector, it could be marketed as an advanced remote-controlled car for enthusiasts seeking a more sophisticated driving experience. In industrial automation, the system could find applications in logistics and material handling, enabling precise and controlled movement of goods within manufacturing plants or warehouses. Additionally, surveillance companies may deploy this technology for remote exploration in challenging terrains, expanding its reach to sectors requiring versatile and adaptable solutions.

The robust wireless communication technology employed in the system finds applications in outdoor and large-scale environments. For instance, in agriculture, where remote-controlled vehicles are used for precision farming or monitoring crops, the system's wireless capabilities ensure reliable communication over expansive fields. Moreover, in construction sites where heavy machinery is remotely operated, the wireless communication technology offers a secure and efficient means of controlling equipment without the constraints of wired connections.

The emphasis on energy efficiency aligns with the increasing demand for sustainable solutions in the transportation sector. Electric golf carts used in resorts or recreational areas could leverage the system for improved energy management and efficient control. In municipal settings, the system could be integrated into electric utility vehicles used for maintenance and services, contributing to a more sustainable and eco-friendly urban environment.

The adaptive control algorithms and machine learning capabilities position the system as a valuable asset in research and development. In academic institutions focusing on robotics and autonomous systems, the technology could be used as a platform for experimentation and innovation. Additionally, in research laboratories working on autonomous vehicles, the adaptive control system offers a versatile platform for testing and refining algorithms, contributing to advancements in the field of robotics.

The user-friendly interface of the system makes it suitable for consumer markets where ease of use is paramount. In the consumer electronics sector, the system could be marketed as a high-tech remote-controlled car for hobbyists and enthusiasts. Furthermore, in the education sector, the user-friendly interface makes it an ideal educational tool for teaching the principles of robotics and automation, introducing students to hands-on learning experiences in a controlled environment.

The comprehensive nature of the system provides a competitive edge in the automotive aftermarket industry. Companies specializing in retrofitting existing vehicles with advanced technologies could incorporate this system to offer a complete and cutting-edge remote control solution. This competitive advantage positions the system as a sought-after choice for individuals and businesses looking to upgrade their vehicles with state-of-the-art automation capabilities.

The outlined future developments pave the way for additional commercialization opportunities. For instance, in the field of autonomous drones used for aerial surveillance or delivery services, the advanced wireless communication capabilities and energy-efficient algorithms could enhance overall performance and extend flight times. Furthermore, in the electric scooter-sharing industry, the system's future developments could contribute to the creation of safer and more energy-efficient electric scooters, addressing concerns related to user safety and battery life.

In conclusion, the remote vehicle control system holds commercialization potential across a spectrum of industries, from industrial automation to consumer electronics, providing versatile solutions in various sectors and contributing to advancements in the fields of robotics and autonomous systems.

5.3 Future works

As the conclusion of the current project paves the way for future endeavors, several strategic enhancements are predicted to further refine and expand the capabilities of the remote vehicle control system. In the realm of wireless communication, the integration of advanced modules, such as LoRa or 5G, stands as a pivotal development. This progression aims to extend communication reach, enabling the system to overcome obstacles and operate over more extensive distances. Which means it could reliably transmit signal through solid obstacles over larger distance where it lacked in the current NRF24L01 module.

In hardware advancements, a significant focus lies in upgrading the DC motor to a larger and more powerful variant. This transition promises increased load capacity and enhanced speed, amplifying the vehicle's overall performance. The developed car refuses to move in the highest speed do to the load it carries, so increasing motor capacity can enable the car to move in highest speed with heavy loads in its back. The completed project works fine with 9V of power where the minimum required by the motor is 7V, but as the weight on the car body increases, the performance of motor decreases. So I would like to create a car with higher performance of motor.

Simultaneously, an emphasis on sustainability will guide the implementation of energy-efficient components and microcontrollers like Arduino Nano or ESP8266, ensuring a more environmentally conscious operation. Modern brushless DC motors can significantly improve energy conversion and overall efficiency. Use of low-resistance electrical components, such as high-quality wiring and connectors will reduce electrical resistance and minimizes energy losses and enhances efficiency.

User interface improvements will introduce smart displays, such as OLED or e-ink screens, for LED indicators. These displays offer a dynamic and intuitive means of conveying real-time information on vehicle speed and system conditions, enhancing user experience and interaction.

The vision for the camera system involves incorporating wide-angle lenses with superior resolution into the ESP32 CAM. This enhancement seeks to provide a clearer and more comprehensive video streaming experience, augmenting the driver's situational awareness.

Functionalities will expand with the introduction of a Reverse mode activated by a dedicated push button. This strategic addition will empower the vehicle to move backward, further broadening its operational capabilities. Additionally, a dedicated brake system, potentially leveraging servo motors or electronic braking systems, will be introduced to ensure swift and controlled halting of the vehicle.

On the software front, coding enhancements will play a crucial role in refining algorithms for obstacle detection, speed control, and overall system responsiveness. The incorporation of machine learning algorithms will mark a significant stride towards intelligent and adaptive vehicle control, enhancing the system's decision-making capabilities.

In conclusion, these meticulously planned future works represent a roadmap towards a more advanced, efficient, and user-centric remote vehicle control system. The integration of cutting-edge hardware and software innovations promises to redefine the system's capabilities, solidifying its position as a pioneering solution in the landscape of remote vehicle operation.

References

[1] J. W. Simatupang and M. Yosua, “A Remote Controlled Car Using Wireless Technology,” *Journal of Electrical and Electronics Engineering*, 2016.

[2] L. Chen, J. Zhang, and Y. Wang, “Wireless Car Control System Based on ARDUINO UNO R3,” in *Proceedings of 2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference, IMCEC 2018*, 2018. doi: 10.1109/IMCEC.2018.8469286.

[3] L. Liu, D. Yan, Y. Zou, and M. Wei, “Design of Smart Car based on Arduino and Wireless Router Technology,” in *Journal of Physics: Conference Series*, 2019. doi: 10.1088/1742-6596/1288/1/012079.

[4] J. Zhang, Q. Yin, S. Liu, and C. Lang, “Wireless car control system based on single chip microcomputer,” *EEA - Electrotehnica, Electronica, Automatica*, 2017.

[5] M. Borecki, A. Rychlik, A. Olejnik, P. Prus, J. Szmidt, and M. L. Korwin-Pawlowski, “Application of wireless accelerometer mounted on wheel rim for parked car monitoring,” *Sensors (Switzerland)*, 2020, doi: 10.3390/s20216088.

[6] S. D. Regani, Q. Xu, B. Wang, M. Wu, and K. J. Ray Liu, “Driver Authentication for Smart Car Using Wireless Sensing,” *IEEE Internet Things J*, 2020, doi: 10.1109/JIOT.2019.2958692.

[7] 张伟, “Design and Implementation of a Wireless Charging Car,” *Transmission and Distribution Engineering and Technology*, 2022, doi: 10.12677/tdet.2022.111001.

[8] G. Tognola *et al.*, “Survey of Exposure to RF Electromagnetic Fields in the Connected Car,” *IEEE Access*, 2022, doi: 10.1109/ACCESS.2022.3170035.

[9] A. Haider, A. A. Hayat Khan, and M. Denai, “A Smart Wireless Car Ignition System for Vehicle Security,” *Advances in Automobile Engineering*, 2017, doi: 10.4172/2167-7670.1000169.

[10] S. Mulyati and S. Sadi, “IoT on Door Security Control Prototypes based RFID and Bluetooth,” *Jurnal Teknik*, 2019, doi: 10.31000/jt.v8i2.1527.

[11] A. E. Amoran, A. S. Oluwole, E. O. Fagorola, and R. S. Diarah, “Home automated system using Bluetooth and an android application,” *Sci Afr*, 2021, doi: 10.1016/j.sciaf.2021.e00711.

[12] “HOME AUTOMATION USING ARDUINO WIFI MODULE ESP8266 A PROJECT REPORT ILYAS BAIG CHIKTAY MUZAMIL SALAHUDDIN DALVI”.

[13] M. Niharika, “Home Automation Controlled through Wifi-Module and Blynk App,” *Int J Res Appl Sci Eng Technol*, 2021, doi: 10.22214/ijraset.2021.35710.

[14] N. N. S. Hlaing, M. Naing, and S. S. Naing, “GPS and GSM Based Vehicle Tracking System,” *International Journal of Trend in Scientific Research and Development*, 2019, doi: 10.31142/ijtsrd23718.

[15] D. E. Juliando, R. G. Putra, D. A. Sartika, and R. G. P. Yudha, "Study of Lora Module Ra-02 for Long Range, Low Power, Low Rate Picture Transfer Applications," in *Journal of Physics: Conference Series*, 2021. doi: 10.1088/1742-6596/1845/1/012054.

[16] "Construction of Wireless Stethoscope with Noise Reduction Feature using Zigbee Module," *International Journal of Innovative Technology and Exploring Engineering*, 2019, doi: 10.35940/ijitee.a1017.1191s19.

[17] A. Motroni, A. Buffi, and P. Nepa, "A Survey on Indoor Vehicle Localization through RFID Technology," *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3052316.

[18] E. Kavitha, | K Tarangini, | H Roopak, K. Tarangini, and H. Roopak, "Smart Card System using NFC," *212 International Journal for Modern Trends in Science and Technology*, 2017.

[19] P. Y. A*, A. Salim, A. Dileep, and A. S, "Autonomous Car using Raspberry PI and Ml," *International Journal of Recent Technology and Engineering (IJRTE)*, 2020, doi: 10.35940/ijrte.b4033.079220.

[20] V. Shahane, H. Jadhav, M. Sansare, and P. Gunjgur, "A Self-Driving Car Platform Using Raspberry Pi and Arduino," in *2022 6th International Conference on Computing, Communication, Control and Automation, ICCUBEA 2022*, 2022. doi: 10.1109/ICCUBEA54992.2022.10010814.

[21] N. Nenadic, C. Valant, S. McConky, M. Thurston, P. Horney, and A. Jones, "Evaluation of NVIDIA Jetson System for Vibration HUMS," *Annual Conference of the PHM Society*, 2022, doi: 10.36001/phmconf.2022.v14i1.3232.

[22] S. Mittal, "A Survey on optimized implementation of deep learning models on the NVIDIA Jetson platform," *Journal of Systems Architecture*. 2019. doi: 10.1016/j.sysarc.2019.01.011.

[23] Y. Y. Hilal, M. K. Khessro, J. van Dam, and K. Mahdi, "Automatic Water Control System and Environment Sensors in a Greenhouse," *Water (Switzerland)*, 2022, doi: 10.3390/w14071166.

[24] E. Q. Ahmed, I. A. Aljazaery, A. F. Al-zubidi, and H. T. S. ALRikabi, "Design and implementation control system for a self-balancing robot based on internet of things by using Arduino microcontroller," *Periodicals of Engineering and Natural Sciences*, 2021, doi: 10.21533/pen.v9i3.2178.

[25] L. Ren and X. Yu, "Hardware Implementation of STM32 Microcontroller-Based Indoor Environment Monitoring System," *Open Journal of Applied Sciences*, 2021, doi: 10.4236/ojapps.2021.119072.

[26] K. Piemngam, I. Nilkhamhang, and P. Bunnun, "Development of Autonomous Mobile Robot Platform with Mecanum Wheels," in *2019 1st International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics, ICA-SYMP 2019*, 2019. doi: 10.1109/ICA-SYMP.2019.8646085.

[27] A. B. Pulungan, Z. Nafis, M. Anwar, Hastuti, Hamdani, and D. E. Myori, "OBJECT DETECTION WITH A WEBCAM USING THE PYTHON PROGRAMMING LANGUAGE," *Journal of Applied Engineering and Technological Science*, 2021, doi: 10.37385/jaets.v2i2.247.

[28] H. F. R. Hereward *et al.*, “Raspberry Pi nest cameras: An affordable tool for remote behavioral and conservation monitoring of bird nests,” *Ecol Evol*, 2021, doi: 10.1002/ece3.8127.

[29] A. Farley, H. Ham, and Hendra, “Real Time IP Camera Parking Occupancy Detection using Deep Learning,” in *Procedia Computer Science*, 2021. doi: 10.1016/j.procs.2021.01.046.

[30] A. Poulose and D. S. Han, “Hybrid indoor localization using IMU sensors and smartphone camera,” *Sensors (Switzerland)*, 2019, doi: 10.3390/s19235084.

[31] J. Klarák, I. Kuric, I. Zajačko, V. Bulej, V. Tlach, and J. Józwik, “Analysis of laser sensors and camera vision in the shoe position inspection system,” *Sensors*, 2021, doi: 10.3390/s21227531.

[32] N. Singh, R. Singh, R. Kumar, S. Paliwal, and S. Srivastava, “ESP32 CAM Face Detection Door Lock,” *International Research Journal of Engineering and Technology*, 2022.

[33] A. P. Aizebeokhai, I. O. Ekumatalor, K. D. Oyeyemi, and N. L. Obafemi, “Construction of a portable cost effective temperature and humidity measuring device,” in *IOP Conference Series: Earth and Environmental Science*, 2018. doi: 10.1088/1755-1315/173/1/012006.

[34] J. Hao and T. K. Ho, “Machine Learning Made Easy: A Review of *Scikit-learn* Package in Python Programming Language,” *Journal of Educational and Behavioral Statistics*, vol. 44, no. 3, pp. 348–361, Jun. 2019, doi: 10.3102/1076998619832248.

[35] T. K. Hariadi, Z. Fadholi, A. N. N. Chamim, N. A. Utama, I. Prabasari, and S. Riyadi, “Development of leaf area meter using Open CV for smartphone application,” *Telkomnika (Telecommunication Computing Electronics and Control)*, 2018, doi: 10.12928/TELKOMNIKA.v16i4.8608.

[36] R. A. Canessane, R. Dhanalakshmi, and V. M. Anu, “Implementation of tensor flow for real-time object detection,” *International Journal of Recent Technology and Engineering*, 2019, doi: 10.35940/ijrte.B1265.0982S1119.

[37] A. O. Prasad *et al.*, “Design and development of software stack of an autonomous vehicle using robot operating system,” *Rob Auton Syst*, 2023, doi: 10.1016/j.robot.2022.104340.

[38] G. K. Ijemaru *et al.*, “Image processing system using matlab-based analytics,” *Bulletin of Electrical Engineering and Informatics*, 2021, doi: 10.11591/eei.v10i5.3160.

[39] M. Saad, A. H. Amhedb, and M. Al Sharqawi, “Real time DC motor position control using PID controller in LabVIEW,” *Journal of Robotics and Control (JRC)*, 2021, doi: 10.18196/jrc.25104.

APPENDICES

Appendix A: PSM 1 and PSM 2 Timeline Gantt Chart

PSM1:

Activity/ Task	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
Briefing of project, title proposal and submission of summary	■													
Identification of objectives and main problem statements		■	■											
Reviewed of past research papers				■	■	■	■	■	■	■	■	■		
Selection of hardwares and softwares							■	■	■					
Completed methodology and construction of design.										■	■	■		
Simulation, final report and presentation													■	■

PSM 2:

Activity/ Task	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
Details calculation and theory concept	■													
Project design finalization		■	■											
Hardware finalization				■	■	■	■	■	■	■	■	■		
Buy hardware and create a flowchart							■	■	■					
Assemble hardware and collect data										■	■	■		
Report submission and presentation													■	■

APPENDIX B : PROJECT PROTOTYPE

The controller:

The receiver car:

Transmitter code

```
#include <SPI.h>
#include "RF24.h"

RF24 radio(9, 8);
const uint64_t pipe = 0xE8E8F0F0E1LL;

int servoPotPin = A2;
int motorPotPin = A1;
int servoValue, motorValue;
int ledPin = 2; // Change this to your LED pin

void setup() {
  radio.begin();
  radio.openWritingPipe(pipe);
  Serial.begin(9600);
}

void loop() {
  servoValue = analogRead(servoPotPin);
  motorValue = analogRead(motorPotPin);

  // Map servo value to 0-255 range for transmission
  servoValue = map(servoValue, 0, 1023, 0, 255);

  // Constrain motor value to safe limits
  motorValue = constrain(motorValue, 0, 255);


  // Send servo and motor values in a single array
  int data[2] = {servoValue, motorValue};
  radio.write(data, sizeof(data));

  Serial.println(data[1]);
// Check for high speed and update LED accordingly
  if (motorValue > 150) {
    digitalWrite(ledPin, HIGH); // Turn LED on for high speed
    Serial.println("VIBRATION DETECTED DUE TO HIGH SPEED");

  } else {
    digitalWrite(ledPin, LOW); // Turn LED off for low/medium
speedSerial.println("VIBRATION DETECTED DUE TO HIGH SPEED");
    Serial.println("SAFE DRIVING");
  }
}
```

Receiver code

```
#include <SPI.h>
#include "RF24.h"
#include <Servo.h>
#include <Wire.h>
#include <MPU6050.h>

RF24 radio(9, 8);
const uint64_t pipe = 0xE8E8F0F0E1LL;

Servo servo;

#define EN1 3
#define IN1 5
#define IN2 4

int minSpeed = 0;
int maxSpeed = 255;

int data[2];

int pwmValues[] = {0, 64, 128, 192, 255};
int rpmValues[] = {0, 300, 600, 900, 1200};

MPU6050 mpu;

void setup() {
  servo.attach(6);

  pinMode(EN1, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);

  Serial.begin(9600);
  radio.begin();
  radio.openReadingPipe(1, pipe);
  radio.startListening();

  Wire.begin();
  mpu.begin();
  mpu.calibrateGyro();
}

void loop() {
  if (radio.available()) {
    radio.read(data, sizeof(data));
```

```
    Serial.println("DATA RECEIVED FROM CONTROLLER");

    int servoValue = data[0];
    int motorValue = data[1];

    int servoPos = map(servoValue, 0, 255, 0, 180);
    servo.write(servoPos);

    if (motorValue >= 2) {
      digitalWrite(EN1, HIGH);
      int mappedSpeed = map(motorValue, 2, 255, minSpeed, maxSpeed);
      analogWrite(EN1, mappedSpeed);

      int estimatedRPM = 0;
      for (int i = 0; i < 5; i++) {
        if (mappedSpeed <= pwmValues[i]) {
          estimatedRPM = map(mappedSpeed, pwmValues[i - 1], pwmValues[i], rpmValues[i - 1],
rpmValues[i]);
          break;
        }
      }
      Serial.print("Motor speed: ");
      Serial.print(estimatedRPM);
      Serial.println(" RPM");

      mpu.update();
      int16_t accelX = mpu.getAccelerationX();
      int16_t accelY = mpu.getAccelerationY();
      int16_t accelZ = mpu.getAccelerationZ();

      if (abs(accelX) > 2000 || abs(accelY) > 2000 || abs(accelZ) > 2000) {
        Serial.println("High speed detected!");
      }

    } else {
      digitalWrite(EN1, LOW);
      digitalWrite(IN1, LOW);
      digitalWrite(IN2, LOW);
      Serial.println("Motor speed: Stop");
    }

    Serial.print("Servo angle: ");
    Serial.print(servoPos);
    Serial.println(" degrees");


    if (servoPos < 100) {
      Serial.println("Servo Direction: Left");
    } else if (servoPos > 150) {
```

```
      Serial.println("Servo Direction: Right");
    } else {
      Serial.println("Servo Direction: Center");
    }
  }
}
```