CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

The integration of gesture recognition technology into assistive devices has marked a significant advancement in the field of human-computer interaction, particularly for individuals with mobility impairments. Traditional input methods, such as joystick controls, may present challenges for users with limited hand dexterity or motor control. Hence, there arises a pressing need to explore alternative interfaces that offer greater accessibility, ease of use, and intuitive interaction. The Hand Gesture-Controlled Wheelchair project is motivated by this need, aiming to harness the power of gesture recognition technology to revolutionize wheelchair navigation and empower individuals with severe physical disabilities.

1.1.2 Technological Advancements in Gesture Recognition:

The Hand Gesture-Controlled Wheelchair project capitalizes on significant advancements in gesture recognition technology, making it possible to create a more intuitive and user-friendly interface for assistive devices. Previously, gesture recognition systems were complex and costly, often requiring specialized hardware and sophisticated algorithms. However, recent innovations have introduced more affordable and practical solutions, making this technology accessible for a wider range of applications, including mobility aids for individuals with physical disabilities.

1.Inertial Measurement Units (IMUs): One of the critical advancements that has made gesture recognition more feasible is the widespread availability of low-cost inertial measurement units (IMUs), such as the MPU-6050 accelerometer and gyroscope sensor. These sensors offer highresolution motion tracking capabilities, enabling precise detection of hand gestures and movements. The MPU-6050 sensor combines a three-axis accelerometer and a three-axis gyroscope in a single chip, providing comprehensive motion data. This integration allows for accurate tracking of hand gestures in three-dimensional space, making it ideal for controlling a wheelchair. The MPU-6050 operates by measuring the acceleration and rotational rate along three axes (X, Y, and Z). The accelerometer detects linear movements, while the gyroscope measures rotational movements. This dual capability provides a detailed representation of the hand's motion, which is crucial for recognizing complex gestures.

2.Microcontroller Integration: Microcontrollers such as the Arduino Uno and Nano have played a significant role in making gesture recognition systems more accessible. These microcontrollers provide a powerful yet affordable platform for integrating various sensors and controlling devices. They offer sufficient processing power to handle the real-time data from the sensors and execute the necessary control commands for the wheelchair. The Arduino Nano, used as the transmitter in this project, processes the sensor data from the MPU-6050 and sends commands wirelessly to the Arduino Uno, which acts as the receiver. The Arduino Uno then controls the wheelchair's movements based on these commands. This setup allows for a seamless and efficient communication system, ensuring that the wheelchair responds accurately and promptly to the user's gestures.

3. Impact of Gesture-Controlled Wheelchairs on User Experience: The introduction of gesture controlled wheelchairs represents a significant improvement in the user experience for individuals with mobility impairments. These wheelchairs allow users to interact with their environment using natural hand gestures, which is far more intuitive and less physically demanding than traditional control methods such as joysticks or push-buttons.

Benefits

Natural Interaction: Gesture-controlled interfaces enable users to control the wheelchair using familiar gestures such as waving, pointing, or swiping. This natural form of interaction reduces the learning curve and makes the system more accessible to a broader range of users. For instance, a simple hand wave to the left can command the wheelchair to turn left, while a forward motion can make it move forward.

Reduced Physical Strain: Traditional wheelchair controls, like joysticks, require precise hand movements and fine motor skills, which may not be feasible for all users. Gesture-based controls reduce the need for such precision, allowing users with limited dexterity to operate the wheelchair more easily. This can be particularly beneficial for individuals with conditions that affect motor skills, such as arthritis or muscular dystrophy.

Enhanced Independence: By offering a more accessible control interface, gesture-controlled wheelchairs empower users to navigate their environment with greater confidence and independence. This can significantly enhance their quality of life and provide them with a greater sense of autonomy. Users can perform daily activities and move around more freely without relying heavily on caregivers.

Customization and Adaptability: Gesture-controlled wheelchairs can be customized to suit the individual needs and preferences of users. The system can be programmed to recognize specific gestures defined by the user, allowing for personalized control schemes. Additionally, the sensitivity of the sensors can be adjusted to accommodate varying levels of mobility or motor control, ensuring that the interface is responsive to the user's capabilities.

1.1.3 Overview of Gesture Recognition Technology

Gesture recognition technology involves various methods and techniques for interpreting human gestures and movements. These technologies enable natural and intuitive interaction with computers, devices, and machines. In the context of the Hand Gesture-Controlled Wheelchair project, gesture recognition technology plays a central role in enabling users to communicate their intentions and preferences through simple hand gestures.

Key Components of Gesture Recognition

1.Sensors: The MPU-6050 accelerometer and gyroscope sensor is the primary sensor used for gesture recognition in this project. It captures motion data, including linear acceleration and angular velocity, which is essential for detecting hand movements and gestures accurately.

2.Microcontroller: The Arduino Nano serves as the transmitter, processing the sensor data and sending commands to the wheelchair's control system. The Arduino Uno acts as the receiver, coordinating the execution of these commands and controlling the wheelchair's movements.

3.Data Transmission: A 433 MHz RF module facilitates wireless communication between the transmitter and receiver. This module ensures reliable and real-time data exchange, enabling seamless control of the wheelchair based on the user's gestures.

System Architecture

The system architecture of the Hand Gesture-Controlled Wheelchair consists of interconnected modules responsible for gesture recognition, motor control, communication, and safety mechanisms. The integration of these modules creates a cohesive system that enables intuitive and responsive wheelchair control.

1.Gesture Recognition Module: The gesture recognition module is responsible for capturing and interpreting hand movements. It comprises the MPU-6050 sensor, which provides high resolution motion tracking. The sensor data is processed by the Arduino Nano, which interprets the gestures and sends corresponding commands to the receiver. The MPU-6050 sensor, mounted on the user's hand or wrist, continuously captures motion data. This data includes the acceleration and rotational velocity along three axes. The Arduino Nano processes this data to recognize specific gestures based on predefined patterns. For example, an upward acceleration followed by a forward tilt might be interpreted as a command to move the wheelchair forward.

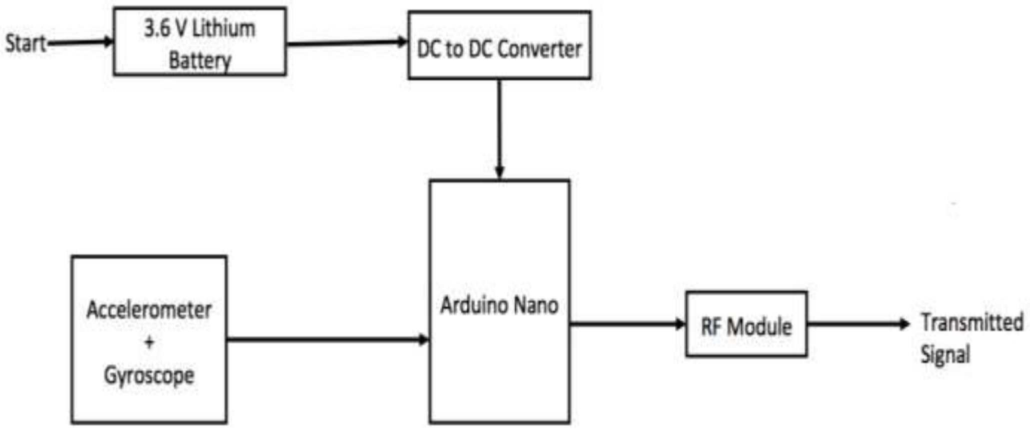

Figure 1. Block Diagram of Remote-Control Unit

2.Motor Control Module: The motor control module translates the gesture commands into wheelchair movements. It uses the L298N dual H-bridge motor driver to interface with the wheelchair's propulsion system, which consists of DC geared motors. The motor driver regulates the speed, direction, and trajectory of the wheelchair based on the received commands, ensuring smooth and responsive control. The Arduino Uno receives the gesture commands from the Arduino Nano and translates them into motor control signals.

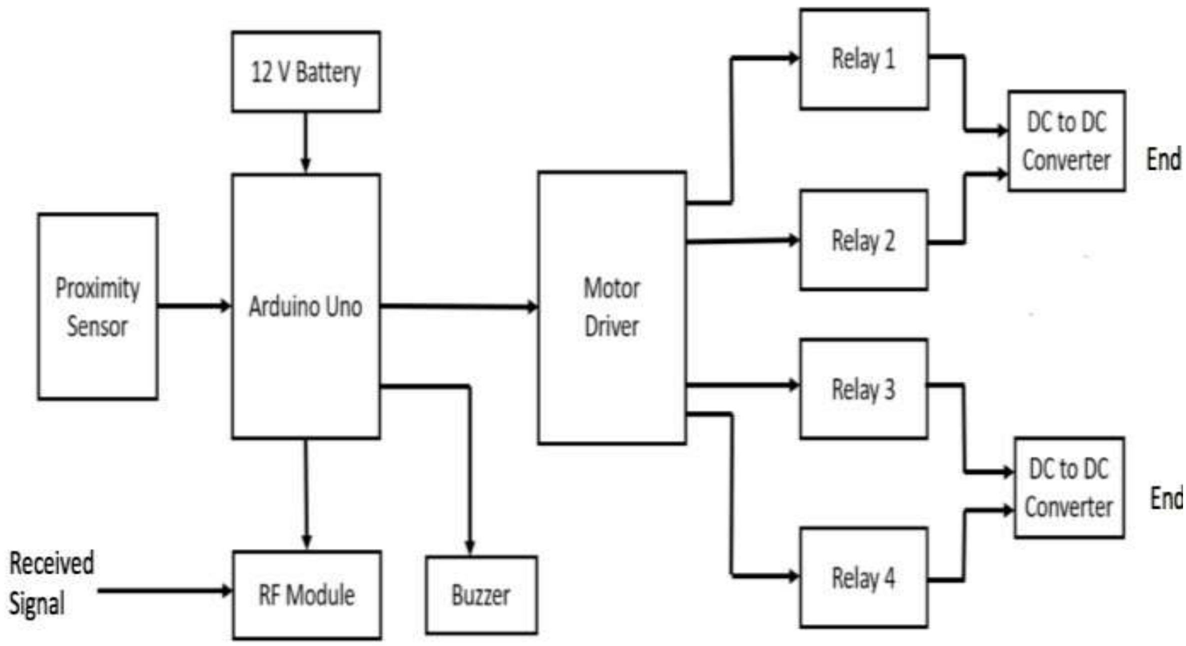

Figure 2. Block Diagram of Wheelchair System

3.Communication Module: The communication module facilitates wireless communication between the transmitter and receiver using the 433 MHz RF module. This module ensures that gesture commands are transmitted reliably and in real-time, enabling seamless control of the wheelchair. The RF module consists of a transmitter connected to the Arduino Nano and a receiver connected to the

Arduino Uno. When a gesture is recognized, the Arduino Nano sends a corresponding command wirelessly to the Arduino Uno. The RF module ensures low-latency and reliable communication, which is crucial for real-time control of the wheelchair.

4.Safety Mechanisms: The safety mechanisms module incorporates various sensors and features to ensure the safe operation of the wheelchair. This includes obstacle detection and fall detection systems that help prevent accidents and ensure user safety.

1.1.4 Challenges and Strategies for Overcoming Them in Hand Gesture-Controlled Wheelchair Design

1.Microcontrollers: Microcontrollers are the central processing units of the wheelchair system, responsible for interpreting sensor inputs, executing control algorithms, and managing communication with external devices. The challenge lies in optimizing the code for efficient performance while minimizing latency and resource usage.

Strategy: To overcome this challenge, developers can employ several strategies:

- Code Optimization: Implement efficient algorithms and data structures to minimize processing overhead and memory usage.
- Modular Design: Break down the software into modular components to facilitate code reuse, maintenance, and debugging.
- Real-Time Task Scheduling: Prioritize critical tasks, such as sensor data processing and motor control, using real-time scheduling techniques to minimize response times.
- Low-Power Modes: Utilize low-power sleep modes when idle to conserve energy and extend battery life.

2. Sensors: Sensors, including accelerometers, gyroscopes, and ultrasonic sensors, provide critical input for gesture recognition, obstacle detection, and fall detection. Challenges include sensor noise, calibration errors, and environmental interference affecting accuracy and reliability.

Strategy: Overcoming sensor-related challenges involves the following strategies:

- Calibration and Compensation: Calibrate sensors regularly to correct for biases, offsets, and environmental effects. Implement compensation algorithms to mitigate drift and improve accuracy over time.

- Sensor Fusion: Integrate data from multiple sensors using sensor fusion techniques, such as Kalman filtering or complementary filtering, to improve accuracy and robustness.
- Filtering and Signal Processing: Apply digital filtering and signal processing techniques to remove noise, smooth sensor data, and extract meaningful information.
- Environmental Monitoring: Monitor environmental conditions, such as temperature, humidity, and electromagnetic interference, to identify and mitigate sources of interference.

3. Motor Control: Motor control systems translate gesture commands into wheelchair movements, requiring precise control over speed, direction, and trajectory. Challenges include motor dynamics, friction, and load variations affecting performance and stability.

Strategy: Overcoming motor control challenges involves the following strategies:

- Closed-Loop Control: Implement closed-loop control algorithms, such as PID control, to continuously adjust motor output based on feedback from encoders or sensors.
- Dynamic Parameter Adjustment: Dynamically adjust control parameters, such as gains and thresholds, based on real-time operating conditions to optimize performance.
- Safety Limitations: Implement safety features, such as current limiting and overtemperature protection, to prevent motor overheating, overloading, or damage.
- Mechanical Design: Optimize mechanical design, including motor selection, gearing, and wheel size, to minimize friction, improve efficiency, and enhance manoeuvrability.

4. Power Supply: The power supply module must provide stable power to all components while managing energy consumption and battery life effectively. Challenges include fluctuating power demands, battery degradation, and charging/discharging cycles affecting reliability and longevity.

Strategy: Overcoming power supply challenges involves the following strategies:

- Efficient Power Conversion: Utilize efficient voltage regulation and power conversion techniques to minimize energy losses and maximize battery life.
- Smart Charging Algorithms: Implement intelligent charging algorithms to optimize battery charging rates, prevent overcharging, and extend battery lifespan.
- Energy Harvesting: Explore energy harvesting technologies, such as solar panels or regenerative braking, to supplement battery power and prolong operating time.
- Battery Monitoring: Implement battery monitoring circuits to track battery voltage, current, and temperature, enabling accurate state-of-charge estimation and preventing overdischarge.

5. Safety Mechanisms: Safety mechanisms, including obstacle detection and fall detection systems, are crucial for preventing accidents and ensuring user safety. Challenges include false alarms, sensor malfunctions, and latency in emergency responses affecting reliability and effectiveness.

Strategy: Overcoming safety mechanism challenges involves the following strategies:

- Redundancy and Fault Tolerance: Implement redundant safety checks and fail-safe protocols to enhance reliability and robustness. For example, use multiple sensors for obstacle detection and cross-validate readings to minimize false alarms.
- Real-Time Monitoring: Continuously monitor sensor inputs and system status in real-time to detect anomalies or deviations from normal operation. Implement predictive algorithms to anticipate potential hazards and preemptively intervene to prevent accidents.
- User Feedback and Intervention: Provide feedback to users through visual or auditory cues to alert them of potential hazards or emergency situations. Enable manual overrides or emergency stop mechanisms to allow users to take immediate action in case of system failure or malfunction.

1.1.5 Overview of Related Work

The Hand Gesture-Controlled Wheelchair project builds upon a rich body of related work in the fields of assistive technology, human-computer interaction, and gesture recognition. Previous research and development efforts have explored various approaches to gesturecontrolled interfaces for wheelchair navigation, each with its own strengths, limitations, and applications. A brief overview of related work in this area is provided below:

1. Gesture Recognition Techniques: Gesture recognition techniques form the foundation of gesture-controlled wheelchair systems, enabling users to convey commands through hand movements and gestures. Computer vision-based methods leverage cameras to track hand movements in real-time, employing image processing algorithms to extract relevant features such as hand position, orientation, and motion trajectories. Wearable sensors offer an alternative approach to gesture recognition, embedding accelerometers, gyroscopes, and magnetometers in gloves or wristbands worn by users. These sensors capture hand movements and gestures directly, providing raw motion data for analysis.

2. Control Algorithms and System Integration: Control algorithms play a critical role in translating recognized gestures into wheelchair movements, ensuring smooth and responsive control. Rule-based systems offer a straightforward approach, mapping specific gestures to predefined control commands based on a set of rules and thresholds. While simple and transparent, rule-based systems may lack adaptability to user preferences or environmental conditions.

3. Sensor Fusion and Environmental Awareness: Sensor fusion techniques combine data from multiple sensors to enhance gesture recognition accuracy and environmental awareness. Fusion of vision-based and inertial sensor data enables robust gesture recognition in diverse lighting conditions and environments, mitigating the limitations of individual sensor modalities. Kalman filters and particle filters are commonly used for sensor fusion, combining visual and inertial cues to estimate hand position and motion accurately. Ultrasonic sensors play a crucial role in obstacle detection and navigation assistance, providing crucial environmental feedback to ensure safe and efficient wheelchair operation. By emitting ultrasonic pulses and measuring their time-of-flight to detect obstacles, these sensors enable the wheelchair to navigate complex environments autonomously.

4. User Studies and Evaluation: User studies and evaluations are essential for assessing the usability, effectiveness, and user experience of gesture-controlled wheelchair systems. Usercentered design methodologies, such as participatory design and co-creation workshops, involve users in the design process from the outset, ensuring that the system meets their needs, preferences, and abilities. Usability testing and performance evaluations measure factors such as gesture recognition accuracy, response time, task completion rates, and user satisfaction, providing valuable feedback for system refinement and improvement. Longitudinal studies and field trials offer insights into the real-world usability and effectiveness of gesture-controlled wheelchair systems.

5. Real-World Deployment and Adoption: Several gesture-controlled wheelchair systems have been deployed in real-world settings, including rehabilitation centres, hospitals, and assistive technology labs. These deployments provide valuable insights into the practical usability and effectiveness of gesture-controlled wheelchair systems, offering users greater independence and mobility in their daily lives. Field trials and long-term deployments allow researchers to evaluate system performance over extended periods, identifying usability challenges and gathering feedback from users and caregivers. By iteratively refining and improving system design and functionality based on real-world usage data, researchers can enhance the usability, reliability, and effectiveness of gesture-controlled wheelchair systems. Adoption studies investigate factors influencing the acceptance and adoption of gesture-controlled wheelchairs among users, caregivers, and healthcare professionals. These studies examine usability, reliability, cost-effectiveness, and integration with existing assistive technology ecosystems, identifying barriers to adoption and opportunities for improvement.

1.1.6 Hardware Components and System Architecture

The Hand Gesture-Controlled Wheelchair project employs a sophisticated array of hardware components and sensors, intricately interconnected to enable seamless gesture-based control of the wheelchair's movements. The system architecture is designed with modularity and scalability in mind, ensuring flexibility for future upgrades and enhancements.

- Microcontroller: At the core of the system lies the Arduino microcontroller, functioning as the brain that orchestrates the various functionalities of the wheelchair. The system utilizes two Arduino boards to ensure efficient communication and processing: an Arduino Uno serves as the receiver, while an Arduino Nano acts as the transmitter.

- Gesture Recognition Module: The robust gesture recognition module employs a suite of motion sensors to capture and interpret hand movements with precision. The primary sensor utilized for this purpose is the MPU-6050 accelerometer and gyroscope sensor, renowned for its high-resolution motion tracking capabilities. Complemented by magnetometers for enhanced orientation detection, this module enables the wheelchair to respond accurately to subtle gestures and commands from the user.

- Motor Control Module: Translating gesture commands into smooth and responsive wheelchair movements is the task of the motor control module. This module interfaces with the wheelchair's propulsion system, comprising DC geared motors, to regulate speed, direction, and trajectory. Leveraging a motor driver such as the L298N dual H-bridge motor driver, the system ensures efficient and reliable control over locomotion, empowering users to navigate their environment with confidence and ease.

- Communication Module: Facilitating seamless interaction between the wheelchair and external devices is the communication module, essential for user convenience and accessibility. Supporting multiple communication protocols such as Bluetooth, Wi-Fi, and RF (Radio Frequency), this module enables wireless connectivity with smartphones, tablets, or remote-control units. The utilization of 433 MHz RF modules ensures robust and reliable communication over extended distances, enhancing the wheelchair's versatility and usability.

- Safety Mechanisms: Prioritizing user safety, the system incorporates advanced safety mechanisms to mitigate risks and prevent accidents. Ultrasonic sensors are strategically deployed for obstacle detection and avoidance, enabling the wheelchair to navigate complex environments with agility and precision. Additionally, a sophisticated fall detection algorithm analyses accelerometer data in real-time, promptly detecting sudden changes in orientation indicative of a fall or tip-over event, and activating safety protocols such as audible alerts or emergency brakes.

- Power Supply: Reliable and sustainable power provision is ensured by the power supply module, which furnishes electrical energy to all system components. A rechargeable lithium-ion polymer (LiPo) battery serves as the primary power source, offering a harmonious balance of energy density, voltage stability, and weight efficiency. Additionally, auxiliary power management circuits and voltage regulators guarantee optimal performance and longevity of the system's electrical components.

1.1.7 Software Implementation and System Operation

The software implementation of the Hand Gesture-Controlled Wheelchair project is a cornerstone of its functionality, seamlessly integrating sensor data processing, gesture recognition, obstacle detection, and safety mechanisms. This intricate orchestration ensures intuitive and responsive control, empowering individuals with physical disabilities to navigate their environments safely and independently through intuitive hand gestures.

- Gesture Recognition and Sensor Integration: At the core of the system lies its ability to accurately interpret user gestures. This is facilitated by motion sensors, notably the MPU6050 accelerometer and gyroscope sensor. These sensors capture changes in acceleration and angular velocity as the user makes hand gestures, providing essential input data for the software to process. Through real-time analysis of sensor readings, the software identifies predefined gestures based on predetermined thresholds and patterns. By leveraging the Arduino IDE and its extensive library ecosystem, developers implement robust algorithms to translate sensor data into actionable commands, ensuring precise and responsive gesture recognition.

- Obstacle Detection with HC-SR04 Ultrasonic Sensors: Navigating environments with obstacles presents significant challenges for wheelchair users. To address this, the Hand Gesture-Controlled Wheelchair project integrates HC-SR04 ultrasonic sensors, strategically positioned to detect obstacles in the wheelchair's path. The software continuously monitors the distance measured by these sensors and calculates proximity to nearby objects. When an obstacle is detected within a predefined range, the software triggers appropriate commands to steer the wheelchair away from the obstruction, ensuring safe and efficient navigation. Complex algorithms handle sensor data fusion and trajectory planning, optimizing obstacle avoidance while minimizing response time and energy consumption.

- Buzzer Integration for Safety Alerts: Enhancing user safety is paramount in the design of the Hand Gesture-Controlled Wheelchair system. In addition to obstacle detection, the project incorporates a buzzer as an additional safety mechanism. Connected to the Arduino microcontroller, the buzzer serves to alert the user and surrounding individuals in the event of a potential fall or collision. The software includes sophisticated logic to analyse accelerometer data in real-time. Sudden changes in orientation indicative of a fall or tipover event prompt the software to activate the buzzer, providing immediate auditory feedback to the user and alerting others to the situation.

- Communication and Control: Facilitating seamless interaction between the wheelchair and external devices, the software establishes wireless communication using standard protocols such as UART or SPI. This enables real-time exchange of control commands, telemetry data, and status updates between the wheelchair and external devices. User inputs, manifested as hand gestures, are interpreted by the software, which then translates these gestures into actionable commands to control the wheelchair's movements. Commands such as forward, backward, left, and right are executed based on the interpreted gestures and obstacle detection data, ensuring safe and efficient navigation through diverse environments.

System Operation: The operation of the Hand Gesture-Controlled Wheelchair system unfolds in a series of steps, showcasing the seamless integration of sensor data processing, gesture recognition, obstacle detection, and safety mechanisms.

- User Interaction: The user initiates interaction by making predefined hand gestures, which are captured by the accelerometer and gyroscope sensors.
- Gesture Recognition: The software processes sensor data to recognize the user's gestures accurately. It identifies predefined gestures based on predetermined thresholds and patterns, leveraging sophisticated algorithms to ensure precision and responsiveness.
- Obstacle Detection: Simultaneously, the software monitors data from the ultrasonic sensors to detect obstacles in the wheelchair's path. By calculating proximity to nearby objects, the system triggers appropriate commands to steer the wheelchair away from obstructions, ensuring safe navigation.
- Safety Mechanisms: In the event of a potential fall or collision, the software activates the buzzer to alert the user and surrounding individuals. Real-time analysis of accelerometer data enables timely intervention, enhancing user safety and awareness.
- Control Execution: Based on the interpreted gestures and obstacle detection data, the software executes control commands to drive the wheelchair's movements. This ensures smooth and efficient navigation through diverse environments, empowering users with confidence and independence.
- Testing and Optimization: Rigorous testing and optimization are paramount to ensure the software's functionality, reliability, and efficiency. Test scenarios simulate various user interactions and environmental conditions to validate the software's performance under diverse circumstances.

1.2 Objectives

1. Enhanced Connectivity:

The primary objective was to establish a robust and reliable communication link between the Arduino Uno (acting as the receiver) and the Arduino Nano (serving as the transmitter). This connection is crucial for the seamless operation of the hand gesture-controlled wheelchair, ensuring that commands are transmitted instantly and reliably from the user's gestures to the wheelchair's control system. This objective was successfully achieved, employing 433 MHz RF modules that facilitated uninterrupted data transfer, despite potential environmental interferences typically found in diverse settings. The implementation of this technology not only meets the project’s requirements but also enhances the user's experience by providing a smooth and responsive control over the wheelchair.

2. Gesture Recognition Enhancement:

The gesture recognition system was engineered to interpret various hand gestures accurately, translating them into movements and navigational commands for the wheelchair. Utilizing the MPU 6050 accelerometer sensor, the system captures motion and orientation data, which is then processed by the Arduino Nano. Our efforts focused on improving the accuracy and responsiveness of this system. Within two months of development, we achieved a sophisticated algorithm capable of interpreting gestures with high precision. This success has substantially improved the interaction between the user and the wheelchair, making it more intuitive and less prone to errors.

3. Obstacle Detection Refinement:

Another critical goal was to refine the obstacle detection capabilities of the wheelchair, which employs ultrasonic sensors to sense and navigate around obstacles. The technology was optimized to detect a variety of obstacles reliably across multiple environments, whether at home or outdoors. This system enhancement was pivotal in ensuring the safety and autonomy of the user, allowing for smooth navigation through tight spaces and crowded areas. The refined obstacle detection system not only prevents collisions but also contributes to the overall confidence of the user while operating the wheelchair.

Summary: The project successfully achieved most objectives, enhancing connectivity, gesture recognition, and obstacle detection, while making progress in fall detection. This led to a highly functional, user-friendly gesture-controlled wheelchair, improving mobility and independence for individuals with physical disabilities. Meticulously tested components, including Arduino platforms for data handling and an RF communication module for real-time gesture data transmission, ensured seamless system integration

CHAPTER 2

LITERATURE REVIEW AND RESEARCH GAP

This research investigates strategies and algorithms to improve the accuracy of Gesture control for wheelchair movements. The literature review examines existing approaches, focusing on outcomes and implications. By identifying research gaps, this project aims to advance Hand Gesture Controlled Wheelchair methods, enhancing mobility and independence for users with physical disabilities.

1. **Name: "Design and Implementation of a Hand Gesture-Controlled Smart Wheelchair System", 2023.**

The literature review examines the article titled "Design and Implementation of a Hand Gesture-Controlled Smart Wheelchair System" by Reshma Anilkumar. This project presents the design and implementation of a hand gesture-controlled smart wheelchair system aimed at empowering individuals with physical disabilities to enhance their mobility and independence. The system comprises two primary sections: the transmitter and receiver sections. In the transmitter section, an accelerometer sensor (adxl335) worn on the wrist detects hand movements and translates them into signals. These signals are transmitted wirelessly via an RF transmitter module to the receiver section integrated into the wheelchair. The receiver section consists of components such as Arduino UNO, RF Receiver module, L293D motor driver, motors, and power supply. Upon receiving the signals from the transmitter section, the receiver section processes them and commands the wheelchair's motors to move accordingly, enabling the wheelchair to navigate in various directions based on the detected gestures.

Research Gap: Despite the innovative approach of the project, several research gaps remain that require further exploration. Firstly, while the project emphasizes the potential of manual gesture-controlled wheelchairs to improve accessibility, there is a need for research to optimize gesture recognition algorithms for enhanced accuracy and responsiveness. Current systems may face challenges in accurately interpreting gestures, especially in dynamic environments with obstacles and varying conditions. Research focusing on refining these algorithms can lead to more reliable and user-friendly gesture-controlled wheelchair systems.

2. **Name: The hand movement gesture wheelchair, 2023**

The literature review examines the article titled “The hand movement gesture wheelchair by the Student of Computer Science and Engineering Vidya Vikas Institute of Engineering and Technology, Mysore”. This project amplifies cutting-edge technology designed to empower individuals with physical limitations by enabling them to control their mobility device through hand gestures. This system utilizes a combination of sensors like accelerometers, gyroscopes, and flex sensors, alongside an ESP32 microcontroller that processes the sensor data to interpret these gestures into commands that control the wheelchair’s movements. Essential components of this system include the ESP32 transmitter and receiver modules that ensure seamless wireless communication, and a motor driver that executes the movement commands by managing the wheelchair's motors. The integration of IoT connectivity enhances this system by enabling remote data logging and monitoring, which can provide additional functionalities such as remote assistance and integration with other IoT devices. However, despite these advancements, the technology presents several research gaps and opportunities for improvement. For instance, the system currently relies on predefined gestures, which may not cater to individual user needs or adapt to unique gesture patterns, suggesting a potential area for the development of adaptive algorithms or the incorporation of machine learning to enhance personalization and system responsiveness.

Research Gap: The hand movement gesture wheelchair, while innovative, exhibits several research gaps and opportunities for advancement that warrant further exploration. One significant gap lies in the system's reliance on predefined gestures, which may not adequately accommodate the unique needs or gesture variations of each user. This limitation suggests an urgent need for the development of adaptive algorithms or the integration of machine learning techniques to enhance personalization and improve the system's ability to respond dynamically to different user inputs. Another area that requires attention is the system's safety and user experience. Current implementations lack comprehensive error management strategies and robust feedback mechanisms, which are vital for ensuring user safety and facilitating effective interaction with the device. Research could explore the development of more intuitive user feedback systems, such as haptic or auditory feedback, that could alert users about recognized gestures or inform them of any system errors.

3. **Name: "A Study on Smart Wheelchair Systems" 2022.**

The literature review examines the article titled "A Study on Smart Wheelchair Systems" Project by Mohammed Hayyan Al Sibai and Sulastri Abdul Manap from the University Malaysia Pahang presents a comprehensive review of recent studies on smart wheelchair systems. It addresses the growing need for enhanced mobility solutions for individuals with disabilities, particularly in severe cases where manual or electrical wheelchairs may not suffice. By introducing the concept of smart wheelchairs equipped with control systems and sensors, the paper highlights advancements aimed at assisting users in various ways. Through an analysis of available technologies, including navigation methods, sensor integration, and user interfaces, the authors provide valuable insights into the current state of research in the field. Moreover, the paper explores commercialization issues and user attitudes towards smart wheelchairs, shedding light on challenges and opportunities for future development. By emphasizing the importance of adaptability and customization in smart wheelchair systems, the paper underscores the need for user-centric design approaches to effectively meet individual user needs and preferences.

Research Gaps: Despite its comprehensive coverage, the paper leaves several research gaps to be addressed. Firstly, while it acknowledges the limited availability of commercial smart wheelchairs and discusses barriers such as technology reliability and cost, it lacks in-depth exploration of strategies to overcome these challenges or potential business models for scaling up production and distribution. Additionally, the paper briefly touches upon user attitudes towards smart wheelchair technologies but does not extensively explore user centred design principles or methodologies for involving end-users in the design and development process. Furthermore, while navigation methods and sensor technologies are discussed, the paper does not delve into specific localization techniques or algorithms used for precise navigation in indoor and outdoor environments. Moreover, the integration of various sensors and sensor fusion techniques are not thoroughly explored, which are crucial for accurate perception and obstacle detection. Finally, the paper does not address broader ethical and social implications, such as privacy concerns and socio-economic disparities in access to assistive technologies, which are essential considerations for equitable and inclusive design and deployment of smart wheelchair systems.

4. **Name: Gesture-Controlled Wheelchair Project by Mahipal Manda, B Shankar Babu, M Abhishek, and J Srikanth,2018.**

The literature review examines the article titled Gesture-Controlled Wheelchair by Mahipal Manda, B Shankar Babu, M Abhishek, and J Srikanth. This project endeavours to develop a hand gesture-based wheelchair utilizing MEMS (Micro Electro-Mechanical Systems) technology, catering to individuals with physical disabilities. By integrating MEMS accelerometer sensors and RF (Radio Frequency) technology, the wheelchair interprets hand gestures to execute movement commands, thereby enhancing mobility and independence for users. Through the utilization of MEMS technology, the project aims to harness the sensitivity of accelerometer sensors to detect subtle hand movements and translate them into precise wheelchair manoeuvres. This approach seeks to provide an intuitive and non-intrusive control mechanism, empowering users with greater autonomy in navigating their surroundings. Furthermore, by incorporating RF technology for wireless communication between the controller and wheelchair, the project aims to eliminate physical tethering, offering users enhanced freedom of movement and convenience. Overall, the project represents a significant step towards the development of assistive technologies that prioritize user-centric design and accessibility, ultimately aiming to improve the quality of life for individuals with physical disabilities.

Research Gap: While the project "Hand Gesture-Controlled Wheelchair" presents commendable strides in assistive technology, it leaves several critical areas unaddressed, signifying significant research gaps. Despite leveraging MEMS accelerometer sensors, there exists a notable lack of detailed exploration into the specific algorithms utilized for gesture recognition. This gap underscores the need for optimization through advanced machine learning or signal processing methodologies to enhance accuracy and reliability in interpreting hand gestures effectively. Additionally, the project overlooks comprehensive evaluations regarding user interface design and user experience, which are essential for ensuring the wheelchair's usability and effectiveness, particularly for individuals with varying degrees of motor impairments. Integrating user-centred design approaches could yield invaluable insights into interface intuitiveness, responsiveness, and overall user satisfaction, thereby enhancing the wheelchair's adoption and usability.

5. **Robotic Wheelchair Technology, Marmara University, 2018.**

The literature review examines the article titled Robotic Wheelchair Technology by Istanbul Bilgi University, Marmara University. This study delves into the transformative potential of robotic wheelchair technology in empowering individuals with disabilities to lead more independent and fulfilling lives. Robotic wheelchairs, equipped with advanced features such as navigation assistance, accident prevention mechanisms, and versatile control systems ranging from traditional joysticks to cutting-edge interfaces, aim to offer users unprecedented levels of autonomy and social integration. Existing prototypes like Accom, SPAM, and Intel wheels exemplify the evolving landscape of assistive robotics. However, there is a pressing need to shift focus towards prioritizing affordability, ergonomic design, and inclusivity, especially in catering to the unique needs of bed-dependent or mentally challenged individuals. The field of robotic wheelchair technology presents a dynamic and promising avenue for ongoing innovation and interdisciplinary collaboration.

Research Gap: While significant strides have been made in robotic wheelchair technology, several gaps persist that warrant attention. Paramount among these is the imperative to embed cost-effectiveness and user-centric design principles into the development process. Although current designs offer compelling functionalities, ensuring affordability and ergonomic comfort remains pivotal for widespread acceptance and utilization. Moreover, there is a distinct lack of tailored solutions catering to the specific requirements of bed-dependent or cognitively challenged users, underscoring a gap in inclusive design methodologies.

6. **Name: "IR Sensor-Based Gesture Control Wheelchair for Stroke and SCI Patients", by Rajesh Kannan Megalingam 2016.**

The literature review examines the article titled "IR Sensor-Based Gesture Control Wheelchair for Stroke and SCI Patients by Rajesh Kannan Megalingam, Senior Member, IEEE. This project introduces HanGes, a novel IR sensor-based gesture recognition system designed for stroke patients and individuals with spinal cord injuries (SCI) to control powered wheelchairs. HanGes consists of a control device called gpaD (Gesture Pad), which incorporates an array of IR sensors, a microcontroller unit (MCU), and a power management circuit. Users make simple hand gestures over the gpaD to navigate the wheelchair, with five predefined gestures corresponding to forward, reverse, left, right, and brake movements. The system architecture, design, and implementation of gpaD, along with the suggested hand gestures, are detailed.

Research Gap: The project addresses a significant gap in assistive device technology by offering a user-friendly and affordable alternative to conventional joystick-based wheelchair controls, specifically tailored for stroke and SCI patients. Existing literature predominantly focuses on complex gesture recognition systems or joystick controls, overlooking the need for simple and accessible solutions for individuals with mobility impairments. While previous research explores various gesture recognition techniques, such as camera-based or accelerometer-based approaches, these often entail high costs and technical complexities, limiting their practicality for widespread adoption. By introducing the gpaD system, which relies on IR sensors and simple hand gestures, the project bridges this gap, providing a promising solution that enhances mobility and independence for a diverse range of users.

7. **Finger-Gesture Controlled Wheelchair with Enabling IoT, 2019.**

The literature review examines the article titled "Finger-Gesture Controlled Wheelchair with Enabling IoT by John, F., Shaju, J., Mathai, M. K., & Seby M ,This project gesture-controlled smart wheelchair system represents a paradigm shift in assistive technology, offering a multifaceted solution to the mobility challenges faced by individuals with disabilities. By leveraging computer vision algorithms and IoT-enabled fall detection mechanisms, the system empowers users to navigate their environments with unprecedented ease and safety. Unlike conventional wheelchairs that rely on manual or joystick controls, this system utilizes intuitive finger gestures captured by an RGB camera, offering a more natural and accessible mode of interaction. Moreover, the integration of IoT technology enables automatic fall detection and emergency messaging, enhancing user security and peace of mind. What sets this project apart is its focus on affordability, with a development cost of under $300, making it accessible to a broader range of users.

Research Gap: While the proposed project presents several innovative features, there are specific areas where further research and development could enhance its effectiveness and usability. Firstly, compared to the provided research paper, the project may benefit from additional studies focused on optimizing gesture recognition algorithms for real-world conditions. This includes improving accuracy in varied lighting environments and refining gesture detection algorithms to accommodate users with different levels of dexterity. Secondly, while the integration of IoT technology for fall detection is a valuable feature, there is a need for research into enhancing the reliability and responsiveness of this functionality. This includes refining algorithms to minimize false positives and exploring alternative sensor configurations to improve detection accuracy.

8. **A new design approach for gesture controlled smart wheelchair utilizing microcontroller, 2018.**

The literature review examines the article titled "A new design approach for gesture controlled smart wheelchair utilizing microcontroller by Noman, A. T., Rashid, H., Khan, M. S., & Islam, 2018, October. The proposed project focuses on developing a cost-effective, gesture-controlled wheelchair aimed at enhancing mobility for individuals with lower body paralysis and limited upper body strength. Traditional wheelchairs often require manual effort or complex control systems like joysticks or voice commands, which may not be suitable for all users due to physical or environmental limitations. This project introduces a smart wheelchair that utilizes the in-built gesture function of smartphones and a capacitive touch sensor to navigate and control the direction of the wheelchair seamlessly.

The core components include an ATMega328 microcontroller, L298N motor driver, DC gear motors, ultrasonic sensors for obstacle detection, a TTP224 capacitive touch sensor, a Bluetooth module for communication, and an IP camera for remote monitoring by caregivers. The user can control the wheelchair by tilting a smartphone or using simple touch gestures, which sends signals to the microcontroller to drive the motors accordingly. The system also incorporates safety features to automatically stop the wheelchair if obstacles are detected, preventing collisions and enhancing user safety. This integration of accessible technology aims to provide a more independent and secure mobility solution for the disabled.

Research Gap: Current solutions for wheelchair mobility predominantly revolve around manual operation or electronic controls via joysticks and voice commands. However, these methods pose significant challenges for individuals with severe physical disabilities, especially those unable to use their upper limbs or speak clearly. The research gap identified in this project lies in the development of a simplified, cost-effective solution that utilizes widely available consumer technology, such as smartphones, to control wheelchairs through gestures and touch.

This project aims to address these gaps by creating a wheelchair that is not only affordable and simple to operate using minimal physical effort but also equipped with an IP camera for continuous monitoring. By leveraging the smartphone's built-in sensors for gesture recognition and control, the system reduces the need for additional, costly hardware components, thereby lowering the overall cost and technical complexity of the wheelchair.

9. Significance of hand gesture recognition systems in vehicular automation, 2014.

The literature review examines the article titled "Significance of hand gesture recognition systems in vehicular automation Francis, J., & Kadan, A. B. (2014, August)". The project aims to develop a robust hand gesture recognition system tailored specifically for vehicular automation. The system will utilize state-of-the-art technologies such as accelerometer-based approaches, glove-based methods, and vision-based techniques employing Kinect sensors. Hand gestures serve as an intuitive and non-intrusive means of interaction between the driver and the vehicle's infotainment systems and other functionalities. By enabling drivers to control various aspects of the vehicle through simple hand movements, the system seeks to enhance both convenience and safety on the road.

The proposed system will consist of three main components: image acquisition, preprocessing, and gesture recognition. Image acquisition involves capturing hand movements using different hardware modules such as accelerometers, data gloves, or Kinect sensors. Preprocessing techniques will be applied to enhance the quality of the acquired data, including filtering, edge detection, and thresholding. Gesture recognition algorithms will then classify the processed hand movements into specific gestures, allowing the system to determine the corresponding operations to be executed. Several methodologies will be explored to address the challenges associated with each approach. For instance, accelerometer-based systems will employ dynamic time warping and affinity propagation algorithms for gesture recognition. Glovebased systems will utilize color-coded gloves and feature extraction techniques followed by classification using learning vector quantization. Vision-based approaches using Kinect sensors will leverage depth sensing and finger detection algorithms based on Finger Earth Mover's Distance (FEMD) for accurate gesture recognition. The project will involve the design, implementation, and evaluation of these gesture detection systems. Performance evaluation will include assessing factors such as accuracy, robustness to variations in lighting and hand orientation, and user-friendliness

Research Gap: While considerable strides have been made in the realm of hand gesture recognition systems, there persist several critical research gaps demanding attention. Despite the extensive exploration of gesture recognition systems across various domains, their seamless integration with vehicular automation remains relatively underexplored. Specifically, there exists a paucity of research dedicated to designing gesture control interfaces tailored explicitly for vehicles.

One notable gap lies in the absence of comprehensive studies evaluating the real-world impact of gesture-based interaction on driver safety and performance during driving scenarios. It is imperative to comprehend how different gesture recognition systems influence driver behavior, visual workload, and overall driving performance to ensure their successful integration into vehicles. Moreover, many existing gesture recognition systems may fall short of meeting the stringent real-time performance requirements essential for seamless integration into vehicles. Hence, there is a pressing need for research focusing on optimizing algorithms and hardware implementations to achieve low-latency, real-time gesture recognition capabilities.

10. Name: Intelligent Gesture Controlled Wireless Wheelchair for The Physically Handicapped, 2013.

The literature review examines the article titled “Intelligent Gesture Controlled Wireless Wheelchair for The Physically Handicapped by SHREEDEEP GANGOPADHYAY, SOMSUBHRA MUKHERJEE & SOUMYA CHATTERJEE. This project endeavours to engineer an intelligent wheelchair system tailored to the mobility needs of individuals grappling with physical disabilities. Departing from conventional methods of wheelchair navigation reliant on keypad inputs, the project pioneers a paradigm shift by harnessing cutting-edge hand gesture recognition technology to facilitate wireless control, liberating users to maneuverer the wheelchair remotely from considerable distances of up to several meters.

Central to the system's architecture are two pivotal components: a sophisticated gesture control module and a seamlessly integrated receiver unit mounted on the wheelchair. The gesture control module, comprising a triple-axis accelerometer sensor (ADXL335), discerns subtle hand orientations and intricate gestures. This module establishes wireless communication with the receiver unit through RF transmitter and receiver modules, affording users the dexterity to dispatch commands to the wheelchair from afar. The receiver unit interprets these commands, orchestrating the locomotion of the wheelchair with precision and finesse.

State-of-the-art edge detection sensors detect precipitous drops or stairwells during backward motion, reorienting the wheelchair's trajectory to avert potential mishaps. Complementing these safety measures is a robust distress call system, enabling users to dispatch SOS messages to preassigned contacts in emergencies.

While the project augments user mobility and fosters independence, it confronts research gaps warranting exploration. Validation of the system's efficacy in real-world environments is imperative, along with scrutiny of the gesture recognition algorithm's accuracy and resilience.

Research Gap:

The project description admirably outlines the components and functionalities of the intelligent wheelchair system, providing a comprehensive overview of its design and capabilities. However, the research gap lies in the lack of detailed discussion regarding the validation of the system's performance and usability in real-world environments. While the project highlights safety features and user-centric design principles, it falls short in substantiating these claims with empirical evidence derived from rigorous testing and evaluation in diverse operational contexts. Moreover, the project could benefit from a deeper exploration of the gesture recognition algorithm's accuracy and resilience, delving into the intricacies of algorithmic design and optimization to enhance robustness and reliability.

Summary: This chapter reviewed ten scholarly papers on assistive mobility technologies, analyzing their objectives, methodologies, and findings. This analysis revealed diverse approaches and identified research gaps in areas like user adaptability, safety, ergonomic design, and the integration of AI and IoT. These gaps highlight the need for innovative solutions to improve device efficacy and accessibility. The next chapter will focus on examining relevant patent applications and grants to uncover emerging trends, innovative solutions, and potential areas for further research in assistive mobility technologies.

CHAPTER 3

IPR PRIOR ART SEARCH

A prior art search is a fundamental component of any project report, crucial for evaluating the novelty and inventiveness of an idea or invention. This comprehensive search involves investigating existing patents, scientific literature, and other publicly available resources to identify prior disclosures related to the project topic. The primary aim is to uncover existing knowledge and inventions that might invalidate or limit the novelty of the project's concept.

By conducting a thorough prior art search, researchers can assess the potential for patentability, identify gaps in the existing knowledge base, and gain valuable insights for further development. This process ensures that the project report accurately acknowledges the current body of knowledge and demonstrates innovation, adding significant value to the field of study. A detailed prior art search helps in avoiding duplication of effort and directs the project toward more unique and unexplored areas. Furthermore, understanding the landscape of existing technologies and solutions can inspire new approaches and improvements. It provides a clearer understanding of the competition and the state of the art in the relevant field. Conducting a comprehensive prior art search substantiates the project's claims of originality, ensuring that it stands on a solid foundation of research and contributes meaningfully to the advancement of the specific research area. This diligent process enhances the credibility of the project and supports its innovative aspects, making it a critical step in the research and development phase.

The insights gained from a prior art search can help in refining the project's objectives and methodologies. By identifying what has already been done, researchers can focus on what hasn't been addressed or can be improved. This can lead to the development of more innovative solutions and can also highlight potential challenges and opportunities within the research area. Moreover, a prior art search can reveal potential collaborators or competing researchers working on similar topics. This information can be valuable for networking, seeking partnerships, or understanding market trends. It can also help in identifying potential licensing opportunities or avoiding infringement on existing patents, thus safeguarding the project's legal standing. Ultimately, a comprehensive prior art search is not just about ensuring the novelty and patentability of the project; it is about building a robust foundation for the research.

IPR 1: WHEELCHAIR MANEUVERING ASSISTIVE SYSTEM

Patent number: 18/2024

Application Number: 202431032600

Applicant name: JIS University

Country: India

Inventor name: Arya Bhattacharyya, Abhishek Kushwaha, Dr. Sandip Roy, Dr. Rajesh Bose, Inderpreet Kaur

Publication Date: 03/05/2024

Abstract: The present invention relates to a wheelchair maneuvering assistive system designed to aid a user in moving a wheelchair with minimal effort. The system features a pair of wearable bodies equipped with a microphone for voice commands, cameras for image capture, motorized grippers for wheel manipulation, and a laser projection unit for path guidance. It also includes a GPS module for real-time location tracking and a microcontroller for processing commands and controlling the system's operations. The system aims to reduce the physical effort required for wheelchair mobility, improve navigation, and enhance safety by detecting potential falls and alerting caregivers.

Claims:

1. A wheelchair maneuvering assistive system, comprising:
    - A pair of wearable bodies developed to be worn by a user over hands, aligning with wheelchair wheels. A microphone for voice commands to aid wheelchair movement, processed by a microcontroller which activates cameras for capturing wheel images.
    - Circular platforms configured via L-shaped telescopically operated rods, extending/retracting to align at the centre of the wheels based on image analysis. Circular plates with motorized grippers installed for gripping hand rims of the wheels, controlled by an intermittent rotation arrangement to maneuver the wheelchair. A laser projection unit for emitting a continuous laser beam to guide the user along an appropriate path, avoiding obstacles on the ground.
2. The system as claimed in claim 1, wherein the microcontroller detects falling conditions of the wheelchair and sends an alert to a computing unit to notify a concerned person for providing aid.
3. The system as claimed in claim 1 and 2, wherein a GPS module is included for detecting real-time location coordinates of the user, sent to the computing unit along with the alert for tracking.
4. The system as claimed in claim 1 and 2, wherein the microcontroller is wirelessly linked with the computing unit via a communication module including Wi-Fi, Bluetooth, and GSM.

5. The system as claimed in claim 1, wherein the intermittent rotation arrangement includes a first disc with a curved member engaging with protrusions on a secondary disc, rotating the plates intermittently.

6. The system as claimed in claim 1, wherein the L-shaped telescopically operated rods are powered by a pneumatic unit comprising an air compressor, air cylinder, air valves, and piston.

7. The system as claimed in claim 1, wherein a battery supplies power to the electrical and electronic components associated with the system.

IPR 2:

Patent number: 2024/11016187

Application Number: 202411016187

Applicant name: Center for Research Impact and Outcome, Chitkara University

Country: India

Inventor name: Jitendra Kumar Katiyar

Publication Date: 05/04/2024

Abstract: The present invention provides an AI-based wheelchair for especially abled persons that includes a diverse array of sensors, such as obstacle detection sensors and environmental sensors, which gather data on the wheelchair's surroundings and user inputs. This data is processed by a microcontroller unit or processor equipped with advanced algorithms specifically designed for navigation and assistance functionalities. The wheelchair also features a communication module enabling interaction with external devices or systems for data exchange and remote monitoring.

Claims:

1. An AI-based wheelchair for especially abled persons, comprising:
   - a plurality of sensors configured to detect environmental factors and user inputs;
   - a processor programmed with advanced algorithms for navigation and assistance functionalities; a communication module enabling interaction with external devices or systems for data exchange and remote monitoring; and a user interface providing personalized controls and feedback to the user.

2. The wheelchair as claimed in claim 1, wherein said plurality of sensors includes:
   - an obstacle detection sensor to identify obstacles and navigate around them safely;
   - an environmental sensor to assess terrain conditions and adjust wheelchair parameters accordingly; and a user input sensor to detect user commands or preferences for personalized navigation.

3. The wheelchair as claimed in claim 1, further comprising a machine learning module trained with user-specific data to adapt and optimize navigation strategies based on individual preferences and behavior patterns.

4. A method for providing personalized navigation and support to especially abled individuals using the AI-based wheelchair of claim 1, comprising the steps of:

   - collecting environmental data and user inputs using the plurality of sensors; processing the collected data using the microcontroller unit or processor to generate navigation commands and assistance recommendations;
   - communicating with external devices or systems to exchange data or receive remote assistance; and presenting personalized controls and feedback to the user via the user interface for enhanced mobility and independence.

5. The method as claimed in claim 4, wherein the method comprises the step of continuously learning and adapting navigation strategies using machine learning techniques based on user feedback and environmental changes.

IPR 3: GESTURE-CONTROLLED SMART WHEELCHAIR

Patent number: 48/2023

Application Number: 202311075599

Applicant name: AJAY KUMAR GARG ENGINEERING COLLEGE

Country: India

Inventor name: MR. NEERAJ GUPTA, SHRUTI GUPTA, TANISHA SINGH, SHRUTI GARG, MUDIT KUMAR SINGH

Publication Date: 01/12/2023

Abstract: The present invention relates to a gesture-controlled smart wheelchair incorporating various sensors and modules such as an accelerometer, IR sensor, proximity sensor, and communication module to interpret user gestures, provide obstacle feedback, and enhance user experience.

Claims:

1. A gesture-controlled smart wheelchair, comprising: a motorized wheelchair designed with specific dimensions and weight capacity; motor controllers to facilitate precise and smooth multidirectional movement; an accelerometer integrated into the system to recognize and interpret user gestures for controlling wheelchair movement; a communication module enabling wireless interactions; a buzzer for providing visual feedback to the user concerning nearby obstacles; a display unit for haptic feedback indicating body temperature and pulse rate; an Infrared (IR) sensor positioned for finger placement to

capture body temperature and pulse rate; a proximity sensor for real-time obstacle detection and prevention of collisions; and user settings allowing adjustments for gesture recognition sensitivity and speed responsiveness.

2. The gesture-controlled smart wheelchair of claim 1, wherein said accelerometer is configured to detect and differentiate gestures corresponding to forward, backward, left, and right movements.
3. The gesture-controlled smart wheelchair of claim 1, wherein said buzzer activates upon detecting an obstacle within a predetermined distance, alerting the user to possible collisions.
4. The gesture-controlled smart wheelchair of claim 1, comprising multiple user profiles, allowing shared use with individualized settings in multi-user environments.
5. The gesture-controlled smart wheelchair of claim 1, wherein said motor controllers, in conjunction with said proximity sensor, ensure real-time adjustments to wheelchair navigation for enhanced safety.
6. A method for controlling a smart wheelchair, comprising the steps of: powering on a motorized wheelchair; calibrating an accelerometer to detect specific user gestures; receiving gesture input from the user via said accelerometer; translating said gesture input into specific movement commands; and activating motor controllers to move the wheelchair in the desired direction.
7. The method of claim 6, further comprising the step of: utilizing a proximity sensor to scan the wheelchair's surroundings, and, upon detecting an obstacle, sending a feedback signal to a buzzer to alert the user.
8. The method of claim 6, wherein body temperature and pulse rate measurements are initiated by: placing a user's finger on an IR sensor, capturing the data, and displaying the readings through haptic feedback on a display unit.
9. The method of claim 6, further comprising: adjusting the gesture recognition sensitivity and wheelchair speed responsiveness based on individual user settings.
10. The method of claim 6, wherein multiple user profiles can be selected and loaded, enabling personalized wheelchair settings for different users in shared environments.

IPR 4: A MIND-CONTROLLED WHEELCHAIR

Patent number: 07/2023

Application Number: 202311008360

Applicant name: Lovely Professional University

Country: India

Inventor name: AKSHAY PAVITHRAN, AMIT KUMAR THAKUR

Publication Date: 17/02/2023

Abstract: The present invention relates to a mind-controlled wheelchair system incorporating EEG sensors, eye tracking technology, a microprocessor with machine learning algorithms, and an Inertial Measurement Unit (IMU) sensor to enable users to control the wheelchair using brain activity and eye movements. The system aims to provide locomotion assistance to paralyzed individuals and aid in motor recovery for those undergoing rehabilitation.

Claims:

1. The mind-controlled wheelchair apparatus comprising:

   a) EEG sensor (101) for recording the brain activity of the user;

   b) Eye tracking camera (102) to detect and track the movement of eyes;

   c) Microprocessor with machine learning algorithm (103) embedded for decision making;

   d) Inertial Movement Unit (IMU) sensor (104) to provide the positional and movement data of the wheelchair (201); D.C. motors for moving the wheelchair (201).

2. The mind-controlled wheelchair apparatus as claimed in claim 1, wherein, signals received from the EEG sensor (101) module is recorded using EEG headband (204).

3.The mind-controlled wheelchair apparatus as claimed in claim 1, wherein, eye tracking (102) is used for the navigation purpose. The mind-controlled wheelchair apparatus as claimed in claim 1, wherein the movement of the wheelchair (201) is integrated with the eye tracking (102).

Summary: This chapter presents the findings of a survey conducted to gather granted patents relevant to the work undertaken in this study. It includes detailed information such as patent numbers, application numbers, applicant and inventor names, as well as a concise description of each patent's work and claims. The subsequent chapter will address the problem statement and initial project design, developed based on the literature survey and prior art search. It will also delve into the rigorous experimentation carried out to achieve the desired outcomes.

CHAPTER 4

PROBLEM STATEMENT AND WORKING

4.1 Problem Statement

Designing a hand gesture-controlled smart wheelchair to enhance independence and accessibility for users with physical disabilities.

Background and Motivation

Individuals with physical disabilities often face significant challenges in mobility and independence. Traditional wheelchairs require manual effort or a joystick control, which can be difficult or impossible for those with severe disabilities. Therefore, the need for an alternative control mechanism that can provide these individuals with the ability to navigate their environment independently is critical.

The primary objective of this project is to design and implement a hand gesture-controlled wheelchair. This system uses intuitive hand gestures to control the wheelchair, providing an accessible solution for individuals with limited motor capabilities. By leveraging advancements in sensor technology, wireless communication, and microcontroller programming, we aim to create a reliable, efficient, and user-friendly device that can significantly improve the quality of life for users with disabilities.

4.2 Technology Used

The hardware components used in the Hand Gesture Controlled Wheelchair project include:

1.Arduino Uno (Receiver)

The Arduino Uno's robust hardware specifications, versatile functionality, and user-friendly development environment make it an ideal choice for the hand gesture-controlled wheelchair project. Serving as the project's central control unit, the Uno orchestrates sensor input, motor control, gesture recognition, and communication, enabling intuitive and safe mobility for individuals with disabilities. Its reliability, flexibility, and ease of use contribute significantly to the project's success and impact.

Specifications:

The Arduino Uno, featuring an Atmel ATmega328P microcontroller, operates at 5 volts with a recommended input voltage range of 7 to 12 volts. It offers 14 digital I/O pins (6 with PWM output) and 6 analog input pins, each capable of sourcing or sinking 20 mA of current. Additionally, the 3.3V pin can supply up to 50 mA. With 32 KB of flash memory (0.5 KB used by the bootloader), 2 KB of SRAM, and 1 KB of EEPROM, the Uno provides sufficient storage and processing capabilities. Operating at a clock speed of 16 MHz, it ensures rapid execution of program instructions. Its compact form factor and versatile connectivity make it suitable for serving as the central control unit in the hand gesture-controlled wheelchair project, orchestrating sensor input, motor control, gesture recognition, and wireless communication functionalities.

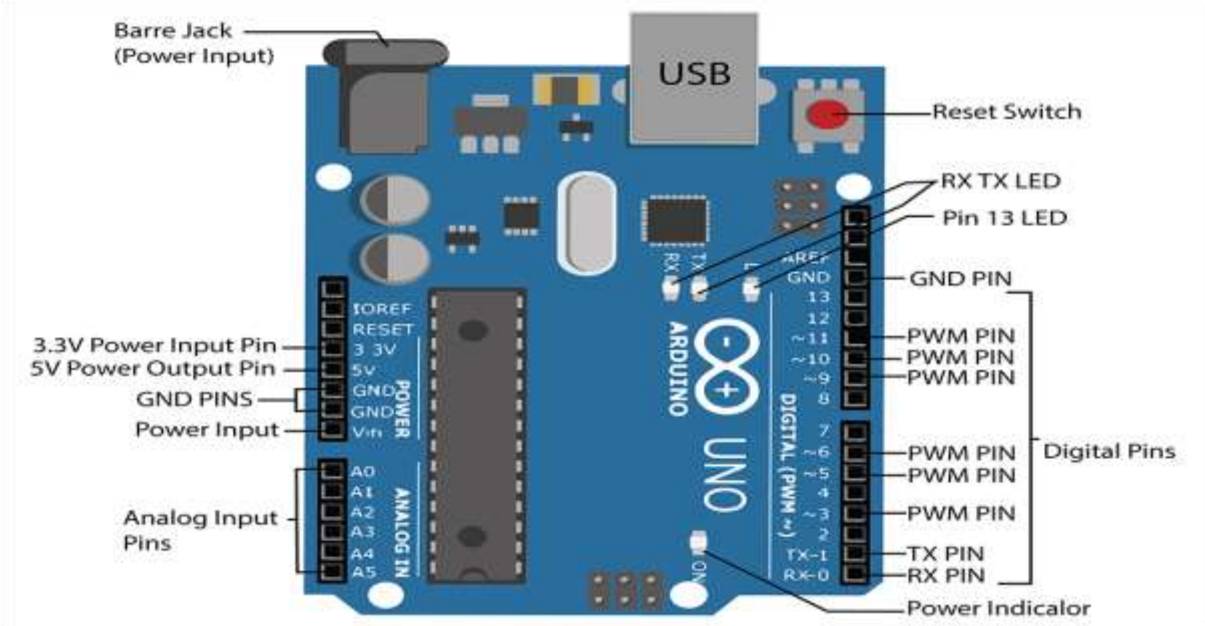

Figure.3 Arduino Uno pinout

Role in the Project:

1. Central Control Unit: The Arduino Uno serves as the central control unit of the hand gesture controlled wheelchair project, orchestrating the operation of various subsystems and components.

2. Sensor Interface: It interfaces with sensors such as the MPU6050 accelerometer and HCSR04 ultrasonic sensor, receiving input data and processing it to make decisions regarding wheelchair movement and navigation.

3. Motor Control: The Uno controls the speed and direction of the wheelchair's DC geared motors through motor drivers such as the L298N, ensuring precise and responsive movement based on user input and environmental feedback.

4. Gesture Recognition: Utilizing data from sensors, the Uno interprets hand gestures made by the user to command the wheelchair's movement, providing an intuitive and user-friendly control interface.

5. Communication Hub: The Uno facilitates communication between various components of the system, such as sensors, motor drivers, and the user interface. It ensures seamless data exchange and coordination to achieve smooth and efficient operation.

6. Safety Monitoring: The Uno incorporates safety features to monitor system status, detect anomalies, and implement fail-safe mechanisms to prevent accidents and ensure user safety during operation.

2. Arduino Nano (Transmitter)

The Arduino Nano serves as the transmitter component in the hand gesture-controlled wheelchair project, facilitating intuitive and hands-free control of the wheelchair's motion. By capturing, processing, and wirelessly transmitting user gestures, it enables seamless interaction between the user and the wheelchair, enhancing mobility and independence for individuals with disabilities. Its compact form factor, powerful microcontroller, and integrated wireless capabilities make it a key enabler of the project's success and usability.

Specifications:

The Arduino Nano (Transmitter) is powered by an Atmel ATmega328P microcontroller, operating at 5 volts with a recommended input voltage range of 7 to 12 volts. It features 14 digital I/O pins (6 with PWM output) and 8 analog input pins, each capable of sourcing or sinking 20 mA of current. Additionally, the 3.3V pin can supply up to 50 mA. With 32 KB of flash memory (2 KB used by the bootloader), 2 KB of SRAM, and 1 KB of EEPROM, the Nano offers ample storage and processing capabilities. Operating at a clock speed of 16 MHz, it ensures fast and efficient execution of program instructions. Its compact dimensions (approximately 18mm x 45mm) make it ideal for wearable applications, such as integration into a glove for capturing hand gestures using the MPU6050 accelerometer and wirelessly transmitting them to the Arduino Uno for controlling the hand gesture-controlled wheelchair.

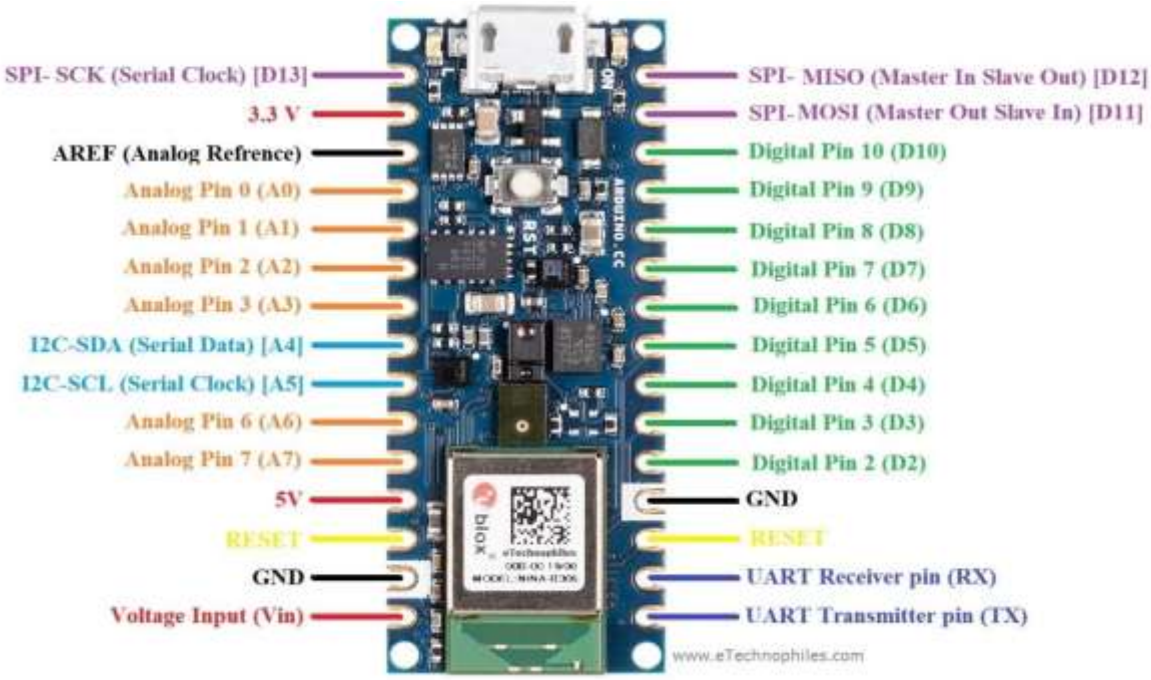

Figure. 4 Arduino NANO Pinout

Role in the Project:

1. Gesture Capture: As the primary interface between the user's hand gestures and the control system, the Arduino Nano plays a crucial role in capturing and interpreting user movements. Its integration with the MPU6050 accelerometer enables precise detection of hand orientation and motion, allowing for intuitive control of the wheelchair.

2. Data Processing: The Nano's microcontroller processes the raw accelerometer data, applying algorithms or thresholds to recognize specific gestures. This processing step is essential for accurately translating physical movements into actionable commands that drive the wheelchair's motion.

3. Wireless Transmission: By wirelessly transmitting gesture commands to the Arduino Uno, the Nano facilitates seamless communication between the user's hand-worn device and the wheelchair's control system. This wireless link enables remote control functionality, freeing the user from physical tethering to the wheelchair and enhancing mobility and convenience.

4. Wearable Integration: With its compact size and lightweight design, the Arduino Nano is well-suited for wearable applications. Mounted on a glove or similar wearable device, it seamlessly integrates into the user's attire, providing unobtrusive control of the wheelchair's movements.

3. L298N Motor Driver

The L298N motor driver serves as a critical component in the hand gesture-controlled wheelchair project, providing robust motor control capabilities essential for propulsion. Its comprehensive specifications, including wide voltage range compatibility, high current handling capacity, and dual H-bridge configuration, enable precise and reliable control over the wheelchair's movements. By amplifying control signals and translating them into motor actions, the motor driver ensures smooth and responsive operation, enhancing the user experience and functionality of the wheelchair system.

Specifications:

The L298N motor driver is a versatile and robust device designed for controlling DC motors in various applications. It operates within a wide voltage range of 5 to 46 volts DC, making it compatible with a diverse range of power sources. With a maximum continuous current handling capacity of 2 amps per channel and a peak current capability of 3 amps per channel, the L298N ensures reliable motor operation even under demanding conditions. Its dual Hbridge configuration allows it to independently control two DC motors or one stepper motor, offering flexibility in motor control options. The control logic operates at 5 volts, facilitating seamless integration with microcontrollers such as the Arduino Uno. Additionally, the L298N features built-in protection mechanisms including over-temperature shutdown, over-current sensing, and under-voltage lockout, enhancing reliability and safety in operation. Its compact dimensions of approximately 43mm x 43mm x 27mm make it suitable for integration into various electronic projects. Overall, the L298N motor driver provides comprehensive motor control capabilities essential for projects requiring precise and reliable motor control.

Figure. 5 L298N Motor driver

Role in the Project:

- High Current Handling for Propulsion: As the primary component responsible for controlling the four DC motors used for propulsion, the L298N motor driver plays a pivotal role in ensuring the wheelchair's mobility. Its ability to handle high continuous and peak currents ensures reliable motor operation, contributing to the smooth and consistent movement of the wheelchair.
- Versatile Motor Control Capability: With its dual H-bridge configuration, the motor driver offers versatile motor control options, enabling precise maneuverability and responsiveness in the wheelchair's movements.
- Integration with Power Source: The L298N motor driver's compatibility with the 3-12V Li-ion batteries utilized in the project ensures efficient utilization of power resources.

4. 433 MHz RF Module

The RF module facilitates wireless communication between the Arduino Nano (transmitter) and the Arduino Uno (receiver). This module is crucial for the real-time transmission of gesture data.

Specifications

The 433MHz frequency is commonly used for short-range wireless communication due to its favourable propagation characteristics.

Employing Amplitude Shift Keying (ASK) modulation, devices operating at this frequency are widely utilized in remote control systems, wireless sensors, and home automation applications for reliable and efficient data transmission.

433MHz RF Transmitter

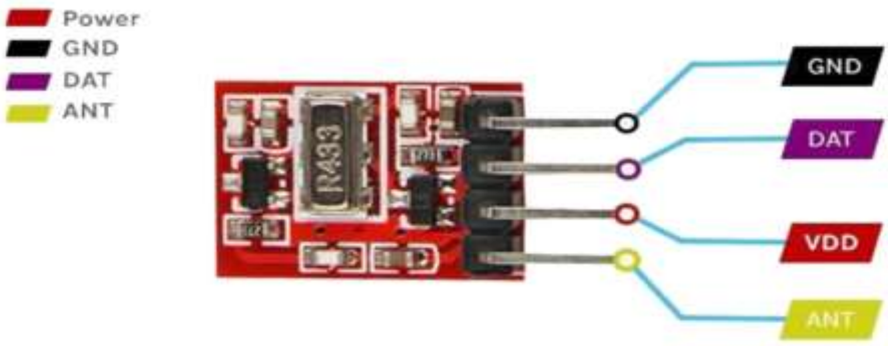

Figure.6 433MHz RF Transmitter

- Operating Frequency: 433MHz
- Operating Voltage: 3.3V to 12V
- Note: The range of the device increases with higher supply voltage.
- Current Consumption: 9mA at 3.3V, up to 40mA at 12V
- Data Rate: Up to 10kbps
- Pin Configuration:
- Data Pin: Accepts digital data to be transmitted.
- VCC: Power pin, operating voltage between 3.3V and 12V.
- GND: Ground pin.
- ANTEENA: External antenna connection, recommended 17.3cm wire for range.

433MHz RF Receiver

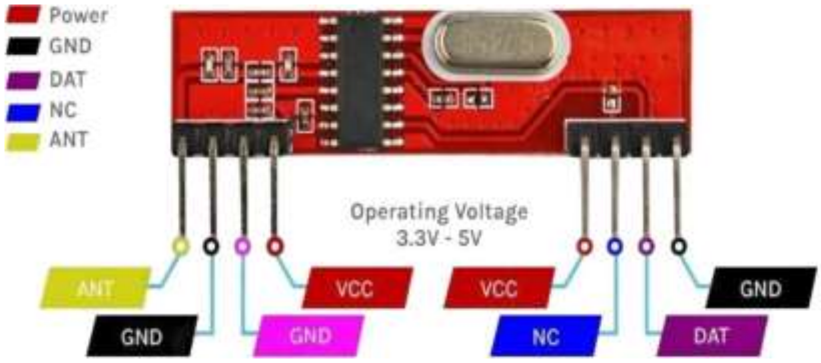

Figure.7 433MHz RF Receiver

- Operating Frequency: 433MHz
- Operating Voltage: 5V
- Current Consumption: 4mA
- Sensitivity: -105 dBm • Data Rate: Up to 10kbps
- Pin Configuration:
- ANT: Exposed pin for RF antenna.
- GND: Three ground pins, internally connected.
- VCC: Two power pins, internally connected, requires 5V.
- Data: Outputs the received digital data.
- NC: Not connected.

Working

Transmitter (Handheld Gesture Recognition Unit):

- Gesture Detection: Integrated with the MPU6050 accelerometer, the transmitter unit captures the user's hand gestures. The MPU6050 senses the orientation and motion of the hand, converting these physical movements into corresponding analog signals.
- Data Processing: The Arduino Nano processes the analog signals from the MPU6050, translating them into digital data that represents specific gestures.
- Wireless Transmission: The digital data is then sent to the 433MHz RF transmitter module. Using ASK modulation, the transmitter converts the digital data into a modulated RF signal and broadcasts it wirelessly.
- Power Utilization: The transmitter operates on a voltage range of 3.3V to 12V, powered by a battery. The operating voltage influences the transmission range, with higher voltages providing greater range.

Receiver (Wheelchair Control System):

- Signal Reception: The 433MHz RF receiver module receives the transmitted RF signal.
  The antenna picks up the signal, which is then processed by the receiver.
- Data Decoding: The receiver demodulates the RF signal, extracting the digital data. This data is sent to the Arduino Uno for further processing.

- Data Interpretation: The Arduino Uno interprets the received digital data, determining the specific hand gestures made by the user. Each gesture is mapped to a corresponding command for the wheelchair.
- Motor Control: Based on the interpreted commands, the Arduino Uno generates control signals for the L298N motor driver, directing the wheelchair’s motors for movement.
- Power Supply: The receiver module requires a 5V power supply, typically provided by the wheelchair's onboard battery system.

Role in the Project

The 433MHz ASK transceiver plays a pivotal role in the hand gesture-controlled wheelchair project, enabling seamless wireless communication between the user’s hand gestures and the wheelchair’s control system. Here’s how it integrates into the project:

- Gesture Transmission: The transceiver facilitates the wireless transmission of hand gesture data from the handheld unit to the wheelchair control system, ensuring real-time communication without the need for physical connections.
- User Mobility: By providing a wireless interface, the transceiver enhances the mobility and convenience of the user, allowing them to control the wheelchair intuitively with hand gestures.
- Motor Control Commands: The transmitted gestures are converted into motor control commands by the Arduino Uno, which directs the wheelchair’s movement. This includes forward, backward, and turning motions, based on the interpreted gestures.
- Enhanced User Independence: The wireless functionality of the transceiver significantly enhances the independence of users with physical disabilities, allowing them to control their wheelchair effortlessly.

5.SENSORS

5.1.MPU 6050 Accelerometer

The MPU-6050 is a 6-axis motion tracking device that includes a 3-axis gyroscope and a 3axis accelerometer. It is capable of capturing both angular velocity and linear acceleration, which makes it ideal for applications involving motion sensing and gesture recognition. The sensor communicates with a microcontroller, such as the Arduino Nano, via the I2C protocol.

Role in the Project

In the hand gesture-controlled wheelchair project, the MPU-6050 is mounted on a wearable device, typically a glove, which the user wears on their hand. The primary function of the MPU-

6050 in this project is to detect and interpret the user's hand gestures, translating these gestures into commands that control the movement of the wheelchair.

The MPU6050 is a very popular accelerometer gyroscope chip that has 6 axis sense with a 16bit measurement resolution. This high accuracy in sense and the cheap cost makes it very popular among the DIY community. Even many commercial products are equipped with the MPU6050. The combination of gyroscope and accelerometers is commonly referred to as an Inertial Measurement Unit or IMU.IMU sensors are used in a wide variety of applications such as mobile phones, tablets, satellites, spacecraft, drones, UAVs, robotics, and many more. They are used for motion tracking, orientation and position detection, flight control, etc.

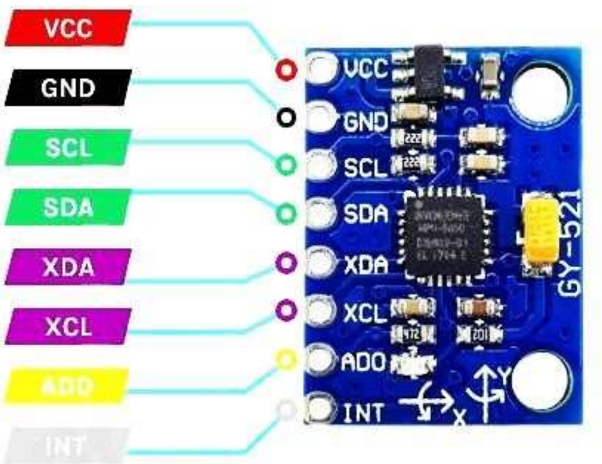

Figure.8 MPU6050 Pinout

- VCC: Provides power for the module, Connect to the 5V pin of the Arduino.
- GND: Ground Connected to Ground pin of the Arduino.
- SCL: Serial Clock Used for providing clock pulse for I2C Communication.
- SDA: Serial Data Used for transferring Data through I2C communication.

- XDA: Auxiliary Serial Data - Can be used to interface other I2C modules with MPU6050.
- XCL: Auxiliary Serial Clock - Can be used to interface other I2C modules with MPU6050.
- ADD/ADO: Address select pin if multiple MPU6050 modules are used.
- INT: Interrupt pin to indicate that data is available for MCU to read.

The MPU6050 module consists of an MPU6050 IMU chip from TDK Intenseness. It comes in a 24-pin QFN package with a dimension of 4mm x 4mm x 0.9mm. The module has a very low component count, including an AP2112K 3.3V regulator, I2C pull-up resistors, and bypass capacitors. There is also a power led which indicates the power status of the module.

How Does the MPU-6050 Module Work?

The MPU-6050 is a Micro-Electro-Mechanical Systems (MEMS) device that integrates a 3axis accelerometer and a 3-axis gyroscope within a single module. This combination allows the MPU-6050 to measure acceleration, velocity, orientation, and displacement, as well as other motion-related parameters of a system or object. Additionally, the module includes a Digital Motion Processor (DMP), which is powerful enough to handle complex calculations, thereby offloading these tasks from the main microcontroller. The MPU-6050 module features two auxiliary pins that can be used to interface with external I2C modules, such as magnetometers, though this is optional. The module's I2C address is configurable, enabling the connection of multiple MPU-6050 sensors to a single microcontroller using the AD0 pin. The MPU-6050 is widely supported with well-documented libraries, making it easy to integrate with popular platforms like Arduino. Its versatility makes it suitable for applications in RC cars, drones, selfbalancing robots, humanoids, bipedal robots, and other motion control systems.

How Does a MEMS Accelerometer Work?

MEMS accelerometers are utilized to measure linear motion, including movement, shock, and vibration, without relying on a fixed reference. These accelerometers measure the linear acceleration of the objects to which they are attached. The working principle of an accelerometer involves a mass attached to a spring. When the object accelerates, the mass tends to remain stationary due to inertia, causing the spring to stretch or compress. This creates a force that is detected and corresponds to the applied acceleration.

In a MEMS accelerometer, precise detection of linear acceleration along two orthogonal axes is achieved using silicon MEMS detectors formed by spring 'proof' masses. Each mass acts as the moving plate of a variable capacitor, formed by an array of interlaced finger-like structures. When the sensor experiences linear acceleration along its sensitive axis, the proof mass resists motion due to its inertia. This resistance causes the mass and its fingers to displace relative to the fixed electrode fingers. The gas between the fingers provides a damping effect. The displacement induces a differential capacitance between the moving and fixed silicon fingers, which is proportional to the applied acceleration. This change in capacitance is measured with a high-resolution Analog-to-Digital Converter (ADC), and the acceleration is calculated from the rate of change in capacitance. The MPU-6050 then converts this into readable values and transfers the data to the I2C master device.

How Does a MEMS Gyroscope Work?

The working principle of a MEMS gyroscope is based on the Coriolis Effect. The Coriolis Effect occurs when a mass moving in a specific direction with a certain velocity experience an external angular motion, generating a force that causes perpendicular displacement of the mass. This generated force is known as the Coriolis Force, and the phenomenon is called the Coriolis Effect. The rate of displacement is directly related to the applied angular motion.

A MEMS gyroscope contains a set of four proof masses that are kept in continuous oscillating movement. When angular motion is applied, the Coriolis Effect induces a change in capacitance between the masses, depending on the axis of angular movement. This change in capacitance is detected and converted into a readable value. The MPU-6050 uses this data to measure the rate of angular velocity around the X, Y, and Z axes. These readings are crucial for detecting rotational movements.

Working Process: -

1. Mounting and Initial Setup: The MPU-6050 is securely mounted on a glove that the user wears. This placement ensures that the sensor can accurately capture the hand's movements. The sensor's VCC and GND pins are connected to the 9Volt and ground pins of the Arduino Nano, respectively. The I2C communication lines (SDA and SCL) of the MPU-6050 are connected to the corresponding pins on the Arduino Nano, with appropriate pull-up resistors to ensure stable communication.
2. Initialization: When the system is powered on, the Arduino Nano initializes the MPU-6050. This involves configuring the sensor's registers via the I2C interface. Key initialization steps include setting the accelerometer and gyroscope ranges (e.g., ±2g for the accelerometer and ±250°/s for the gyroscope), enabling the Digital Motion Processor (DMP) if needed, and calibrating the sensor to eliminate any offsets.
3. Data Acquisition: The MPU-6050 continuously measures linear acceleration along the X, Y, and Z axes using its accelerometer, and angular velocity around the X, Y, and Z axes using its

gyroscope. The sensor's 16-bit analog-to-digital converters (ADCs) convert these measurements into digital values, which are then sent to the Arduino Nano over the I2C bus.

4. Gesture Recognition: The Arduino Nano reads the raw accelerometer and gyroscope data from the MPU-6050 at regular intervals. This raw data includes values for acceleration (ax, ay, az) and angular velocity (gx, gy, gz). The Arduino Nano processes this data to determine the orientation and movement of the hand. For example, a forward tilt of the hand increases the value of ay, while ay backward tilt decreases it. By applying filtering techniques, such as lowpass filters, the Arduino Nano removes noise from the sensor data to improve accuracy.
5. Interpreting Gestures: The Arduino Nano uses predefined algorithms to interpret the processed sensor data as specific gestures. For instance:
6. Forward Gesture: If the ay value exceeds a certain threshold while the ax and az values remain relatively stable, the system interprets this as a forward movement gesture.
7. Backward Gesture: If the ay value drops below a certain threshold, it is recognized as a backward movement gesture.
8. Left and Right Gestures: These are detected by changes in the ax values, indicating lateral hand movements.
9. Transmitting Gesture Data: Once a gesture is recognized, the Arduino Nano encodes this information into a data packet and transmits it wirelessly to the Arduino Uno (receiver) using the 433 MHz RF module. The data packet includes the type of gesture and its corresponding command like move forward, turn right, backward ,stop, turn left.
10. Wheelchair Control: The Arduino Uno receives the data packet from the Arduino Nano and decodes the gesture command. Based on the received command, the Arduino Uno generates appropriate control signals for the L298N motor driver, which in turn controls the DC motors to move the wheelchair in the desired direction.
11. Continuous Feedback Loop: The system operates in a continuous feedback loop, where the MPU-6050 continuously captures hand movements, the Arduino Nano processes and interprets these movements, and the Arduino Uno adjusts the wheelchair's motion accordingly. This realtime interaction ensures that the wheelchair responds promptly to the user's gestures, providing an intuitive and seamless control experience.
12. Calibration and Optimization: Calibration is crucial to ensure the accuracy of the MPU-6050. This involves placing the sensor in known orientations and adjusting the output to match expected values. During calibration, the Arduino Nano measures the offset values for the accelerometer and gyroscope when the sensor is at rest. These offsets are subtracted from the raw sensor data to eliminate bias and improve accuracy.

13. Power Utilization: The MPU-6050 and other remote components are directly connected to the 9V battery. The Arduino Nano, which has an onboard voltage regulator, steps down the 9V to 3.3V required by the MPU-6050. This regulated power supply ensures the sensor operates reliably, capturing accurate hand gesture data.

5.2. Ultrasonic Sensor

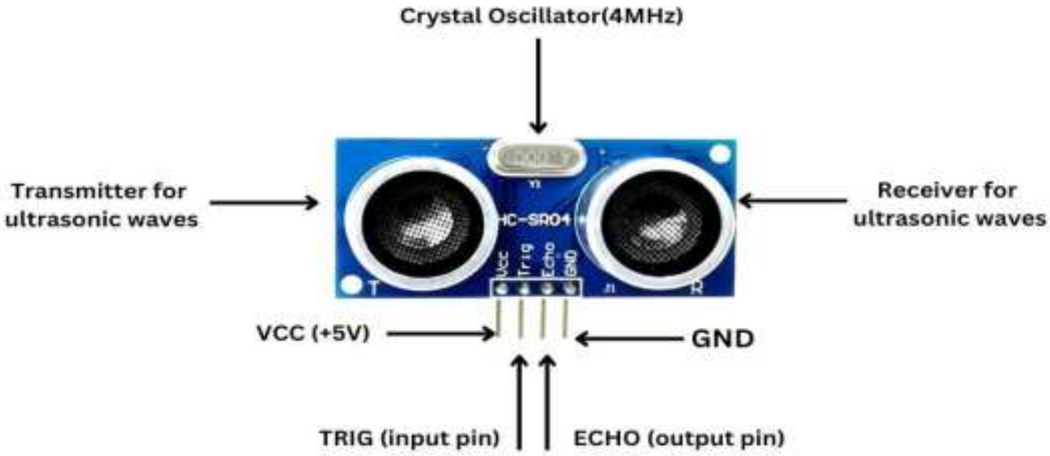

Figure. 9 Ultrasonic Sensor

Specifications:

1.The HC-SR04 ultrasonic sensor operates at a standard voltage of 5V DC, making it compatible with a wide range of electronic systems and microcontrollers. This ensures seamless integration into various projects without the need for additional voltage regulation or modification.

2.With a low quiescent current of less than 2mA and a moderate working current of approximately 15mA, the HC-SR04 sensor demonstrates efficient power utilization. This minimal power consumption is advantageous, especially in battery-powered applications, where conserving energy is crucial for prolonged operation and enhanced usability.

3. The sensor emits ultrasonic waves at a frequency of 40kHz, which is well-suited for distance measurement and proximity sensing tasks. This frequency ensures reliable performance in detecting objects and obstacles within its operational range, providing accurate distance readings for various applications, including robotics, automation, and obstacle avoidance systems.

4.The HC-SR04 sensor offers a wide measuring range, capable of accurately detecting distances from as close as 2 centimetres to as far as 4 meters. This versatility makes it suitable for a diverse array of applications, ranging from close-range proximity sensing to long-range distance measurement, enabling its use in projects such as object detection, liquid level sensing, and smart parking systems.

5. With a resolution of 0.3 centimetres, the HC-SR04 sensor delivers precise distance measurements, allowing for detailed localization of objects within its detection range. This high resolution enables finer control and accuracy in applications where precise distance information is essential, such as robotics navigation, gesture recognition, and interactive installations.

6. Featuring a narrow measuring angle of 15 degrees, the HC-SR04 sensor provides focused and directional distance detection. This narrow angle ensures that the sensor's ultrasonic waves are concentrated within a specific field of view, enhancing accuracy and reducing interference from surrounding objects or surfaces. As a result, the sensor can effectively detect objects with high precision, even in complex environments with multiple obstacles or reflective surfaces.

Working Principle of ultra sonic sensor

The HC-SR04 operates based on the principle of echolocation, similar to how bats navigate and locate objects in their environment. Here's a detailed breakdown of its working principle:

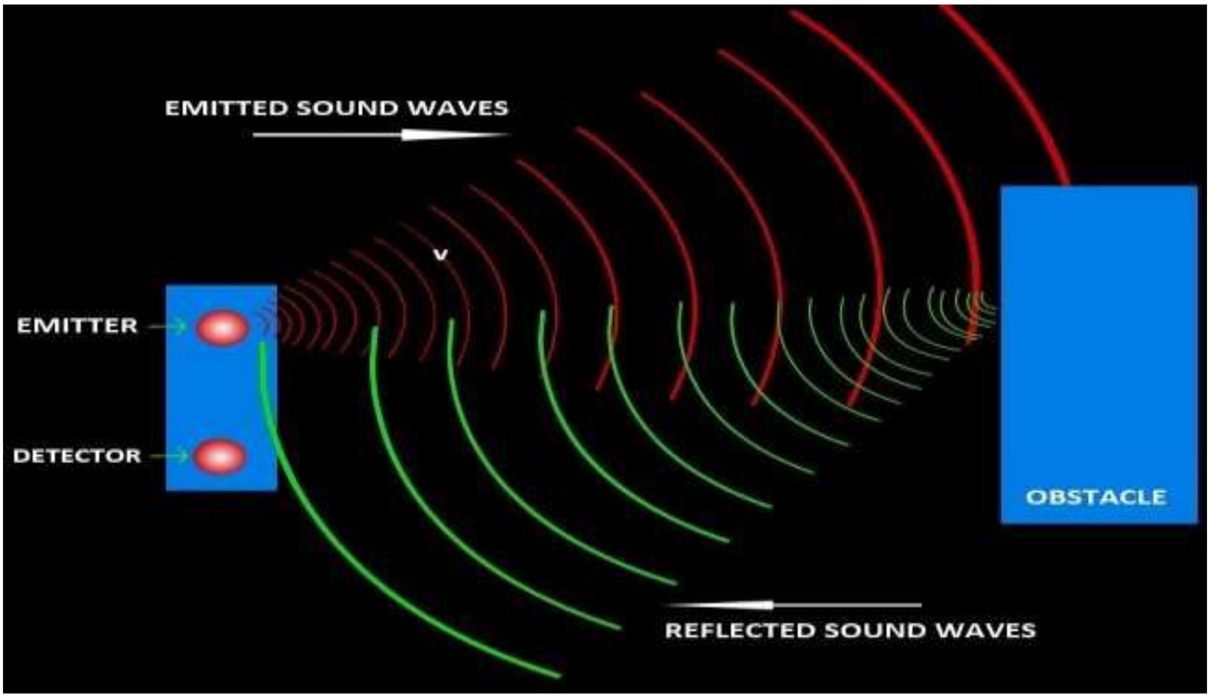

Figure. 10 Working of ultra sonic sensor

1. Transmission of Ultrasonic Waves: The sensor consists of two main components: a transmitter and a receiver. The transmitter emits ultrasonic waves at a frequency of 40kHz when triggered by a pulse from the microcontroller.

2. Propagation and Reflection: Once emitted, ultrasonic waves propagate through the air until they encounter an obstacle. Upon striking the obstacle, the waves reflect back towards the sensor, creating an echo. The sensor's receiver then detects this echo, enabling distance calculation and obstacle detection in applications such as the hand gesture-controlled wheelchair project.

3. Reception of Echo: The HC-SR04 sensor's receiver detects echoes after emitting ultrasonic waves, indicating obstacle proximity. These echoes are generated when waves bounce off obstacles and return to the sensor. The microcontroller processes these signals to calculate distances, crucial for collision avoidance in the hand gesture-controlled wheelchair project. By measuring the time taken for echoes to return, the system determines obstacle distances and initiates appropriate responses to ensure user safety.

4. Distance Calculation: The sensor measures the time interval between transmitting the ultrasonic wave and receiving the echo. Using the known speed of sound in air (~343 meters per second at room temperature), the sensor calculates the distance to the obstacle based on the time taken for the ultrasonic wave to travel to the obstacle and back.

5. 5.Output Signal: After emitting ultrasonic waves, the HC-SR04 sensor receives echoes upon encountering obstacles. It sends an echo pulse to the microcontroller, conveying distance information. Analysing the pulse duration, the microcontroller calculates obstacle distance.
6. This feedback enables accurate obstacle detection, crucial for collision avoidance.

Role in the Project

1. Sensor Integration and Data Acquisition: The HC-SR04 ultrasonic sensor is strategically mounted on the wheelchair's front frame to provide a clear view of the surroundings. Through integration with the Arduino Uno microcontroller, the sensor continuously sends and receives ultrasonic waves to detect nearby obstacles.
2. Real-Time Distance Calculation: When the Arduino Uno triggers the sensor, it emits ultrasonic waves, which propagate until they encounter an obstacle. Upon hitting an obstacle, the waves reflect back to the sensor, where the receiver detects the echo. The Arduino calculates the distance to the obstacle based on the time taken for the ultrasonic waves to travel to the obstacle and back, using the speed of sound in air (~343 meters per second at room temperature).
3. Obstacle Detection Logic: Once the distance to the obstacle is calculated, the Arduino compares it to a predefined safety threshold. If the calculated distance falls below the threshold, indicating the presence of an obstacle within the wheelchair's path, the Arduino identifies the situation as a potential collision hazard.
4. Activation of Collision Avoidance Mechanism: Upon detecting an obstacle, the Arduino triggers the collision avoidance mechanism to ensure the safety of the wheelchair user. The mechanism typically involves halting the wheelchair's movement or altering its direction to avoid the obstacle.
5. Wheelchair Movement Halt: One common approach to collision avoidance is to stop the wheelchair's movement altogether when an obstacle is detected. To achieve this, the Arduino sends signals to the motor control system, instructing it to cease propulsion. By cutting off power to the

wheelchair's motors, the Arduino effectively brings the wheelchair to a complete stop, preventing any further movement towards the obstacle.

6. Power Utilization: The HC-SR04 ultrasonic sensor operates at a standard voltage of 5V, typically supplied by the Arduino Uno microcontroller. Its low quiescent and working currents ensure efficient power utilization, making it suitable for battery-powered applications like the hand gesture-controlled wheelchair. The sensor's power requirements are well within the capabilities of the wheelchair's electronic system, ensuring reliable operation without significant power consumption.

6. Propulsion System:

The propulsion system comprising DC geared motors, L298N motor driver, and battery pack serves as the driving force behind the wheelchair's movement. With efficient torque generation, speed and direction control, and reliable power supply.

Figure. 11 DC Motors and wheels

- Speed: The DC geared motors operate at approximately 100 RPM, providing sufficient speed for wheelchair movement.
- Voltage: They are designed to operate at a nominal voltage of 12 volts, ensuring compatibility with the power source.
- Working Voltage: The working voltage range of the motors is 3-6 volts, allowing flexibility in power supply configurations.
- Torque and Quality: Selected for their high torque output and durability, these motors offer reliable performance in driving the wheelchair.
- Wheel Diameter: With a wheel diameter of approximately 65mm, these motors facilitate smooth movement and traction on various surfaces.

7. Power Management

Power management is essential for ensuring the effective operation of the hand gesture-controlled wheelchair project. This involves managing the power supply to various components of the system, including the Arduino Nano, Arduino Uno, DC motors, L298N motor driver, MPU 6050 sensor, and the 433MHz RF modules. Proper power management ensures that each component operates within its specified voltage and current range, thereby enhancing the overall efficiency, reliability, and safety of the project.

Battery Specifications: - **1. Primary Battery System for Wheelchair:**

Figure. 12 Li-PO Battery

Type: Three 12V Li-ion batteries.

- Capacity: Typically, between 1000mAh to 1200mAh per battery, depending on the specific model used.
- Configuration: Batteries connected in series to provide a stable 12V output with increased capacity. Voltage Rating: 12V nominal voltage.

2. Secondary Power Sources:

Figure. 13 9v Battery

9V Battery for Arduino Uno: Provides the necessary power to the microcontroller, ensuring it can process data and control other components.

- 9V HW Battery for Remote Control: Powers the Arduino Nano and the 433MHz RF transmitter module in the remote.

Power Distribution and Management

Voltage Regulation: Step-Down Converters: DC-DC buck converters are used to step down the 12V supply from the batteries to 5V, required by most of the low-power electronic components. Buck converters are preferred due to their high efficiency, typically above 90%, which is crucial for conserving battery life.

Linear Regulators: In some cases, linear voltage regulators are used for components that require very stable voltage with minimal noise, albeit with lower efficiency compared to buck converters.

On/Off Switch: integration with Battery Pack: On/off buttons are integrated into the battery packs to provide a convenient way to disconnect power. This prevents battery drain during inactivity and allows for safe maintenance. Pressing the button completes the circuit, allowing current to flow and powering the device. Releasing the button interrupts the circuit, cutting off the power supply.

Role in the Project

- Stable Power Supply: Ensures that all components receive the correct voltage and current, maintaining stable operation and preventing damage. This is essential for the smooth and reliable functioning of the wheelchair.
- Extended Battery Life: Optimized Power Consumption: Through efficient voltage regulation and power conservation strategies, the overall battery life is extended. This reduces the frequency of recharges, enhancing the wheelchair's usability and independence for the user.
- Protection Mechanisms: The power management system includes fuses and thermal protection to safeguard against overcurrent and overheating, ensuring user safety and protecting sensitive components.
- On/Off Buttons: Provide a quick way to disconnect power, allowing for immediate shutdown in emergencies and safe handling during maintenance.

Performance Optimization

- Efficient Motor Control: Ensures smooth and responsive movement of the wheelchair, providing a better user experience. Proper power management ensures that the motors receive stable power, enhancing performance.
- Sensor Responsiveness: Maintains the responsiveness of sensors for obstacle detection and gesture recognition, crucial for the wheelchair's functionality.
- Heat Management: Proper heat management strategies, such as using heatsinks and ensuring adequate ventilation, are implemented to dissipate heat generated by the highpower components. This prevents overheating and potential damage to the system.

Working of the Battery System

Primary Battery System (Three 12V Li-ion Batteries): Powering High-Power Components: The 12V Li-ion batteries primarily power the DC motors via the L298N motor driver. These motors require substantial power to drive the wheelchair, especially when navigating slopes or rough terrain.

Voltage Regulation: The voltage from the 12V batteries is regulated to lower voltages for other components. For example, the Arduino Uno and sensors operate at 5V, requiring step-down voltage regulation.

Secondary Power Sources (9V Batteries):

- Arduino Uno: A 9V battery powers the Arduino Uno, which acts as the central control unit. This microcontroller processes data from the sensors and remote control and sends control signals to the motor driver.

- Remote Control (Arduino Nano): Another 9V battery powers the remote control, including the Arduino Nano and the 433MHz RF transmitter module. This setup ensures reliable communication between the user's hand gestures and the wheelchair's control system. **Detailed Analysis of Power Management**

Battery Selection:

- Capacity and Discharge Rate: The capacity of the Li-ion batteries is chosen based on the power requirements of the DC motors and the expected usage duration. Highdischarge rate batteries are preferred to handle the sudden surges in current demand during starting or rapid acceleration.

- Safety Features: Li-ion batteries with built-in protection circuits are selected to prevent overcharging, over-discharging, and short circuits. This enhances the safety and longevity of the battery packs.

Power Distribution Strategy:

- Centralized Control: The power distribution board centralizes power management, simplifying wiring and reducing the risk of electrical faults. It also allows for easy integration of protection features like fuses and circuit breakers.

- Separate Power Rails: High-power components (motors) and low-power components (microcontrollers and sensors) are powered by separate rails. This separation prevents interference and ensures that the voltage drops caused by high current draw from the motors do not affect the low-power components.

On/Off Switch Integration

- User Convenience: On/off buttons integrated into the battery packs allow users to easily control the power state of the wheelchair and remote control. This feature is particularly useful for conserving battery life during periods of inactivity.

- Emergency Shutdown: The on/off buttons provide a quick way to disconnect power in case of an emergency, ensuring user safety and preventing damage to the electronic components

Detailed Connections in the Project

1.MPU 6050 to Arduino Nano:

- VCC to 3.3V: Powers the MPU 6050.
- GND to GND: Provides a common ground.
- SDA to A4: I2C data line.
- SCL to A5: I2C clock line.

2. Arduino Nano to 433MHz Transmitter:

- Data to D2: Sends data to be transmitted.
- VCC to 5V: Powers the transmitter.
- GND to GND: Provides a common ground.

3. 433MHz Receiver to Arduino Uno:

- Data to D8: Receives data from the transmitter.
- VCC to 5V: Powers the receiver.
- GND to GND: Provides a common ground.

4. Arduino Uno to L298N Motor Driver:

- IN1 to D3: Control signal for Motor 1.
- IN2 to D4: Control signal for Motor 1.
- IN3 to D5: Control signal for Motor 2.
- IN4 to D6: Control signal for Motor 2.
- ENA to D9: PWM control for Motor 1 speed.
- ENB to D10: PWM control for Motor 2 speed.
- 12V to Battery Positive: Power supply for motors.
- GND to Battery Negative: Common ground.

5. Ultrasonic Sensor to Arduino Uno:

- VCC to 5V: Powers the sensor.
- GND to GND: Provides a common ground.
- Trig to D11: Sends trigger signal.
- Echo to D12: Receives echo signal.

4.3 Block Diagram and Working

The block diagram section offers a comprehensive visual representation of the system architecture for both the Remote-control unit and the wheelchair. It includes high-level and component-specific diagrams that detail key components and their interconnections. These diagrams map out the signal flow and functionality, from initial gesture detection to final motor activation. Each block is thoroughly discussed to explain its role, working principles, and contribution to the overall system functionality. This in-depth analysis highlights the seamless integration of hardware and software, ensuring a clear understanding of the intuitive and reliable operation of the hand gesture-controlled wheelchair system.

4.3.1. Remote Control Unit

The Remote-control Unit captures hand gestures using the MPU6050 sensor, processes them with the Arduino Nano, and wirelessly transmits commands to the wheelchair via the 433 MHz RF transmitter.

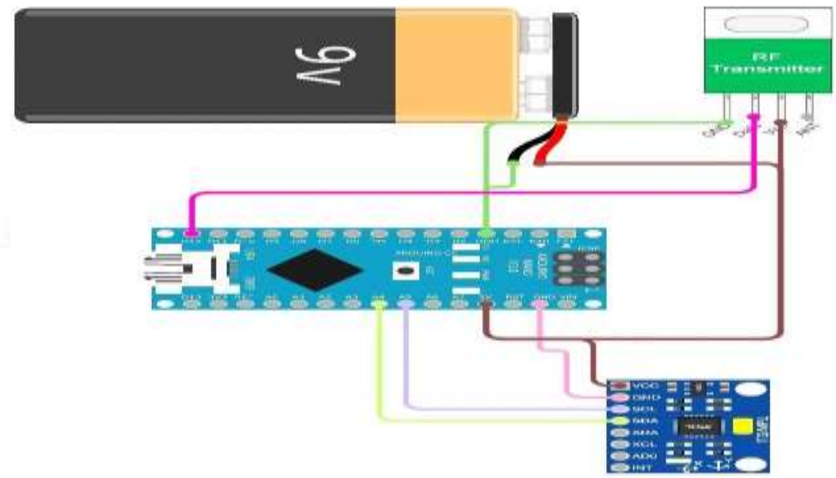

Figure. 14 Circuit Diagram of Remote-Control Unit

1. Key Components and Integration:

- MPU6050 Sensor: The MPU6050 is a six-axis motion tracking device that integrates a three-axis gyroscope and a three-axis accelerometer. It is mounted on the back of the glove to capture precise hand movements. The sensor provides raw acceleration and angular velocity data which are used to detect gestures. The sensor's orientation is calibrated to ensure accurate capture of tilts and movements in the plane of the hand. It uses the I2C communication protocol to send data to the Arduino Nano.
- Arduino Nano: The Arduino Nano, a compact microcontroller board based on the ATmega328P, is mounted at the wrist section of the glove. It is responsible for initializing the MPU6050, reading data via the I2C bus, processing this data to recognize gestures, and then sending the appropriate control signals to the RF transmitter. The Arduino Nano operates at 5V and provides 14 digital I/O pins, of which 6 can be used as PWM outputs. It also has 8 analog input pins, 16 MHz quartz crystal, a USB connection, a power jack, and a reset button.
- 433 MHz RF Transmitter: The RF transmitter is connected to one of the Arduino Nano's digital I/O pins. It sends control signals wirelessly to the wheelchair's RF receiver. The transmitter modulates the digital data using Amplitude Shift Keying (ASK).
- Power Supply (9V Battery): The glove is powered by a 9V battery connected to the VIN and GND pins of the Arduino Nano. This battery also powers the RF transmitter, ensuring the glove operates continuously for extended periods. The battery capacity, typically around 500 mah, determines the operational time, which is sufficient for several hours of continuous use.

2. Block diagram working of Remote-control unit:

- Sensor Data Acquisition: The MPU6050 captures real-time data on the hand's orientation and acceleration. It continuously monitors the acceleration along the X, Y, and Z axes and the angular velocity around these axes. Data acquisition is performed at a high sampling rate to ensure that even quick and subtle movements are accurately detected.

- Data Processing and Command Generation: The Arduino Nano reads the sensor data through the I2C interface. It periodically polls the MPU6050's registers to obtain the latest readings. An embedded algorithm processes the raw data to detect specific gestures. For instance, a forward tilt beyond a certain threshold is interpreted as a command to move forward. The algorithm filters out noise and applies smoothing techniques to ensure that only deliberate gestures are recognized.

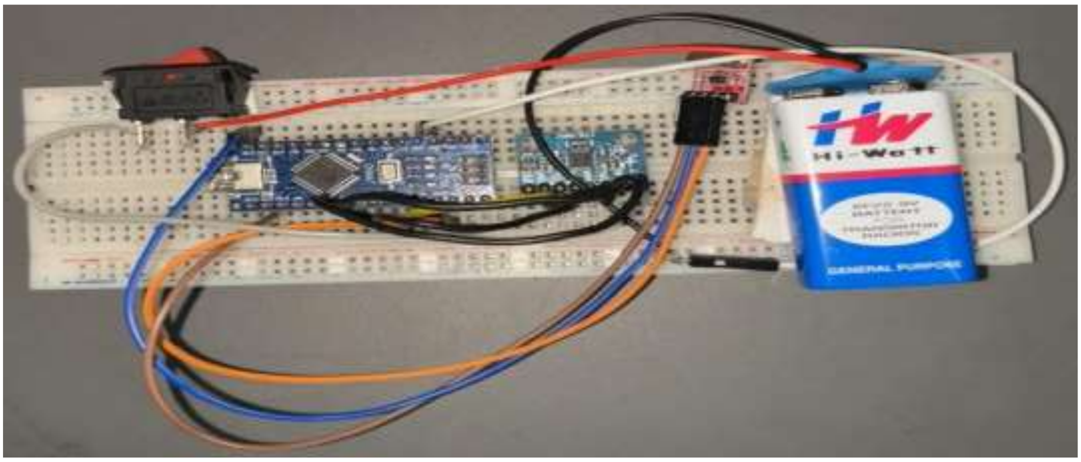

Figure. 15 Remote Control Unit design

- Wireless Data Transmission: The Arduino Nano sends the control command to the 433 MHz RF transmitter. The transmitter modulates this data using ASK before transmitting it. The transmitter converts the digital command into a high-frequency signal, which is then transmitted as a radio wave to the receiver on the wheelchair.

- Arduino Nano: During the initialization phase, the Arduino Nano configures the MPU6050 sensor by setting the appropriate registers. This includes setting the accelerometer and gyroscope ranges, initializing the I2C bus, and performing a self-test. The Arduino Nano enters a continuous loop where it reads data from the MPU6050 at regular intervals. The frequency of this loop is optimized to balance between responsiveness and power efficiency. Data is read from the MPU6050's registers and stored in variables for further processing.

- Gesture Recognition Algorithm: The core of the Arduino's programming is the gesture recognition algorithm. This algorithm analyses the sensor data to detect specific movements such as tilting forward, backward, or sideways.

Movement of motors according to hand gesture		
Direction of hand gesture	Movement of left Motor	Movement of right Motor
Forward	Forward	Forward
Backward	Backward	Backward
Right	Forward	Stop
Left	Stop	Forward

Figure. 16 Motor Movements

- Command Encoding and Transmission: Once a gesture is recognized, the Arduino encodes it into a specific command format. This command is then sent to the RF transmitter. The command format includes start and stop bits, ensuring that the receiver correctly identifies the beginning and end of each command.

- Power Management: The Arduino Nano's power management features are utilized to extend battery life. This includes entering low-power sleep modes during inactivity and waking up periodically to read sensor data. Power consumption is further reduced by optimizing the sensor read frequency and minimizing unnecessary operations.

- 433 MHz RF Transmitter: The RF transmitter uses Amplitude Shift Keying (ASK) to encode the digital command onto a carrier wave. For logic 1, the carrier wave is transmitted; for logic 0, it is not. The modulation process involves turning the carrier signal on and off at the rate of the data bits, creating a waveform that represents the command. An antenna, typically a 17.3 cm wire, is connected to the ANT pin of the transmitter. This wire acts as a dipole antenna, enhancing the transmission range. Proper antenna placement and orientation are crucial for maximizing the transmission range and ensuring reliable communication.

- Power Supply Considerations: The glove's 9V battery powers both the Arduino Nano and the RF transmitter. The Arduino's onboard regulator steps down the 9V to 5V, suitable for the microcontroller and sensor. The battery's capacity (typically around 500 mAh) determines the operational time, which is sufficient for several hours of continuous use. The system employs various power management techniques to extend battery life. This includes using sleep modes for the Arduino Nano and optimizing the sensor read frequency. Efficient coding practices ensure that the microcontroller performs only necessary operations, reducing power consumption.

4.3.2 Wheelchair System

The Wheelchair System receives wireless commands from the remote, processes them using the Arduino Uno, and controls the DC motors through the L298N motor driver, enabling precise and responsive wheelchair movement.

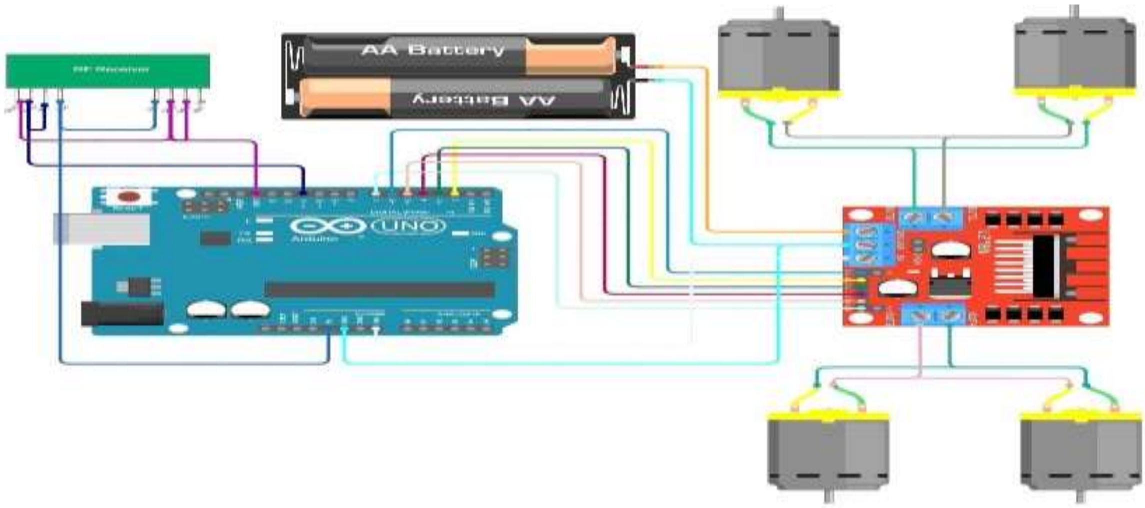

Figure. 17 Block Diagram of Wheelchair System

1. Key Components and Integration:

- 433 MHz RF Receiver: The RF receiver captures the transmitted signals from the glove’s RF transmitter. It demodulates these signals to extract the digital control commands. The receiver's antenna is designed to maximize signal reception, ensuring reliable communication with the glove’s transmitter.

- Arduino Uno: The Arduino Uno decodes the received signals and sends control commands to the motor driver. It processes the commands to generate PWM signals for motor speed control and digital signals for direction control. The microcontroller’s I/O pins are used to interface with the RF receiver and motor driver, ensuring seamless communication and control.

- L298N Motor Driver: The L298N motor driver controls the four DC motors based on the PWM and digital signals from the Arduino Uno. It can handle high current loads, making it suitable for driving the motors. The motor driver receives power from the 12V LiPo batteries, providing the necessary voltage and current for motor operation.

- DC Motors: Each motor is attached to one wheel of the wheelchair, providing independent control of each wheel’s speed and direction. This allows for precise navigation and manoeuvrability. The motors are selected for their high torque and efficiency, ensuring that the wheelchair can handle various terrains and inclines.

- Power Supply: Three 12V LiPo Batteries Connected in parallel, these batteries provide a stable 12V supply with increased capacity, ensuring sufficient power for motor operation. One 9V Battery Powers the Arduino Uno and RF receiver, ensuring stable operation of the control electronics.

2. Block Diagram and Working of Wheelchair System:

- Signal Reception and Processing: The RF receiver captures the wireless control signals transmitted by the remote control unit's RF transmitter. It demodulates these signals to extract the digital control commands. The received commands are then forwarded to the Arduino Uno microcontroller for further processing. The Arduino Uno's digital pins interface with the RF receiver to receive the command signals.

- Command Decoding and Interpretation: The Arduino Uno decodes the received signals to determine the required action. It processes the data to identify the specific command, such as moving forward, turning left, or stopping. The microcontroller’s program includes a state machine that maps each command to specific motor control actions. For example, a forward command activates both motors to move the wheelchair forward, while a left turn command activates one motor to turn left while keeping the other motor stationary.

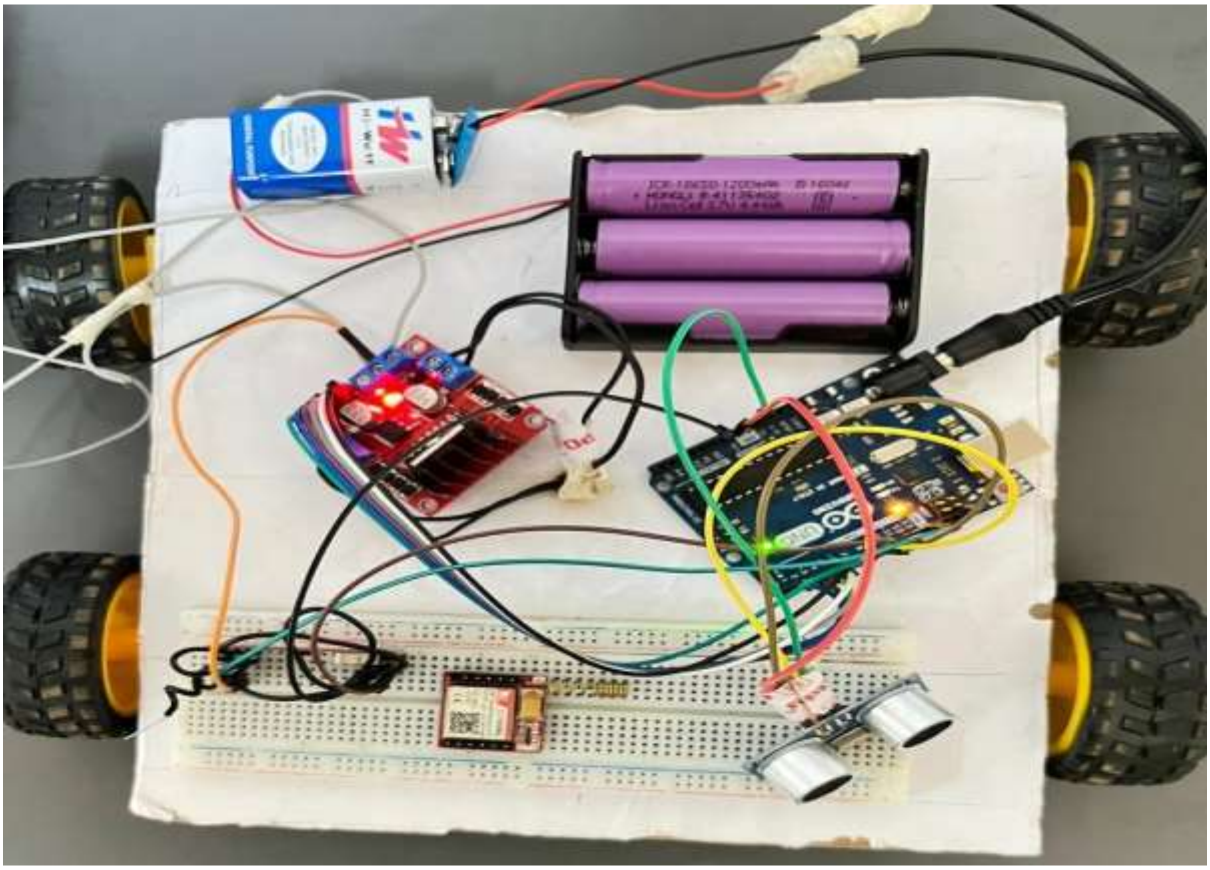

Figure. 18 Wheelchair System Design

- Motor Control Logic: Once the command is decoded, the Arduino Uno generates Pulse Width Modulation (PWM) signals to control the speed of the DC motors. The PWM signals control the duty cycle, determining the motors' rotational speed. Additionally, the microcontroller sends digital signals to the L298N motor driver to control the direction of rotation for each motor. By varying the combination of high and low signals on the motor driver's input pins, the Arduino Uno can control forward, backward, left, and right movements of the wheelchair.
- L298N Motor Driver: The L298N motor driver is a dual H-bridge motor driver integrated circuit (IC) that allows bi-directional control of two DC motors. Each H-bridge consists of four MOSFETs arranged in an H configuration, enabling independent control of motor speed and direction. The motor driver operates on a supply voltage range of 5V to 35V and can deliver a continuous current of up to 2A per channel, with peak currents up to 3A. The L298N motor driver receives control signals from the Arduino Uno microcontroller to control the direction and speed of the DC motors. The control inputs consist of two signals per motor: one for direction control and one for PWM speed control. The direction control signals determine the rotational direction of the motor, while the PWM signals control the motor speed by varying the duty cycle of the PWM waveform.
- The PWM speed control input accepts a PWM signal generated by the Arduino Uno. The duty cycle of the PWM signal determines the effective voltage applied to the motor, thereby controlling its speed. By adjusting the duty cycle of the PWM signal, the Arduino Uno can regulate the speed of the DC motors connected to the L298N motor driver. The direction control inputs determine the rotational direction of the motor. By configuring the direction control pins appropriately, the Arduino Uno can control whether the motor rotates clockwise or counterclockwise. The direction control pins of the L298N motor driver are interfaced with the digital output pins of the Arduino Uno. By toggling these pins HIGH or LOW, the microcontroller can change the motor direction as required.
- DC Motors: The wheelchair is equipped with four high-torque DC motors, with each motor driving one wheel. These motors are selected for their high torque output and efficient performance, making them suitable for wheelchair propulsion. The motors operate on a nominal voltage of 12V, matching the voltage supplied by the battery pack. They are designed to deliver sufficient torque to propel the wheelchair over various terrains and inclines. The DC motors are responsible for propelling the wheelchair forward, backward, and turning left or right. Each motor is independently controlled by the Arduino Uno microcontroller, allowing for precise manoeuvrability. The motors are coupled to the wheelchair wheels via gearboxes, which provide mechanical advantage and help optimize the motor torque output for efficient wheelchair movement.

- Power Supply Considerations: The wheelchair is powered by three 12V Lithium Polymer (LiPo) batteries connected in parallel. This configuration provides a stable 12V supply with increased capacity, ensuring sufficient power for motor operation. The batteries are rechargeable and offer high energy density, making them ideal for powering motorized devices such as wheelchairs. A 9V battery is used to power the Arduino Uno microcontroller and the RF receiver. This battery provides a stable voltage supply for the control electronics, ensuring reliable operation of the wheelchair's control system. The 9V battery is connected to the VIN and GND pins of the Arduino Uno, providing the necessary power for microcontroller operation. The wheelchair system employs various power management techniques to optimize energy usage and extend battery life. This includes implementing efficient motor control algorithms to minimize power consumption during operation. The Arduino Uno microcontroller utilizes low-power sleep.

4.4 Arduino Code Implementation

This section provides the detailed Arduino code used to implement the functionality of the remote-control glove, focusing on both the transmitter (Tx) and receiver (Rx) modules.

4.4.1 Transmitter (Tx) Code

The following Arduino code handles the transmission of control signals from the glove based on detected tilt angles using the MPU6050 sensor.

```
#include <Wire.h>

#include <MPU6050_tockn.h>

#include <VirtualWire.h>

// Define MPU6050 object

MPU6050 mpu6050(Wire);

// Define transmitter pin

#define TRANSMITTER_PIN 12 // Change to the desired pin for transmitter char data[10]; // Buffer
to store the movement direction

// Define PWM duty cycles for different speeds

#define PWM_DUTY_CYCLE_MAX 255   // Maximum duty cycle (full speed) #define
PWM_DUTY_CYCLE_LOW 100   // Low duty cycle for slow speed void setup() {

// Initialize serial communication

Serial.begin(9600);

// Initialize MPU6050 sensor Wire.begin(); mpu6050.begin(); mpu6050.calcGyroOffsets(true);

// Print accelerometer initial values
```

```
Serial.println("MPU6050 Initializing...");

Serial.print("AcX = ");

Serial.print(mpu6050.getAccX());

Serial.print(" | AcY = ");

Serial.print(mpu6050.getAccY());

Serial.print(" | AcZ = ");

Serial.println(mpu6050.getAccZ()); // Initialize VirtualWire library

vw_set_tx_pin(TRANSMITTER_PIN); // Set the transmit pin vw_setup(2000); // Bits per second

} void loop() {

// Read accelerometer data mpu6050.update(); // Get accelerometer values float accX =
mpu6050.getAccX(); float accY = mpu6050.getAccY(); float accZ = mpu6050.getAccZ();

// Calculate tilt angle

float tiltAngleX = atan(accX / sqrt(accY * accY + accZ * accZ)) * 180 / PI; float tiltAngleY =
atan(accY / sqrt(accX * accX + accZ * accZ)) * 180 / PI;

// Print tilt angles to serial monitor

Serial.print("Tilt Angle X: ");

Serial.print(tiltAngleX);

Serial.print(" | Tilt Angle Y: ");

Serial.println(tiltAngleY);

// Send commands based on tilt angles with different speeds if (tiltAngleY > 20) { // Forward with
slow speed

sendCommand('F', PWM_DUTY_CYCLE_LOW);

} else if (tiltAngleY < -20) { // Backward with slow speed

sendCommand('B', PWM_DUTY_CYCLE_LOW);

} else if (tiltAngleX > 20) {

// Right with slow speed

sendCommand('R', PWM_DUTY_CYCLE_LOW);

} else if (tiltAngleX < -20) {

// Left with slow speed

sendCommand('L', PWM_DUTY_CYCLE_LOW);

} else { // Stop

sendCommand('S', 0); // Stop command with zero duty cycle
```

```
}
// Delay to prevent spamming delay(100);
}
// Function to send command wirelessly with specified PWM duty cycle void sendCommand(char command, uint8_t pwm_duty_cycle) {
// Send command vw_send((uint8_t *)&command, 1); vw_wait_tx(); // Wait for transmission to complete // Send PWM duty cycle
vw_send((uint8_t *)&pwm_duty_cycle, 1); vw_wait_tx(); // Wait for transmission to complete
}
```

4.4.2 Receiver (Rx) Code

The following Arduino code handles the reception of control signals and manages the movement of the wheelchair based on the received commands.

```
#include <VirtualWire.h>
#include <Wire.h>
#include <MPU6050_tockn.h>
#include <Ultrasonic.h>
// Define motor control pins
#define MOTOR_ENA 6
#define MOTOR_IN1 5
#define MOTOR_IN2 4
#define MOTOR_ENB 7
#define MOTOR_IN3 3
#define MOTOR_IN4 2
// Define Ultrasonic sensor pins
#define TRIGGER_PIN 8
#define ECHO_PIN 9
#define MAX_DISTANCE 20 // Maximum distance to detect obstacles, adjust as needed
// Define Ultrasonic object
Ultrasonic ultrasonic(TRIGGER_PIN, ECHO_PIN); speedChairA = 500;   // Default speed for motor A int speedChairB = 500;   // Default speed for motor B
// Define MPU6050 object
MPU6050 mpu6050(Wire);
```

```
void setup() {
  // Initialize serial communication
  Serial.begin(9600);
  // Initialize motor control pins
  pinMode(MOTOR_ENA, OUTPUT);   pinMode(MOTOR_IN1, OUTPUT);
pinMode(MOTOR_IN2, OUTPUT);   pinMode(MOTOR_ENB, OUTPUT);
pinMode(MOTOR_IN3, OUTPUT);   pinMode(MOTOR_IN4, OUTPUT);
  // Initialize MPU6050 sensor   Wire.begin();   mpu6050.begin();
mpu6050.calcGyroOffsets(true);   // Initialize VirtualWire library   vw_set_rx_pin(11);   // Set the
RF receiver pin   vw_setup(2000);      // Bits per sec   vw_rx_start();      // Start the receiver
} void loop() {
  // Check for obstacle detection   if (!isObstacleDetected()) {
    // No obstacle detected, proceed with receiving instructions
    uint8_t buf[VW_MAX_MESSAGE_LEN];     uint8_t buflen = VW_MAX_MESSAGE_LEN;

    if (vw_get_message(buf, &buflen)) {      // Process received command      switch (buf[0]) {
      case 'F':
        moveWheelchairForward();
        break;        case 'B':
        moveWheelchairBackward();
        break;        case 'R':
        moveWheelchairRight();
        break;        case 'L':
        moveWheelchairLeft();
        break;        case 'S':
        stopWheelchair();          break;
      }
    }
  } else {
    // Obstacle detected, stop wheelchair     stopWheelchair();
  }
  // Display wheelchair status in serial monitor   displayWheelchairStatus();
```

```
} bool isObstacleDetected() {   // Read distance from ultrasonic sensor   int distance =
ultrasonic.read();   Serial.println(distance);    if (distance > 0 && distance <= MAX_DISTANCE)
{     Serial.println("Obstacle detected within the specified range");

 // Obstacle detected within the specified range     return true;

  } else {

    Serial.println("No obstacle detected or out of range");

    // No obstacle detected or out of range     return false;

  } }   void stopWheelchair() {   digitalWrite(MOTOR_ENA, LOW);
digitalWrite(MOTOR_ENB, LOW);

}   void moveWheelchairForward() {   digitalWrite(MOTOR_IN1, HIGH);
digitalWrite(MOTOR_IN2, LOW);   digitalWrite(MOTOR_IN3, HIGH);
digitalWrite(MOTOR_IN4, LOW);   analogWrite(MOTOR_ENA, speedChairA);
analogWrite(MOTOR_ENB, speedChairB);

}  void moveWheelchairBackward() {   digitalWrite(MOTOR_IN1, LOW);
digitalWrite(MOTOR_IN2, HIGH);   digitalWrite(MOTOR_IN3, LOW);
digitalWrite(MOTOR_IN4, HIGH);   analogWrite(MOTOR_ENA, speedChairA);
analogWrite(MOTOR_ENB, speedChairB); }  void moveWheelchairRight() {
digitalWrite(MOTOR_IN1, LOW);   digitalWrite(MOTOR_IN2, HIGH);
digitalWrite(MOTOR_IN3, HIGH);   digitalWrite(MOTOR_IN4, LOW);
analogWrite(MOTOR_ENA, speedChairA);   analogWrite(MOTOR_ENB, 0);

}

void moveWheelchairLeft() {   digitalWrite(MOTOR_IN1, HIGH);   digitalWrite(MOTOR_IN2,
LOW);   digitalWrite(MOTOR_IN3, LOW);   digitalWrite(MOTOR_IN4, HIGH);
analogWrite(MOTOR_ENA, 0);   analogWrite(MOTOR_ENB, speedChairB);

}

void displayWheelchairStatus() {

  Serial.println("Wheelchair is running.");

}
```

Summary: This chapter delves into the development and implementation of a hand gesture controlled wheelchair system, highlighting the innovative integration of gesture recognition technology with traditional wheelchair mechanics. The next chapter will discuss the experimental results in detail.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Results

In This Chapter we present the results achieved through the development and testing of the 433MHz ASK transceiver, demonstrating significant enhancements in communication efficiency, range, and reliability for assistive mobility technologies. These improvements have direct implications for the performance and user experience of assistive devices, showcasing the potential of our project in advancing the field of assistive mobility.

5.1.1 Remote Control Glove: Design and Implementation

Figure. 19 Aseemled Remote Control Unit

Conceptualization and Design:

The Hand Gesture-Controlled Wheelchair project began with a clear vision: to create a userfriendly, intuitive interface that allows individuals with mobility impairments to control a wheelchair using simple hand gestures. The cornerstone of this system is the remote-control glove, which integrates essential components such as the Arduino Nano microcontroller and the MPU 6050 accelerometer and gyroscope sensor.

The initial design phase involved detailed sketches and prototyping to ensure the glove was both comfortable and functional. The Arduino Nano was selected for its compact size and capability to process the gesture recognition algorithms required for this project. The MPU 6050 sensor was chosen for its high accuracy in capturing hand movements, including acceleration and angular velocity data.

1.Embedded Hardware Integration: Embedding the hardware components within the glove required meticulous planning. The Arduino Nano and MPU 6050 were carefully positioned to maintain the glove's flexibility and comfort. The wires connecting these components were routed through the glove's fabric in a way that minimized bulk and maximized durability. To ensure ease of use, connectors were placed in accessible locations, allowing for straightforward maintenance and updates. This thoughtful integration ensured that the glove remained lightweight and did not hinder the user's natural hand movements.

2.Gesture Recognition and Calibration: The gesture recognition system relied on data from the MPU 6050 sensor. The raw accelerometer and gyroscope data were processed by the Arduino Nano, which interpreted specific hand movements into commands. Gestures such as swipes, tilts, and rotations were identified through predefined thresholds and patterns in the sensor data. Calibration was a critical step, involving users performing a series of gestures to establish baseline readings. This process was essential to accommodate individual differences in hand sizes and movement styles, ensuring accurate and consistent gesture recognition.

3.Real-Time Feedback Mechanisms: To enhance user interaction, the glove was equipped with visual and haptic feedback mechanisms. LEDs embedded in the glove provided visual confirmation of recognized gestures, while small vibrating motors delivered tactile feedback. These real-time feedback mechanisms were crucial in building user confidence and ensuring successful command execution.

5.1.2 Wheelchair System: Integration and Performance

1.Motor Control and Propulsion: The wheelchair's propulsion system was powered by four high-torque DC motors controlled by an L298N motor driver. This setup provided the necessary power and control to navigate various terrains and inclines. The motor driver was selected for its reliability and ease of integration with the Arduino Nano.

Extensive testing ensured that the wheelchair responded promptly to commands from the glove. The motors provided smooth acceleration and deceleration, allowing for precise control of the wheelchair's movements. This responsiveness was vital for user confidence and safety.

Figure. 20 Assembled Wheelchair System

2.Obstacle Detection: The HC-SR04 ultrasonic sensor was a key component in the wheelchair's obstacle detection system. Mounted on the front of the wheelchair, the sensor emitted ultrasonic waves and measured the time taken for the waves to bounce back from nearby objects. This data was processed to determine the distance to obstacles, allowing the wheelchair to adjust its path accordingly. During testing, the wheelchair was subjected to various scenarios, including navigating through cluttered rooms and narrow corridors. The system demonstrated robust performance, effectively detecting and avoiding obstacles to prevent collisions.

3.Fall Detection Mechanism: To ensure user safety, the wheelchair was equipped with a fall detection system utilizing the MPU 6050 accelerometer. The sensor continuously monitored the wheelchair's orientation, detecting sudden changes indicative of a fall or tilt. When such an event was detected, the system triggered an audible buzzer alarm and halted the wheelchair to prevent further movement. Testing involved simulating falls and abrupt drops to verify the system's reliability. The fall detection mechanism proved effective, quickly identifying potential falls and responding appropriately to safeguard the user.

5.1.3 User Testing and Feedback

1.Training Sessions: Users participated in comprehensive training sessions designed to familiarize them with the glove's interface and the gesture-based control system. These sessions included step-by-step instructions, hands-on practice, and real-time feedback to help users master the gestures. User feedback from these sessions was invaluable in identifying areas for improvement. Suggestions included adjustments to gesture sensitivity and enhancements to the feedback mechanisms, which were incorporated into subsequent iterations of the system.

2.Usability Testing: Usability testing involved navigating through simulated environments using the glove interface. Metrics such as task completion time, error rates, and user satisfaction scores were collected and analysed. These metrics provided insights into the system's performance and highlighted areas for further refinement. The tests demonstrated that users could effectively control the wheelchair using the glove, with high levels of accuracy and low error rates. Users reported increased confidence and satisfaction with the gesture-based control system.

3.Comfort and Ergonomics: Feedback on the glove's comfort and ergonomics was critical in guiding design improvements. Users reported on factors such as the glove's breathability, flexibility, and overall comfort during extended use. Iterative design changes included adjustments to the glove's material, sensor placement, and fit to enhance comfort and usability.

4.Gesture Accuracy and Precision: The system's accuracy in recognizing gestures was evaluated through rigorous testing. Users performed a series of gestures, and the system's ability to accurately interpret these gestures was measured. Adjustments to the gesture recognition algorithm and sensor calibration improved accuracy and minimized errors, ensuring reliable operation in various conditions.

5.2 Discussion

5.2.1 Key Findings

The Hand Gesture-Controlled Wheelchair project successfully demonstrated the feasibility and effectiveness of using hand gestures for wheelchair control. Key findings include:

The glove-based interface of the wheelchair system proved to be user-friendly, with users quickly adapting to its intuitive gesture-based control system. The ultrasonic sensor provided reliable obstacle detection, enabling the wheelchair to navigate safely through various environments. Additionally, the accelerometer-based fall detection system effectively identified potential falls, triggering alarms and halting the wheelchair to ensure user safety. Users reported high levels of satisfaction with the system, noting significant improvements in mobility and independence.

5.2.2 Challenges Encountered and Strategies to Overcome Them

1.Gesture Variability: One of the primary challenges was accommodating the variability in hand gestures. Users exhibited different speeds, amplitudes, and styles in their movements, making it challenging to develop a one-size-fits-all gesture recognition algorithm. Strategy: To address this, the system incorporated a robust calibration procedure that tailored the gesture recognition to each user's specific motion range. This customization ensured accurate and consistent gesture interpretation, enhancing the overall user experience.

2.Environmental Sensitivity: The system's sensitivity to environmental factors, such as ambient lighting and interference from other devices, posed challenges. Ensuring reliable performance in different conditions required rigorous testing and refinement. Strategy: Advanced filtering techniques were implemented in the sensor algorithms to reduce the impact of environmental noise and interference.

3.User Adaptation: Some users initially struggled to adapt to the gesture-based control system. Addressing this required designing intuitive training programs and providing ample practice opportunities. Strategy: Comprehensive user training programs were developed, including guided practice sessions, instructional materials, and real-time feedback.

Summary: This chapter presents the results and discussion of the Hand Gesture-Controlled Wheelchair project, detailing the design, implementation, and testing of the remote-control glove and wheelchair system. Key findings include successful gesture recognition, reliable obstacle detection, and effective fall detection, with positive user feedback. The next chapter will describe the conclusions of the study and potential future enhancements for the system.

CHAPTER 6

CONCLUSION AND SCOPE FOR FUTURE WORK

6.1 Conclusion

The Hand Gesture-Controlled Wheelchair project represents a significant advancement in assistive technology, offering a more intuitive and accessible means of mobility for individuals with disabilities. By leveraging readily available hardware components such as the Arduino microcontrollers, MPU-6050 accelerometer and gyroscope sensor, HC-SR04 ultrasonic sensor, and integrating them with a glove-based gesture recognition system, the project successfully demonstrated the feasibility and effectiveness of gesture-based wheelchair control.

The Model developed in this project was rigorously tested to ensure that it met the key objectives of reliability, user-friendliness, and safety. The remote-control glove allowed users to perform a range of gestures, which were accurately interpreted by the Arduino Nano and transmitted to the wheelchair's Arduino Uno for execution. The L298N motor driver ensured precise control of the wheelchair's movements, translating gestures into smooth and responsive navigation.

Obstacle detection was a critical component of the system, achieved through the integration of the HC-SR04 ultrasonic sensor. This sensor effectively scanned the environment for obstacles, allowing the system to dynamically adjust its path and avoid collisions. The fall detection mechanism, utilizing the MPU-6050 sensor, monitored the wheelchair's orientation and triggered an alarm in case of sudden tilts or drops, ensuring user safety.

User feedback was overwhelmingly positive, highlighting the system's ease of use and the increased sense of independence it provided. The successful implementation of this project underscores the potential of gesture-controlled interfaces in enhancing the quality of life for individuals with mobility impairments.

6.2 Scope for Future Work

1.Integration of Advanced Sensors: Depth Cameras and LiDAR for Enhanced Navigation The current system uses the HC-SR04 ultrasonic sensor for obstacle detection, which, while effective, has limitations in range and resolution. Integrating advanced sensors such as depth cameras (e.g., Microsoft Kinect) or Light Detection and Ranging (LiDAR) systems could significantly enhance the wheelchair's ability to map and navigate complex environments. Depth cameras can provide a detailed three-dimensional view of the surroundings, enabling the system to detect obstacles at varying heights and distances with greater accuracy. LiDAR, commonly used in autonomous vehicles, can generate precise, high-resolution maps of the environment, allowing for more sophisticated path planning and collision avoidance. Real-World Application: Incorporating these advanced sensors would enable the wheelchair to operate more effectively in diverse environments, such as crowded urban areas, uneven terrain, and indoor settings with tight spaces. The enhanced environmental awareness provided by these sensors could lead to more fluid and natural navigation, improving the user's overall experience.

2.Enhanced Gesture Recognition

Complex Gesture and Multi-Modal Input Integration: The current prototype relies on basic hand gestures for control. Future work could expand the gesture library to include more complex gestures and combinations of gestures for a broader range of commands. Additionally, integrating multi-modal inputs, such as voice commands and facial expressions, could provide a more versatile and inclusive interface. For instance, users with limited hand mobility could use voice commands in conjunction with gestures to control the wheelchair.

Technological Implementation: Implementing this enhancement would involve upgrading the gesture recognition hardware and software to process multiple input types simultaneously. Advanced microcontrollers or single-board computers, such as the Raspberry Pi, could be employed to handle the increased computational load. This multi-modal approach would make the system more adaptable to the diverse needs and preferences of users.

3.Improved User Interface

Comprehensive and Customizable User Interface: A more sophisticated user interface (UI) could greatly enhance the user experience. Developing a companion smartphone app or a dedicated touch screen interface on the wheelchair could offer users real-time feedback, customization options, and system diagnostics. Features could include battery life monitoring, system health checks, and the ability to adjust gesture sensitivity settings.

User-Centric Design: This UI would allow users to personalize their interaction with the wheelchair, making it more intuitive and responsive to their specific needs. For example, the app could provide tutorials and visual feedback to help users learn and refine their gestures. Implementing a user-friendly UI would require collaboration with UI/UX designers to ensure accessibility and ease of use for individuals with varying levels of technical proficiency.

4.Machine Learning for Adaptive Control

Personalized Gesture Recognition through Machine Learning: Although this project did not use machine learning (ML) algorithms, incorporating ML could significantly enhance the system's adaptability and accuracy. Machine learning algorithms could be trained to recognize individual user patterns and preferences, continually improving the system's responsiveness over time. By learning from each user's unique movements, the wheelchair could adjust its gesture recognition parameters to optimize performance.

Adaptive Learning Models: Developing adaptive learning models would involve collecting data from users over time and using this data to train the ML algorithms. Techniques such as supervised learning, where the system learns from labelled gesture data, and reinforcement learning, where the system receives feedback on its performance, could be employed. This approach would result in a highly personalized and efficient control system, further empowering users.

5.Enhanced Safety Features

Sophisticated Fall Detection and Emergency Systems: Safety is paramount for assistive devices like wheelchairs. The current system includes basic fall detection and obstacle avoidance features. Future enhancements could include more sophisticated fall detection algorithms that can distinguish between minor bumps and significant falls. Additionally, integrating automatic emergency braking systems would provide an extra layer of safety.

Emergency Response Integration: In the event of a detected fall or collision, the system could be designed to automatically alert caregivers or emergency services. This could be achieved through a connected app that sends real-time notifications or by integrating the wheelchair with smart home systems. Implementing these features would involve developing more advanced sensors and algorithms, as well as ensuring reliable communication channels for emergency alerts.

Summary: This chapter describes the conclusions that were reached after a detailed and comprehensive study and experimentation. The future scope of the study has also been explained in brief.

References: -

[1] Bourhis, G., Moumen, K., Pino, P., Rohmer, S., & Pruski, A. (2002). Assisted navigation for a powered wheelchair. IEEE Xplore. Retrieved November 4, 2022, from https://ieeexplore.ieee.org/document/385072.

[2] Mahipal Manda, B Shankar Babu, M Abhishek, & J Srikanth. (2018, September). HAND GESTURES CONTROLLED WHEELCHAIR. Retrieved November 4, 2022, from https://www.jetir.org/papers/JETIRE006094.pdf.

[3] Kalantri, R. A., & Chitre, D. K. (2013, August). Automatic Wheelchair using Gesture Recognition. www.ijeat.org. Retrieved November 4, 2022, from https://www.ijeat.org/wpcontent/uploads/papers/v2i6/F1907082613.pdf.

[4] Megalingam, R. K., Sreekanth, S., A, G., Teja, C. R., & Raj, A. (2017, January). Wireless gesture-controlled wheelchair. IEEE Xplore. Retrieved November 4, 2022, from https://ieeexplore.ieee.org/document/8014621.

[5] Gangopadhyay, S., Mukherjee, S., & Chatterjee, S. (2013, September). Intelligent gesture controlled wireless wheelchair for the physically handicapped - IRAJ. Retrieved October 9,

2022, from http://iraj.in/journal/journal_file/journal_pdf/1-7-139029269940-45.pdf.

[6] Mahmood, M., Rizwan, M. F., Sultana, M., & Imam, M. H. (2020, June 7). Design of a low-cost hand gesture controlled automated wheelchair. IEEE Xplore. Retrieved November 4,

2022, from https://ieeexplore.ieee.org/document/9230849.

[7] Francis, J., & Kadan, A. B. (2014, August). Significance of hand gesture recognition systems in vehicular automation-A survey. IJCA. Retrieved November 4, 2022, from https://www.ijcaonline.org/archives/volume99/number7/17389-7931.

[8] John, F., Shaju, J., Mathai, M. K., & Seby, M. (2019, November 3). Gesture controlled wheelchair. Journal of Embedded Systems and Processing. Retrieved November 4, 2022, from http://matjournals.in/index.php/JoESP/article/view/4419.

[9] Noman, A. T., Rashid, H., Khan, M. S., & Islam, M. E. (2018, October). A new design approach for gesture controlled smart wheelchair utilizing microcontroller. IEEE Xplore.

Retrieved November 4, 2022, from https://ieeexplore.ieee.org/document/8745607.