AUTOMATIC WHEELCHAIR USING EYEBALL CONTROL

PROJECT REPORT

Submitted by

KALYANI S	**(LIDK20EE058)**
SHIBIN K GEORGE	**(LIDK20EE067)**
SNEHA ET	**(LIDK20EE068)**
SUMAYYA PAREED	**(LIDK20EE070)**

to

The APJ Abdul Kalam Technological University

In Partial Fulfillment of the Requirements for the award of the degree

of

Bachelor Of Technology

in

Electrical And Electronics Engineering

Department of Electrical and Electronics Engineering

Government Engineering College, Idukki

May 2024

Institute Vision

To be a premier technical institution imparting knowledge, to carve technically competent and research minded professionals with social responsibilities.

Institute Mission

1. Facilitate quality engineering education through state-of-the-art facilities and qualified vibrant teachers.
2. Transform students to responsible professionals with ethical and social values capable of providing innovative solutions to the problems faced by the country and to enhance the quality of life of the people
3. Accomplish a conducive learning environment to equip students for higher education and life-long learning.
4. To instil managerial skill and entrepreneurial capabilities.

Department Vision

To impart state-of-the-art technological education and facilitate students pursue excellence in electrical and electronics engineering and research through ingenuity, innovations, and teamwork.

Department Mission

1. Ensure quality technical education to students in electrical and electronics engineering through state-of-the-art facilities and well qualified teachers, encouraging research and innovation.
2. Equip students to be professionally competent and ethically compliant in order to accomplish self-development through higher education and provide service to the industry and society.
3. Encourage lifelong learning and team-oriented problem solving through modern tools. and cutting-edge technology.

AUTOMATIC WHEELCHAIR USING EYEBALL CONTROL

PROJECT REPORT

Submitted By

KALYANI S	**(LIDK20EE058)**
SHIBIN K GEORGE	**(LIDK20EE067)**
SNEHA ET	**(LIDK20EE068)**
SUMAYYA PAREED	**(LIDK20EE070)**

to

The APJ Abdul Kalam Technological University

In partial fulfillment of the requirements for the award of the degree

of

Bachelor of Technology

in

Electrical And Electronics Engineering

Department of Electrical and Electronics Engineering

Government engineering college idukki

May 2024

DECLARATION

We undersigned hereby declare that the Project report (**"AUTOMATIC WHEELCHAIR USING EYEBALL CONTROL**"), submitted for partial fulfilment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under the supervision of Dr.Arun Kishore W C Head of the department, Government Engineering College, Idukki. This submission represents ideas in my own words and where ideas or words of others have been included; I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

PAINAVU

03/05/2024

KALYANI S (LIDK20EE058)

SHIBIN K GEORGE(LIDK20EE067)

SNEHA ET (LIDK20EE068)

SUMAYYA PAREED (LIDK20EE070)

Department of Electrical and Electronics Engineering

Government engineering college, Idukki

May 2024

CERTIFICATE

This is to certify that the project report entitled **"Automatic Wheelchair Using Eyeball Control"** submitted by **KALYANI S (LIDK20EE058), SHIBIN K GEORGE (LIDK20EE067), SNEHA ET (LIDK20EE068), SUMAYYA PAREED (LIDK20EE070)**, to the APJ Abdul Kalam Technological university in partial fulfillment of the Beach degree in Electrical & Electronics Engineering is a Bonafede record of the project presentation under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Dr. Arun Kishore W C	Dr. Binoj Kumar A C	Dr. Sunil Kumar P R
Project Guide	Project Coordinator	Project Coordinator
Head of the department	Professor	Professor

Dr.Joseph K D	Dr. Shanifa Beevi S
Project Coordinator	Project Coordinator
Associate Professor	Associate Professor

ACKNOWLEDGEMENT

First of all, we are indebted to the god almighty for giving me an opportunity to excel in our effort to complete this project on time.

We would like to extend gratitude to our guide for the project **Dr. Arun Kishore WC,** Head of the Department for all the guidelines and support for the successful completion of the project .

Our heartfelt thanks to the project coordinators **Dr. Joseph K D**, Associate Professor

Dr. Binoj Kumar A C, Professor, **Dr. Shanifa Beevi S** Associate professor, and **Dr. Sunil Kumar P R** professor, for their valuable suggestions and guidance in the preparation of the project report.

I sincerely thank **Dr. Suresh K**,the principal of Government Engineering College, Idukki, for providing me with the best facilities and atmosphere for the partial fulfilment of the project.

We express our thanks to all staff members and friends for all the help and co-ordination extended in bringing out this project successfully in time.

We will be failing in duty if we do not acknowledge with grateful thanks to the authors of the references and other literature referred to in this project till now. Last but not the least; we are very much thankful to our parents who guided us in every step which we took.

KALYANI S (LIDK20EE058)
SHIBIN K GEORGE (LIDK20EE067)
SNEHA E T (LIDK20EE068)
SUMAYYA PAREED (LIDK20EE070)

ABSTRACT

Paralysis is one of the biggest curses to mankind. In worst case paralysis the person could move only his eyes. The head movement or voice-based wheelchairs will not hold good in that situation. So, an eyeball movement-based wheelchair would help the best for those people. This would be more accurate when compared to other automated wheelchairs. A method for eyeball localization is proposed for controlling wheelchair. An algorithm is furnished with various processing steps and develops an efficient system to reduce both the cost and the computational complexity. Primary goal was to detect eyes in real-time and to keep track on it. The idea is to create an Eye Monitored System which allows movement of the patient ‘s wheelchair depending on the eye movements. A patient looks directly at the camera mounted on a head gear and is able to move in a direction just by looking in that direction.

CONTENTS

LIST OF FIGURES

LIST OF TABLE

CHAPTER 1
INTRODUCTION

Several Wheelchair Systems have been made for disabled persons. Some of the wheelchair systems present till now are discussed below. Hand Gesture based Wheelchair System uses MEMS Accelerometer Sensor which is attached to the hand. Then based on hand gesture, the Wheelchair system is controlled. Voice Operated Wheelchair System uses the voice of the user to operate Wheelchair. Head and Finger based Automated Wheelchair System uses Accelerometer and Flex Sensor to operate Wheelchair. But all the above systems require much of a human effort and none of these systems help people suffering from Quadriplegia. In Quadriplegia, Paralysis is of the extreme level in which a person can only move his eyes. In order to help such disabled persons, Eye Movement based Electronic Wheelchair System using MATLAB came into existence.

1.1 MOTIVATION

Designing an automatic wheelchair with eyeball control can be a transformative and empowering solution for individuals with mobility challenges. The motivation behind such a technology is rooted in enhancing independence, improving accessibility, and providing a more intuitive means of mobility for users And Help people with total paralysis to move without any muscle stress, making their life easier

Individuals with severe mobility impairments often rely on others for assistance in navigating their surroundings. An eyeball-controlled wheelchair can empower users to move independently, reducing dependence on caregivers and promoting a sense of autonomy.

Mobility is closely linked to the overall quality of life. Enabling individuals to control their wheelchairs effortlessly using eye movements can significantly enhance their daily experiences, allowing them to engage more actively in various activities and social interactions. Eyeball control can potentially enable faster and more efficient wheelchair navigation. Users can simply look in the direction they want to go, streamlining the control process and allowing for quicker response times.

1.2 OBJECTIVES

Develop a eye tracking system capable of accurately capturing and interpreting user eye movements in real-time. Integrate sensors and computer vision capabilities to enable the wheelchair to detect obstacles in its path and autonomously navigate around them and user-friendly interface that allows users to calibrate, customize, and control the wheelchair using their eye movements. Implement safety mechanisms, such as emergency braking and collision avoidance, to enhance user safety during operation.

1.3 PROBLEM DEFINITION

The project aims to design and implement an innovative automatic wheelchair system that utilizes eye tracking technology for control. Individuals with severe physical disabilities often face challenges in manoeuvring conventional wheelchairs, making it essential to explore alternative control mechanisms. The proposed system leverages eye tracking to provide a reliable and intuitive method for users to navigate the wheelchair.

1.4 METHODOLOGY

A head mount camera detects the eye movement and wheelchair is moved accordingly. The head mount camera is connected to the laptop where a continuously running and script processes the image and gives command to the microcontroller to control the wheels of a Wheelchair. This system came as a boon for such people. But the constraint was that you had to carry your laptop every time along with the Wheelchair System. That was bulky and costly. To remove the bulkiness and costliness of the Eye Movement based Electronic Wheelchair System, which uses MATLAB, people came up with ideas of using Raspberry Pi to control the whole Wheelchair System. Since Raspberry Pi has its own OS and it is easily portable, people switched to using Raspberry Pi based Wheelchair System. Although in the existing Raspberry Pi based Wheelchair System, latency (delay in response) is the biggest issue. Hence we have come up with a system that uses efficient algorithms for image processing using OpenCV and reduces the latency as much as possible. OpenCV processes the eye and by applying the two algorithms (Centroid and Threshold), movement of wheelchair is initiated. Python is used for programming the Raspberry Pi.

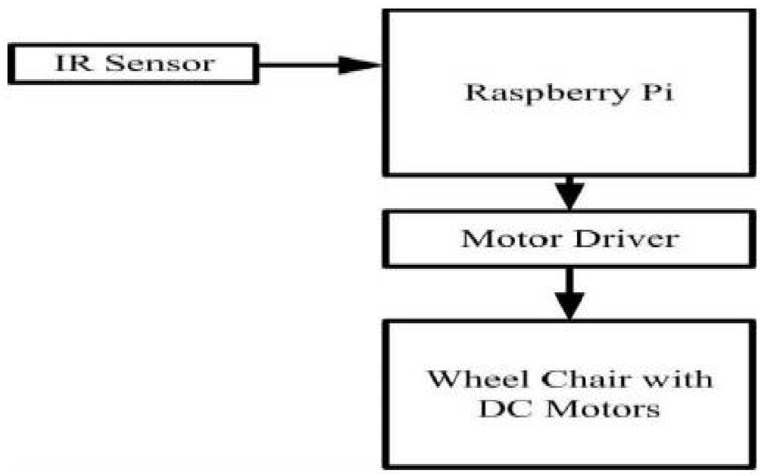

Fig 1: Diagram of Eyeball Movement Based Wheelchair

The project based on eye ball controlled wheelchair for fully paralyzed people Literature survey based on other types of wheelchair. But the all existing wheelchair requires need human efforts for their mobility Designed the required motor for the wheelchair Various components required in our project and their brief explanation about the components and their specifications Block diagram explanation and we can understand what we are doing in the project The image captured by webcam used to capture real time image of person's Eye and send to open CV Wheelchair movement controlled by Arduino nano and it's simulation on the Proteus software and their programming

CHAPTER 2
LITERATURE SURVEY

[1]. In this study, researchers explore the replacement of traditional joystick control in electric wheelchairs with a Human-Computer Interface (HCI) system. Utilizing fuzzy set theory, the system enables gaze detection and recognition of unknown environments. Laser Range Finder (LRF) technology is integrated to calculate the wheelchair's direction and speed based on the user's eye gaze. Consequently, the wheelchair autonomously moves in the direction of the user's gaze while avoiding obstacles in real time within unfamiliar surroundings.

[2]. This paper investigates a wheelchair control system entirely driven by eye movements and blinks, employing deep convolutional neural network (CNN) learning for classification. By capturing images of either eye, divided into four frames, the system determines the direction of motion through a classifier. Movement initiation requires the user to sustain gaze in a particular direction for a set duration, with wheelchair motion ceasing upon detection of closed eye lids for 16 consecutive frames. Convolutional neural networks process the input images to provide directional information for the wheelchair's movement.

[3]. A novel approach is presented where an electric wheelchair's manual control module is substituted with an eye tracker system, specifically Tobii eye tracker. Tobii X120 eye tracker collects eye movement signals and translates them into coordinates of gazing points. To enhance accuracy and stabilize gaze fluctuations, a Kalman filter algorithm is implemented. The system determines 2-Dimensional coordinates of the user's eye gaze on a screen, converting this information into wheelchair movement control signals.

[4]. Researchers devise a smart eye tracking system tailored for individuals with disabilities and elderly persons. By mounting a camera on the user's head, the system captures eye images for

digital image processing using open-source Computer Vision (CV) techniques. Raspberry Pi microcontroller processes the images, initiating wheelchair movement based on eye movements and blinks. Additionally, servo motors facilitate 2-Dimensional wheelchair movement and obstacle avoidance.

[5]. This study introduces a microcontroller-based wheelchair control system receiving signals from relays via serial communication. The microcontroller then transmits appropriate signals to an L293D motor driver, initiating movement in the desired direction without delay. Infrared sensors detect obstacles, prompting the microcontroller to halt movement or adjust direction accordingly.

[6]. To aid individuals with conditions like SMA and Cerebral Palsy, a simplified robotic arm with four degrees of freedom is proposed for wheelchair-mounted operation. Utilizing visual serving and computer vision algorithms, the arm assists in daily tasks by detecting and reaching selected objects. A Human-Machine Interface (HMI) based on a touch screen minimizes user interactions, enhancing ease of control.

[7]. In this paper, an Electrically Powered Wheelchair (EPW) is controlled via an eye tracker system integrated with a fuzzy logic controller. Eye region extraction using Viola and Jones technique facilitates pupil position detection. Adaptive thresholding converts captured images to binary, determining eye movement direction. A Mamdani-type fuzzy logic controller processes pupil position information, dictating wheelchair movement.

[8]. A power assisting control system incorporating monocular fish-eye vision for electric wheelchairs is proposed for highly disabled individuals. A fish-eye camera captures eye pupil images, with optical flow vector calculations determining required travel distance. Implemented on a robotic wheelchair, this system facilitates movement between specific areas based on the optical flow vector obtained from the floor area in the fisheye image.

[9]. This paper evaluates the performance of an intelligent wheelchair motion control system utilizing eyeball movement signals acquired via electrooculography (EOG). Signal conditioning and processing enable fuzzy logic control, dividing into two categories: fuzzy classifier for direction control and fuzzy-proportional derivative for angular speed modulation. This comprehensive approach ensures precise and adaptable wheelchair movement based on user eye movements.

CHAPTER 3

WHEELCHAIR ARCHITECTURE

In the initial stage of designing, a block diagram of the wheelchair is created, with the aim of addressing the challenges faced by fully paralyzed individuals who can only control movement through eye motion or individuals whose paralysis limits movement beyond their head. In our design, the wheelchair is operated by tracking the movement of the eye ball. This is achieved through image processing utilizing the OpenCV library, which enables real-time monitoring of the pupil. Import serial and MediaPipe libraries are used to translate the movement of the pupil into actionable commands. This approach ensures that individuals with severe physical limitations can navigate their daily lives with greater independence and ease using the wheelchair.

3.1 BLOCK DIAGRAM

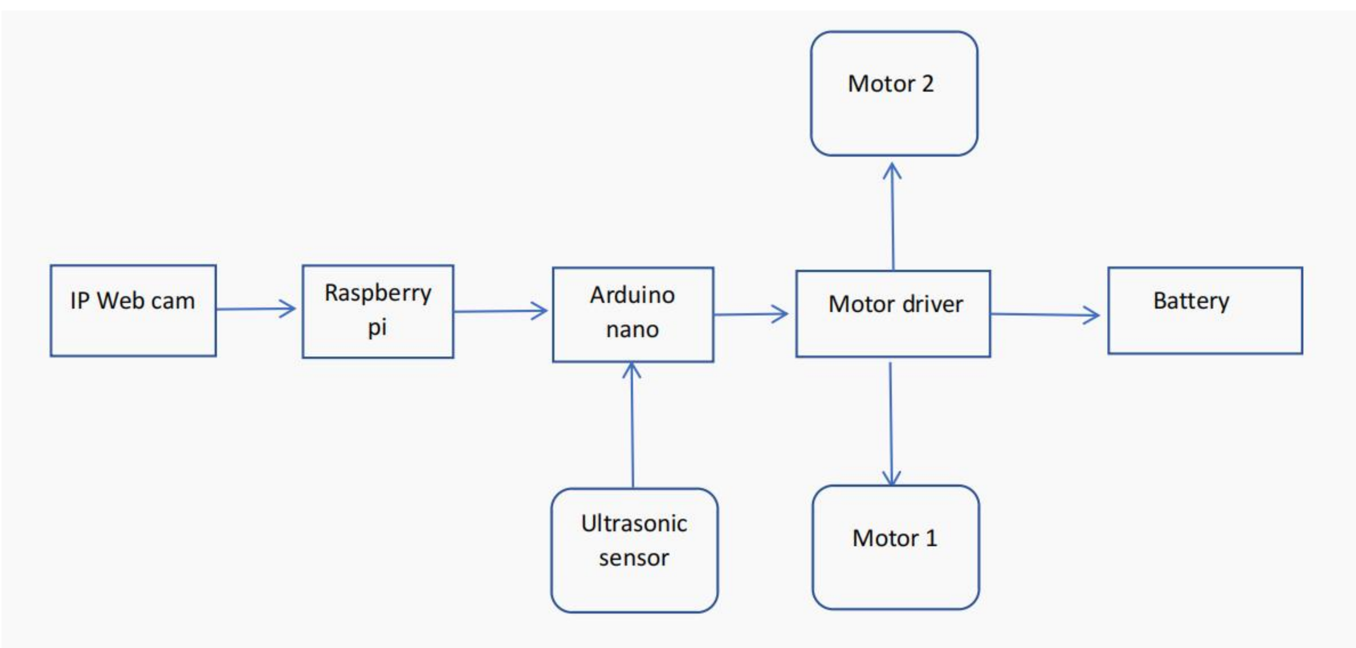

Fig 2: Block Diagram Working of a Wheelchair

In this system, an IP Webcam is utilized for real-time monitoring of eye movement. The webcam is connected to a Raspberry Pi via internet or Wi-Fi. The Raspberry Pi is programmed with OpenCV and Media Pipe, along with imported serial libraries, enabling real-time eye movement monitoring and command generation. Commands generated by the Raspberry Pi are transmitted to an Arduino Nano via USB connection. PWM signals generated by the Arduino Nano, corresponding to the commands received, are then sent to motor drivers, with one driver allocated for each motor. This arrangement allows for the control of both the direction and movement of the wheelchair. The PWM signals regulate the motor drivers by adjusting the input power from the battery. Additionally, speed control is facilitated by PWM signals from the Arduino Nano. Furthermore, an ultrasonic sensor connected to Arduino Nano serves for obstructive detection within the system.

3.2 IMAGE PROCESSING WITH OPENCV

Visual information is the most important type of information perceived, processed and interpreted by the human brain. Image processing is a method to perform some operations on an image, in order to extract some useful information from it. An image is nothing more than a two-dimensional matrix (3-D in case of coloured images) which is defined by the mathematical function $f(x, y)$where x and y are the two co-ordinates horizontally and vertically. The value of $f(x, y)$ at any point is gives the pixel value at that point of an image, the pixel value describes how bright that pixel is, and/or what colour it should be. For grayscale images the pixel value is a single number that represents the brightness of that pixel, the most common pixel format is the byte image, which is stored as an 8-bit integer giving a range of possible values from 0 to 255. As a convention is taken to be black, and 255 is taken to be white the values in between make up the different shades of Gray. To represent colour images, separate red, green and blue components must be specified for each pixel (assuming a RGB colour model), and so the pixel 'value' becomes a vector of three numbers. Often the three different components are stored as three separates 'grayscale' images known as colour planes (one for each of red, green and blue), which have to be recombined when displaying or processing.

It contains a Raspberry Pi in which OpenCV has been installed. IP Camera has been interfaced with the Raspberry Pi. IP Camera is used to capture real time images of person 's eye and send it to continuously running OpenCV Python script. OpenCV Python script processes the image using Iris Detection Algorithm (Centroid Algorithm) and Threshold Algorithm. Then the command is given by the Raspberry Pi to the Motor Driver circuit regarding the position and direction in which the wheelchair has to move.

3.3 WHEELCHAIR MOVEMENT CONTROL

EYEBALL MOVEMENT	DIRECTION FOR MOVEMENT OF WHEELCHAIR
UP	Straight
RIGHT	Right
LEFT	Left
DOWN	Down

Table 1: wheel chair movement instruction table

To control the Electric Wheelchair (EWC), we have designed a layout with four functional keys positioned invisibly. These keys correspond to moving forward, backward, turning right, and turning left, each located at specific portions. If the user's pupil remains at the center portion, the EWC is automatically stopped for safety reasons. There's no need to display the key layout visually; users intuitively understand the location of the desired key and can select it accordingly.

For example, if the user gazes to the right for more than 0.7 seconds, the EWC initiates a right turn until the user shifts their gaze direction. If the user continues looking in the same direction, the EWC maintains the previously determined moving action. This design ensures seamless control of the wheelchair based on the user's eye movement, enhancing safety and usability.

3.4 HARDWARE SIMULATION

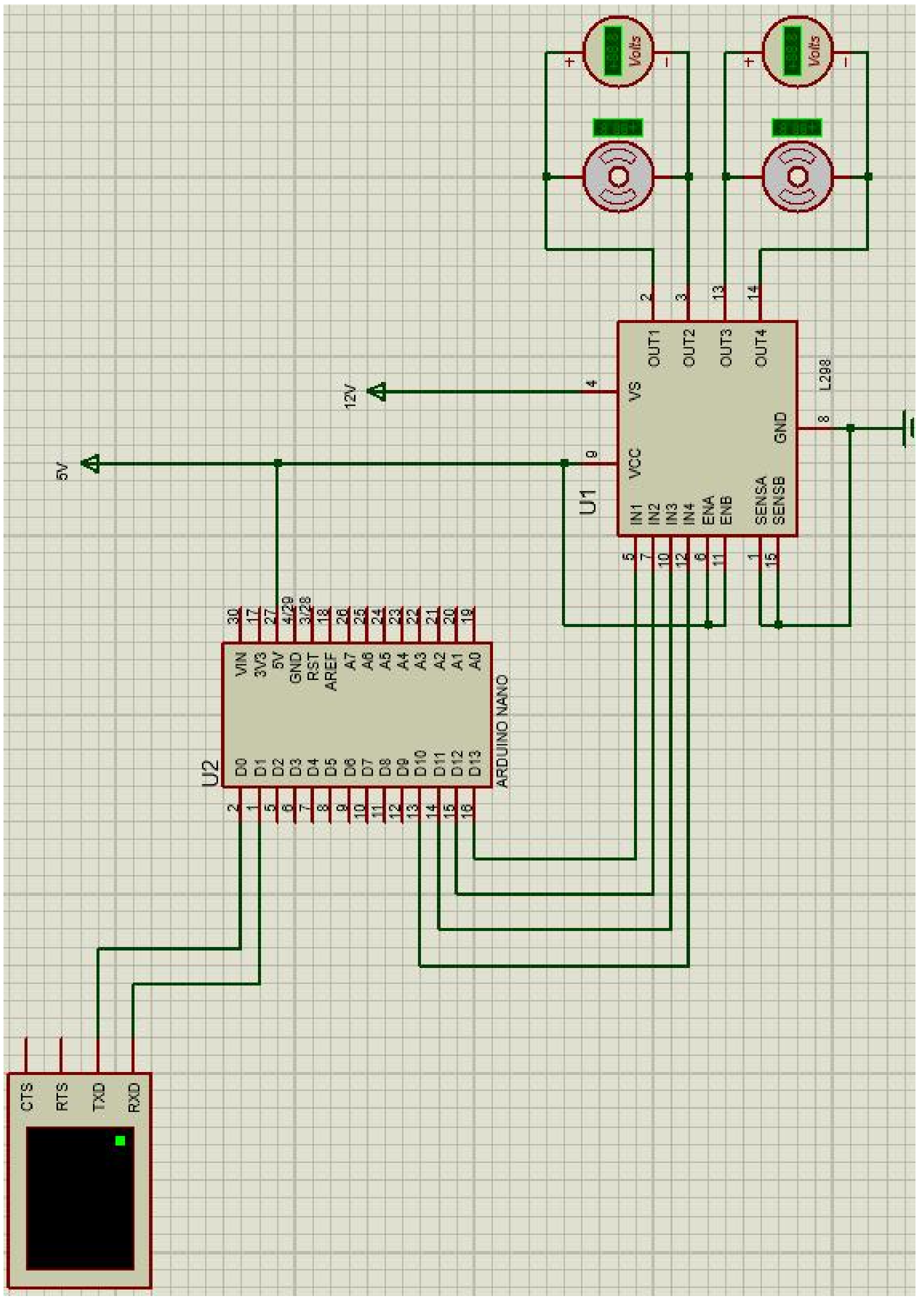

Fig 3: Hardware simulation diagram

The picture above depicts the hardware simulation of the wheelchair conducted during Phase 1 of our project. In this phase, the design includes an Arduino Nano connected to a single motor driver, which in turn drives both motors. Each motor driver comprises an H Bridge circuit dedicated to controlling each motor. The hardware simulation is carried out on the Simulink platform. Through this simulation, the control of both motors corresponding to the received commands is thoroughly tested and evaluated for effectiveness and accuracy. Using this L293D motor driver IC is very simple. The IC works on the principle of Half H-Bridge, let us not go too deep into what H-Bridge means, but for now just know that H bridge is a set up which is used to run motors both in clock wise and anti clockwise direction. As said earlier this IC is capable of running two motors at the any direction at the same time, the circuit to achieve the same is shown

3.5 ARDUINO IDE

The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment. The program or code written in the Arduino IDE is often called as sketching. We need to connect the Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '. ino.' The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low-cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Faire, for example. Arduino is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step-bystep instructions of a kit or sharing ideas online with other members of the Arduino community. There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic

Stamp, Netmedia's BX-24, Phidgets, MIT's Handy board, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems.

WRITING SKETCHES

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension. ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom right-hand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

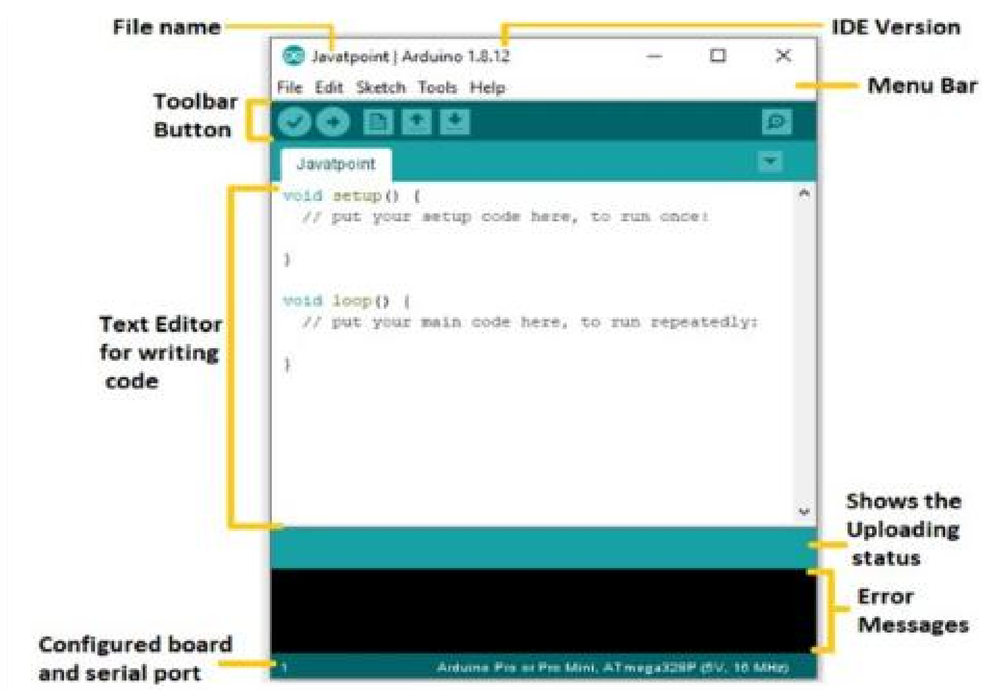

Fig 4: Arduino ide interface

CHAPTER 4

STRUCTURE AND DESIGN

4.1 COMPONENTS

Components required for eye controlled wheelchair are, Arduino nano, Raspberry pi5, IP Web Cam, ultra sonic sensor ,MY1016Z2 Motor, Li-ion Battery, BTS7960 Motor driver

4..1.1 ARDUINO NANO

The Arduino Nano is a compact and versatile microcontroller board based on the Atmega328P or Atmega168 (depending on the version). It's part of the Arduino family and is commonly used in electronics projects for its small size and functionality.

Microcontroller	Atmega 328
Architecture	AVR
Operating voltage	5v
EEPROM	1 KB
SRAM	2KB
Clock Frequency	16 MHZ
Analog Input Pins	8
Input Voltage	7-12V

Table 2: Specifications of arduino nano

Fig 5: Arduino nano expansion board

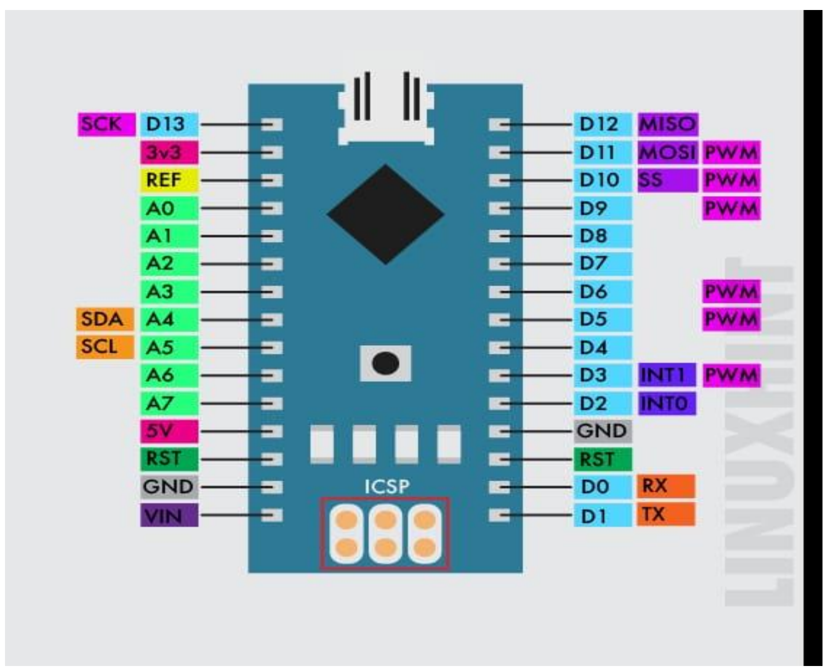

Fig 6: pin diagram of Arduino nano

Technical specification:

- Digital Pins (D2 to D13): These pins can be configured as digital input or output. D2 to D13 can also be used as PWM (Pulse Width Modulation) output. D0 (RX) and D1 (TX) are also used for serial communication with the computer.
- Analog Pins (A0 to A7):These pins can be used as analog inputs to read values from analog sensors. A6 and A7 are analog input only and do not have digital I/O capabilities.
- Power Pins: Vin: This is the input voltage to the Nano when it is powered using an external power source. 5V: This pin outputs 5V when the Nano is powered via USB or an external 5V source. 3.3V: Provides a regulated 3.3V output .GND: Ground pins.
- Reset Button: The reset button resets the microcontroller, restarting the program.
- ICSP Control Serial Programming (ICSP) header for programming the microcontroller.
- UART (Universal Asynchronous Receiver-Transmitter):TX (D1): Transmit pin for serial communication. RX (D0): Receive pin for serial communication.
- External Interrupts: D2 (INT0) and D3 (INT1) can be configured to trigger an interrupt on a low or high signal.
- PWM Pins: D3, D5, D6, D9, D10, and D11 can generate PWM signals.
- AREF: Analog Reference voltage pin.
- Crystal Oscillator Pins: XTAL1 and XTAL2 are the pins for connecting an external crystal oscillator for precise timing.
- I2C Pins: A4 (SDA) and A5 (SCL) are used for I2C communication.
- LED Indicators: There is a built-in LED connected to pin D13, which is often used for basic debugging

4.1.2 RASPBERRY PI:

Raspberry Pi is a series of small, affordable single-board computers developed by the Raspberry Pi Foundation. These versatile devices can be used for various projects, from simple tasks like web browsing to more complex applications like home automation. Raspberry Pi typically consists of a system-on-a-chip (SoC), which includes a processor, GPU, and RAM.It has various ports for connectivity, such as USB ports, HDMI for video output, audio jack, Ethernet, GPIO (General Purpose Input/Output) pins for interfacing with other hardware.

Fig 7: Raspberry pi

4.1.3 IP WEBCAM:

webcam, short for web camera, is a digital camera typically built into or connected to a computer. Its primary purpose is to capture video and sometimes audio, allowing for real-time communication or recording. we have attached an IR web camera onto the handle part of the Wheelchair that is used to detect the eye motion. Then we have designed an algorithm to track the iris of the eye using centroid calculation method and implemented the same in the Open CV. Once the iris is tracked, then the threshold is set.

Fig 8: IP cam

4.1.4 ULTRASONIC SENSOR:

The ultrasonic sensor is a device that uses sound waves to measure distance. It emits ultrasonic pulses and calculates the time it takes for the waves to bounce back after hitting an object, allowing it to determine the object's distance. Commonly used in robotics and automation for proximity sensing. energy takes place over a wide solid angle which might be as high as 180 degrees Thes, some fraction of the incident energy is reflected back to the transducer in the form of echoes. If the object is very close to the sensor, the sound waves return quickly, but if the object is far away from the sensor, the sound waves take longer in return. But if objects are too far away from the sensor, the signal takes so long to come back for is very weak when it comes back that the receiver cannot

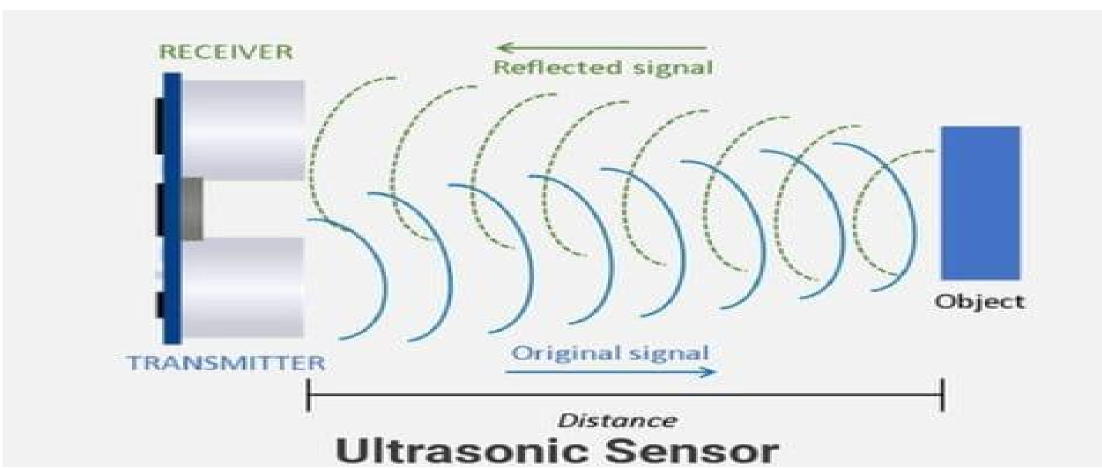

Fig 9:Ultrasonic sensor

4.1.5 MOTOR:

DC motor MY1016Z2 that has a power capacity of 250W, 24 V and can run at 300 rpm. The maximum torque that can be achieved is 22 Nm. This motor can be powered by a 24 V Lithium-ion battery.

Fig 10: MY1016Z2 motor

Motor Specifications:

- Voltage: 24 Volt DC.
- Output: 250 Watt.
- RPM (after Reduction) – 300.
- Full load current – 13.4A.
- No load Current – 2.2A.
- Torque Constant – 8 N.m (80 kg-cm).
- Torque stall – 40 N.m (400 kg-cm).
- Sprocket: 9Tooth only fits #410 bicycle chains.
- For Chain Size: Pitch 0.5 inch.
- Roller Diameter 0.3 inch.
- Roller Width 0.16 inch.

4.1.6 BATTERY:

Lithium ion batteries

Fig 11: Lithium ion battery

Lithium ion batteries have high energy density Lithium ion batteries are considered the best among all battery types because of it's superior Chara and performance Lithium ion batteries are more preferred because their high energy density long life and low cost In this project we use 2 batteries of 12V ,15000 ,MAh batteries.

4.1.7 MOTOR DRIVER

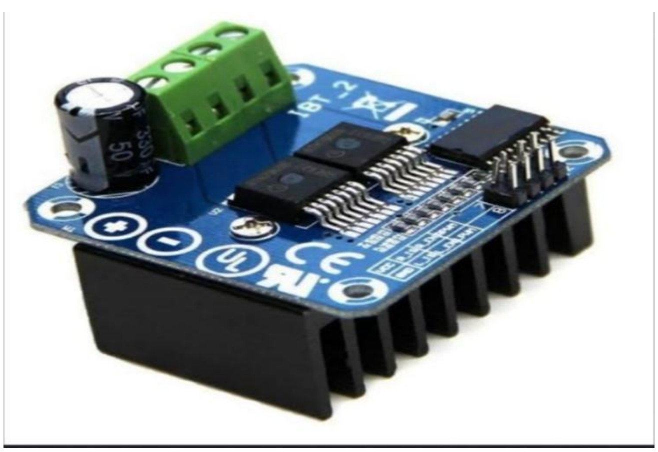

Fig 12: BTS7960 Motor Driver

The BTS7960 is a fully integrated high current H bridge module for motor drive applications. Interfacing to a microcontroller is made easy by the integrated driver IC which features logic level inputs, diagnosis with current sense, slew rate adjustment, dead time generation and protection against overtemperature,

overvoltage, undervoltage, overcurrent and short circuit. The BTS7960 provides a cost optimized solution for protected high current PWM motor drives with very low board space consumption.

4.2 INTERNAL DIAGRAM OF MOTOR DRIVER

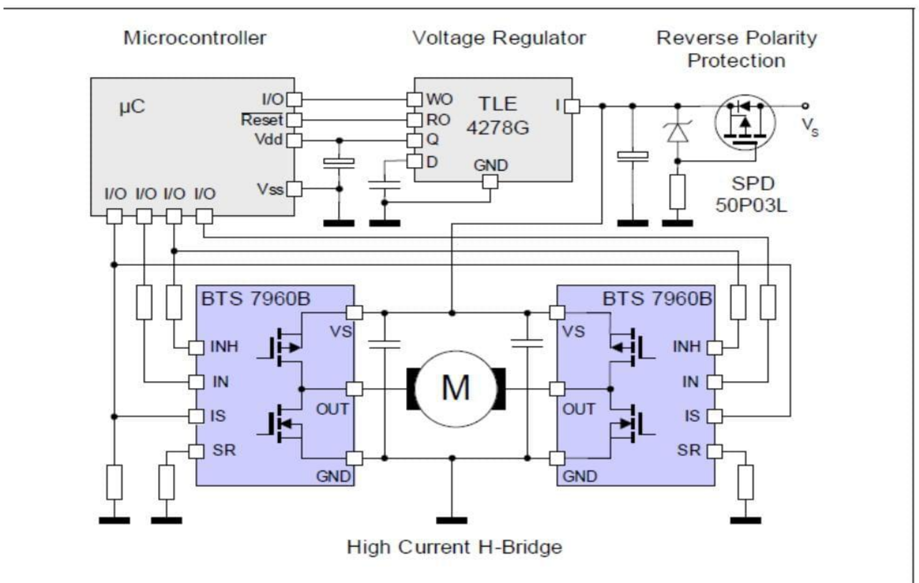

Fig 13:Internal diagram of motor driver

Principle of Work:

The BTS7960 Motor Driver Module is based on the BTS7960B H-bridge driver IC, which allows bi-directional control of DC motors or bipolar stepper motors. The module consists of two H-bridge circuits, one for each motor channel, which can be controlled using digital signals from a microcontroller or other control circuit. To control the speed and direction of a motor, the module uses pulse width modulation (PWM) signals. The duty cycle of the PWM signal determines the motor speed, while the direction is determined by the logic level of two digital input pins. When the PWM signal is applied to one of the H-bridge circuits, the module switches the output of the circuit between the positive and negative supply voltage, which creates a voltage across the motor terminals. By varying the duty cycle of the PWM signal, the average voltage across the motor can be controlled, and hence the motor speed. The module also includes two independent current-sensing channels that can detect over-current or over-temperature conditions. If the current or temperature exceeds a set threshold, the module will disable the H-bridge circuit to protect the module and the connected motor.

4.3 DESIGN OF REQUIRED MOTOR

Standard slop in Indian hospital environment is 7.1°

Total mass of the wheelchair =145Kg (self-weight = 45 kg, load carrying capacity =100 kg)

Radius of the wheel = 10 inches = 0.254 m (based on availability and cost)

Circumference of the wheel is the linear distance that will be covered in

one revolution = $2\pi r$ =2*3.14*0.254=1.59m

1rps covers = 1.59 m in 1 second

Revolution per second =0.94

RPM = 0.94*60 =56.4~57rpm

<u>Case-1</u>

Travelling on flat path

Torque = (Push force +rolling resistance) * radius of the wheel

Push is the minimum amount of force needed to Start the motion of any automobile

it is given by = m*a

rolling resistance is the resistance offered by the tire due to its visco-elasticity

it is given by = umg cos α

u = coefficient of Rolling resistance =0.01

Therefore T = [(m*a) +(umg cosα)] *r= [(43.5 + 14.22) *0.254] = 14.66 N/m

Consider factor for safety= 1.5

T= 14.66 * 1.5 = 21.99

Either 11N/m at each wheel

Calculating power required at 1wheelPower = torque *angular velocity (v/r) = 11 *(1.5/.254) =64.9w

4.4 SELECTION OF BATTERY

The battery voltage needs to match the motor rating. The controller voltage rating needs to be the same or higher.The battery AH rating should be chosen based on the motor power rating motor voltage rating x 1hr.

In our project 24V 250W motor should be paired with a 24V battery that has an AH rating of at least 250W ÷ 24V x 1hr = 10.4AH.

This helps assure that the battery will not be over stressed when driving the motor at max power. A higher AH rating will equate to longer range and extended battery life. The controller current rating should be comfortably higher than the motor's maximum current draw.

CHAPTER 5

HARDWARE IMPLEMENTATION

The hardware implementation of our project involved several key steps. Firstly, we combined two wheelchairs obtained from a scrap shop and a child care home in Kattappana. Next, we constructed a wooden piece to house the motor and control circuits. Gear and sprocket mechanisms were utilized to connect the motor to the wheelchair wheels. The motor speed was regulated using PWM signals generated by the Arduino Nano, which were tested in the lab using a miniature model of our project. The motor driver circuit and Arduino Nano were connected according to a predetermined layout, and the program was developed accordingly. Initially, testing was conducted using a laptop in place of the Raspberry Pi and camera module. After confirming the functionality of the program, it was uploaded to the Raspberry Pi. The final testing phase involved running the program with a mobile camera instead of an IP Webcam. It's worth noting that the camera module required for this purpose must have at least 8 MB picture clarity, making a Raspberry camera unsuitable for our needs.

5.1 WHEELCHAIR CONSTRUCTION

The wheelchair frame is made up of aluminium and MS metal frame and seating arrangement is done by placing a plastic chair. The power source, motor and driver unit are placed at the bottom of the wheelchair. The commands for the motor driver are given by pwm generated by Arduino nano which is connected to Raspberry pi through USB cable. The place of motor is done as one motor rotate in one forward direction and other in backward direction.

Fig 14.1: wheel chair frame

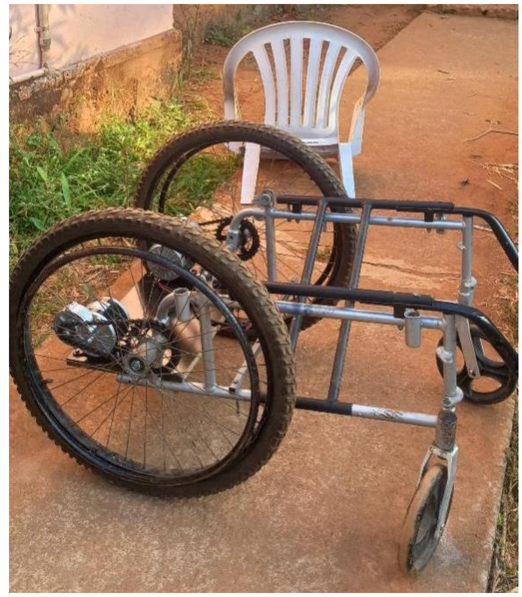

fig 14.2: connecting motor & sprocket

Fig 14.3: final structure

The wheelchair has been made with locally available materials such as plastic for seat and backrest, aluminium and mild steel used for building the body of wheelchair, commercially available shaft and sprocket gear was used.

5.2 GEAR AND CHAIN DRIVE

Based on top speed of 5 km/hr and wheelchair wheel with 50 cm radius, output speed of drive system has maximum speed equal to150 rpm. Chain drive has one main advantage over a traditional gear train. Two gear wheels which are connected to shaft of the wheel and a chain that connect gear wheel with motor shaft are needed for the movement of the motor.

Fig15: gear and sprocket

5.3 SPEED CONTROL USING PWM

Fig 16:Testing of speed control using pwm

By Theoretical

Time period of PWM = 200

Rated speed of the motor = 300 RPM

The maximum PWM signal given to the motor in the program =150

Therefore 200: 300 =150:(150*300)/200=225 RPM

The teeth ratio of motor and gear Sprocket 9:28

That is 225: 72.3 RPM

Therefore, the speed is approximately 70 to 75 RPM

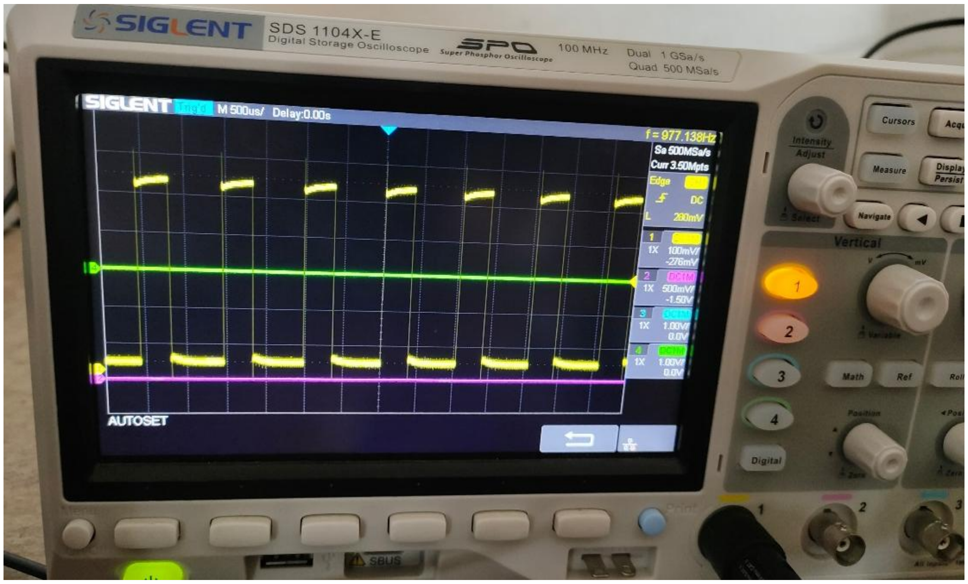

Fig 17.1: waveform of pwm signal on 40% duty cycle

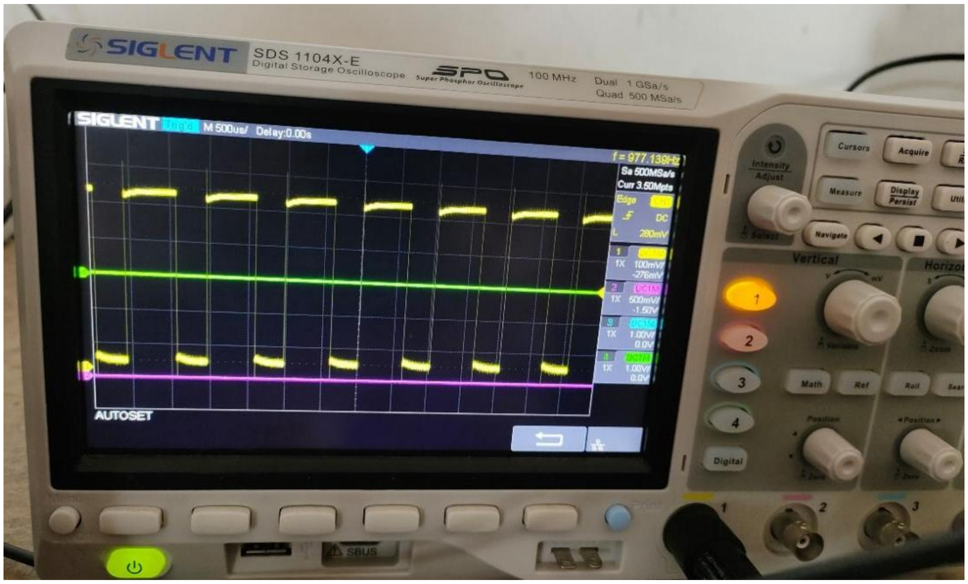

Fig 17.2 waveform of pwm on 75% duty cycle

Controlling the speed of a DC motor using a limited resistor in series can result in energy loss and excess heat production. To mitigate these issues, the most common technique employed is Pulse Width Modulation (PWM). PWM involves altering the average DC value of a signal by passing it through a low pass filter. When this modified signal is supplied to a DC motor, its speed can be adjusted by varying the duty cycle of the PWM signal.

In PWM, the pulse width, or on-time (HIGH value), of the signal is increased while simultaneously reducing the off-time (LOW value) by the same amount, maintaining a constant frequency. By increasing the on-time, the average DC voltage value of the signal rises, thereby increasing the motor speed, and vice versa.

This method essentially converts digital signals into analog signals by modulating the pulse width. A square wave is generated between 0 and 5 volts, with the PWM frequency determining how rapidly the signal switches between high and low states. Ultimately, PWM provides an efficient and effective means of controlling the speed of DC motors while minimizing energy loss and heat generation.

Duty cycle =Ton/ (T total) *100%

5.4 MOTOR DRIVER IMPLEMENTATION

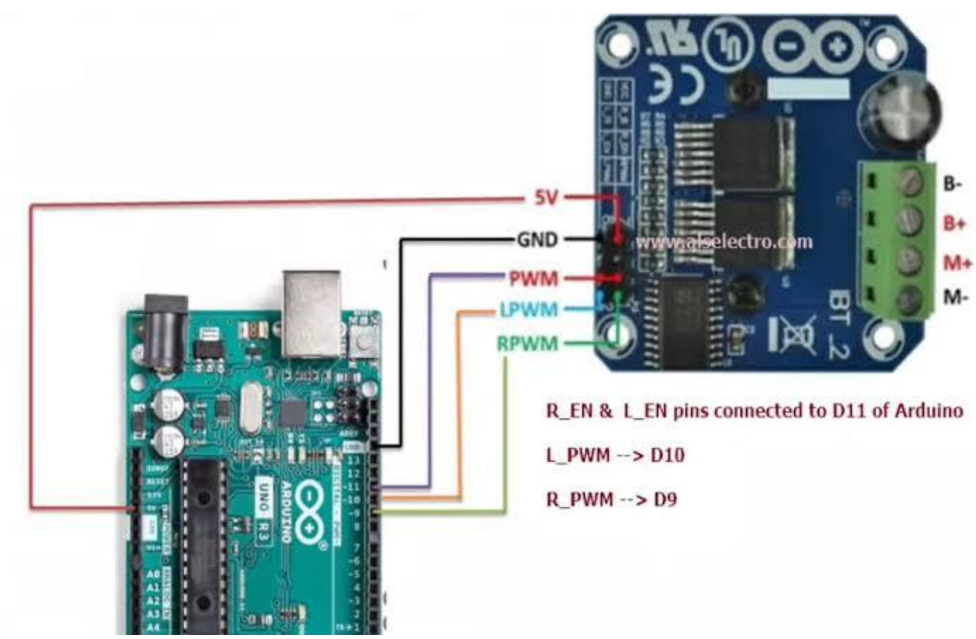

Fig 18: Connecting terminals of Arduino nano and motor driver

The +5V terminal of the Arduino Nano is linked to the +5V of the BST7960 motor driver. Pin 11 of the Arduino Nano is connected to the short-circuited terminals of the PWM of the motor driver. Pins 10 and 9 of the Arduino Nano are connected to the left and right PWM points of the driver, where PWM is required to change direction to left and right. Terminals M+ and M -ve denoted in the driver are connected to the motor. One of the motors is connected in negative polarity due to mechanical construction. The power supply is connected to terminal B+ and B-ve And the ground is connected

5.5 HARDWARE CIRCUIT DIAGRAM

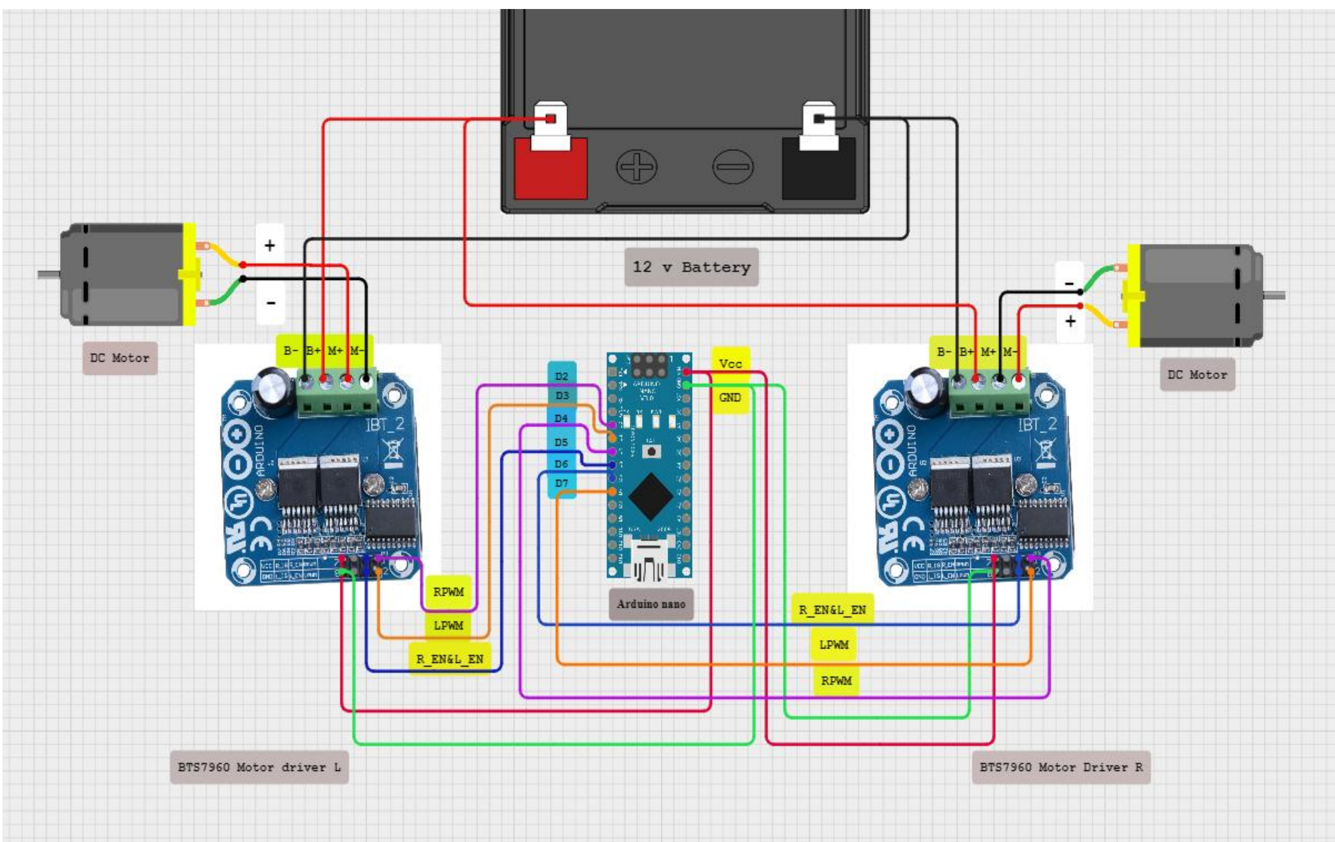

Fig19: Hardware circuit diagram

Power connection:

Connect the B+ pins of both BTS7960s to the positive terminal of the power supply (e.g., 12V from a 12V 7Ah battery). Connect the B- pins of both BTS7960s to the negative terminal of the power supply. Connect the VCC pins of both BTS7960s to the VIN pin of the Arduino Nano if the supply voltage is within the Arduino's input voltage range. Otherwise, connect VCC directly to a 5V supply that is within the BTS7960's operating range.

Connect all GND pins of the BTS7960s to the GND pin of the Arduino Nano to establish a common ground.

Control connection:

Connect the power supply's positive and negative terminals to both BTS7960 modules' VCC and GND, respectively.

Connect each motor's leads to the outputs of each BTS7960 module (OUT1 and OUT2).

Connect the Arduino Nano's GND to the GND of both BTS7960 modules (common ground).

Connect the Arduino Nano's digital pins to the control pins of both BTS7960 modules.

Arduino D5 -> BTS7960 #1 L_EN and R_EN (Enable pins for the first motor driver)

Arduino D6 -> BTS7960 #2 L_EN and R_EN (Enable pins for the second motor driver)

Arduino D2 -> BTS7960 #1 RPWM (Right PWM input for speed control of the first motor)

Arduino D3 -> BTS7960 #1 LPWM (Left PWM input for speed control of the first motor)

Arduino D4 -> BTS7960 #2 RPWM (Right PWM input for speed control of the second motor)

Arduino D7 -> BTS7960 #2 LPWM (Left PWM input for speed control of the second motor)

Not connected -> BTS7960 #1 R_IS (Right current sense for the first motor, optional for feedback)

Not connected -> BTS7960 #1 L_IS (Left current sense for the first motor, optional for feedback)

Not connected -> BTS7960 #2 R_IS (Right current sense for the second motor, optional for feedback)

Not connected -> BTS7960 #2 L_IS (Left current sense for the second motor, optional for feedback)

CHAPTER 6

RESULTS AND DISCUSSION

6.1 EYE TRACKING SIMULATION

Eye tracking Simulation refers to software or technology that mimics how a person’s eye move and focus on a virtual environment using open cv. To control eyeball control wheelchair, we design four functional keys visible layout, move forward, backward, turn right and turn left. We are constructing four boxes and placed around the eye gaze. When the eyeball enters any boxes which show the direction of wheelchair motion, top box direct forward direction, Bottom box direct Backward motion, Left & Right boxes direct Left & Right Movement of the wheelchair.

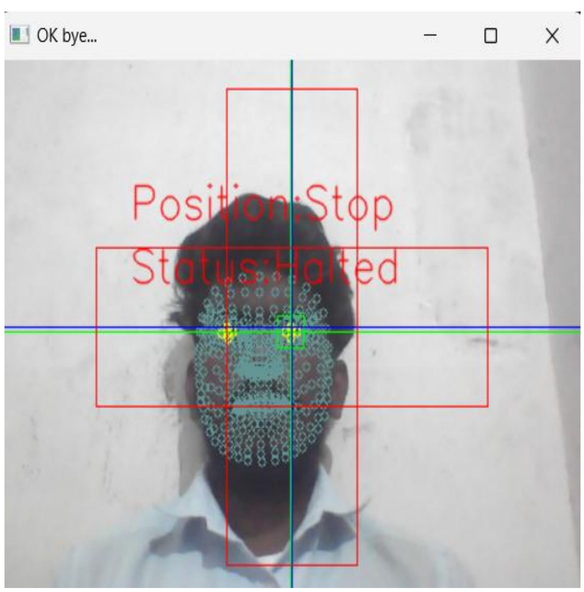

Fig 20.A: stop

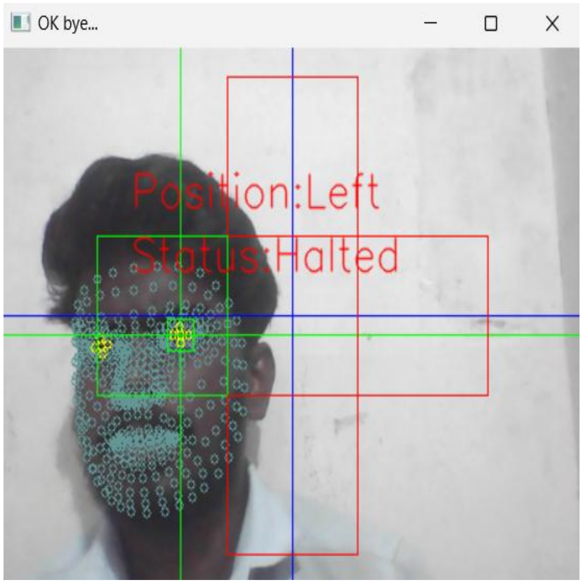

Fig 20 B: Left

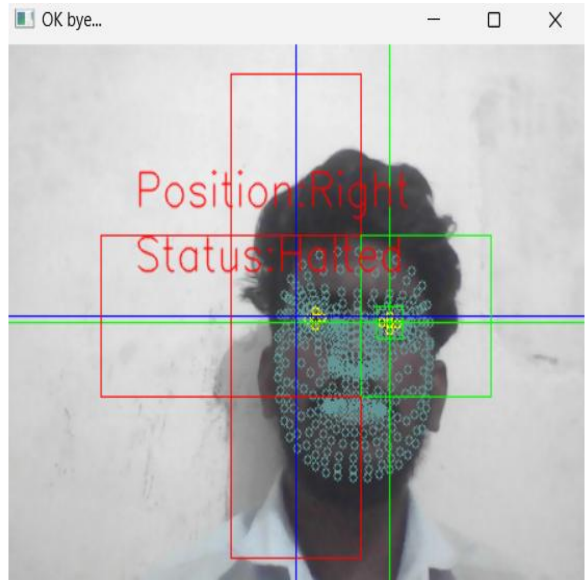

fig 20 C: Right

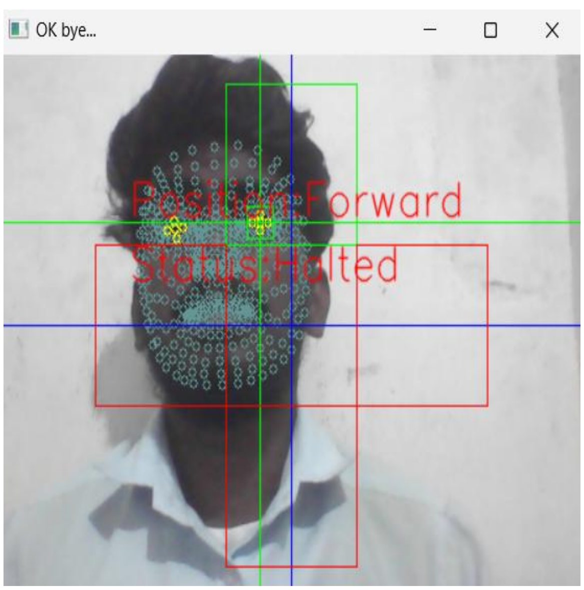

fig 20 D: Forward

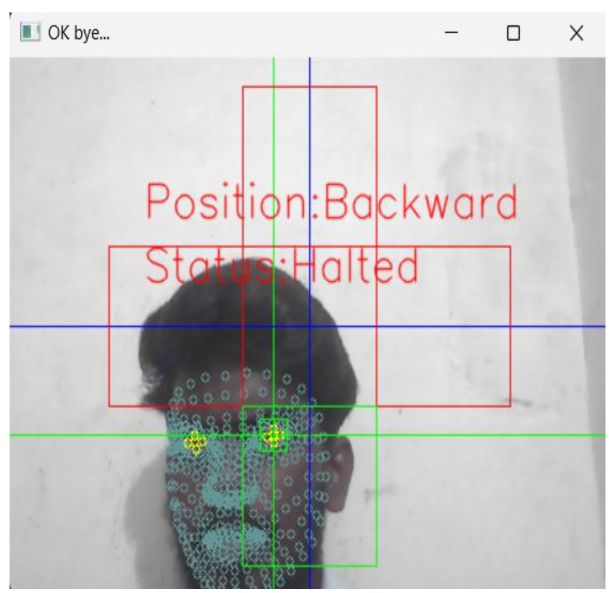

fig 20 E: Backward

6.2 FINAL STRUCTURE OF EYE BALL CONTROLLED WHEEL CHAIR

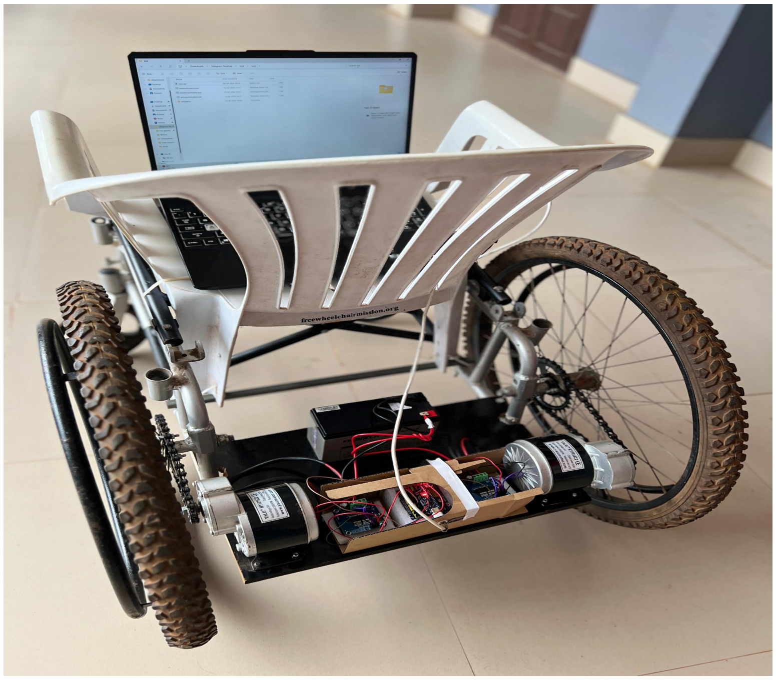

Fig 21 : Back view of eye ball controlled wheelchair

Fig 22: Front view of eye ball controlled wheelchair

An eyeball controlled wheelchair that meets the needs of individuals with paralysis has beendesigned,manufactured,and tested. It allows user to move using eyeball control in direction by capturing the eye movement using camera.Inorder to control the wheel chair movement we designed 4 functional keyways forward ,backward,right,left.If user looks for any other direction wheelchair will stop.The wheelchair can move automatically with the help of that are controlled by 2 motors that are controlled by motor driver and arduino.Motor speed can be controlled by using pwm

CHAPTER 7

CONCLUSION

Eyeball movement based wheelchair control system is designed for paralysed people. This system makes the lives of the people independent. Raspberry pi technology is used to improve the accuracy compared to other systems. Obstacle detection using IR sensors which are fixed at the base of wheel chair allows the user to know the obstacles in his path. Camera module is fixed for backward tracking. The system is efficient with minimum cost compared to other systems. The idea of this system could better the lives of many people across the globe.

The project focused on the development and implementation of an automatic wheelchair system controlled by eye movements. This innovative technology leverages eye-tracking technology to offer a more accessible and intuitive means of mobility for individuals with physical disabilities. Through the integration of advanced sensors and intelligent algorithms, the wheelchair can interpret the user's eye movements and translate them into precise commands for navigation. The project not only addresses the practical challenges faced by those with physical disabilities but also highlights the potential for technological solutions to create positive societal impact. The integration of eye-tracking technology in assistive devices opens up new possibilities for inclusive design and fosters a more inclusive and accessible world.

The key advantages of the eye-controlled wheelchair include enhanced independence and improved quality of life for users. By eliminating the need for manual controls, the system empowers individuals with limited mobility to navigate their surroundings with greater ease. The real-time responsiveness of the technology ensures a seamless and efficient user experience, allowing users to navigate through both indoor and outdoor environments effortlessly.

REFERENCE

[1]N Wanluk, S Visitsattapongse, A Juhong . "Smart wheelchair based on eye tracking." 2016 9th Biomedical Engineering International Conference (BMEiCON). doi: 10.1109/BMEiCON.2016.7859594 (2016)

[2]MA Eid, N Giakoumidis, A El Saddik(2016). "A novel eye-gaze- controlled wheelchair system for navigating unknown environment"s: case study with a person with ALS. IEEE Access (Volume: 4). doi: 10.1109/ACCESS.2016.2520093

[3]A Ishizuka, A Yoroz. "Motion control of a powered wheelchair using eye gaze in unknown environments". 2017 11th Asian Control Conference (ASCC). doi: 10.1109/ASCC.2017.8287148 (2016)

[4]A Rajesh, M Mantur ." Eyeball gesture controlled automatic wheelchair using deep learning." 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC). doi: 10.1109/R10- (2017)
HTC.2017.8288981

[5]D Bai, Z Liu, Q Hu, J Yang, G Yang, C Ni." Design of an eye movement-controlled wheelchair using Kalman filter algorithm". 2016 IEEE International Conference on Information and Automation (ICIA). doi: 10.1109/ICInfA.2016.7832085 (2016)

[6]Aniwat Juhong, T Treebupachatsakul, C Pintavirooj "Smart eye-tracking system". International Workshop on Advanced Image Technology (IWAIT). doi: 10.1109/IWAIT.2018.8369701 2018 (2018)

[7]SN Patel, V Prakash . "Autonomous camera based eye controlled wheelchair system using raspberry-pi". International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS). doi: 10.1109/ICIIECS.2015.7192876 (2015)

[8]P Arora, A Sharma, AS Soni. "Control of wheelchair dummy for differently abled patients via iris movement using image processing in MATLAB", Annual IEEE India Conference (INDICON). doi: 10.1109/INDICON.2015.7443610 (2015)

[9]A Palla, L Sarti, A Frigerio)." Embedded implementation of an eye- in-hand visual servoing control for a wheelchair mounted robotic arm". IEEE Symposium on Computers and Communication (ISCC). doi: 10.1109/ISCC.2016.7543753 2016

[10]FB Taher, NB Amor, M Jallouli . An extended eye movement tracker system for an electric wheelchair movement control.2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA). doi: 10.1109/AICCSA.2015.7507101 (2015).

[11]N Oda. Development of Power Assist Wheelchair Control with Driving Assistance using Fish-eye Vision. 12th France-Japan and 10th Europe-Asia Congress on Mechatronics. doi: 10.1109/MECATRONICS.2018.8495884 2018

[12]H Zatla, A Hadj-Abdelkader, Y Morere . OPCM model application on a 3D simulator for powered wheelchair. 2015 International Conference on Virtual Rehabilitation (ICVR). 2015
doi: 10.1109/ICVR.2015.7358611

[13]NMM Noor, S Ahmad . Performance analysis of an electrooculography-based on intelligent wheelchair motion control. 10th Asian Control Conference (ASCC). doi: 10.1109/ASCC.2015.7244813 201

APPENDIX

5.1 HARD WARE SIMULATION

```
#pragma GCC push_options
#pragma GCC optimize ("Os")
 #pragma GCC pop_options
 // Peripheral Constructors
 void peripheral_setup () {
}
 void peripheral_loop() {
}
//---CONFIG_END---
 char data;
void setup() {
  peripheral_setup();
  pinMode(13,OUTPUT); //left motors forward
  pinMode(12,OUTPUT); //left motors reverse
  pinMode(11,OUTPUT); //right motors forward
  pinMode(10,OUTPUT); //right motors reverse
  pinMode(9,OUTPUT); //blade motor
  Serial.begin(9600);
}
void loop() {
  peripheral_loop();
  if(Serial.available()){
    data = Serial.read();
    Serial.print("Command :");
    Serial.println(data);
```

```
  }
  if(data == 'F'){ //move forward(all motors rotate inforward direction)
    digitalWrite(12,LOW);
    digitalWrite(10,LOW);
    digitalWrite(13,HIGH);
    digitalWrite(11,HIGH);
  }
  else if(data == 'B'){ //move reverse (all motors rotate inreverse direction)
    digitalWrite(13,LOW);
    digitalWrite(11,LOW);
    digitalWrite(12,HIGH);
    digitalWrite(10,HIGH);
  }
  else if(data == 'L'){ //turn right (left side motors rotate inforward direction, right side motors
doesn't rotate)
    digitalWrite(13,LOW);
    digitalWrite(10,LOW);
    digitalWrite(11,HIGH);
    digitalWrite(12,HIGH);
  }
  else if(data == 'R'){ //turn left (right side motors rotate inforward direction, left side motors
doesn't rotate)
    digitalWrite(11,LOW);
    digitalWrite(12,LOW);
    digitalWrite(10,HIGH);
    digitalWrite(13,HIGH);
  }
  else if(data == 'S'){ //STOP (all motors stop)
    digitalWrite(13,LOW);
    digitalWrite(12,LOW);
```

```
    digitalWrite(11,LOW);
    digitalWrite(10,LOW);
  }    delay(100);    }
```

// CODE OF RASPBERRY PI //

```
import cv2
import mediapipe as mp
import sys
ip=sys.argv[1]
ser=sys.argv[2]
link='http://'+ip+'/video'
if ip=='0':
   cam = cv2.VideoCapture(int(ip))
else:
   print('\n\nCamera : '+link)
   cam = cv2.VideoCapture(link)
if ser=='-s':
   serial_port=sys.argv[3]
   serial_port='com'+serial_port
   print('Serial Port : '+serial_port+'\n\n')
   import serial
   datas=serial.Serial(serial_port, 115200, serial.EIGHTBITS, serial.PARITY_NONE,
     serial.STOPBITS_ONE)
```

```
face_mesh = mp.solutions.face_mesh.FaceMesh(refine_landmarks=True)
while True:
    _, frame = cam.read()
    frame = cv2.flip(frame, 1)
    frame = cv2.resize(frame, (0, 0), fx = 0.7, fy = 0.7)
    frame_h, frame_w, _ = frame.shape
    # frame = frame[0:frame_h, int((frame_w/2)-(frame_h/2)):int((frame_w/2)+(frame_h/2))]
    frame_h, frame_w, _ = frame.shape
    frame_h_mid, frame_w_mid=int(frame_h/2),int(frame_w/2)
    rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    output = face_mesh.process(rgb_frame)
    landmark_points = output.multi_face_landmarks
    if landmark_points:
        landmarks = landmark_points[0].landmark
        for landmark in landmarks:
            x = int(landmark.x * frame_w)
            y = int(landmark.y * frame_h)
            cv2.circle(frame, (x, y), 3, (150, 150, 100))
        for landmark in landmarks[468:473]:
            x = int(landmark.x * frame_w)
            y = int(landmark.y * frame_h)
            cv2.circle(frame, (x, y), 3, (0, 255, 255))
```

```
  for landmark in landmarks[473:478]:
      x = int(landmark.x * frame_w)
      y = int(landmark.y * frame_h)
      cv2.circle(frame, (x, y), 3, (0, 255, 255))
  text='Stop'
  status_text='Halted'
  forward_box=(0,0,255)
  left_box=(0,0,255)
  right_box=(0,0,255)
  backward_box=(0,0,255)
  cpx = int(landmarks[473].x * frame_w)
  cpy = int(landmarks[473].y * frame_h)
  forward_box_point_1,forward_box_point_2=   (frame_w_mid-50,  frame_h_mid-150),
(frame_w_mid+50, frame_h_mid-50)
  right_box_point_1,right_box_point_2=      (frame_w_mid+50,  frame_h_mid-50),
(frame_w_mid+150, frame_h_mid+50)
  backward_box_point_1,backward_box_point_2= (frame_w_mid-50,  frame_h_mid+50),
(frame_w_mid+50,  frame_h_mid+150)
  left_box_point_1,left_box_point_2=        (frame_w_mid-150, frame_h_mid-50),
(frame_w_mid-50, frame_h_mid+50)
  if
(cpx>forward_box_point_1[0])&(cpx<forward_box_point_2[0])&(cpy>forward_box_point_1[1])
&(cpy<forward_box_point_2[1]):
      forward_box=(0,255,0)
```

```
    text='Forward'
    # datas.write(b'f')
 if
(cpx>right_box_point_1[0])&(cpx<right_box_point_2[0])&(cpy>right_box_point_1[1])&(cpy<ri
ght_box_point_2[1]):
    right_box=(0,255,0)
    text='Right'
    # datas.write(b'r')
 if
(cpx>backward_box_point_1[0])&(cpx<backward_box_point_2[0])&(cpy>backward_box_point
_1[1])&(cpy<backward_box_point_2[1]):
    backward_box=(0,255,0)
    text='Backward'
    # datas.write(b'b')
 if
(cpx>left_box_point_1[0])&(cpx<left_box_point_2[0])&(cpy>left_box_point_1[1])&(cpy<left_b
ox_point_2[1]):
    left_box=(0,255,0)
    text='Left'
    # datas.write(b'l')
 cv2.rectangle(frame, (cpx-10, cpy-10), (cpx+10, cpy+10),   (0, 255, 0), 1) #Crosshair
 cv2.line(frame,     (int(cpx), 0),    (int(cpx),frame_h), (0, 255, 0), 1) #Vertical
 cv2.line(frame,     (0, int(cpy)),    (frame_w,int(cpy)), (0, 255, 0), 1) #Horizontal
 cv2.line(frame, (0, frame_h_mid), (frame_w,frame_h_mid), (255, 0, 0), 1)
```

```
    cv2.line(frame, (frame_w_mid,0),  (frame_w_mid,frame_h), (255, 0, 0), 1)
    cv2.rectangle(frame, forward_box_point_1,  forward_box_point_2,  forward_box,  1)
    cv2.rectangle(frame, right_box_point_1,    right_box_point_2 ,   right_box,    1)
    cv2.rectangle(frame, left_box_point_1,     left_box_point_2 ,    left_box,     1)
    cv2.rectangle(frame, backward_box_point_1, backward_box_point_2, backward_box, 1)
    if ser=='-s':
        if text=='Forward':
            datas.write(b'f')
        if text=='Right':
            datas.write(b'r')
        if text=='Left':
            datas.write(b'l')
        if text=='Stop':
            datas.write(b's')
        if text=='Backward':
            datas.write(b'b')

    cv2.putText(frame, "Status:"+status_text, (100,140), cv2.FONT_HERSHEY_SIMPLEX, 1,
  (0,0,255), 1, cv2.LINE_AA)
    cv2.putText(frame, "Position:"+text,      (100,100), cv2.FONT_HERSHEY_SIMPLEX, 1,
  (0,0,255), 1, cv2.LINE_AA)
cv2.imshow('OK bye...', frame)
cv2.waitKey(1)
```

// CODE OF AURDINO NANO //

```
#define EN1 5 //Driver 1, R_EN&L_EN

#define EN2 6 //Driver 2, R_EN&L_EN

#define IN1 2 //Driver 1

#define IN2 3 //Driver 1

#define IN3 4 //Driver 2

#define IN4 7 //Driver 2

#define TRIG_PIN 8

#define ECHO_PIN 9

#define DISTANCE 25

long dis;

unsigned long tim1;

long dur;

unsigned long tim,lag_timer_1;

uint16_t duration;

uint8_t ultrasonic_flag;

char a;

int motor_pwm_1,motor_pwm_2;

int motor_pwm_lag_1,motor_pwm_lag_2;

void setup() {

  Serial.begin(115200);
```

```
    pinMode(EN1,OUTPUT);
    pinMode(EN2,OUTPUT);
    pinMode(IN1,OUTPUT);
    pinMode(IN2,OUTPUT);
    pinMode(IN3,OUTPUT);
    pinMode(IN4,OUTPUT);
    pinMode(TRIG_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);
    analogWrite(EN1,0);
    analogWrite(EN2,0);
    moveStop();
   }
void loop() {
    if(millis()-lag_timer_1>10){
     lag_timer_1=millis();
     if(motor_pwm_lag_1>motor_pwm_1){
      motor_pwm_lag_1=motor_pwm_lag_1-4;
     }
     if(motor_pwm_lag_1<motor_pwm_1){
      motor_pwm_lag_1=motor_pwm_lag_1+2;
     }
     if(motor_pwm_lag_2>motor_pwm_2){
      motor_pwm_lag_2=motor_pwm_lag_2-4;
```

```
  }
  if(motor_pwm_lag_2<motor_pwm_2){
    motor_pwm_lag_2=motor_pwm_lag_2+2;
  }
}

Serial.print(motor_pwm_lag_1);
Serial.print("   ");
Serial.println(motor_pwm_lag_2);
analogWrite(EN1,motor_pwm_lag_1);
analogWrite(EN2,motor_pwm_lag_2);
if(millis()-tim1>=100){
  // dis=ultrasonic()/29/2;
  dis=50;
  tim1=millis();
}
if(dis<DISTANCE){
  if(a=='f'){
    moveStop();
  }
  ultrasonic_flag=1;
}
else if(dis>DISTANCE+5){
```

```
  ultrasonic_flag=0;
}
if(millis()-tim>=duration){
  // moveStop();
  motor_pwm_1=0;
  motor_pwm_2=0;
}
if(Serial.available()){
  a=Serial.read();
  if(a=='f'||a=='r'||a=='b'||a=='l'){
    tim=millis();
  }
  if(a=='f'&&ultrasonic_flag==0){
    duration=200;
    moveForward();
    motor_pwm_1=150;
    motor_pwm_2=150;
  }
  if(a=='r'){
    duration=200;
    turnRight();
    motor_pwm_1=150;
    motor_pwm_2=150;
```

```
    }
    if(a=='b'){
      duration=200;
      moveBackward();
      motor_pwm_1=150;
      motor_pwm_2=150;
    }
    if(a=='l'){
      duration=200;
      motor_pwm_1=150;
      motor_pwm_2=150;
      turnLeft();
    }
    if(a=='s'){
      duration=0;
      motor_pwm_1=150;
      motor_pwm_2=150;
      moveStop();
    }
  }
}

void moveStop() {
```

```
  analogWrite(EN1,0);
  analogWrite(EN2,0);
  digitalWrite(IN1,0);
  digitalWrite(IN2,0);
  digitalWrite(IN3,0);
  digitalWrite(IN4,0);
}
void moveForward() {
  digitalWrite(IN1,0);
  digitalWrite(IN2,1);
  digitalWrite(IN3,0);
  digitalWrite(IN4,1);
}
void moveBackward() {
  digitalWrite(IN1,1);
  digitalWrite(IN2,0);
  digitalWrite(IN3,1);
  digitalWrite(IN4,0);
}
void turnRight()
{
  digitalWrite(IN1,0);
  digitalWrite(IN2,1);
```

```
  digitalWrite(IN3,1);
  digitalWrite(IN4,0);
}
void turnLeft() {
  digitalWrite(IN1,1);
  digitalWrite(IN2,0);
  digitalWrite(IN3,0);
  digitalWrite(IN4,1);
}
long ultrasonic(){
  pinMode(TRIG_PIN, OUTPUT);
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(5);
  digitalWrite(TRIG_PIN, LOW);
  pinMode(ECHO_PIN, INPUT);
  dur = pulseIn(ECHO_PIN, HIGH);
  return dur;
}
```

Program Educational Objectives (PEO)

1.To mould graduates with sound professional knowledge in the field of electrical and electronics engineering and allied areas to meet the needs of industry and society

2.To build graduates who pursue higher studies and contribute significant progress in the field of research and development in the arena of electrical and electronics engineering

3. To frame graduates with ethical values and leadership qualities who are successful in team work and self-motivated in life-long learning objectives.

Course Outcomes |COs| :

After successful completion of the course, the students will be able to:

CO1	Model and solve real world problems by applying knowledge across domains Cognitive knowledge level: Apply).
CO2	Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: Apply).
CO3	Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: Apply).
CO4	Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: Apply).
CO5	Identify technology/research gaps and propose innovative/creative solutions(Cognitive knowledge level: Analyse).
CO6	Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge (level : Apply).

Program Outcomes

Engineering Graduates will be able to

1. **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems
2. **Analysis:** Identify, formulate review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.
3. **Design/Development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations
6. **The Engineer and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environmental and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and Team work:** Function effectively as an individual, and as a member or leader in diverse teams and in multi-disciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write

effective reports and design documentation, make effective presentation, and give and receive clear instruction.

11. **Project management and finance**: Demonstrate knowledge and undertaking of engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects in multi-disciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs) Engineering Graduates will be able to	
1.	Technical Knowledge: Apprehend, analyse and solve specific engineering problems of circuits, power systems, control, power electronics, electric machines, drives and signal processing by applying the knowledge of basic sciences, mathematics and engineering principles.
2.	Industry and Society: Transform and excel as solution providers for developing efficient electric drives, clean, green and renewable energy systems to cater the needs of industry and society.
3.	Professionalism: Assimilate professional competency, ,promote design and development skill set in concurrent needs