

**JOMO KENYATTA UNIVERSITY OF AGRICULTURE AND TECHNOLOGY**

**SCHOOL OF ELECTRICAL, ELECTRONIC AND INFORMATION ENGINEERING**

**DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING**

**FINAL YEAR PROJECT REPORT**

**DENZEL MURAGE NG'ENDO**

ENE211-0203/2018

**SUPERVISOR:** MR. S. CHEGE

**PROJECT: RF MUTLICONTROLLED WHEELCHAIR**

*A Final Year Project Report submitted to the Department of Electrical and Electronic Engineering in partial fulfillment of the requirements for the award of a Bachelor of Science Degree in Electrical and Electronic Engineering.*

**DECEMBER 2023**

# **Declaration**

This project report is my original work, except where due acknowledgment is made in the text, and to the best of my knowledge has not been previously submitted to Jomo Kenyatta University of Agriculture and Technology or any other institution for the award of a degree or diploma.

**NAME:** NG’ENDO DENZEL MURAGE

**REG. No:** ENE211-0203/2018

**SIGNATURE………………………………………… DATE ………………………………**

**TITLE OF PROJECT**: DESIGNING AN RF MULTI-CONTROLLED WHEELCHAIR.

**Supervisor Confirmation:**

This project report has been submitted to the Department of Electrical and Electronic Engineering, Jomo Kenyatta University of Agriculture and Technology, with my approval as the supervisor:

A blue line in the sky

Description automatically generated**NAME:** MR. S. CHEGE

**SIGNATURE………………………………………… DATE ………………………………**

T**able of Contents**

[**Declaration** i](#_Toc152507104)

[**List of Abbreviations** v](#_Toc152507105)

[**Abstract** vi](#_Toc152507106)

[**Chapter One** 1](#_Toc152507107)

[**1.1 Introduction** 1](#_Toc152507108)

[**1.2 Problem Statement** 2](#_Toc152507109)

[**1.3 Problem Justification** 3](#_Toc152507110)

[**1.4 Objectives** 4](#_Toc152507111)

[**1.4.1 Main Objective** 4](#_Toc152507112)

[**1.4.2 Specific Objectives** 4](#_Toc152507113)

[**1.5 Scope of the Project** 4](#_Toc152507114)

[**Chapter Two** 5](#_Toc152507115)

[**2.1** **Literature Review** 5](#_Toc152507116)

[**2.1.1 The Voice Control Technique** 5](#_Toc152507117)

[**2.1.1 Touch control technique** 7](#_Toc152507118)

[**2.1.3 Hand Gesture control technique** 9](#_Toc152507119)

[**2.2 Microcontrollers** 11](#_Toc152507120)

[**2.2.1 The Raspberry Pi** 11](#_Toc152507121)

[**2.2.2 ATmega 328** 12](#_Toc152507122)

[**2.3 Components and Software** 13](#_Toc152507123)

[**2.3.1 Display Module** 13](#_Toc152507124)

[**2.3.2 Motors** 16](#_Toc152507125)

[**2.3.3 Batteries** 20](#_Toc152507126)

[**2.3.4The Bluetooth** 25](#_Toc152507127)

[**2.3.5The mic sensor** 26](#_Toc152507128)

[**2.3.6 RF Transmitter Receiver module** 27](#_Toc152507129)

[**Chapter Three** 29](#_Toc152507130)

[**3.0 Methodology** 29](#_Toc152507131)

[**3.1 Block Diagram** 29](#_Toc152507132)

[**3.1.1 Functional Description.** 30](#_Toc152507133)

[**3.2 System Flow Diagram.** 31](#_Toc152507134)

[**3.2.1 Libraries Used** 33](#_Toc152507135)

[**3.3.1 Schematic Layout** 34](#_Toc152507136)

[**3.3.2 Power Circuit** 35](#_Toc152507137)

[**3.3.3 Pin Configuration** 35](#_Toc152507138)

[**Chapter Four** 41](#_Toc152507139)

[**4.0 Results** 41](#_Toc152507140)

[**4.0.1 Navigation** 41](#_Toc152507141)

[**4.0.2 Obstacle Detection.** 44](#_Toc152507142)

[**4.0.3 Financial Viability.** 44](#_Toc152507143)

[**Chapter 5** 45](#_Toc152507144)

[**5.0 Challenges** 45](#_Toc152507145)

[**5.0.1 Radio Frequency Transmission.** 45](#_Toc152507146)

[**5.0.2 Funding.** 46](#_Toc152507147)

[**5.1 Conclusion** 46](#_Toc152507148)

[**5.3 Recommendations** 46](#_Toc152507149)

[**Chapter Six** 47](#_Toc152507150)

[**6.0 Project Time Plan** 47](#_Toc152507151)

[**Chapter Seven** 48](#_Toc152507152)

[**7.0Proposed Budget** 48](#_Toc152507153)

[**Chapter Eight** 49](#_Toc152507154)

[**References** 49](#_Toc152507155)

[**Appendix** 52](#_Toc152507156)

[**Touch Navigation code** 52](#_Toc152507157)

[**Voice Navigation Code** 1](#_Toc152507158)

[**Gesture Navigation Code** 1](#_Toc152507159)

[**Transmitter Code** 1](#_Toc152507160)

[**Receiver Code** 2](#_Toc152507161)

**List of Figures**

[Figure 1 : Raspberry Pi CPU. 12](#_Toc152506854)

[Figure 2: The Raspberry Pi Display Module. 16](#_Toc152506855)

[Figure 3: Pole Configuration in a Brushless Motor. 19](#_Toc152506856)

[Figure 4: Structure of an AGM Battery. 23](#_Toc152506857)

[Figure 5: The Microphone sensor. 26](#_Toc152506858)

[Figure 6: RF Transmitter and Receiver 27](#_Toc152506859)

[Figure 7: Block Diagram 29](#_Toc152506860)

[Figure 8: Flowchart 32](file:///C:\Users\Denzel%20Murage\OneDrive\Desktop\Project%20Report%20edit%201.docx#_Toc152506861)

[Figure 9: Software Libraries 33](#_Toc152506862)

[Figure 10: Schematic diagram 34](#_Toc152506863)

[Figure 11: Power Circuit 35](#_Toc152506864)

[Figure 12: Hardware Setup 37](#_Toc152506865)

[Figure 13: Resistor 38](#_Toc152506866)

[Figure 14: Jumper wires 39](#_Toc152506867)

[Figure 15: L298N Driver 39](#_Toc152506868)

[Figure 16: Raspberry Pico 40](#_Toc152506869)

[Figure 17: Pi 3 and Display 40](#_Toc152506870)

[Figure 18: Touch Graphical User Interface 41](#_Toc152506871)

[Figure 19:Speech System Response 42](#_Toc152506872)

[Figure 20: Gesture Input Controls 43](#_Toc152506873)

[Figure 21: Ultrasonic Installation 44](#_Toc152506874)

[Figure 22:RF Transmitter/Receiver 45](#_Toc152506875)

[Figure 23: nRF24Lo1 Transceivers 45](#_Toc152506876)

# **List of Abbreviations**

|  |  |
| --- | --- |
| AGM | Absorbed Glass Mat |
| AVR | Advanced Virtual RISC: |
| AC | alternating current |
| ASK | Amplitude Shift Keying |
| API | Application Program Interface |
| BLE | Bluetooth Low Energy |
| BLDC | Brushless DC electric motor |
| CRT | cathode-ray tube |
| DC | Direct current |
| EEPROM | Electrically Erasable Programmable Read-Only Memory |
| GUI | graphical user interface |
| LED | light-emitting diode |
| LCD | Liquid Crystal display |
| MEMS | Micro-Electo mechanical Systems |
| OLED | organic light-emitting diode |
| PANs | personal area networks |
| PCB | printed circuit board |
| ROS | Robot OS |
| SLA | Sealed Lead Acid |
| SBCs | single-board computers |
| SIG | Special Interest Group |
| Tx/Rx | transmitter/receiver |

# **Abstract**

The Multi-controller-based Wheelchair System was a project aimed at enhancing the safety, mobility, and independence of wheelchair users by replacing traditional joystick-based controls with touch, speech, and gesture control. The system enabled users to control their wheelchairs through voice commands, hand gestures, or touch, providing a more accessible and intuitive interface. Additionally, the system incorporated safety features such as obstacle detection and emergency stop controls to prevent accidents and injuries. By addressing the limitations of joystick-based controls, including the difficulty of use for individuals with limited hand or arm mobility, fatigue and repetitive stress injuries, and potential damage to the joystick, the project aimed to improve user comfort and effectiveness. The expected outcomes of the project included a functional prototype of the Multi-controlled-based Wheelchair System and an evaluation of its usability, safety, and effectiveness. The project involved hardware and software development, testing, and evaluation, including the integration of a motor controller with a suitable microcontroller and the development of control algorithms, user interface, and safety features. Through iterative design, testing, and feedback, the system was enhanced to meet the needs of wheelchair users.

# **Chapter One**

## **1.1 Introduction**

A person's capacity to move and perform physical activities can be limited by a motor-physical impairment. Mobility presents a significant challenge for those with motor disabilities. Fortunately, supportive tools like electric wheelchairs enable these individuals to become mobile and independent. If not, they will be reliant on other people to go around. The task of steering a wheelchair is simple. The speed and direction of the motor's motion are managed using a standard joystick. Similar to a gas pedal in a car, the wheelchair will go quicker the further you push in a particular direction [1].

These individuals require a different wheelchair's control. One of the main functions of a multi-controlled wheelchair is to let users with severe impairments who are unable to operate the chair with a regular joystick have autonomous mobility.

Compared to average people, those who have mobility issues are more likely to experience depression or anxiety. Consequently, gaining more autonomy in their mobility could greatly enhance their quality of life [2]. The advantage of having multiple mobility options is having an alternative since each option may not be ideal for given situations. Take for example when in a noisy area the voice control option would not be suitable for the environment hence the user can opt to use the other options available.

Regular joystick-controlled wheelchairs suffer from several drawbacks that hinder the safety, mobility, and independence of wheelchair users. The reliance on joystick controls poses numerous challenges for wheelchair users. Firstly, individuals with limited hand or arm mobility encounter difficulties in operating traditional joysticks effectively. This limitation restricts their ability to control their wheelchairs and compromises their independence.

Additionally, the constant pushing of the joystick during extended periods of use can lead to fatigue and repetitive stress injuries, further impacting the user's well-being. Moreover, traditional joystick controls are prone to damage due to the constant force inflicted on them during regular use. This susceptibility to wear and tear necessitates frequent repairs or replacements, causing inconvenience for users and potential financial burdens [3].

To address the existing limitations, the project proposed a Multi-controlled based Wheelchair Safety system that provides users with an innovative and accessible control interface. By incorporating touch, speech, and gesture control, it aimed to offer wheelchair users a selection of intuitive and customizable modes of navigation. This multi-modal approach eliminated the dependency on joystick controls and opens up a world of possibilities for individuals with varying levels of hand or arm mobility.

The solution extends beyond control enhancements. The proposed system was to include safety features such as obstacle detection, collision avoidance, and emergency stop controls. These safety measures aim to prevent accidents and injuries, instilling confidence and peace of mind in both users and their caregivers.

## **1.2 Problem Statement**

Individuals with mobility limitations face numerous challenges in their daily lives, with limited accessibility and usability of traditional joystick-based controls in motorized wheelchairs being a significant obstacle. The current joystick controls suffer from drawbacks such as limited functionality, usability issues, and high costs, hampering the safety, mobility, and independence of wheelchair users [4]. This project aims to address these limitations and develop a Multicontroller-based Wheelchair Safety system that revolutionizes the control interface and affordability of motorized wheelchairs.

The reliance on joystick controls poses challenges for wheelchair users, particularly those with limited hand or arm mobility. Operating joysticks effectively becomes difficult, restricting users' ability to control their wheelchairs and compromising their independence. Additionally, continuous use of the joystick can lead to fatigue and repetitive stress injuries, affecting users' well-being. Moreover, traditional joystick controls are prone to damage due to constant force, requiring frequent repairs or replacements, causing inconvenience and financial burdens for users.

The limited availability of alternative control methods further compounds the problem, restricting users' ability to navigate their environment comfortably and effectively. This inadequacy in control interfaces significantly impacts the safety, mobility, and overall quality of life for wheelchair users.

To overcome the limitations of traditional joystick controls, the project proposes the development of a Multi-controlled based Wheelchair Safety system. This system will provide users with an innovative and accessible control interface by incorporating touch, speech, and gesture control. The multi-modal control options will offer wheelchair users intuitive and customizable modes of navigation, reducing their dependence on joystick controls.

In addition to control enhancements, the proposed system will include safety features such as obstacle detection, collision avoidance, and emergency stop controls. These safety measures aim to prevent accidents and injuries, instilling confidence and peace of mind in both users and caregivers [2].

## **1.3 Problem Justification**

The multi-controlled wheelchair is set to make navigation of wheelchair users by overcoming several shortcomings of the already existing motorized wheelchairs. The first issue to resolve would be Limited Control Options. Traditional joystick controls are not suitable for individuals with limited hand or arm mobility, as they face challenges in operating them effectively. This limitation restricts their ability to control their wheelchairs, resulting in decreased independence and compromised mobility. A more inclusive control interface is needed to accommodate users with varying levels of mobility impediments. The availability of options provides convenience for users in terms of comfort since they can easily switch to the option that suits a given situation or environment.

The second issue to resolve would be the Physical Strain and Fatigue caused by the continuous pushing of joysticks during extended periods of wheelchair use can cause fatigue and repetitive stress injuries. This physical strain not only impacts the user's well-being but also hampers their ability to operate the wheelchair comfortably and effectively. A solution that reduces physical strain and provides alternative control methods is essential.

The third thing tackled would be the Wear and Tear of Joystick Controls since they are prone to damage due to the constant force inflicted on them. This susceptibility to wear and tear necessitates frequent repairs or replacements, causing inconvenience for users and potentially adding financial burdens. A more durable and reliable control system is required to mitigate these issues.

The multi-controlled wheelchair project is set to solve the mentioned shortcomings to revolutionize the navigation of wheelchair users by improving the efficiency of motorized wheelchairs.

## **1.4 Objectives**

### **1.4.1 Main Objective**

1. To develop a Multi-controlledbased Wheelchair Safety system that enhances the safety, mobility, and independence of wheelchair users by the end of the project.

### **1.4.2 Specific Objectives**

1. To develop and implement touch, speech, and gesture control interfaces for wheelchair navigation by the end of the project.
2. To integrate safety features such as obstacle detection, collision avoidance, and emergency stop controls by the end of the project.
3. To develop an inexpensive system concept.

## **1.5 Scope of the Project**

The scope of the project was to develop a multi-controlled wheelchair system utilizing various components. The wheelchair was to be equipped with a microcontroller unit, which was to receive input from a microphone sensor, enabling voice commands for navigation. An RF Tx Rx module was also to be incorporated to facilitate wireless communication with external devices. An accelerometer provided motion-sensing capabilities, allowing the wheelchair to detect and respond to changes in direction or speed.

A display module served as the vital information and feedback interface. The integration of capacitors, other components and cables ensured proper electrical connectivity and signal processing throughout the system. PCBs and breadboards were utilized for circuit design testing and prototyping. A battery set provided a power supply, while push buttons and switches offered manual control options. ICs and IC sockets were utilized for additional functionality and ease of maintenance. Overall, this project aimed at creating a wheelchair system that is adaptable, controllable through multiple input methods and enhanced the mobility and independence of individuals with physical disabilities.

# **Chapter Two**

## **Literature Review**

There are numerous electric wheelchair control kinds available on the market and in research. Each one has a different usage and price. The most well-known and widely used type is possibly the conventional joystick-controlled type. Power wheelchairs have the same basic structural components, which are as follows [1]: Wheels, a propulsion system, a battery, and a controller for a chair. A 2-pole motor is used by lightweight power wheelchairs, whereas a 4-pole motor is used by heavy-duty power wheelchairs. The 4-pole motor has a greater carrying capacity and gives more useful alternatives.

Sealed Lead Acid (SLA) batteries are used in power wheelchairs. They can have an output of 4 to 5 amps and can be either wet or dry cell batteries. When the chair is not in use, the battery can be recharged using a regular electrical socket. A wheelchair's drive system consists of at least one clutch and at least one gear set housed in a central hub housing that is operationally connected to the drive surface of the primary wheel [2] being driven. The electric wheelchair is steered and controlled using the joystick or grip.

The typical components of a joystick are a gimbal knob, an on/off switch, a speed control, and a battery gauge. The user pushes the wheelchair's gimbal in the desired direction to operate the joystick; the further he pushes in one direction, the faster the [3] wheelchair will move in that direction. Other methods are employed to control the [4]wheelchair because it is known that a standard joystick is not ideal for a person who is tetraplegic and cannot use his hands. every wheelchair

With the exception of the controller, elements remain unchanged. Even the joystick control may be abandoned with the addition of a different specialized [4] [5]controller. Both the drive system and the joystick controller can be connected to the extra controller. A list of some electric wheelchair control techniques is provided below:

### **2.1.1 The Voice Control Technique**

Through voice recognition, the system is intended to control a wheelchair by speech commands. This is by the use of a mic to perceive sound. The design can incorporate several wheelchair-friendly features, including low power consumption (about 20 watts), small dimensions (10"x11"x14"), low weight (roughly 22 lb), and the ability to connect to existing joystick control cable connectors directly into this system [6].

To maintain chair speed and direction, the system employs an eight-word vocabulary (e.g., "forward," "slower," "right") and a feedback control system. According to previous tests, the microphone and speaker can achieve a recognition rate of more than 90% [6]. Future system enhancements are anticipated to improve the chair's recognition rate and practical utility. Voice instructions are possibly a viable and practical form of wheelchair control, according to the results of laboratory testing with normal and quadriplegic participants [7].

#### Advantages of Voice Controls

The multi-controlled wheelchair with touch, voice, and gesture controls offers significant advantages, particularly for individuals with disabilities. Firstly, it enables individuals without limbs to become self-sufficient by providing them with an accessible and intuitive control system. Through touch, voice, or gesture controls, users can navigate and operate the wheelchair independently, enhancing their mobility and overall independence.

In addition, the multi-controlled wheelchair requires less hardware and is smaller in size compared to traditional wheelchair systems. The integration of advanced control mechanisms allows for a more compact and streamlined design, improving portability and maneuverability.

Furthermore, the use of speech commands reduces the need for manual labor in operating the wheelchair. By controlling the wheelchair through speech, individuals can navigate and control their movements with minimal physical exertion, reducing the reliance on external assistance.

The multi-controlled wheelchair also has the potential for automated operation. By implementing advanced strategies and technologies, such as automation, the wheelchair can be transformed into a fully automated system. This opens up possibilities for autonomous navigation, obstacle detection, and other intelligent functionalities, enhancing convenience and safety for users.

Moreover, the integration of Bluetooth technology in the multi-controlled wheelchair results in less wiring complexity. Wireless communication via Bluetooth reduces the need for extensive wiring connections, simplifying the overall design and maintenance of the wheelchair.

#### Disadvantages of Voice Controls

While the multi-controlled wheelchair with touch, voice, and gesture controls offers numerous benefits, there are a couple of considerations to keep in mind. Firstly, in a noisy atmosphere, there may be challenges with accurate command recognition. Ambient noise can interfere with the voice recognition system, leading to potential misinterpretation or incomplete understanding of commands. This limitation can affect the responsiveness and reliability of the wheelchair's control system in noisy environments.

Additionally, the control system of the multi-controlled wheelchair may be limited to basic commands such as Back, Left, Right, and Stop. However, this can be improved by integrating Application Program Interface (API) modules, which can expand the range of recognized commands and enhance the versatility of the wheelchair's control capabilities. By incorporating additional modules and functionalities, users can enjoy a broader range of commands and interactions with the wheelchair, making the control system more comprehensive and adaptable to their specific needs.

Addressing these considerations, such as improving noise tolerance and expanding command recognition through API modules, can further enhance the performance and usability of the multi-controlled wheelchair, ensuring a more efficient and customized user experience.

### **2.1.1 Touch control technique**

It is an easy-to-use technology that runs on a touch screen. A wheelchair can be operated with a single finger, which requires less force. The touch screen technology uses less power for the user to control it, so even the patient can use it without fear of running out of power. Wheelchair movement can be regulated in four directions: forward, reverse, left, and right.

Embedded instructions are used to program a microcontroller. Programming languages such as C or Python are used to create the embedded instructions. This microcontroller can communicate with both input and output modules. The information is provided to the Microcontroller (onboard computer) through the touchscreen-based sensor.

The system employs anLiquid Crystal display (LCD) that shows graphics of directions in the form of arrow keys. The controller communicates with dc motors attached to the wheelchair to regulate the wheelchair's direction.

#### Advantages of Touch Controls

The multi-controlled wheelchair with touch, voice, and gesture controls offers several advantages over traditional joystick controls. Firstly, the operation of the wheelchair becomes faster and more efficient as instructions can be keyed in at a quicker pace compared to using joysticks. By simply touching the screen or using voice commands, users can input their desired movements with greater speed and precision.

In addition, the touch controls require lesser physical force as users only need to touch the screen to control the wheelchair, rather than exerting pressure on a joystick. This lighter touch reduces the physical effort required to operate the wheelchair, particularly for individuals with limited strength or mobility.

Furthermore, the multi-controlled wheelchair is designed to be user-friendly, making it easier for individuals to navigate and control. The intuitive nature of touch, voice, and gesture controls simplifies the user experience, allowing for a more accessible and enjoyable operation of the wheelchair.

Another advantage is the reduced complexity of wiring in the design of the multi-controlled wheelchair. Compared to traditional joystick controls, the touch, voice, and gesture controls require fewer wired connections, resulting in a simpler and more streamlined design. This not only facilitates the manufacturing process but also contributes to the overall reliability and ease of maintenance of the wheelchair.

#### Disadvantages of Touch Controls

While the multi-controlled wheelchair offers various advantages, there are a couple of considerations to be aware of. Firstly, one limitation is the absence of auto braking. However, this can be mitigated by installing obstacle detection features, which can provide additional safety measures by automatically detecting and avoiding obstacles in the wheelchair's path. These features enhance the overall control and maneuverability of the wheelchair, reducing the risk of collisions or accidents.

Another aspect to consider is the power consumption of touch screens. Depending on the type of touch screen used, they can be power-hungry, which may affect the battery life of the wheelchair. If left on without being used for an extended period, the touch screen's continuous power consumption can drain the battery more quickly. To optimize battery life, it is recommended to turn off or minimize the use of the touch screen when not actively needed, conserving energy for other essential functions of the wheelchair.

By addressing these considerations and implementing obstacle detection features while managing touch screen power consumption, the multi-controlled wheelchair can continue to provide enhanced safety and efficient operation while ensuring optimal battery performance.

### **2.1.3 Hand Gesture control technique**

Hand gesture controls on wheelchairs can be very beneficial to the users since they would help reduce difficulties encountered during the motion of people with motor disabilities. Outlined below are some of these strategies that have been used in gesture controls for understanding purposes:

Kutbi et al. [8], for example, suggested a wheelchair control model based on head movement monitoring. The images of the head in this piece were captured using an egocentric camera. The wheelchair was modeled using a TI-TAN18CS, an Arduino Mega was utilized as a command processor, and Robot OS (ROS) was used as the framework. The system achieved a performance of approximately 85.7%. The system's cost remains considerable, and wearing an egocentric camera on one's helmet is inconvenient.

Tejonidhi et al. [9] suggested an eye-pupil tracking-based wheelchair movement system. For recognizing the eyeballs in RGB photos, this system uses a Philips microcontroller and the Viola-Jones MATLAB method. However, for real-time applications, the detection procedure is crucial, and performance ranges from 70% to 90%. As a result, it cannot be used in a real-world context.

Utaminingrum et al. created a wheelchair-controlling mechanism by employing RGB pictures to track a target item in front of the [10] wheelchair. Human detection using the HOG algorithm [11], interested human tracking using the CAMSHAFT algorithm [12], and movement detection flow the wheelchair movement control. In this system, target selection among several human objects is a difficult and problematic scenario; performance varies by roughly 80% and has to improve for real-world applications.

The proposed system involves the use of Micro-Electomechanical Systems (MEMS) technology by incorporating an accelerometer into a hand wearable in order to control the movement of the wheelchair, flex sensors gloves, and Raspberry Pi. The performance ranges between 90% and 100%.

#### Advantages of Hand Gesture Controls

The multi-controlled wheelchair with touch, voice, and gesture controls offers several notable benefits. Firstly, it allows users to control the wheelchair in a natural and convenient manner. By incorporating touch, voice, and gesture controls, users can interact with the wheelchair's interface in intuitive ways, enabling smooth and easy navigation.

In environments with high levels of noise, the touch and gesture controls can serve as a complementary substitute for voice commands. This ensures that users can still operate the wheelchair effectively even when verbal communication may be challenging or ineffective.

Another advantage is that the multi-controlled wheelchair reduces the need for unnecessary contact with areas like dirty tires when attempting to move by pushing oneself. With the touch, voice, and gesture controls, users can navigate and adjust the wheelchair without having to physically interact with potentially unclean or inconvenient areas.

Moreover, the multi-controlled wheelchair offers flexibility and customization options. Users can adjust the wearable components, such as the armrests or control interfaces, to enhance their comfort and optimize the wheelchair's ergonomics based on their specific needs and preferences.

#### Disadvantages of Hand Gesture Controls

While the multi-controlled wheelchair with touch, voice, and gesture controls offers various advantages, there are a few considerations to keep in mind. Firstly, there can be instances where errors occur during the interpretation of commands, leading to unintended movements, such as moving in the opposite direction of the intended one. This highlights the importance of precise and accurate control mechanisms to ensure the desired movements are executed correctly.

Secondly, the applicability of these controls may be limited for individuals with upper limb motor disabilities. People who have difficulty with fine motor control or limited upper limb mobility may face challenges in effectively utilizing touch, voice, or gesture controls, which rely on specific gestures or movements.

Additionally, using the multi-controlled wheelchair with touch, voice, and gesture controls may result in more fatigue compared to other control options within the proposed system. Continuous engagement with touch interfaces, vocal commands, or repetitive gestures can strain the user physically and mentally over time.

Lastly, there is a learning curve associated with using the controls effectively and avoiding errors. Users may need time and practice to become proficient in operating the wheelchair using touch, voice, and gesture controls, as coordination and familiarity with the specific control methods are crucial to minimizing mistakes.

While these considerations should be acknowledged, the multi-controlled wheelchair still provides a versatile and personalized experience for individuals who can effectively utilize the touch, voice, and gesture controls, offering increased independence and mobility.

Below will be a delve into the systems and components that can be used to design and enable the functionality, of the proposed multi-controlled wheelchair.

## **2.2 Microcontrollers**

A printed circuit board (PCB) with hardware and circuitry on it is called a microcontroller development board. Its purpose is to enable experiments with a particular microcontroller board function. The development boards include a CPU, memory, chipset, as well as on-board peripherals with debugging capabilities, such as an LCD, keypad, USB, serial port, ADC, RTC, motor driver ICs, SD card slot, Ethernet, etc. Microcontroller boards' specifications include bus, processor, memory, number of ports, port type, and operating system. Programs for embedded devices, such as different controllers, home appliances, robots, point-of-sale (PoS) terminals, kiosks, and information appliances, are evaluated using these.

### **2.2.1 The Raspberry Pi**

The Raspberry Pi Foundation and Broadcom collaborated to create a line of compact single-board computers (SBCs) called Raspberry Pi in the UK [13]. The initial focus of the Raspberry Pi project was to encourage the study of fundamental computer science in schools [14]. The initial model sold outside of its intended market for applications like robotics because it was more widely used than planned. Because of its inexpensive cost, versatility, and open design, it is frequently utilized in numerous fields, including weather monitoring [15]. Due to its support of HDMI and USB standards, computer and electronic enthusiasts frequently utilize it.

It does not have an integrated hard drive or solid-state drive; instead, it boots from an SD card and stores data on it for a long time. Raspbian, an operating system based on the Linux kernel, is meant to run on this board, to install different packages including Node.js, Java, the LAMP stack, Python, and much more.

Currently, The Linux community is constantly working to expand Linux's robust, adaptable kernel and open runtime platform to support new processors, buses, devices, and protocols. Embedded hardware initiatives that must be cost-effective and timely can benefit from its independence or free availability. By utilizing the strength and adaptability that a genuine multi-tasking operating system gives to embedded devices, embedded device projects can frequently minimize hardware costs.

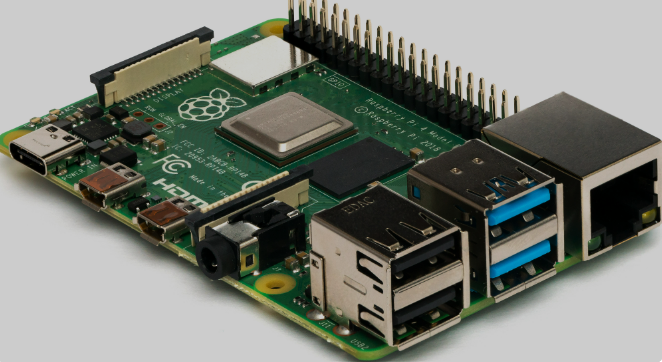


Figure 1 : Raspberry Pi CPU.

The specifications of a Raspberry Pi include;

1. Processor: 1.2GHz, 64-bit quad-core ARMv8 CPU
2. 802.11n Wireless LAN
3. Bluetooth 4.1
4. Bluetooth Low Energy (BLE)
5. 1GB RAM
6. 4 USB ports
7. 40 GPIO pins
8. Full HDMI port
9. Combined 3.5mm audio jack and composite video
10. Camera interface (CSI)
11. Display interface (DSI)
12. Micro SD card slot
13. video Core IV 3D graphics core

### **2.2.2 ATmega 328**

An Advanced Virtual RISC (AVR) microcontroller is the ATmega328. It supports the processing of 8-bit data. The ATmega-328 features inbuilt flash memory of 32 KB.

Electrically Erasable Programmable Read-Only Memory (EEPROM) on the ATmega328 [16] is 1KB. This feature demonstrates that the microcontroller can still store data and provide outcomes after receiving an electric source, even if the electric supply is disconnected. Additionally, the ATmega-328 contains 2KB of static randomaccess memory. Later explanations will cover additional traits. The ATmega 328 is the most well-liked product on the market right now thanks to a variety of features. These characteristics include an advanced RISC architecture, good performance, low power consumption, programmable Serial USART [17], 6 PWM pins, a real-time counter with a separate oscillator, a programming lock for software security, a throughput of up to 20 MIPS, etc.

The ATmega328 is widely utilized in numerous projects and autonomous systems that require a basic, low-power, low-cost microcontroller. This chip is perhaps most commonly used on the popular Arduino development platform, specifically the Arduino Uno, Arduino Pro Mini [18], and Arduino Nano versions.

## **2.3 Components and Software**

### **2.3.1 Display Module**

A display module is a highly integrated real-time embedded device that is designed to interact and communicate with its surroundings efficiently. Displays are used for a variety of purposes, including debugging, GUI, and just showing messages. There are primarily two types of screens available: LCD and OLED. Both displays differ in terms of working, output, and operating voltage yet perform the same function of displaying images/text. TFT LCD displays are an additional type of display. These monitors are typically equipped with touch panels and are used for graphical user interface (GUI) applications.

#### LCD Screens

A liquid-crystal display (LCD) is a flat-panel display or other electronically controlled optical device that exploits the light-modulating characteristics of liquid crystals in conjunction with polarizers. Liquid crystals do not emit light directly [19], but rather utilize a backlight or reflector to produce color or monochromatic pictures LCDs can display arbitrary images (as in a general-purpose computer display) or fixed images with minimal information content that can be displayed or hidden. Preset words, digits, and seven-segment displays, such as in a digital clock, are all examples of devices with these displays. They both use the same underlying technology, with the exception that arbitrary images are created from a matrix of small pixels, whilst other displays contain larger elements.

LCDs are utilized in a variety of applications, including LCD televisions, computer monitors, instrument panels, cockpit displays in airplanes, and indoor and outdoor signs. LCD projectors with small LCD screens are widespread, as are portable consumer gadgets such as digital cameras, watches, calculators, and mobile phones, including smartphones. In practically all applications, LCD screens have replaced heavy, large, and inefficient cathode-ray tube (CRT) displays.

#### OLED Displays

An organic light-emitting diode (OLED), also known as an organic electroluminescent (organic EL) diode, is a light-emitting diode (LED) with an emissive electroluminescent layer that emits light in response to an electric current. This organic layer is sandwiched between two electrodes, at least one of which is transparent. OLEDs are used to make digital displays for devices like television screens, computer monitors, and portable systems like smartphones and handheld game consoles.

There are two types of OLEDs: those that use tiny molecules and those that use polymers. When mobile ions are added to an OLED, a light-emitting electrochemical cell (LEC) is formed with a somewhat different mode of operation. A passive-matrix (PMOLED) or active-matrix (AMOLED) control method can be used to power an OLED display. Each row and line in the display is controlled sequentially, one by one, in the PMOLED system, whereas AMOLED control employs a thin-film transistor (TFT) backplane to directly access and switch each individual pixel on or off, allowing for better resolution and larger display sizes.

#### The Raspberry Pi Display Module

The Raspberry Pi Touch Display is a touch-screen display that connects to the Raspberry Pi via the DSI connector. In some cases, it is possible to use both the HDMI and LCD screens at the same time (software support is required). The DSI display A 7-inch touchscreen Raspberry Pi display for interactive projects like tablets, entertainment systems, and information dashboards, is meant to work with all Raspberry Pi models; [15] however, early models that lack mounting holes (the Raspberry Pi 1 Model A and B) will require additional mounting hardware to suit the display PCB's HAT-dimensioned stand-offs. The Raspberry Pi Touch Display and Raspberry Pi are configured by default to perform best when viewed from slightly above, such as on a desktop. When viewing from below, physically rotate the display and then instruct the system software to adjust by operating the screen upside down. Furthermore, by adding a*dtoverlay* command during configuration, one can modify the rotation of the touchscreen independently of the display.

**Specifications**

Raspberry Pi OS includes touchscreen drivers that support ten-finger touch and an on-screen keyboard, allowing you to use the device without connecting a keyboard or mouse. An adapter board that handles power and signal conversion links the 800 x 480 monitor to the Raspberry Pi. On all Raspberry Pi computers except the Raspberry Pi 400 [15] and the Raspberry Pi Zero line, just two connections are required: power from the GPIO port and a ribbon cable that connects to the DSI port.

* 7-inch diagonal display size
* Display resolution: 800 (RGB) 480 pixels
* The active area is 154.08mm by 85.92mm.
* Touch panel: True multi-touch capacitive touch panel with up to ten absolution points.

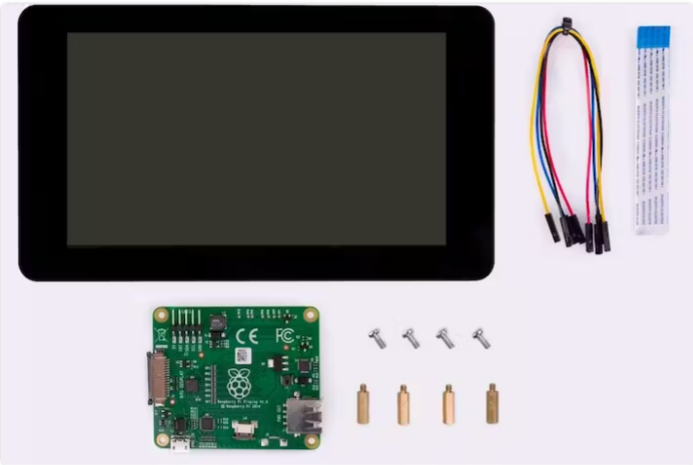


Figure 2: The Raspberry Pi Display Module.

### **2.3.2 Motors**

An electric motor is a device that converts electrical energy into mechanical energy. Most electric motors generate force in the form of torque imparted to the motor's shaft by interacting between the magnetic field of the motor and electric current in a wire winding. Mechanically, an electric generator is identical to an electric motor, but it runs with a reversed flow of power, transforming mechanical energy into electrical energy

Direct current (DC) sources, such as batteries or rectifiers, or alternating current (AC) sources, such as a power grid, inverters, or electrical generators, can power electric motors [20]. To accomplish the long-term performance and reliability that customers expect from medical mobility equipment, long-life, high-torque DC brush or brushless gear motors are required. The proposed system intends to use the DC type of motor.

#### DC Brush motor

A brushed DC electric motor is an internally commutated electric motor that uses an electric brush for contact and is powered by a direct current power source. The speed of a brushed DC motor can be changed by adjusting the operating voltage or the strength of the magnetic field [20]. The speed and torque characteristics of a brushed motor can be changed depending on how the field is connected to the power source to produce a constant speed or speed that is inversely proportional to the mechanical load. Brushed motors are still employed in electric propulsion, cranes, paper machines, and steel rolling mills.

Current-carrying conductors are always found on the rotating component. In practice, these wires are powered by a direct current power source. It transfers electricity to the coil via metallic brushes that rotate in tandem with the rotor. When a battery or any other DC source delivers power to the motor, energy passes from the source to the armature [20] via the brushes, which are typically located on opposing sides of the motor's shaft. Brushes are a critical component of this motor. Through physical contact with the commutator, these brushes deliver electric current to the armature. The armature coil begins to behave like a magnet as soon as it receives energy or power. At this moment, its poles begin to repel the poles of the permanent magnet that makes up the stator. As the poles repel, the motor shaft, to which the armature coil is linked, begins to revolve at a high rate and with torque [21]. The magnetic field strength around the armature determines the speed and torque.

##### **Advantages of DC Brushed Motors**

Brushed DC motors offer several advantages that make them a preferred choice in certain applications. One notable benefit is the overall low building costs associated with these motors. They are relatively simple in design, which contributes to cost savings during the manufacturing process. Additionally, the life of brushed DC motors can often be extended through rebuilding, where components such as brushes and commutators can be replaced, allowing for continued use and cost-effectiveness.

The controller used with brushed DC motors is typically simple and economical, making it easy to implement and maintain. In applications where a fixed speed is required, no additional controller is necessary, simplifying the system setup and reducing overall complexity and cost.

Furthermore, brushed DC motors are well-suited for harsh operational conditions. Their robust construction enables them to withstand challenging environments, including high temperatures, vibration, and dust, making them suitable for various industrial and rugged applications.

##### **Disadvantages of DC Brushed Motors**

While brushed DC motors have their uses, there are a few drawbacks to consider. Firstly, brushed DC motors are generally less effective compared to other motor types in terms of efficiency and overall performance. One notable issue is the electrical and electromagnetic noise they generate. The continuous establishment and breaking of inductive circuits by the commutators' switching activity creates significant levels of electrical and electromagnetic noise, which can be undesirable in certain applications.

Another disadvantage is the relatively short lifespan of brushed DC motors. This is primarily due to the brushes and commutators experiencing wear and tear as they are in constant direct contact with the motor shaft. Over time, this contact leads to the degradation of the brushes and commutators, resulting in decreased motor efficiency and eventual failure.

Brushless DC motors employing power electronic devices have superseded brushed motors in many applications since the brushes wear out and must be replaced.

#### DC Brushless motor

A brushless DC electric motor (BLDC), often known as an electronically commutated motor, is a synchronous motor that operates on direct current (DC). It employs an electronic controller to switch DC currents to the motor windings, resulting in magnetic fields that revolve in space and are followed by the permanent magnet rotor. To control the speed and torque of the motor, the controller modifies the phase and amplitude of the DC current pulses [22]. This control system replaces the mechanical commutator (brushes) found in many traditional electric motors.

A brushless motor system is often built in the same way as a permanent magnet synchronous motor (PMSM), although it can also be a switching reluctance motor or an induction (asynchronous) motor. They are built in a different way than brushed motors. Instead of a mechanical commutator and brushes, electronic commutation rotates the stator's magnetic field. Active control electronics are required for this. They can also be out-runners (the rotor is surrounded by the stator), in-runners (the rotor is surrounded by the stator), or axial (the rotor and stator are flat and parallel). [22].

The number of phases refers to the number of windings in a brushless motor. Though brushless motors can be built with a variety of phase counts, three-phase brushless motors are the most popular. Small cooling fans that require only one or two phases are an exception. A brushless motor's three windings are coupled in either a "star" or a "delta" configuration. In either case, three wires connect to the motor, and the drive technique and waveform are the same.

Motors with three phases can be built using various magnetic configurations known as poles. The rotor of the simplest three-phase motors contains only one set of magnetic poles, one North and one South. Motors with more poles can also be manufactured, which necessitates more magnetic sections in the rotor and more windings in the stator. Higher pole counts can provide a better performance, but smaller pole counts can achieve very high speeds.

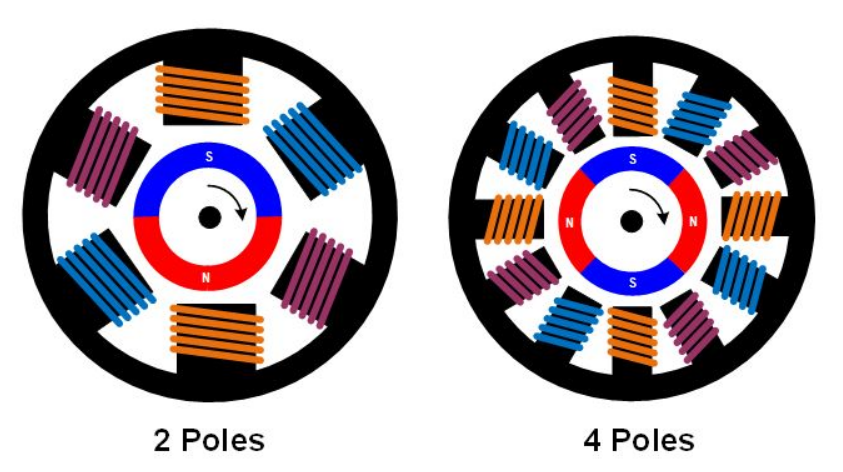


Figure 3: Pole Configuration in a Brushless Motor.

##### **Advantages of Brushless Motors**

Brushless DC (BLDC) motors offer several advantages over brushed DC (BDC) motors. One significant benefit is the absence of carbon brushes in BLDC motors, resulting in reduced brush replacement frequency and lower maintenance expenses. In terms of performance and efficiency, BLDC motors outshine BDC motors due to their electronic control capabilities. This electronic control allows for precise speed and position management, leading to superior overall motor performance. Furthermore, the lifespan of a brushless DC motor is approximately six times longer than that of a brushed DC motor.

Brushes in BDC motors can generate strong sparks, which can cause the motor to have a short life or even burn out. In contrast, brushless DC motors eliminate the spark issue, minimizing the risk of burnout associated with sparking concerns. Additionally, BLDC motors are compact in size and have a high torque-to-weight ratio, making them well-suited for various robotics and medical applications, including robotic arms and legs.

Compared to motors of the same rating, brushless DC motors operate with relatively low noise levels, contributing to a quieter working environment. This noise reduction is beneficial in applications where noise control is essential. Overall, brushless DC motors offer improved longevity, efficiency, reliability, and compactness, making them a preferred choice for many applications across industries.

##### **Disadvantages of Brushless Motors**

There are a few considerations when comparing brushless DC (BLDC) motors to brushed DC (BDC) motors. One factor to note is that BLDC motors are generally more expensive than BDC motors due to their more complex design and the addition of an electronic controller, which increases the overall cost of the system. In contrast, traditional motors typically employ low-cost mechanical commutation configurations using brushes.

At low speeds, BLDC motors may experience small vibrations during rotation. However, these vibrations tend to diminish as the motor operates at higher speeds. This characteristic should be taken into account when considering the specific application requirements.

Another aspect to be aware of is the intrinsic natural vibration frequency of BLDC motors. In some cases, this natural frequency may align or come close to the vibration frequency of the surrounding body or plastic parts, leading to resonance phenomena. However, through proper adjustment and tuning, it is possible to reduce these resonances. It is worth noting that resonance can be observed in many devices that utilize BLDC motors.

### **2.3.3 Batteries**

A battery is a device that uses an electrochemical oxidation-reduction (redox) cycle to transfer chemical energy contained within its active components directly into electric energy. An electric circuit is used to transmit electrons from one material to another in this type of reaction. On-demand batteries and similar devices accept, store, and release electricity. Batteries, like many other common energy sources, [23]employ chemistry to store energy in the form of chemical potential.

There are a few things to consider when selecting a battery for powering a wheelchair. This includes power density, power/battery capacity, and the Discharge rate. The power density of a device refers to how quickly it can discharge its energy, whereas the energy density of a device refers to how much energy it contains. Batteries have a low power density but a high energy density, which means they hold a lot of energy that is slowly discharged. Power density is computed by dividing the battery's power delivered (Vo Io) by its mass (kg) as follows;

|  |  |  |
| --- | --- | --- |
|  |  | (2.1) |

Battery capacity is also key,it is given in ampere-hours and is defined as the total quantity of electricity generated by electrochemical reactions in the battery. A 5 Ah battery, for example, can provide a steady discharge current of 1 C (5 A) for 1 hour. For the same battery, a discharge current of 0.1 C (500 mA) can be pulled for 10 hours.

Discharge rate: Divide the battery capacity by the time it takes to deplete the battery to get the discharge rate.

|  |  |
| --- | --- |
|  | (2.2) |

A 500 Ah battery, for example, that is entirely discharged in 20 hours has a discharge rate of 500 A h 20 h = 25 A.

Deep cycle AGM (short for Absorbed Glass Mat) batteries and gel batteries are the two types of batteries that can power electric wheelchairs. Both are Sealed Lead Acid (SLA) batteries.

#### Absorbed Glass Mat Batteries

Absorbed Glass Mat (AGM) is an abbreviation for absorbent glass mat, which refers to the power generation mechanism used by these devices. This fiberglass mat is placed between your battery's positive and negative lead plates. The battery acid is then absorbed and held, as the name implies. This prevents it from freely flowing around the battery. [23]

Absorbent Glass Mat is a high-performance lead-acid battery that can meet the increasing electrical demands of today's cars and start-stop applications. AGM batteries are exceptionally vibration resistant, completely sealed, non-spillable, and require no maintenance. When compared to ordinary lead-acid batteries, AGM provides superior cycling performance, less gassing, and less acid leakage. Superior life performance is the result of all the advantages of AGM technology.

The positive and negative plates of AGM batteries are separated by an absorbent glass mat that absorbs and traps the battery's acid, preventing it from flowing freely inside the battery. The plates are packed securely inside each cell and kept under pressure in the plastic box. Internal compression reduces plate material shedding caused by cycling and allows for much longer life. The element compression also reduces internal resistance while increasing pulse power output. Throughout the life of the battery, a hard container maintains the appropriate compression. The basic design also provides relief venting to relieve pressure caused by mild gassing during charging. To prevent air from escaping into the battery, the valves open at low pressure and close automatically.

Applications for AGM batteries include start-stop vehicles, large audio systems, heated seats, and other electronic accessories. If you want to power a car with a lot of technological features or plug-in extras, you should think about a deep-cycle battery or a battery produced with modern technology like AGM. AGM batteries are an excellent premium option for high-end and advanced fuel-efficient vehicles with high power demands, as well as for consumers looking for more reliability and longer battery life. DVD players, GPS, heated seats, and audio systems increase the demands on a battery. It would be a great pick to also power a motored wheelchair.

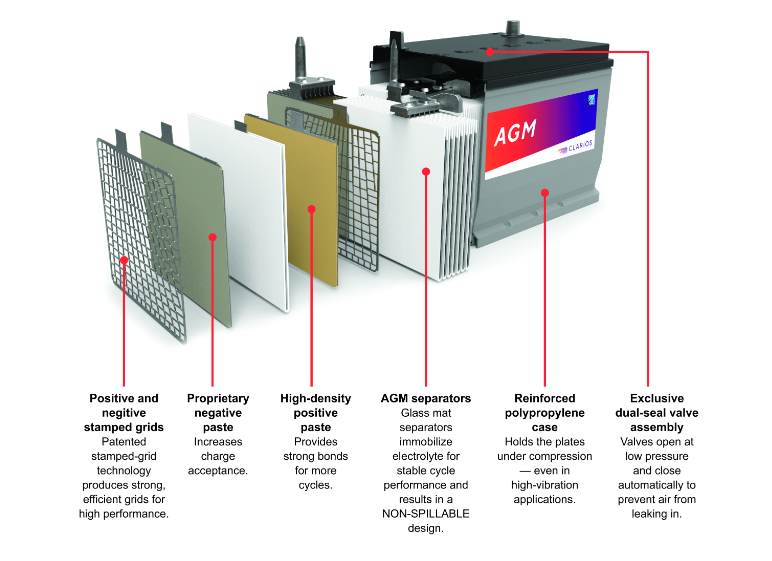


Figure 4: Structure of an AGM Battery.

##### **Advantages of AGM Batteries**

AGM batteries offer several advantages over conventional flooded batteries. One notable benefit is their significantly extended longevity, often lasting up to twice as long. This increased lifespan makes them a reliable choice, especially in demanding situations and extreme climates where durability is crucial. Deep-cycle batteries are designed to handle high electrical loads, making them suitable for applications that require sustained power output. Moreover, they possess the ability to recharge quickly, effectively extending their cycle life and ensuring continuous energy availability. These batteries are also well-suited for start-stop applications, providing reliable power for frequent on-off cycles. Additionally, they exhibit excellent resistance to vibration, making them ideal for use in environments with high mechanical stress or movement. With their non-spillable and low-maintenance design, deep-cycle batteries offer convenience and ease of use. Furthermore, they provide greater mounting adaptability, allowing for flexible installation in various equipment and systems. These combined features make deep-cycle batteries a reliable and versatile energy storage solution for a wide range of applications.

##### **Disadvantages of AGM Batteries**

While AGM batteries have numerous advantages, there are a few considerations to bear in mind. Firstly, they can be sensitive to overcharging, so it is important to use appropriate chargers and regulators to prevent damage. Additionally, these batteries can have a relatively steep production cost compared to other battery types, which may factor into the initial investment. Over time, AGM batteries may experience a gradual reduction in capacity, which means their ability to hold a charge may decrease over their lifespan. It's essential to monitor battery performance and plan for potential capacity loss as part of regular maintenance.

#### Gel Batteries

A gel battery is a valve-regulated lead acid battery that contains a predetermined amount of electrolyte, sulphuric acid, and silica gasses. This chemical reaction produces an immovable gel-like substance, which gives these batteries their name. Because the gel batteries use one-way open valves that allow the internal gasses to recombine into water, so there is no need to check to top up distilled water or monitor the water level [24]. Gel batteries are incredibly durable and adaptable. These batteries emit low emissions and can be utilized in areas with limited ventilation.

Because gel batteries are charged at lower voltages, special care should be taken when selecting chargers. Overvoltage can cause failure and poor performance. GEL cells are sealed, maintenance-free batteries that are sometimes referred to as a setting on charge controllers. This might be perplexing and may result in the improper selection of chargers or inappropriate charging settings. Other charging techniques, such as alternators, require the installation of suitable regulators to maintain control over charging [24] voltage. The charging voltage range of batteries is 14.0 to 14.2 volts, while the float voltage range is 13.1 to 13.3 volts.

##### **Advantages of Gel Batteries.**

Gel batteries are the optimal choice for applications requiring sustained power, as they are best suited for such purposes and can endure 500 to 5000 cycles throughout their lifespan. One notable advantage of these batteries is their maintenance-free and spill-proof nature, alleviating concerns about regular upkeep and potential accidents. Moreover, they exhibit minimal corrosion, making them compatible with sensitive electronic equipment that may be adversely affected by corrosive substances. These batteries are also designed to withstand rugged conditions and are vibration-proof, ensuring reliable performance even in challenging environments. Safety is another significant benefit, as the risk of sulphuric acid burns is minimal. Additionally, deep-cycle batteries offer the lowest monthly cost in terms of cost per month of their lifespan and the lowest cost per cycle, providing cost-effective long-term energy solutions.

##### **Disadvantages of Gel Batteries.**

While Gel batteries offer numerous advantages, there are certain considerations to keep in mind. Firstly, they require a high initial investment due to their advanced technology and long-lasting performance. Another factor to be cautious about is overcharging, as once the water in the battery has been overcharged and evaporated, it cannot be refilled. Special chargers and regulators are necessary to ensure proper charging and prevent any potential damage. Additionally, hot temperatures can pose a challenge, as they may cause the acid within the battery to harden and shrink away from the plates, potentially reducing the battery's overall effectiveness. It is essential to monitor and manage temperature conditions to maintain optimal battery performance.

### **2.3.4The Bluetooth**

Bluetooth is a wireless technology standard for transmitting data between fixed and mobile devices over short distances and for constructing personal area networks (PANs). Transmission power is limited to 2.5 milliwatts in the most commonly used mode, resulting in a fairly narrow range of up to 10 meters (33 feet). It makes use of UHF radio waves in the ISM bands ranging from 2.402 GHz to 2.48 GHz [25]. It is mostly used to exchange files between nearby portable devices and to connect cell phones and music players with wireless headphones.

Bluetooth is controlled by the Bluetooth Special Interest Group (SIG), which has over 35,000 members from the telecommunications, computing, networking, and consumer electronics industries. The IEEE standardized Bluetooth as IEEE 802.15.1, however, the standard is no longer maintained. The Bluetooth SIG monitors specification development, maintains the qualification procedure, and protects trademarks [26]. To market a Bluetooth device, a company must meet Bluetooth SIG criteria. The technology is covered by a network of patents [27], which are licensed to particular eligible devices. Annually, 4.7 billion Bluetooth integrated circuit chips will be shipped by 2021.

Bluetooth has made it possible to use microcontroller CPUs like the Raspberry Pi to work with other components with fewer to no cables. Using the Raspberry Pi's Bluetooth capability, you may also operate other Bluetooth devices, such as smart home gadgets or smartphones. You can also transfer data, create Bluetooth beacons, and broadcast audio.

### **2.3.5The mic sensor**

15 voice instructions can be held in this module. There are 3 groups of those 15 pieces, each with 5 pieces. The voice instructions should first be recorded, group by group. Then, before it could detect the five voice commands contained in that group, we had to import one group using a serial command. We should import the group first if we need to implement instructions in other groups. The speech recognition board on this voice recognition module is small and simple to use. A speaker-dependent voice recognition module, this product. In total, it can support up to 80 voice commands. Maximum 7 voice commands can be active at once. Any sound can be taught to serve as a command. Before allowing the module to recognize any voice commands, users must first train it. Serial Port (full function) and General Input Pins (part of a function) are the two methods for controlling this board. When the corresponding voice command was acknowledged, general output pins on the board may produce a variety of waves.

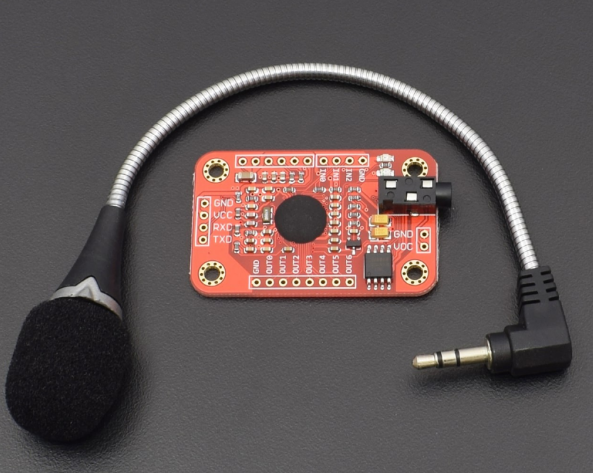


Figure 5: The Microphone sensor.

**Specifications and features**

* 4.5 to 5.5 V.
* 40 mA of current.
* 5V TTL level UART interface for digital interface.
* mm mono-channel microphone connector with an interface for microphone pins (analog interface).

the features include;

* Support up to 80 voice commands at a distance of 1500 meters each.
* A library for Arduino is provided.
* Easy control: GPIO and UART.
* General Pin Output with User Control.
* 7 voice commands can be effectively used at once.

### **2.3.6 RF Transmitter Receiver module**

Radio Frequency is used by the RF module. This frequency band ranges from 30 kHz to 300 GHz. The digital data in this RF system is represented as fluctuations in the amplitude of the carrier wave. Amplitude Shift Keying (ASK) is a method of modulation. It is a hybrid of an RF transmitter and a receiver. The frequency of the transmitter/receiver (Tx/Rx) pair is 433 MHz

The RF transmitter receives serial data and wirelessly delivers it via its RF antenna. The transmission rate ranges between 1 and 10 Kbps. The transmitted data is received by the RF receiver, which operates at the same frequency as the transmitter.

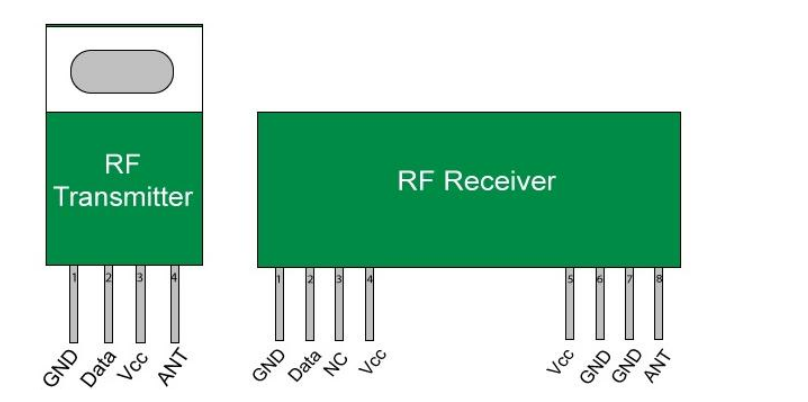


Figure 6: RF Transmitter and Receiver

As indicated in figure 6 above, the Transmitter module has three pins: Vcc, Din, and ground. The Vcc pin accepts input voltages ranging from 3V to 12V. During transmission, the transmitter uses a minimum of 9mA and can reach a maximum of 40mA. The data pin in the center is used to transfer the signal. This signal was modulated with ASK and broadcast over the air at 433MHz.

As illustrated in figure 6 above, the RF receiver module has four pins: Vcc, Dout, Linear out, and Ground. A controlled 5V supply should be used to power the Vcc pin. This module's operational current is less than 5.5mA. To receive the 433Mhz signal from the air, the pins Dout and Linear out are shorted together.

**Features**

* The frequency of reception is 433MHz.
* 105Dbm is the usual frequency of the receiver.
* 3.5mA receiver supply current
* Power usage is minimal.
* 5 volts is the operating voltage of the receiver.
* The frequency range of the transmitter is 433.92MHz.
* 3v-6v transmitter Vin (supply)
* 4v-12v transmitter Pout

# **Chapter Three**

## **3.0 Methodology**

The chapter shows a block diagram, flowchart and the step to step implementation of RF Multi-controlled wheelchair system. The components involved are briefly discussed and the flow chart of the system is also shown.

## **3.1 Block Diagram**

Figure 7: Block Diagram

Power Supply

Regulation

Ultrasonic Sensor

Raspberry PI

RF TX/RX module

Hand Gesture

System

Touch screen

navigation

Voice/Mic system

Motor Drive

Motor

## **3.1.1 Functional Description.**

The functionality of the multi-controlled wheelchair shown in the above block diagram for people with mobility problems is as below;

**Power Supply**

The power supply system starts with two rechargeable 18650 batteries. These batteries are designed to store electrical energy and provide it to the various components of the wheelchair. They are typically a lithium-ion battery. To replenish the energy in the battery, a battery charger is connected to an external power source, such as a wall outlet.

**Power Regulation.**

This is the process of converting AC power to DC for use by the components of the system. The charger converts the AC power from the outlet into DC power suitable for charging the battery. It ensures that the battery remains charged and ready for use. Once the battery is charged, the power is distributed to the different components of the wheelchair.

**Ultrasonic Obstacle detection system**

This involves two ultrasonic sensors meant to detect obstacles at the front and at the back. They help stop the system to avoid obstacle collision.

**Raspberry Pi Module**

In the multi-controlled wheelchair system that utilizes voice commands, touch commands, and gesture commands, a Raspberry Pi can be employed as the prime CPU (Central Processing Unit). Here's a brief explanation of how it would work:

The Raspberry Pi is a small, single-board computer that can serve as the main control unit for the wheelchair system. It offers sufficient processing power, I/O capabilities, and flexibility to handle the various input commands and control the wheelchair's movements.

**Voice Command Integration**

To enable voice commands, the Raspberry Pi can is equipped with a microphone to capture the user's voice input. Speech recognition software or libraries were implemented on the Raspberry Pi to process and interpret voice commands. These commands were mapped to specific actions or wheelchair movements.

**Touch Command Integration**

For touch commands, the Raspberry Pi was connected to a touch-sensitive display, the LCD Raspberry touch screen display. The touch screen displays a user interface that allows the user to interact with the wheelchair system through touch gestures. The written software can interpret these touch inputs and translate them into corresponding actions or movements.

**Gesture Command Integration**

Gesture commands are captured using the accelerometer that detect the tilt motion of the upper limb and send a control signal to the CPU to process the command given. The Raspberry Pi is connected to this sensor to capture and process gesture data. Together with software and the sensor to recognize specific gestures, The Raspberry Pi translates this into control signals for the wheelchair.

**Motor Drive**

Once the input commands are captured, whether voice, touch, or gesture commands, the Raspberry Pi processes the data, recognizes the type of command received, and makes decisions based on the input. It then communicates with motor drive to control the wheelchair's motors. It generates the necessary signals to drive the motors based on the interpreted commands. These signals include direction, and other control parameters like stop. By controlling the motors, the Raspberry Pi enables the wheelchair to move according to the desired commands.

**RF Module**

The RF module serves as a wireless communication interface between the Raspberry Pi and the motor drive. It consists of two transceivers one as for transmitting signal and the other receiving signal, allowing bidirectional communication between the devices. The RF module needs to be wirelessly connected to both the Raspberry Pies and the motor drive. This involves establishing a Radio frequency connection between the devices.

The Raspberry Pi needs to be configured to communicate with the RF module via software. This typically involves configuring the Raspberry Pi's system to transmit and receive gesture signals. Once the radio frequency connection is established, the Raspberry Pi can send control commands to the motor drive through the RF module.

## **3.2 System Flow Diagram.**

Based on the flowchart below a microcontroller program was developed using a Raspberry Pi IDE software.

Figure : Flowchart

Obstacle?

Hold Stop

Forward?

Backward?

Left?

Right?

Recognize Voice

Recognize Touch

Move Rightward

Move Leftward

Move Backward

Move Forward

Recognize Tilt Motion

Stop

Yes

Yes

Yes

Yes

No

No

No

No

Voice Mode

Touch Mode

Gesture Mode

The development approach was based on the Modular Technique and thereafter a merge of all modules will be done using hardware in order for there to be an integrated sequenced flow of the entire system. The software is going to be explained below:

### **3.2.1 Libraries Used**

The following libraries were used to help program this system software

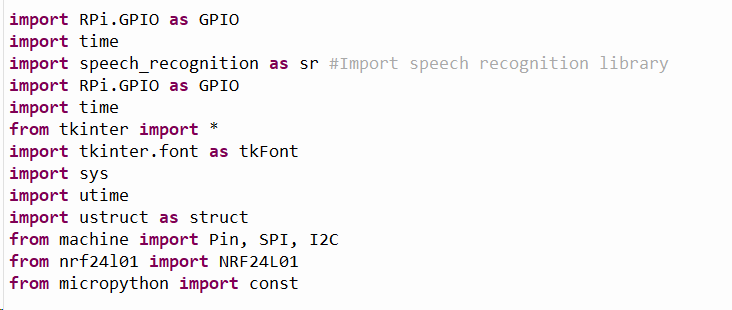


Figure 9: Software Libraries

The code will be availed in the appendix part of this report.

* 1. **RF Multi-controlled wheelchair**

### **3.3.1 Schematic Layout**

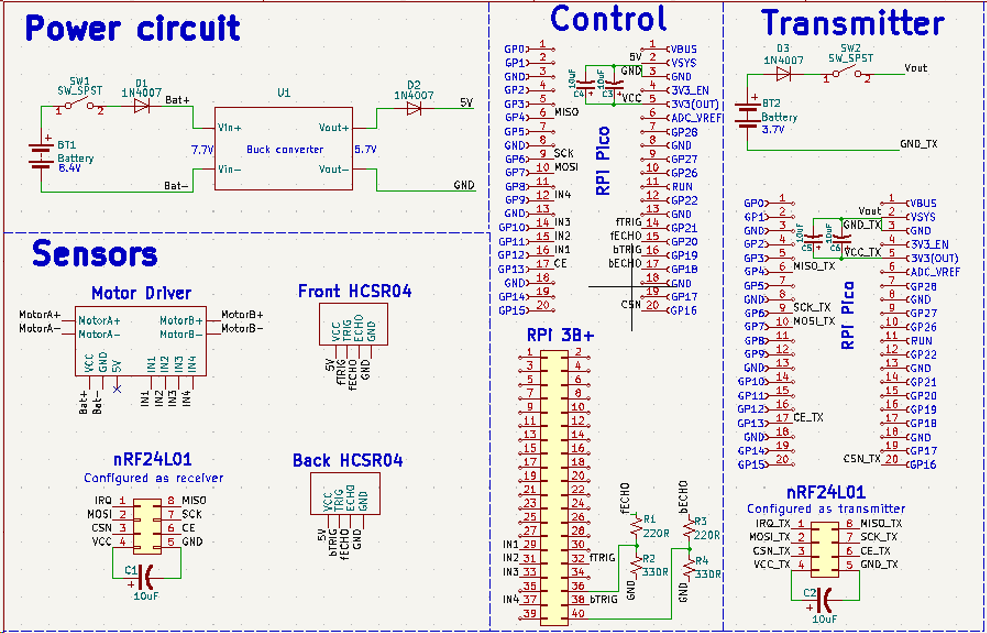


Figure : Schematic diagram

The diagram above is a show of components with their sectional connections to help them work together cohesively as a unit. This schematic was developed using software called KiCAD.

### **3.3.2 Power Circuit**

The power unit combined as follows; The power supply is from a battery pack of two batteries (18650) rated 4.2V each bring out a total of 8.4V. The power flows to a buck converter through a diode and out through to another diode to ensure power to the components is five volts.

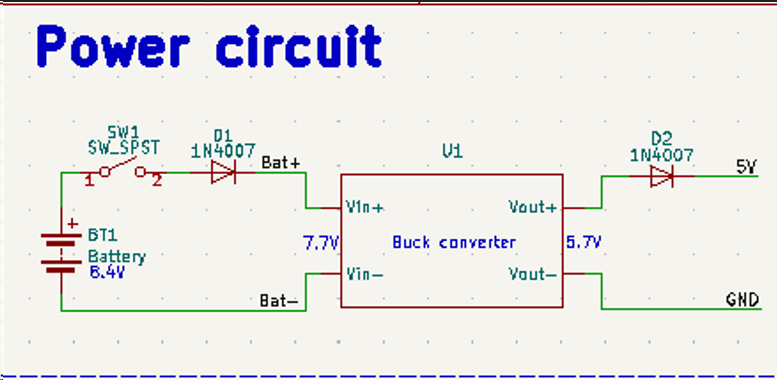


Figure 11: Power Circuit

### **3.3.3 Pin Configuration**

Pin configuration was an essential process to ensure that all components worked correctly since manipulation of the components needed code instructions to target various pins. Pins configured were for elements/components like: the ultrasonic sensors, raspberry pi, picos, nRF24Lo1 transceivers, and the ADXL345.

**Ultrasonic Sensors**

|  |  |  |
| --- | --- | --- |
| Pins | Connections U1 (PICO) | Connections U2 (PICO) |
| VCC | 5V | 5V |
| TRIG | 14 | 16 |
| ECHO | 15 | 17 |
| GND | GND | GND |
| Pins | Connections U1 (PI) | Connections U2 (PI) |
| VCC | 5V | 5V |
| TRIG | 32 | 38 |
| ECHO | 36 | 40 |
| GND | GND | GND |

The ultrasonic sensors are used to detect obstacles and initiate a stop to help avoid collision.

**nRF24l01(Transceiver)**

|  |  |  |
| --- | --- | --- |
| Pins | Connections (PICO) | Definitions |
| IRQ | 1 | Interrupt pin |
| MOSI | 2 | Master Out Slave In |
| CSN | 3 | Chip Select Next |
| VCC | 4 | Voltage in |
| MISO | 8 | Master In Slave Out |
| SCK | 7 | Serial Clock |
| CE | 6 | Chip Enable |
| GND | 5 | Ground |

Since there are two transceivers, they operate as Master for the one on the transmission side and Slave for the one on the Reception side.

**ADXL 345 Accelerometer**

|  |  |  |
| --- | --- | --- |
| Pins | Connections U1(PICO) | Description |
| GND | GND | Ground |
| VCC | Vsys | Voltage in |
| SDA | 0 | Serial Data |
| SCL | 1 | Serial Clock |

This sensor is responsible for capturing gesture data by registering tilt angles to be processed for motion control.

**3.4 Hardware Configuration**

The prototype was designed to put together all the modules of the system together to work cohesively. The diagram below shows the prototype that will be used to demonstrate the envisioned wheelchair system.



Figure 12: Hardware Setup

**Resistors**

The resistors used in the circuit are meant to help the raspberry pi 3 achieve voltage level shifting for the ultrasonic sensor usage since unlike a microcontroller the Pi 3 is a CPU and doesn’t have provision for automatic voltage level shifting. The resistors chosen were decided upon after performing a voltage divider calculation to get values; 220Ω, 220Ω, 330Ω, and 330Ω. They helped achieve a Vout of 3.3V



Figure 13: Resistor

**Capacitors**

These were used to help achieve voltage stabilization, with the aim of also reducing noise during RF transmission and reception. The values of capacitors used are 10 µF for the Picos two were put in parallel to make them sum up to a value of 20µF.

#### Jumper wires

#### These were used to connect sensors, switches, Motor drive, MCUs, the Buck Converter, PI CPU and power supply together to ensure cohesive working of the system.

#### jumper wire

Figure 14: Jumper wires

**Motor drive**

The motor drive used was an L298N and was connected to two motors (Left and Right) and the IN1-IN4 pins were connected to Pi and Pico for control instructions. It was also directly connected to the batteries to ensure motors we sufficiently powered to facilitate movement of the prototype.

# 

Figure 15: L298N Driver

**Raspberry Pico**

The prototype used two pico MCUs to help realize the gesture navigation option. One pico was on the transmitter side connected to a 3.7V battery, nRF24L01 as the transmitter and the ADXL345 accelerometer responsible for sensing direction gesture. The other Pico was on the receiver side. Connected to the motor drive to send commands issued to the motors.



Figure 16: Raspberry Pico

**Raspberry Pi 3**

This CPU is used to run a 3.5 inch touch display for the touch navigation and also establish a microphone communication for the Speech/Voice navigation. Code for both navigation was written in this Pi 3 CPU.



Figure 17: Pi 3 and Display

# **Chapter Four**

## **4.0 Results**

### **4.0.1 Navigation**

The proposed system prototype was designed to implement three types of navigation control. They are as documented below;

**Touch navigation**

This navigation option needed the raspberry Pi 3 to be connected to a 3.5 LCD screen in order to display a user interface that allowed to key in instructions. Programming was done to achieve this using Python language. And the result was as the image below:

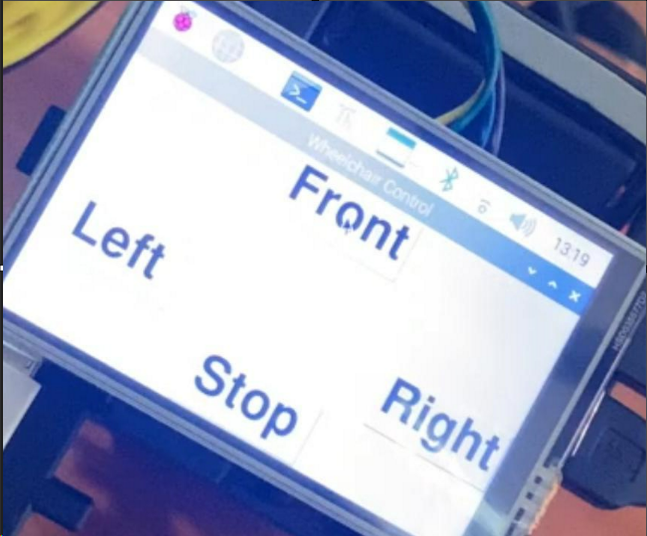


Figure 18: Touch Graphical User Interface

**Voice Navigation.**

The Raspberry Pi 3 provided a USB port to connect a detachable microphone necessary for entry of commands. The program for this navigation mode was done and the result was a shown below:



Figure 19:Speech System Response

**Gesture navigation**

This mode used two MCUs to help transmit and receive tilt gesture information from the ADXL345 accelerometer. The mode of control was on a detached transmitter circuit as shoen below:

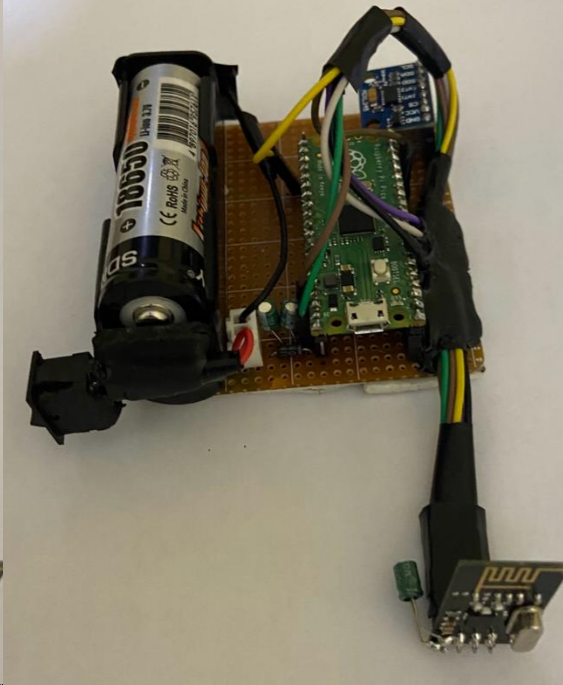


Figure 20: Gesture Input Controls

### **4.0.2 Obstacle Detection.**

The system was fitted with two ultrasonic sensors, one at the front and the other at the back to help detect obstacles. Once obstacles were registered to be 10 cm in front of the prototype, the emergency stop feature was applied to avoid collision.

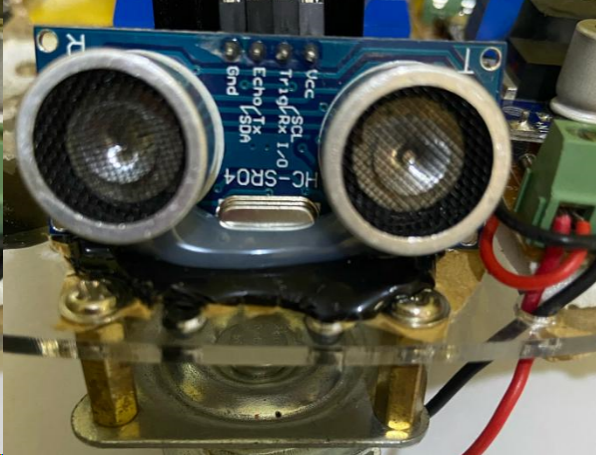


Figure : Ultrasonic Installation

### **4.0.3 Financial Viability.**

Compared to a joystick controlled wheelchair, if this system was to be incorporated to a wheel chair it would be cheaper than purchasing a joystick wheelchair. The total cost including batteries would be 47,000/- whereas the cost of a joystick wheelchair is from 120,000/-. Therefore it will make this more affordable than the latter.

# **Chapter 5**

## **5.0 Challenges**

### **5.0.1 Radio Frequency Transmission.**

During implementation, the RF module that had earlier been obtained had issues with noise interference and this was making it difficult to obtain the desired results. The first option was the 433Mhz RF Transmitter/Receiver. The transmitter side had the crystal containing a local oscillator circuit that helps mitigate noise related issues. The receiver used and inductor and resistor for a local oscillator bringing about a lot of noise in the reception side. The solution was to purchase some components that would shield communication from noise errors. This was by buying two transceivers (nRF24L01) for smoother RF communications. The two look as shown below:

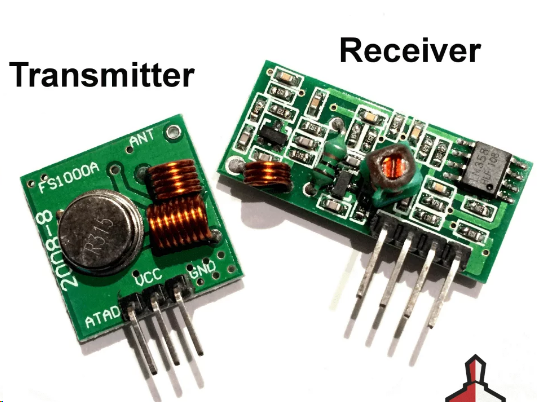


Figure :RF Transmitter/Receiver

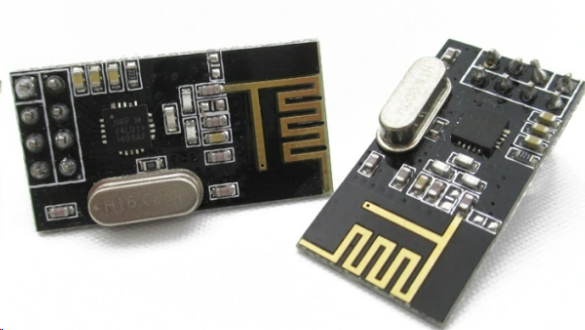


Figure : nRF24Lo1 Transceivers

### **5.0.2 Funding.**

Due to shortage of funds the smooth operation envisioned was affected since it was also difficult to access school funding. Additional funding would help improve the system’s seamlessness.

## **5.1 Conclusion**

In this project, a Radio Frequency multi-controlled wheelchair was developed to help achieve more autonomy for people with motor disabilities. Three different options of navigation control were implemented in the project. They were; voice, speech, and gesture control options. In addition to this the wheelchair incorporated an obstacle detection feature to help prevent collisions.

The system developed seems to also be a cheaper solution motorized wheelchairs. With this system being used it would make it available to more people than before hence bring a solution to a problem.

## **5.3 Recommendations**

To improve the overall efficiency and functionality of the system, the following would be good to consider:

1. Incorporating a multiplexing and demultiplexing system would be good to ensure the whole system worked seamlessly when it comes to execution of commands to the motor drive.

# **Chapter Six**

## **6.0 Project Time Plan**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ACTIVITIES** | **MAY** | **JUN** | **JUL** | **AUG** | **SEP** | **OCT** | **NOV** | **DEC** |
| **Documentation** |  |  |  |  |  |  |  |  |
| **Proposal Writing** |  |  |  |  |  |  |  |  |
| **Proposal Presentation** |  |  |  |  |  |  |  |  |
| **Design and coding** |  |  |  |  |  |  |  |  |
| **Hardware configuration, testing and adjustment** |  |  |  |  |  |  |  |  |
| **Final Report writing** |  |  |  |  |  |  |  |  |
| **Final Presentation** |  |  |  |  |  |  |  |  |

# **Chapter Seven**

## **7.0Proposed Budget**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item | Description | Quantity | Rate | Amount |
| 1 | Raspberry PI 4 | 1 | 7000 | 7000 |
| 2 | Raspberry PI Display | 1 | 5300 | 5300 |
| 3 | Accelerometer& Ultrasonic sensor | 1 | 550 | 550 |
| 4 | Microphone | 1 | 450 | 450 |
| 5 | RF transmitter/receiver | 1 | 1000 | 1000 |
| 6 | Motor | 2 | 750 | 1500 |
| 7 | BT transmitters | 2 | 550 | 1100 |
| 8 | Battery | 1 | 900 | 900 |
| 9 | Miscellaneous |  |  | 700 |
| TOTAL | | | | 18500 |

# **Chapter Eight**

# **References**

apps, B. t.-r. (2001). Bluetooth: technology for short-range wireless apps. *IEEE Internet computing, 5*, 96-103.

Bharani, M. a. (2014). An Embedded System Based Monitoring System For Industries By Interfacing Sensors With ATmega Microcontroller. *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), 3*, 1472-1474.

Boninger, M. (2011). *The power wheelchair: What the .* University of Washington Model Systems Knowledge Translation Centre (MSKTC).

Brock, J. D. (2013). Changing the world with a Raspberry Pi. *Journal of Computing Sciences in Colleges, 29*, 151-153.

Carment, L. a.-L. (2018). Manual dexterity and aging: a pilot study disentangling sensorimotor from cognitive decline. *Frontiers in neurology*, 910.

Caruso, M. a. (2015). Design and experimental characterization of a low-cost, real-time, wireless AC monitoring system based on ATmega 328P-PU microcontroller. *International Annual Conference (AEIT)*, pp. 1-6.

Chauhan, R. a. (2016). Study of implementation of voice controlled wheelchair. In *International Conference on Advanced Computing and Communication Systems (ICACCS)* (Vol. 1, pp. 1-4). IEEE.

Clark, J. a. (1997). Voice controlled wheelchair. *Archives of physical medicine and rehabilitation, 58*, 169-177.

Demers, L. a. (2009). A conceptual framework of outcomes for caregivers of assistive technology users. *American journal of physical medicine \& rehabilitation, 88*(8), 645-655.

Halfacree, G. a. (2012). *Raspberry Pi user guide.* John Wiley \& Sons.

Hassan, S. S. (2018). Security threats in Bluetooth technology. *Computers \& Security*, 308-322.

Huang, S.-S. a.-H.-Y. (2019). Combining Weighted Contour Templates with HOGs for Human Detection Using Biased Boosting. *Sensor, 19*, 1458.

karki, P., Shahi, P. V., Sapkota, K. P., Bhandari, R., Adhikari, N., & Shrestha, B. (2023). Depressive symptoms and associated factors among persons with physical disabilities in disability care homes of Kathmandu district, Nepal: A mixed method study. (T. P. Rakesh Singh, Ed.) *Global Public Health*, 1.

Kumar, R. a. (2015). Arduino ATMEGA-328 microcontroller. *Int. J. Innov. Res. Electr. Electron. Instrum. Control Eng, 3*, 27-29.

Kumawat, R. a. (2021). Implementation of Self-Controlled Wheelchair based on joystick, gesture, and voice recognition. In *2021 IEEE International Symposium on Smart Electronic Systems (iSES)* (pp. 240-243). Jaipur , India. doi:{10.1109/iSES52644.2021.00062}

Kutbi, M. a. (2017). Hands-free wheelchair navigation based on egocentric computer vision: A usability study. *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada*, (pp. 24-28). Vancouver.

Li, J. (2015). Research on camera-based human body tracking using improved cam-shift algorithm. *International journal on smart sensing and intelligent systems, 8*, 1104-1122.

Mekonnen, Y. a. (2018). Life cycle prediction of Sealed Lead Acid batteries based on a Weibull model. *Journal of Energy Storage, 18*, 467-475.

Oh, J. H. (2019). Motion control of joystick interfaced electric wheelchair for improvement of safety and riding comfort. *Mechatronics*, 104-144. doi:https://doi.org/10.1016/j.mechatronics.2019.03.005

Phillips, R. a. (2012). Development of brushed and brushless DC motors for use in the ExoMars drilling and sampling mechanism. *Proceedings of the 41st Aerospace Mechanisms Symposium*, (pp. 16-18).

Purwanto, D. R. (2009). *Electric wheelchair control with gaze direction and eye blinking.* (14 ed.). Artificial Life Robot . doi: 10.1007/s10015-009-0694-x

Radcliffe, P. a. (2015). Sensorless speed measurement for brushed DC motors. *IET Power Electronics, 8*, 2223-2228.

Richardson, M. a. (2012). *Getting started with raspberry PI.* O'Reilly Media, Inc.

Riman, C. (2018). *Implementation of a low cost hands free wheelchair controller.* IIC.

Rossen, C. B. (2012). *Everyday life for users of electric wheelchairs - a qualitative interview study.* (7 ed.). Disability Rehabilit. Assistive Technol. doi: 10.3109/17483107.2012.665976

Saharia, T. J. (2017). *Joystick Controlled wheelchair* (Vol. 4). Int. Res. J. Eng. Technol.

Shorey, R. a. (2000). The Bluetooth technology: merits and limitations. In *IEEE International Conference on Personal Wireless Communications. Conference Proceedings (Cat. No. 00TH8488)* (pp. 80-84). IEEE.

Song, J. a. (1999). Review of gel-type polymer electrolytes for lithium-ion batteries. *Journal of power sources, 77*, 183-197.

Tejonidhi, M., & Vinod, A. (2017). Oculus Supervised Wheelchair Control for People with Locomotor Disabilities. *International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, 254-259.

Toliyat, H. A. (2018). *Handbook of electric motors* (Vol. 120). CRC press.

Utaminingrum, F. a. (2017). Determining direction of moving object using object tracking for smart weelchair controller. *5th International Symposium on Computational and Business Intelligence (ISCBI)* (pp. 6-9). IEEE.

Yeh, P. a. (2009). *Optics of liquid crystal displays.* John Wiley \& Sons.

# **Appendix**

## **Touch Navigation code**

import RPi.GPIO as GPIO

import time

from tkinter import \*

import tkinter.font as tkFont

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BOARD)

#Control pins for the motor driver

in1 = 29

in2 = 31

in3 = 33

in4 =37

GPIO.setup(in1, GPIO.OUT)

GPIO.setup(in2, GPIO.OUT)

GPIO.setup(in3, GPIO.OUT)

GPIO.setup(in4, GPIO.OUT)

win = Tk()

myFont = tkFont.Font(family = 'Helvetica', size = 36, weight = 'bold')

def forward():

GPIO.output(in1, GPIO.LOW)

GPIO.output(in2, GPIO.HIGH)

GPIO.output(in3, GPIO.LOW)

GPIO.output(in4, GPIO.HIGH)

def reverse():

GPIO.output(in1, GPIO.HIGH)

GPIO.output(in2, GPIO.LOW)

GPIO.output(in3, GPIO.HIGH)

GPIO.output(in4, GPIO.LOW)

def right():

GPIO.output(in1, GPIO.LOW)

GPIO.output(in2, GPIO.HIGH)

GPIO.output(in3, GPIO.HIGH)

GPIO.output(in4, GPIO.LOW)

time.sleep(1)

def left():

GPIO.output(in1, GPIO.HIGH)

GPIO.output(in2, GPIO.LOW)

GPIO.output(in3, GPIO.LOW)

GPIO.output(in4, GPIO.HIGH)

time.sleep(1)

def stop():

GPIO.output(in1, GPIO.LOW)

GPIO.output(in2, GPIO.LOW)

GPIO.output(in3, GPIO.LOW)

GPIO.output(in4, GPIO.LOW)

win.title('Wheelchair Control')

win.geometry('480x320')

frontButton = Button(win, text = 'Front', font = myFont, command = forward, height = 1, width = 4)

frontButton.pack(side = TOP)

leftButton = Button(win, text = 'Left', font = myFont, command = left, height = 1, width = 4)

leftButton.pack(side = LEFT)

rightButton = Button(win, text = 'Right', font = myFont, command = right, height = 1, width = 4)

rightButton.pack(side = RIGHT)

stopButton = Button(win, text = 'Stop', font = myFont, command = stop, height = 1, width = 4)

stopButton.pack(side = BOTTOM)

win.mainloop()

## **Voice Navigation Code**

"""

This program will be used to control the wheelchair by voice commands.

We use the SpeechRecognition library as well as Pyaudio for this.

We may experience a lag due to the RAM or the WiFi network.

The words front, back, left, right, stop, and quit will be use for control.

"""

import RPi.GPIO as GPIO

import time

import speech\_recognition as sr #Import speech recognition library

r = sr.Recognizer() #Create an instance of the SpeechRecognition library

mic = sr.MicroPhone() #Create an instance of the pyaudio library

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BOARD)

#Control pins for the motor driver

in1 = 29

in2 = 31

in3 = 33

in4 =37

#Set the motor control pins as output pins

GPIO.setup(in1, GPIO.OUT)

GPIO.setup(in2, GPIO.OUT)

GPIO.setup(in3, GPIO.OUT)

GPIO.setup(in4, GPIO.OUT)

#Function to move the wheelchair forward

def forward():

GPIO.output(in1, GPIO.LOW)

GPIO.output(in2, GPIO.HIGH)

GPIO.output(in3, GPIO.LOW)

GPIO.output(in4, GPIO.HIGH)

#Function to move the wheelchair backwards

def reverse():

GPIO.output(in1, GPIO.HIGH)

GPIO.output(in2, GPIO.LOW)

GPIO.output(in3, GPIO.HIGH)

GPIO.output(in4, GPIO.LOW)

#Function to move the wheelchair to the right

def right():

GPIO.output(in1, GPIO.LOW)

GPIO.output(in2, GPIO.HIGH)

GPIO.output(in3, GPIO.HIGH)

GPIO.output(in4, GPIO.LOW)

time.sleep(1)

stop()

#Function to move the wheelchair to the left

def left():

GPIO.output(in1, GPIO.HIGH)

GPIO.output(in2, GPIO.LOW)

GPIO.output(in3, GPIO.LOW)

GPIO.output(in4, GPIO.HIGH)

time.sleep(1)

stop()

#Function to stop the wheelchair

def stop():

GPIO.output(in1, GPIO.LOW)

GPIO.output(in2, GPIO.LOW)

GPIO.output(in3, GPIO.LOW)

GPIO.output(in4, GPIO.LOW)

#The main program to run forever

while True:

with mic as source:

print("Listening to commands...")

audio = r.listen(source)

command = r.recognize\_google(audio)

print("You said: ", command)

#Control the wheelchair based on the audio command

if command == "front":

print("Moving forward...\n")

forward()

if command == "back":

print("Moving backwards...\n")

reverse()

if command == "left":

print("Turning left...\n")

left()

if command == "right":

print("Turning right...\n")

right()

if command == "stop":

print("Stopping...\n")

stop()

if command == "quit":

print("Bye bye! Have a great time.\n")

break

## **Gesture Navigation Code**

### **Transmitter Code**

import sys

import utime

import ustruct as struct

from machine import Pin, SPI, I2C

from nrf24l01 import NRF24L01

from micropython import const

# Slave pause between receiving data and checking for further packets.

\_RX\_POLL\_DELAY = const(15)

# Slave pauses an additional \_SLAVE\_SEND\_DELAY ms after receiving data and before

# transmitting to allow the (remote) master time to get into receive mode. The

# master may be a slow device. Value tested with Pyboard, ESP32 and ESP8266.

\_SLAVE\_SEND\_DELAY = const(10)

if sys.platform == "pyboard":

cfg = {"spi": 2, "miso": "Y7", "mosi": "Y8", "sck": "Y6", "csn": "Y5", "ce": "Y4"}

elif sys.platform == "esp8266": # Hardware SPI

cfg = {"spi": 1, "miso": 12, "mosi": 13, "sck": 14, "csn": 4, "ce": 5}

elif sys.platform == "esp32": # Software SPI

cfg = {"spi": -1, "miso": 32, "mosi": 33, "sck": 25, "csn": 26, "ce": 27}

elif sys.platform == "rp2": # pico

cfg = {"spi": 0, "miso": 4, "mosi": 7, "sck": 6, "csn": 17, "ce": 13}

else:

raise ValueError("Unsupported platform {}".format(sys.platform))

# Addresses are in little-endian format. They correspond to big-endian

# 0xf0f0f0f0e1, 0xf0f0f0f0d2

pipes = (b"\xe1\xf0\xf0\xf0\xf0", b"\xd2\xf0\xf0\xf0\xf0")

#The ADXL345 accelerometer module

# Constants

ADXL345\_ADDRESS = 0x53 # address for accelerometer

ADXL345\_POWER\_CTL = 0x2D # address for power control

ADXL345\_DATA\_FORMAT = 0x31 # configure data format

ADXL345\_DATAX0 = 0x32 # where the x-axis data starts

# Initialize I2C

i2c = I2C(0, sda=Pin(0), scl=Pin(1), freq=400000)

def getSPI():

if sys.platform == "rp2":

return SPI(cfg['spi'],baudrate=400000,sck=Pin(cfg["sck"]), mosi=Pin(cfg["mosi"]), miso=Pin(cfg["miso"]))

else:

if cfg["spi"] == -1:

return SPI(-1, sck=Pin(cfg["sck"]), mosi=Pin(cfg["mosi"]), miso=Pin(cfg["miso"]))

return SPI(cfg["spi"])

# Initialize ADXL345

def init\_adxl345():

i2c.writeto\_mem(ADXL345\_ADDRESS, ADXL345\_POWER\_CTL, bytearray([0x08])) # Set bit 3 to 1 to enable measurement mode

i2c.writeto\_mem(ADXL345\_ADDRESS, ADXL345\_DATA\_FORMAT, bytearray([0x0B])) # Set data format to full resolution, +/- 16g

# Read acceleration data

def read\_accel\_data():

data = i2c.readfrom\_mem(ADXL345\_ADDRESS, ADXL345\_DATAX0, 6)

x, y, z = struct.unpack('<3h', data)

return x, y, z

# Main loop

init\_adxl345()

def master():

csn = Pin(cfg["csn"], mode=Pin.OUT, value=1)

ce = Pin(cfg["ce"], mode=Pin.OUT, value=0)

spi = getSPI()

nrf = NRF24L01(spi, csn, ce, payload\_size=8)

nrf.open\_tx\_pipe(pipes[0])

nrf.open\_rx\_pipe(1, pipes[1])

nrf.start\_listening()

while True:

# Stop listening, time to send a message

nrf.stop\_listening()

x, y, z = read\_accel\_data()

print('--------------------')

print("X: ", x)

print("Y: ", y)

print("Z: ", z)

# Determine the position of the accelerometer

position = 0

if x > 200:

#Send 1

position = 1

elif x < -200:

#Send 2

position = 2

elif y > 200:

#Send 3

position = 3

elif y < -200:

#Send 4

position = 4

elif z < -200:

#Send 5

position = 5

else:

#Send 6

position = 6

print("sending:", position)

try:

nrf.send(struct.pack("i", position)) # sending the message

except OSError:

pass

# Listen if the other Pico answers us

nrf.start\_listening()

# Wait for 250ms max

start\_time = utime.ticks\_ms()

timeout = False

while not nrf.any() and not timeout:

if utime.ticks\_diff(utime.ticks\_ms(), start\_time) > 250:

timeout = True

if timeout: # no response received

print("failure, no response")

else: # a response has been received

(response,) = struct.unpack("i", nrf.recv())

print ("response recue:", response)

# Wait a second before sending the next message

utime.sleep\_ms(1000)

#Call the master function

master()

### **Receiver Code**

import sys

import utime

import ustruct as struct

from machine import Pin, SPI, I2C

from nrf24l01 import NRF24L01

from micropython import const

#Define the pins for driving the wheelchair via the motor driver

in1 = Pin(12, Pin.OUT)

in2 = Pin(11, Pin.OUT)

in3 = Pin(10, Pin.OUT)

in4 = Pin(9, Pin.OUT)

# Slave pause between receiving data and checking for further packets.(Synchronization)

\_RX\_POLL\_DELAY = const(15)

# Slave pauses an additional \_SLAVE\_SEND\_DELAY ms after receiving data and before

# transmitting to allow the (remote) master time to get into receive mode. The

# master may be a slow device. Value tested with Pyboard, ESP32 and ESP8266.

\_SLAVE\_SEND\_DELAY = const(10)

if sys.platform == "pyboard":

cfg = {"spi": 2, "miso": "Y7", "mosi": "Y8", "sck": "Y6", "csn": "Y5", "ce": "Y4"}

elif sys.platform == "esp8266": # Hardware SPI

cfg = {"spi": 1, "miso": 12, "mosi": 13, "sck": 14, "csn": 4, "ce": 5}

elif sys.platform == "esp32": # Software SPI

cfg = {"spi": -1, "miso": 32, "mosi": 33, "sck": 25, "csn": 26, "ce": 27}

elif sys.platform == "rp2": # pico

cfg = {"spi": 0, "miso": 4, "mosi": 7, "sck": 6, "csn": 17, "ce": 13}

else:

raise ValueError("Unsupported platform {}".format(sys.platform))

# Addresses are in little-endian format. They correspond to big-endian(numbering format)

# 0xf0f0f0f0e1, 0xf0f0f0f0d2

pipes = (b"\xe1\xf0\xf0\xf0\xf0", b"\xd2\xf0\xf0\xf0\xf0")

#Define the functions for moving the wheelchair

#Move forward

def forward():

in1.value(0)

in2.value(1)

in3.value(0)

in4.value(1)

def reverse():

in1.value(1)

in2.value(0)

in3.value(1)

in4.value(0)

def stop():

in1.value(0)

in2.value(0)

in3.value(0)

in4.value(0)

def right():

in1.value(0)

in2.value(1)

in3.value(1)

in4.value(0)

utime.sleep(1)

stop()

def left():

in1.value(1)

in2.value(0)

in3.value(0)

in4.value(1)

utime.sleep(1)

stop()

def getSPI():

if sys.platform == "rp2":

return SPI(cfg['spi'],baudrate=400000,sck=Pin(cfg["sck"]), mosi=Pin(cfg["mosi"]), miso=Pin(cfg["miso"]))

else:

if cfg["spi"] == -1:

return SPI(-1, sck=Pin(cfg["sck"]), mosi=Pin(cfg["mosi"]), miso=Pin(cfg["miso"]))

return SPI(cfg["spi"])

def slave():

csn = Pin(cfg["csn"], mode=Pin.OUT, value=1)

ce = Pin(cfg["ce"], mode=Pin.OUT, value=0)

spi = getSPI()

nrf = NRF24L01(spi, csn, ce, payload\_size=8)

nrf.open\_tx\_pipe(pipes[1])

nrf.open\_rx\_pipe(1, pipes[0])

nrf.start\_listening()

print("NRF24L01 slave mode, waiting for packets... (ctrl-C to stop)")

command = ""

while True:

if nrf.any(): # we received something

print("Listening...")

while nrf.any():

buf = nrf.recv()

command = struct.unpack("i", buf)

print("Message received:", command[0])

utime.sleep\_ms(\_RX\_POLL\_DELAY) # delay before next listening

#Control the wheelchair based on the RF accelerometer command

if command[0] == 1:

print("Moving forward...\n")

forward()

if command[0] == 2:

print("Moving backwards...\n")

reverse()

if command[0] == 3:

print("Turning left...\n")

left()

if command[0] == 4:

print("Turning right...\n")

right()

if command[0] == 5:

print("Stopping...\n")

stop()

if command[0] == 6:

print("Holding state...\n")

response = command[0]%2 # preparing the response

utime.sleep\_ms(\_SLAVE\_SEND\_DELAY) # Give the other Pico a brief time to listen

nrf.stop\_listening()

try:

nrf.send(struct.pack("i", response))

except OSError:

pass

print("reply sent:", response)

nrf.start\_listening()

command = ""

slave()