Hand Gesture Controlled Wheelchair Prototype

Group Members:

Sumit Jangir (210003073)

Tarun Kumar Patel (210003075)

Teja Ram (210003076)

Yashwant Jilowa (210003086)

1. INTRODUCTION

Problem Statement:

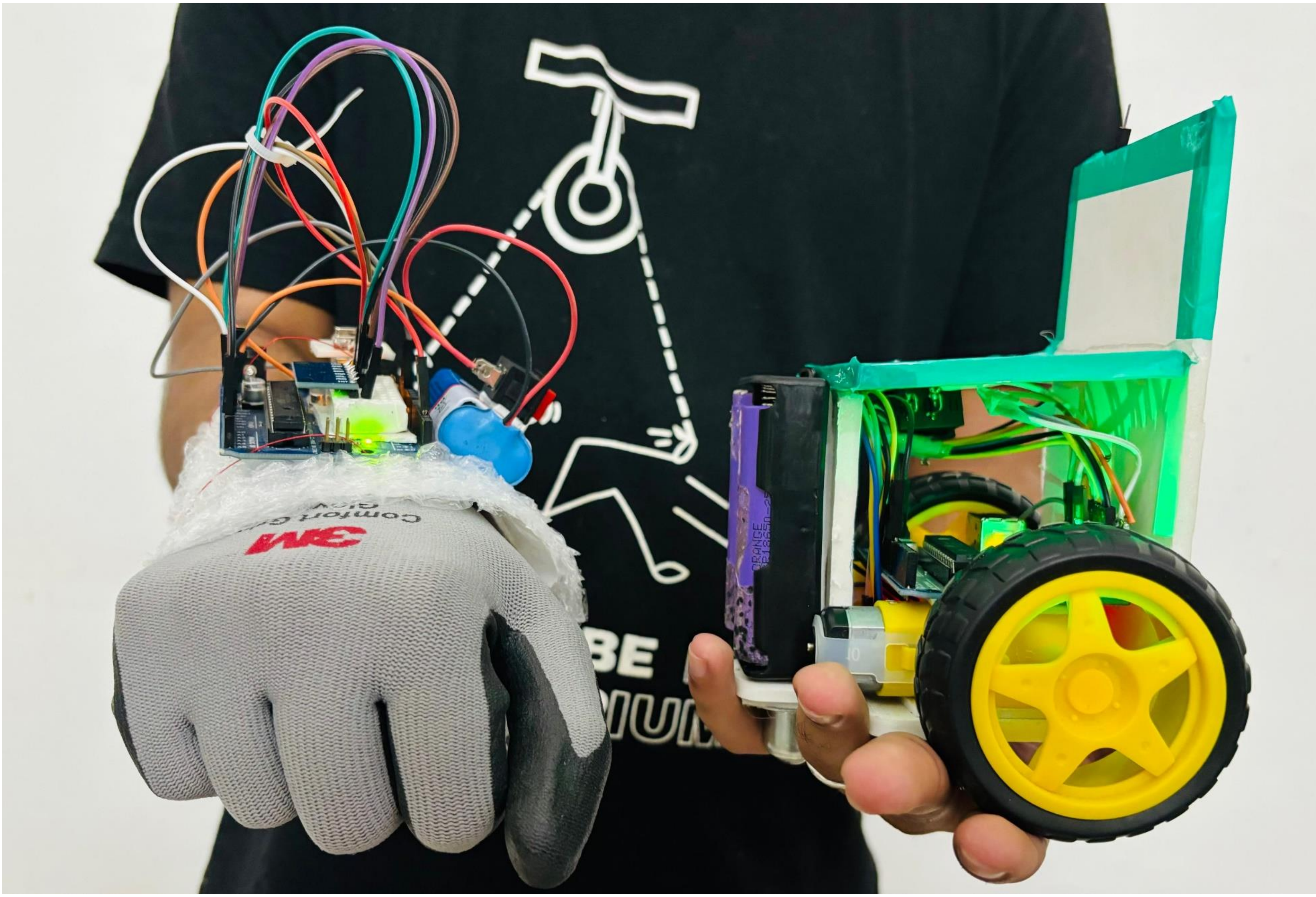
The traditional control mechanisms of wheelchairs present challenges for individuals with limited mobility, relying on manual controls that can be cumbersome and restrictive. This project addresses these limitations by developing a Hand Gesture Controlled Wheelchair.

Objectives:

- Design and develop a prototype of a wheelchair controlled by hand gestures.
- Enhance the independence and accessibility of wheelchair users.
- Provide a more intuitive and user-friendly alternative to traditional controls.

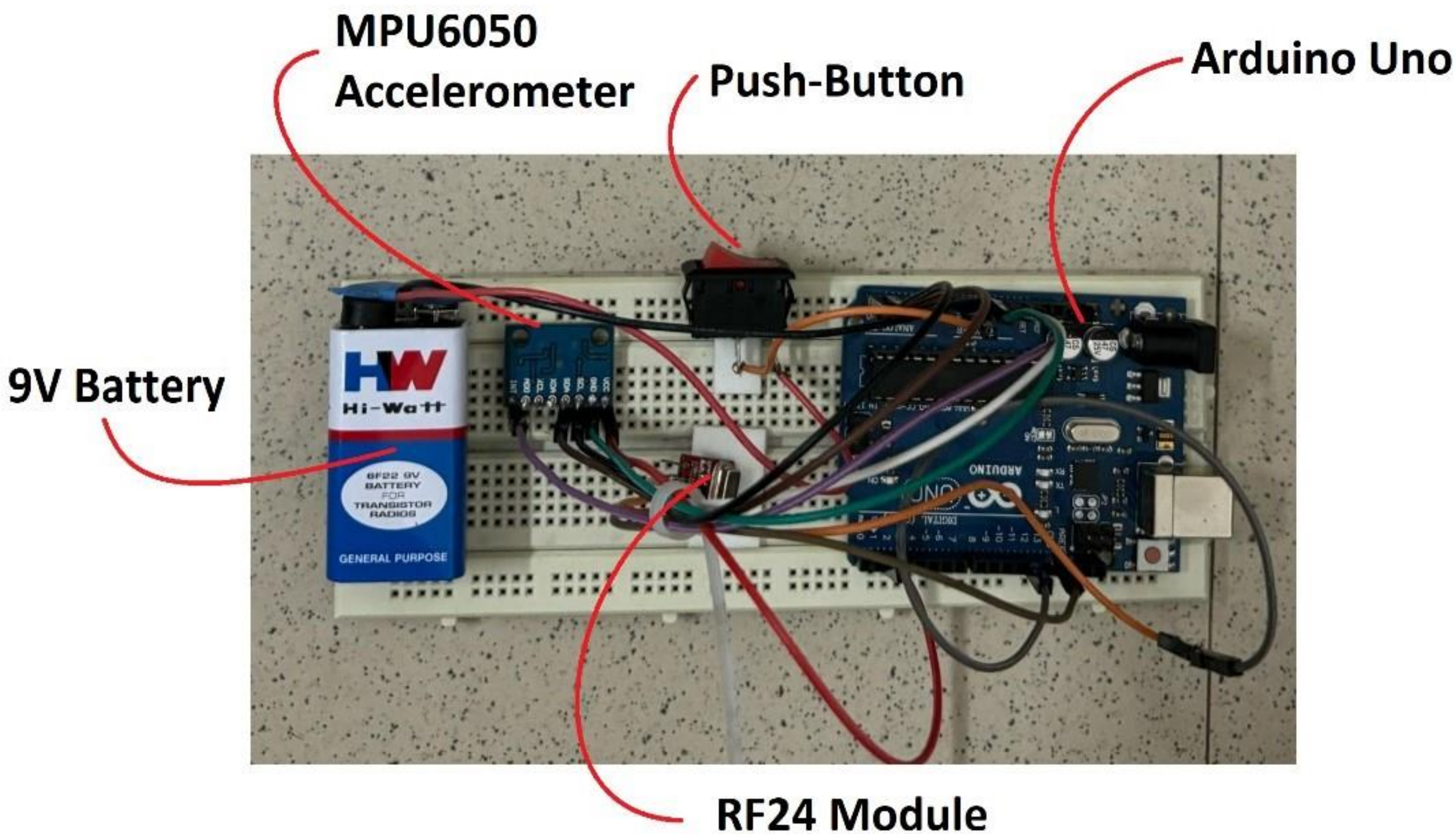
Importance:

This project has the potential to significantly improve the quality of life for wheelchair users by offering a more natural and ergonomic method of control. It aims to bridge the gap between technology and accessibility.

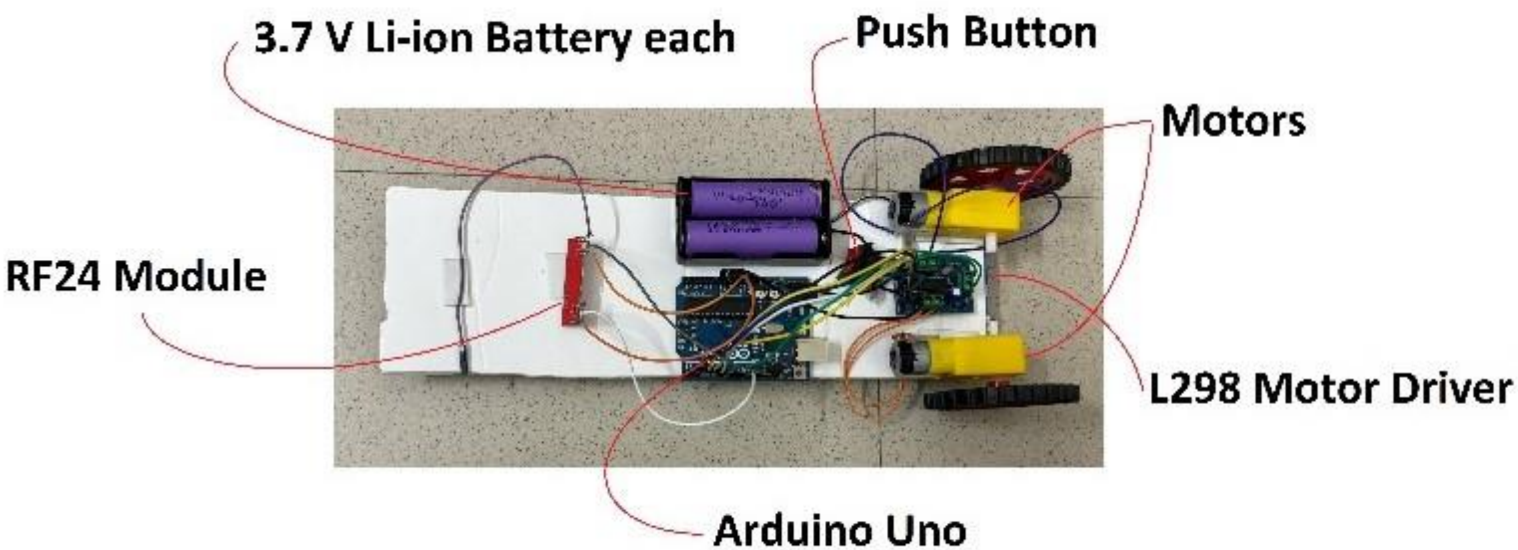

3M

2. TECHNOLOGY USED:

- In the transmitter part of the system, the project utilized an MPU6050 accelerometer as a sensor responsible for detecting the motion of the user's hand. This accelerometer, integrated with a gyroscope, provided precise measurements of acceleration and angular velocity along three axes. The MPU6050 was connected to an Arduino Uno microcontroller, serving as the brain of the transmitter module. The Arduino Uno processed the sensor data and executed the transmitter code, which interpreted hand gestures based on the detected motion. Through the transmitter code, specific gestures such as forward, backward, left, and right were defined, each corresponding to distinct patterns of hand movement.



- In the receiver part of the system, the project employed Dual H-Bridge motor drivers L298. These motor drivers are renowned for their capability to control the speed and direction of two DC motors simultaneously. Connected to another Arduino Uno microcontroller, the receiver module received the interpreted gestures from the transmitter wirelessly. The Arduino Uno on the receiver end executed the receiver code, which decoded the received gestures and converted them into control signals for the Dual H-Bridge motor drivers.



- This allowed for precise and efficient control of the wheelchair's movement in response to the user's hand gestures.

- Furthermore, the project utilized RF24 modules for wireless communication between the transmitter and receiver parts. The RF24 transmitter module was integrated into the transmitter module, while the RF24 receiver module was part of the receiver module. These modules facilitated the reliable transmission of gesture data from the transmitter to the receiver in real-time, ensuring seamless communication between the user's hand gestures and the wheelchair's movement. The use of RF24 modules provided a robust and efficient wireless link, enabling effective control of the wheelchair without the need for physical wires.
- In addition to the control system, the project incorporated two DC motors to drive the wheelchair's wheels. These motors were connected to the Dual H-Bridge motor drivers, allowing for independent control of each wheel's speed and direction. The combination of the MPU6050 accelerometer, Arduino Uno microcontrollers, Dual H-Bridge motor drivers L298, and RF24 modules resulted in a sophisticated yet user-friendly system for controlling the wheelchair through intuitive hand gestures.

3. WORKING DIRECTION OF WHEELCHAIR:

Direction of hand gesture	Movement of left Motor	Movement of right Motor
Forward	Forward	Forward
Backward	Backward	Backward
Right	Forward	stop
left	stop	Forward

When both the wheels of wheelchair rotate in forward direction then the wheelchair moves in forward direction. When both the wheel rotates in reverse direction then the wheel chair moves in backward direction. When left motor rotates and right motor shaft is stationary then the wheelchair moves in right direction, and when right motor rotates and left motor is stationary then the wheelchair moves in left direction.

4. CODE:

Transmitter Code:-

```
#include <RH_ASK.h> // Include RadioHead Amplitude Shift Keying Library
#include <SPI.h>    // Include dependent SPI Library
#include <SPI.h>        //SPI library for communicate with the nRF24L01+
#include  "RF24.h"       //The main library of the nRF24L01+
#include "Wire.h"       //For  communicate
#include "I2Cdev.h"     //For communicate with MPU6050
```

```
#include  "MPU6050.h"    //The main library of the MPU6050

MPU6050 mpu;
int16_t  ax, ay, az;
int16_t gx, gy, gz;
int data[2];
// Create Amplitude Shift Keying Object
RH_ASK rf_driver;

void setup() {
  // Initialize ASK Object
   Wire.begin();
  rf_driver.init();
  // Setup Serial Monitor
  Serial.begin(9600);
   mpu.initialize();
}

void loop() {

   mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);


 data[0] = map(ax,  -17000, 17000, 300, 400 ); //Send X axis data
  data[1] = map(ay, -17000, 17000,  100, 200);
  Serial.println(data[0]);
  Serial.println(data[1]);
```

```
  // Numeric value to be sent
 int arrayToSend[2] = {data[0], data[1]};

  // Set buffer to size of the array
  uint8_t buf[sizeof(arrayToSend)];
  memcpy(buf, &arrayToSend, sizeof(arrayToSend));

  // Send the array
  rf_driver.send(buf, sizeof(arrayToSend));
  rf_driver.waitPacketSent();
  delay(1000);
}
```

Receiver Code:-

```
#include <RH_ASK.h> // Include RadioHead Amplitude Shift Keying Library
#include <SPI.h>    // Include dependent SPI Library
#include <nRF24L01.h>
#include <printf.h>

#include <RF24.h>
#include   <RF24_config.h>
#include <SPI.h>      //SPI library for communicate with the   nRF24L01+
#include "RF24.h"     //The main library of the nRF24L01+


const   int enbA = 3;
const int enbB = 5;
```

```
const int IN1 = 2;    //Right Motor   (-)
const int IN2 = 4;    //Right Motor (+)
const int IN3 = 7;    //Left Motor   (+)
const int IN4 = 6;    //Right Motor (-)

int RightSpd = 130;
int   LeftSpd = 130;

int data[2];
// Create Amplitude Shift Keying Object
RH_ASK rf_driver;

void setup() {
  // Initialize ASK Object
  pinMode(enbA, OUTPUT);
 pinMode(enbB, OUTPUT);
  pinMode(IN1, OUTPUT);
   pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  rf_driver.init();
  // Setup Serial Monitor
  Serial.begin(9600);
}

void loop() {
  // Set buffer to size of expected message
```

```
 int receivedArray[2]; // Assuming the array contains 2 integers
uint8_t buf[sizeof(receivedArray)];
uint8_t buflen = sizeof(buf);

// Check if received packet is correct size
if (rf_driver.recv(buf, &buflen)) {
  // Message received with valid checksum

  // Interpret received bytes as an integer array
  memcpy(&receivedArray, buf, sizeof(receivedArray));
  int n1= receivedArray[0];
  int n2= receivedArray[1];
  Serial.println(n1);
  Serial.println(n2);
        if(n1 > 380){
    //forward
    analogWrite(enbA,   RightSpd);
    analogWrite(enbB, LeftSpd);
    digitalWrite(IN1, HIGH);
     digitalWrite(IN2, LOW);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4,   LOW);
  }

  if(n1 < 310){
    //backward
     analogWrite(enbA, RightSpd);
    analogWrite(enbB, LeftSpd);
```

```
    digitalWrite(IN1,   LOW);
    digitalWrite(IN2, HIGH);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4,   HIGH);
  }

  if(n2 > 180){
    //left
    analogWrite(enbA,   RightSpd);
    analogWrite(enbB, LeftSpd);
    digitalWrite(IN1, HIGH);
     digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4,   HIGH);
  }

  if(n2< 110){
    //right
    analogWrite(enbA,   RightSpd);
    analogWrite(enbB, LeftSpd);
    digitalWrite(IN1, LOW);
     digitalWrite(IN2, HIGH);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4,   LOW);
  }

  if(n1 > 330 && n1< 360 && n2 > 130 &&   n2 < 160){
    //stop car
```

```
        analogWrite(enbA, 0);
        analogWrite(enbB,   0);
      }

    }
}
```

5. CONCLUSION:

The wheelchair is fully capable of moving in accordance to the gesture given by the person who is using the wheel chair. Certain improvisation and improvement can be done to make the wheelchair more reachable to those whose whole body is paralyzed. Certain eyes gesture or brain signals reader can be imparted on the wheelchair system so as to make it better.