Available onlinewww.ejaet.com

European Journal of Advances in Engineering and Technology, 2022, 9(4s):523-536
International Conference on Tech Trends in Science & Engineering (ICTTSE) 2022
Suryodaya College of Engineering & Technology, Suryodaya Polytechnic, Nagpur, Maharashtra, India

Research Article ISSN: 2394 - 658X

Design & Fabrication of Android-based Wheelchair

Ketan Tonpe[1], Ram Wayzode[2], Rajesh Sharma[3], Pranaysunny Mathews[4], Amit Singare[5], Prashant Hardas[6]

[1,2]Assistant Professor, Mechanical Engineering, SCET, Nagpur
[3, 4, 5,6] Student, Mechanical Engineering SCET, Nagpur

ABSTRACT

Key words Design, Fabrication, Android-based Wheelchair

INTRODUCTION

Patients who could not walk be it any reason need wheelchair so that they can move from one place to another. A wheelchair is a chair fitted with wheels. The rear wheels are larger in diameter, generally rear wheels are used by the patient to operate the wheelchair. Front wheels are castor wheels with bearings which adjust themselves as per the movement of rear wheels.

Smart wheelchair is something which makes it easy for the patient to access things they need and reduces their dependency on another person.

Smartphone is nowadays like heartbeat of people and available with every individual. Smartphone is like a minicomputer which can perform a wide range of operations using application available on the Play store. Various wearables are nowadays becoming wireless. This is possible with the help of Bluetooth available in our smartphone. We can send commands to the other Bluetooth devices using our smartphone Bluetooth signal.

In this project, we have integrated a smartphone with a wheelchair. We have used Bluetooth Module HC-05 and Arduino Mega and program the Arduino to control DC Motors with the Monster Motor Shield. The prototype smart wheelchair is able to perform basic movement operations like Front, Left and Right. Also, there are 4 speed variations provided.

There are 3 different positions provided for the patient's comfort: sitting, sleeping and resting. the patient could be easily transferred to bed by switching the wheelchair to the sleeping position.

PROBLEM IDENTIFICATION

Disabled person who is already weak needs to apply efforts to run a traditional mechanical wheel chair. The traditional wheelchair doesn't provide easy chair-to-bed transfer. Also, the patient needs to be dependent on other person for his needs and mobility. All these problems faced by a disabled person can be solved using technology. This will make the patient independent to some extent and also ease the work of attendant so that he/she can provide more effective service to the patient.

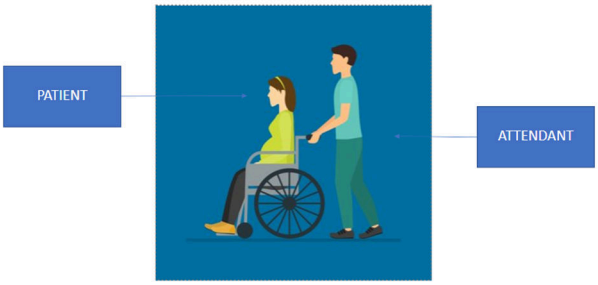

Fig. 1 Patient & Attendant

OBJECTIVE

The aim of this project is:

- to eliminate human effort to move a wheel chair
- to reduce dependency of a patient on another person
- to provide comfort to the patient operating the wheelchair
- to provide easy transfer to bed from wheelchair

DESCRIPTION OF COMPONENTS

1. Inch square bar

Fig. 2 Material used for making chair

- Cross section: 1 inch
- Weight: 8.68 kg

2. Arduino Mega:

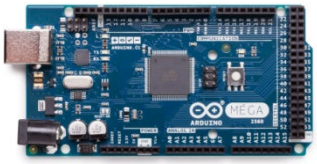

Fig. 3 Arduino Mega 2560

Microcontroller: ATmega2560
Operating Voltage: 5V
Input Voltage (recommended): 7-12V
Input Voltage (limit): 6-20V
Digital I/O Pins: 54 (of which 15 provide PWM output)
Analog Input Pins: 16
DC Current per I/O Pin: 20 mA
DC Current for 3.3V Pin: 50 mA
Flash Memory: 256 KB of which 8 KB used by bootloader
SRAM: 8 KB
EEPROM: 4 KB
Clock Speed: 16 MHz

LED_BUILTIN: 13
Length: 101.52 mm
Width: 53.3 mm
Weight: 37 g

- The Mega 2560 is a microcontroller board based on the ATmega2560.
- It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button.
- It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.
- The Mega 2560 board is compatible with most shields designed for the Uno and the former boards Duemilanove or Diecimila.

3. Monster Motor Shield:

Fig. 4 Monster Motor Shield VNH3ASP30

Voltage Max: 16V
Maximum current rating: 30 A
Practical Continuous Current: 14 A
MOSFET on-resistance: 19 mΩ (per leg)
Maximum PWM frequency: 20 kHz
Current sensing available to Arduino analog pin: Yes
Thermal Shutdown: Yes
Undervoltage and Overvoltage shutdown: Yes

- The Dual Monster Moto Shield VNH3ASP30 DC Motor Driver 2x14A (Peak 30A) is essentially a ramped-up version of Ardumoto motor driver shield. The VIN and motor out are pitched for our 5mm screw terminals (not included), making it easy to connect larger gauge wires. The L298 H-bridge is no longer available in the Motor Driver Shield Monster Moto since it has been replaced for a pair of VNH2SP30 full-bridge motor drivers. The Motor Driver Shield Monster Moto has been enhanced to drive a pair of high-current motors while the VIN and motor have been adjusted for connecting larger wires more easily. VNH3ASP30 is a full bridge motor driver intended for a wide range of automotive applications. The device incorporates a dual monolithic high side driver and two low side switches.
- The high side driver switch is designed using the STMicroelectronics well known and proven proprietary VIPower M0 technology which permits efficient integration on the same die of a true Power MOSFET with an intelligent signal/protection circuitry. The VIN and motor out are pitched for 5mm screw terminals, making it easy to connect larger gauge wires. INA and INB control the direction of each motor, and the PWM pins turn the motors on or off.
- For the VNH3ASP30, the current sense (CS) pins will output approximately 0.13 volts per amp of output current

4. HC-05 6 pin Bluetooth Module with Button:

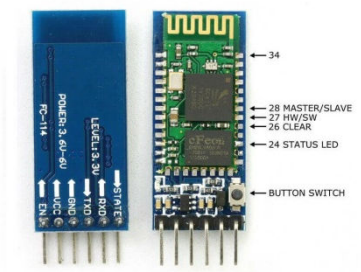

Fig. 5 Bluetooth Module HC-05

Bluetooth protocol: Bluetooth Specification v2.0+EDR.
Frequency: 2.4GHz ISM band.
Modulation: GFSK (Gaussian Frequency Shift Keying).
Emission power: =4dBm, Class 2.
Sensitivity: =-84dBm at 0.1% BER.
HC-05 6 Pin Wireless Serial Bluetooth Module is a Bluetooth module for use with any microcontroller.

- It uses the UART protocol to make it easy to send and receive data wirelessly.
- The HC-06 module is a slave only device. This means that it can connect to most phones and computers with Bluetooth but it cannot connect to another slave-only device such as keyboards and other HC-06 modules. To connect with other slave devices a master module would be necessary such as the HC-05 version which can do both master and slave.

5. Male/Female Jumper Wire:

Fig. 6 Male/Female Jumper Wire

Length: 100mm
Weight: 20 gm each
Compatible with: 2.54mm spacing pin headers

- 10pcs chromatic colour jump wire
- High quality and in good working condition
- Durable and reusable
- Easy to install and use

High Torque DC Motor with Gear Box:

Fig. 7 High Torque DC Motor with Gear Box

- The Rectangular GearBox Motor is mechanically commutated DC electric motor which simply means that these motors equip Gearboxes for providing a great balance between torque-speed as per your requirement.

- So, Rectangular GearBox Motor is a simple DC motor of 6000 RPM (Base Motor RPM) which features a heavy-duty metal gearbox. This Rectangular gearBox DC motor is a very high torque motor which should be used to make big robots or robotized platform.
- The motor will run at 100 RPM when powered with 12V DC supply. It produces the massive torque of 14kg-cm at 100 RPM.
- The motor shaft is made up of good quality Engineering steel with nickel plating to handle high Torsion Stress. The shaft of the motor equips metal bushes which makes these DC gear motors Shaft wear resistant. The Rectangular GearBox Motor's shaft is the D-type shaft which ensures strong and safe coupling.
- The Rectangular GearBox Motor is very easy to use and available in standard size. Their wide application areas include Robot Development, centralized air conditioning valve, amusement equipment, coin refund devices, grill, oven, peristaltic pumps, atm bank automatic system, medical equipment, office equipment, household appliances, automatic actuator and many other Industrial applications.

Multistranded Wires:

Fig. 8 Multistranded wire

- Multi-strand wiring is more flexible and less susceptible to cracking and metal fatigue than single stranded conductors. This makes it the preferable solution for wiring that will need to manoeuvre and bend without experiencing metal fatigue.
- The increased surface area of multi-stranded conductors decreases the amount of resistance that currents or signal passing through the wire will encounter.
- Multi-stranded wiring can create challenges when soldering connectors or terminals as it can be difficult to ensure that all of the small strands have been soldered properly. If this type of wire is soldered, it is good practice to tin the wire first. This type of wiring is better suited for use with an application that requires a crimp connector. The small, flexible strands of wire typically crimp easily and mold to the crimp connector very well.

6. SMPS:

Fig. 9 SMPS 12V 40A

Voltage:	12V
Current:	40A
Power:	480W

7. Type-B USB Cable for Arduino:

Fig. 10 Type-B USB Cable for Arduino

USB Type: Type B
Length: 3m

- This cable is used for feeding program to the Arduino.
- This cable can also be used to power the Arduino via your Laptop.

8. Arduino DC Battery Power cable plug:

Fig. 11 Arduino DC Battery Power cable plug

9. Extension Board:

Fig. 12 Extension Board

10. Front & Rear wheels:

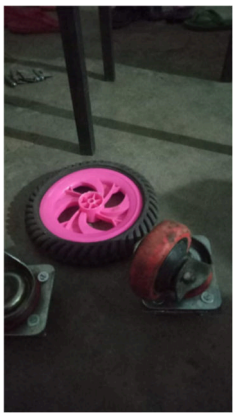

Fig. 13 Front (Red) and Rear (Pink) wheels

Front Wheels:

- Diameter: 55cm
- Wheel Thickness: 30cm
- Type: Caster wheels
- Ball Bearing: Single
- Basic Plate Size (mm): 95x75
- Load Capacity: 100 kg
- Material: Polyurethane
- Surface Finish: Zinc Plated

Rear Wheels:

- Type: Tricylce wheels
- Diameter: 8 inch/ 20cm
- Thickness: 3cm
- Bore Diameter: 7mm

CONSTRUCTION/DIMENSIONS:

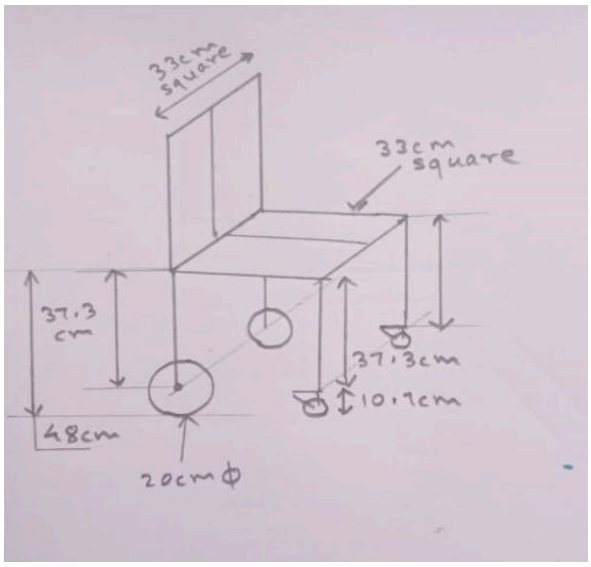

Fig. 14 Wheelchair dimensions

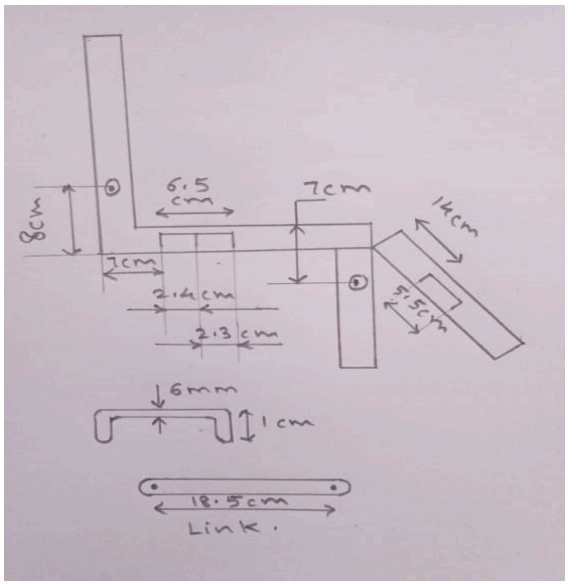

Fig. 15 Dimensions of Links and Slots for setting wheelchair positions

11. Arduino IDE (Integrated Development Environment):

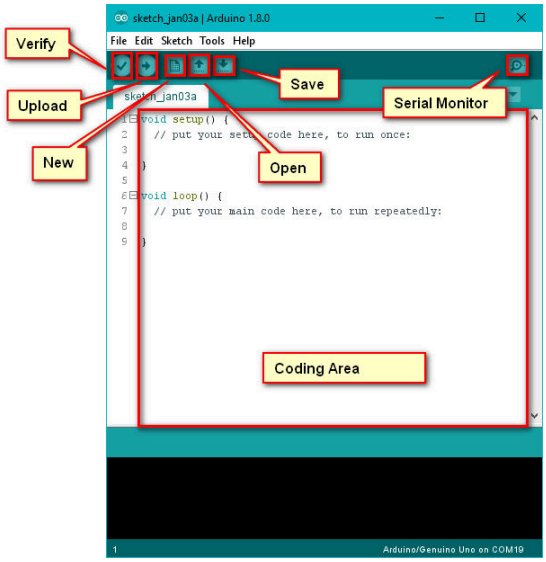

Fig. 16 Arduino IDE

12. Joystick Mobile Application used:

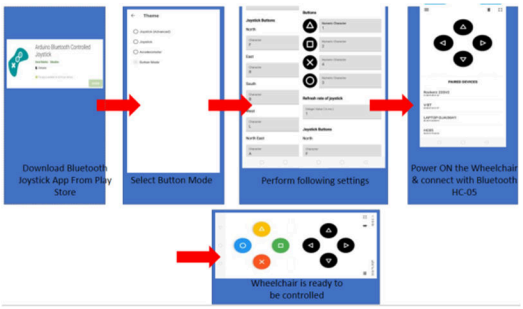

Fig. 17 Schematics for setting up Bluetooth connection with wheelchair

BLOCK DIAGRAM

Fig. 18 Block Diagram

CIRCUIT DIAGRAM

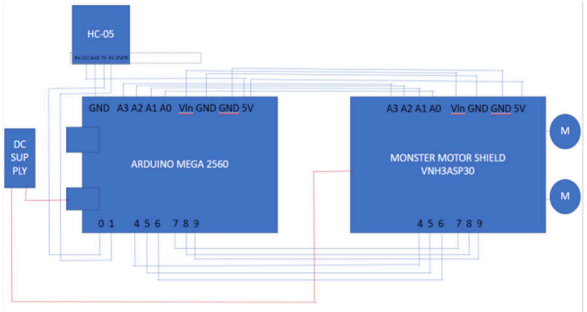

Fig. 19 Circuit Diagram

PROGRAM

```
#include <string.h>
#define BRAKE 0
```

```
#define CW    1
#define CCW   2
#define CS_THRESHOLD 15   // Definition of safety current (Check: "1.3 Monster Shield Example").

#define MOTOR_A1_PIN 8
#define MOTOR_B1_PIN 7
#define MOTOR_A2_PIN 9
#define MOTOR_B2_PIN 4
#define PWM_MOTOR_1 5
#define PWM_MOTOR_2 6
#define CURRENT_SEN_1 A2
#define CURRENT_SEN_2 A3
#define EN_PIN_1 A0
#define EN_PIN_2 A1
#define MOTOR_1 0
#define MOTOR_2 1

unsigned short usMotor_Status = BRAKE;
char recv_char;
int default_speed = 255;
static int usSpeed = default_speed;
int incremental_speed = 50;
int turning_speed = 180;

void setup() {
  pinMode(7, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(A2, OUTPUT);
  pinMode(A3, OUTPUT);
  pinMode(A0, OUTPUT);
  pinMode(A1, OUTPUT);

  digitalWrite(A0,HIGH);
  digitalWrite(A1,HIGH);
  digitalWrite(A2, HIGH);
  digitalWrite(A3, HIGH);

  Serial.begin(9600);
  Serial.println("Begin motor control");
  Serial.println("Ready to connect\nDefualt password is 1234");
}

void turn(String msg) {

  if(msg == "LEFT")
  {
      Serial.println("Left");
```

```
      digitalWrite(A0,HIGH);
      digitalWrite(A1,HIGH);

      analogWrite(PWM_MOTOR_1, turning_speed);
      analogWrite(PWM_MOTOR_2, 0);

      digitalWrite(MOTOR_B1_PIN, HIGH);
      digitalWrite(MOTOR_A1_PIN, LOW);

      digitalWrite(MOTOR_B2_PIN, LOW);
      digitalWrite(MOTOR_A2_PIN, HIGH);

   }else if(msg == "RIGHT")
   {
      Serial.println("Right");

      digitalWrite(A0,HIGH);
      digitalWrite(A1,HIGH);

      analogWrite(PWM_MOTOR_1, 0);
      analogWrite(PWM_MOTOR_2, turning_speed);

      digitalWrite(MOTOR_B1_PIN, HIGH);
      digitalWrite(MOTOR_A1_PIN, LOW);

      digitalWrite(MOTOR_B2_PIN, LOW);
      digitalWrite(MOTOR_A2_PIN, HIGH);
   }
}

void Forward()
{
  Serial.println("Forward");
  digitalWrite(A0,HIGH);
  digitalWrite(A1,HIGH);

  analogWrite(PWM_MOTOR_1, usSpeed);
  analogWrite(PWM_MOTOR_2, usSpeed);

  digitalWrite(MOTOR_B1_PIN, HIGH);
  digitalWrite(MOTOR_A1_PIN, LOW);
  digitalWrite(MOTOR_B2_PIN, LOW);
  digitalWrite(MOTOR_A2_PIN, HIGH);
}

void loop() {


 if(Serial.available() > 0){
      recv_char = Serial.read();
      Serial.println(recv_char);
```

```
      if(recv_char == 'F'){    //Forward
         Forward();
      }else if(recv_char == 'B'){ //Reverse
         Serial.println("Reverse");
         //Reverse();

      }else if(recv_char == 'L'){  //Left
         turn("LEFT");

      }else if(recv_char == 'R'){ //Right
         turn("RIGHT");

      }else if(recv_char == '1') {
         Serial.print("Changed speed to: ");Serial.println(60);
         usSpeed = 60 ;//+= incremental_speed;

      }else if(recv_char == '2'){
         Serial.print("Changed speed to: ");Serial.println(120);
         usSpeed = 120;  // -= incremental_speed;
      }else if(recv_char == '3'){
         Serial.print("Changed speed to: ");Serial.println(180);
         usSpeed = 180;  // -= incremental_speed;
      }else if(recv_char == '4'){
         Serial.print("Changed speed to: ");Serial.println(255);
         usSpeed = 255;  // -= incremental_speed;
      }
      else{
         Stop();
      }


   }else
   {
    //If data transmission stopped, set speed 0
    Serial.println("Data not received sending zero");
    Stop();
   }
  delay(40);
  Serial.flush();
}
```

WORKING

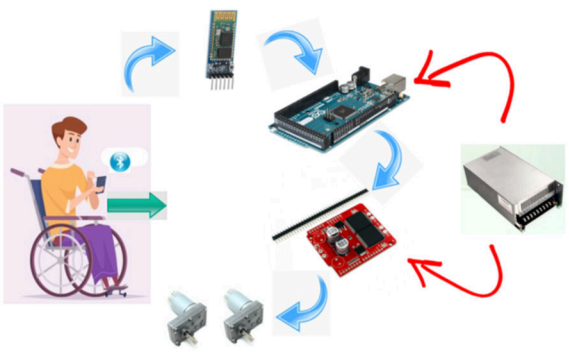

Fig. 20 Schematic Diagram to explain the working of the system

First the user needs to install any Bluetooth joystick application from play store. Then change the button settings for Forward, Left and Right to F, L and R respectively. and for speed settings put 1, 2, 3, 4. After that when we will press the forward button it will send F via Bluetooth to the Arduino.

The Arduino is uploaded with a program made on the Arduino IDE. This program receives input from the Bluetooth Module, processes the input and provides its output to the motor driver shield. The Motor driver receives input from Arduino and then runs the motors as per input received.

A 12V 40A SMPS supplies power to the Arduino and motor driver shield. Power LED of Motor driver (red) and Arduino (green) glows. Bluetooth module LED(LED) also start blinking.

Turn ON mobile Bluetooth. Open The Bluetooth Joystick App. Connect with HC-05. the Bluetooth module LED blinking pattern changes as soon as the devices are connected and paired.

password is generally 0000 or 1234

Now the wheelchair is ready to be controlled.

If user sends F, then both the motors run in the same direction with same rpm

If user sends L, respective motor runs with set rpm and the vehicle turns left

If user sends R, respective motor runs with set rpm and the vehicle turns right

the shield does not allow reverse rotation of the motors so for moving backward movement A 360 degree left or right could be taken.

CAD Model:

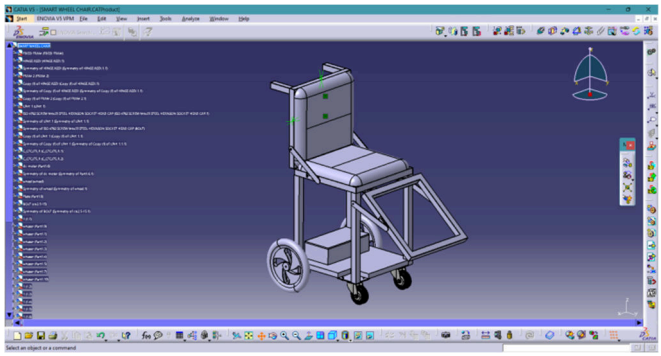

Fig. 21 CAD Model of the wheelchair

Three Chair Resting Positions for Patient's comfort

While designing the wheelchair we have considered the need of the patient. The patient could not continuously sit for long time. He/she may need to change position for better comfort. The patient sometimes may need to be transported in unconscious condition and for this a rigid wheelchair design may cause problem.

To set the wheelchair to different comfortable positions, we have used link and slot mechanism. The link is provided with a hole and a screw at one end. This end is fixed to the vertical member and can rotate about the screw. The other end is provided with a rod. This rod engages with the slot made on the square bar.

The slots are made in such a way that the upper one allows three different locking positions. And the lower one allows two locking positions.

Due to this arrangement, we can set the wheelchair in three different useful positions as shown below

Fig. 22 Sleeping Position

Fig. 23 Sitting Position

Fig. 24 Resting Position

Level Adjustment Bolt on One Leg:

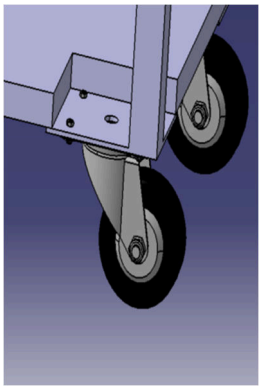

Fig. 25

Fig. 26 Bolt for adjusting level of wheel with respect to the ground

RESULT & DISCUSSION

The wheelchair has been tested and results have been found on 16g load including the external weight. The wheelchiar is designed to travel straight, left and right direction. To travel backward one has to take a 360-degree left or right turn then move straight forward. this has been done because the VNH3ASP30 that we have, do not allow rotation of wheels in reverse direction. Overall, the functioning of the wheelchair is smooth and efficient on flat surface.

FUTURE SCOPE

This prototype wheelchair makes it easy for a user to handle the wheelchair. The mobile application interface is much similar to video game joystick. Thus, the whole system is user-friendly. the motors were successfully controlled as per the command. The 3 modes of wheelchair provide comfort to the user and allows easy chair to bed transportation. If the motors are replaced with higher torque motor and accordingly power supply is provided, it can move a higher load. Thus, real life implementation of the project is feasible and easy.

REFERENCES

[1]. Deepak Kumar, Reetu Malhotra, S. R. Sharma, "Design and Construction of a small wheelchair", 9th World Engineering Education forum, WEEF 2019

[2]. Umang Garg, R. C. Joshi, Rahul Chauhan, "Design and Implementation of smart wheelchair for Quadriplegia using IOT", 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)

[3]. Shaolin Kataria, Aditya Sunil Menon, Aditya Sunil Menon, Prerna Sultania, Kakelli Anil Kumar, "A Novel IOT Based Smart Wheelchair Design for Cerebral Palsy Patient", Article in International Journal of Scientific Research in Science and Technology · August 2021 DOI: 10.32628/CSEIT2174124

[4]. Divya Jennifer Dsouza, Sanket Srivastava, Ruth Prithika, Sahana, "IoT Based Smart Wheelchair for HealthCare", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-2, July 2019

[5]. Harsh Vora, Aime Gupta, Chintan Pamnani, Tushar Jaiswal, "Multimodal Smart Wheelchair Integrated", with Safety Alert System International Journal of Engineering and Advanced Technology (IJEAT), ISSN: 2249 – 8958 (Online), Volume-9 Issue-4, April 2020