# Hand Gesture Controlled Car (HGCC)

## AIM:

The aim of this project is to design and implement a hand gesture-controlled car using Arduino. The car will interpret hand gestures to control its movements wirelessly.

## Components Required:

* 1. Arduino NANO
* 2. Arduino UNO
* 3. L298 Motor Driver
* 4. Car Chassis
* 5. RF Module
* 6. ROOFER INR 18650 25000 mAh Battery
* 7. Wheels
* 8. Gear Motor
* 9. Connectors
* 10. PCB
* 11. Adopter
* 12. Soldering Wire
* 13. Glove
* 14. Velcro Tape

### Arduino Nano

The Arduino Nano is an open-source breadboard-friendly microcontroller board based on the Microchip ATmega328P microcontroller (MCU) and developed by Arduino.cc. It offers the same connectivity and specs as the Arduino Uno board in a smaller form factor.

### Arduino Uno

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller (MCU) and developed by Arduino.cc.

### L298N Motor Driver

The L298N motor driver controls both the speed and direction of rotation of a DC electric motor. It uses an L298N PWM system, which can control voltage using square wave pulses.

### MPU-6050 (Accelerometer)

The MPU-6050 is a 6-axis motion tracking device combining a 3-axis Gyroscope and a 3-axis Accelerometer. It detects changes in motion, acceleration, and rotation.

### RF Module

RF transmitters and receivers are the backbone of wireless communication systems, enabling devices to transmit voice, data, and multimedia content over the airwaves.

## Circuit Diagrams

The circuit involves connecting the components as follows:

1. Connect the L298 motor driver to the gear motors and power supply.

2. Connect the Arduino Uno to the motor driver for control signals.

3. Use the RF module to enable communication between the Arduino Nano (glove) and Arduino Uno(car).

4. Interface the gesture sensors on the glove with the Arduino Nano.

5. Ensure proper connections using jumper wires and soldering where necessary.Refer to the circuit diagram for precise connections.

## Receiver Code

Code includes initialization of RF modules, setting up pins, and functions for controlling motors based on commands. Commands include FORWARD, BACKWARD, LEFT, RIGHT, and STOP.  
void loop() {

uint8\_t buf[VW\_MAX\_MESSAGE\_LEN];

uint8\_t buflen = VW\_MAX\_MESSAGE\_LEN;

if (vw\_get\_message(buf, &buflen)) {

String command = "";

for (int i = 0; i < buflen; i++) {

command += (char)buf[i];

}

Serial.println("Command received: " + command);

if (command == "FORWARD") {

moveForward();

} else if (command == "BACKWARD") {

moveBackward();

} else if (command == "LEFT") {

turnLeft();

} else if (command == "RIGHT") {

turnRight();

} else {

topMotors();

}

}

}

#include <Wire.h>

#include <MPU6050.h>

#include <VirtualWire.h>

MPU6050 mpu;

void setup() {

Serial.begin(9600);

Wire.begin();

mpu.initialize();

if (!mpu.testConnection()) {

Serial.println("MPU6050 connection failed!");

while (1);

}

Serial.println("MPU6050 initialized.");

vw\_set\_tx\_pin(10);

vw\_setup(2000);

}

## Transmitter Code

The transmitter utilizes an MPU6050 accelerometer to detect hand gestures, converting them into commands such as FORWARD, BACKWARD, LEFT, and RIGHT, which are sent wirelessly.

void moveForward() {

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

Serial.println("Moving forward");

}

void moveBackward() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

Serial.println("Moving backward");

}

void turnLeft() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

Serial.println("Turning left");

}

void turnRight() {

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

Serial.println("Turning right");

}

void stopMotors() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, LOW);

Serial.println("Motors stopped");

}

void loop() {

int16\_t ax, ay, az;

mpu.getAcceleration(&ax, &ay, &az);

String command = "";

if (ay > 15000) command = "FORWARD";

else if (ay < -15000) command = "BACKWARD";

else if (ax > 15000) command = "LEFT";

else if (ax < -15000) command = "RIGHT";

else command = "STOP";

vw\_send((uint8\_t \*)command.c\_str(), command.length());

vw\_wait\_tx();

Serial.println("Command sent: " + command);

delay(500);

}

## Working:

1. Utilizes an MPU6050 accelerometer to detect hand gestures.  
2. Transmitter converts hand tilts into commands like FORWARD, BACKWARD, LEFT, or RIGHT.  
3. Commands are sent wirelessly to the receiver via an RF module.  
4. The receiver processes commands to control the car's motors using a motor driver.  
5. Hand gestures direct the car's

## Output:

Upon successful completion of the project, the car should be able to respond to hand gestures transmitted from the glove.

For Example :-

* Moving forward when the gesture is recognized as 'forward'.
* Turning left/right based on corresponding gestures.
* Stopping when no gesture is detected.