**Hand Gesture Controlled Vehicle**

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering

in

Information Technology

by

|  |  |
| --- | --- |
| Deshmukh Vikrant Vilasrao | 218A3066 |
| Mulik Digamber Govind | 218A3069 |
| Patil Mehul Narendra | 218A3071 |

Under the Guidance of:

Ms. Leena V Ladge



Department of Information Technology

SIES graduate School of Technology

2019-20

**CERTIFICATE**

This is to certify that the IoT Mini project entitled **“Hand Gesture Controlled Robot Vehicle”** is a bonafide work of the following students, submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology.**

|  |  |
| --- | --- |
| **Deshmukh Vikrant Vilasrao** | **218A3066** |
| **Digamber Mulik Govind** | **218A3069** |
| **Mehul Patil Narendra** | **218A3071** |

Ms. Leena V Ladge Dr. Lakshmi Sudha Dr. Atul N Kemkar

Internal Guide Head of Department I/C Principal

**PROJECT REPORT APPROVAL**

This IoT Mini project report entitled ***Hand Gesture Controlled Robot Vehicle*** by following students is approved for the degree of ***Bachelor of Engineering*** in ***Information Technology.***

|  |  |
| --- | --- |
| ***Deshmukh Vikrant Vilasrao*** | ***218A3066*** |
| ***Mulik Digamber Govind*** | ***218A3069*** |
| ***Patil Mehul Narendra*** | ***218A3071*** |

**Name of External Examiner: --------------------------------**

**Signature: --------------------------------**

**Name of Internal Examiner: --------------------------------**

**Signature: --------------------------------**

**Date:**

**Place:**

**DECLARATION**

I declare that this written submission represents my ideas in my own words and where others’ ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

|  |  |  |
| --- | --- | --- |
| Deshmukh Vikrant Vilasrao | 218A3066 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Mulik Digamber Govind | 218A3069 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Patil Mehul Narendra | 218A3071 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Signature

Date:

**ACKNOWLEDGEMENT**

It gives us immense pleasure to thank Dr. Atul N Kemkar, our I/C Principal and HOD EXTC for extending his support to carry out this project and also providing us the target boards to work and develop the project. We also thank to our Head of Department Dr. Lakshmi Sudha for her support in completing the project. We wish to express our deep sense of gratitude and thank to our Internal Guide, Ms. Leena V Ladge for her guidance, help and useful suggestions, which helped in completing our project work in time. We are thankful to our Department Project coordinator Ms. Savita Lohiya, for her guidance provided whenever required.

Also, we would like to thank the entire faculty of Information Technology Department for their valuable ideas and timely assistance in this project, last but not the least, we would like to thank our non-teaching staff members of our college for their support, in facilitating timely completion of this project.

**Project Team**

Vikrant Deshmukh

Digamber Mulik

Mehul Patil

**ABSTRACT**

In this Project we are going to develop s a Hand Gesture Controlled Robot using Arduino, which can be controlled by simple hand gesture. According to the movement of the person hand, the accelerometer start moves. It is based on 3axis of accelerometer and robot move in four direction forward, backward, left and right. For sensing Human motion, we use infrared sensor, its range is 790nm wavelength from human body.

This type of robot widely used in military application, industrial robotic, construction field. In such a field, it is very risky and complicated to handle the machines through switches or remote, sometimes operator may be confused so this new concept introduces to control the machine with the movement of hand which will simultaneously control the robot.

The model projected is controlled through a motion device that is mounted on the hand gloves. This style helps physically challenged folks and additionally for sure tasks educated by human. the most aim of this style is to manage the automaton victimization hand gesture. measuring device utilized in the planning senses the direction of hand movement and sends an indication to Arduino Nano. Four main Hand gesture movements like FORWORD, BACKWORD, LEFT and RIGHT area unit detected and enforced.

In this system, a gesture driven robotic vehicle is developed, in which how the vehicle is moving i.e., control and handling is depend on user gesture. This type of control is mostly used in virtual world compute games. This control make switching system is more real and give more freedom to user.

**Contents**

|  |  |  |
| --- | --- | --- |
|  |  | **Page No.** |
| **Chapter 1** | **Introduction** | **1** |
| **Chapter 2** | **Review of Literature** | **4** |
| **Chapter 3** | **Study of Various Target Boards** | **12** |
| **Chapter 4** | **Study of various target boards** | **13-15** |
| **Chapter 5** | **Report on Proposed system and its implementation** | **16-18** |
| **Chapter 6** | **Code** | **18-21** |
| **Chapter 7** | **Test and Procedures** | **22** |
| **Chapter 8** | **Final Result** | **23** |
| **Chapter 9** | **Conclusions** | **24** |
|  | **9.1 Future Scope** | **24** |
| **References** |  | **25** |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure**  **No.** | **Figure Caption** | **Page**  **No.** |
| 4.1 | Receiver Station | 15 |
| 4.2 | Raspberry Pie | 16 |
| 5.1 | Transmitter Block Diagram | 16 |
| 5.2 | Receiver Block Diagram | 17 |
| 5.3 | Circuit diagram of Transmitter Section | 18 |
| 5.4 | MPU6050 Functioning | 18 |
| 5.5 | Transmission Timing for HT12E | 20 |
| 5.6 | Circuit Diagram of Receiver Section | 22 |
| 8.1 | Robot Vehicle | 25 |
| 8.2 | Hand Gesture Controller | 25 |

**Chapter 1**

**Introduction**

Robotics is the system which deals with construction, design and operation. This system is related to robot and their design, manufacturer, application. Robotics research today is focused on developing systems that modularity, flexibility, redundancy, fault- tolerance and some other researchers are on completely automating a manufacturing process or a task, by providing sensor based to the robot arm. In this highly developing industry and man power are critical constraints for completion of task. To save human efforts the automation playing important role in system. This system is useful for regular and frequently carried works. One of the major and most commonly performed works is picking and placing of jobs from source to destination.

In the earlier system, the motion of the human hand is sensed by the robot through sensors and it follow the same. As the person moves their hand, the accelerometer also start moving accordingly motion of the hand sensor displaces and this sensor senses object or parameter according to motion of hand.

In this system, a gesture driven robotic vehicle is developed, in which how the vehicle is moving i.e., control and handling is depend on user gesture. This type of control is mostly used in virtual world compute games. This control make switching system is more real and give more freedom to user.

1

**HAND GESTURE TECHNIQUES**

For a wide range of applications dynamic, non-contact hand gestures are used. From a selective literature review the following applications have been found: remote crane control; aircraft traffic control; human computer Interaction; virtual environments; remote robot manipulation; wearable human computer interfaces home appliance control TV control music; room lighting hearing aids weather forecasting presentations mobile phone translation jukebox and 3D Kiosk .The two common factors in all of the above applications are the use of dynamic and non-contact hand gestures.

**A. Contact and Non-Contact Hand Gestures**

It is recognized that contact-based hand gestures using a touch pad and expansion of existing handwriting recognition techniques is a possible gesture-based interface for in-vehicle secondary controls which are providing safety benefits. The contact-based hand gestures allow a more in-depth analysis of non-contact gesture recognition technologies and possible automotive applications. The three different factors are offers by non-contact gesture recognition. firstly, no working in-vehicle non-contact dynamic hand gesture-based system could be found maximum research efforts. Secondly, non-contact gestures meant there was no physical interface at all, and thirdly, dynamic noncontact gestures could possibly be used outside the vehicle, although this does not offer any safety benefits, it does offer the opportunity for further experimentation with new ideas and concepts.

**B. Dynamic and Static Hand Gestures**

According to the research, it was understood that for replace existing secondary controls by using static hand gestures, the driver would have to recall potentially hundreds of individual hand gestures each of which would map to a particular in-vehicle secondary control. This static hand gestures create too many problems because drivers are unlikely to work all these gestures and if they did the additional mental workload that providing safety benefit. Finally, use of dynamic hand gestures appears to be much clearer than static gestures with less ambiguity, only when observing human-to-human communications are present. For these basic common-sense reasons, to concentrate on researching dynamic hand gestures only was initially decided.

**C. Gesture Driver Interaction**

When reviewing previous research, it is interesting and instructive that when gestures are used the differences in different approach with driver interaction is present. In particular visual reminders, gesture location and system feedback, these are now briefly described.

**D. Gesture Location**

It is possible to perform a hand gesture practically anywhere within the drivers reach zone, there are three zones like dynamic, non- contact hand gestures can be performed for in-vehicle controls.

If a specific gesture zone is not to be used and gestures are to be used as a supplementary input method, then it could be argued there is a fourth potential location, namely at or adjacent to the relevant tactile control, this may help users with mental modeling of the gesture and aid recall.

No research has been identified on the best location of hand gestures that would provide maximum safety for in-vehicle applications, ease of use and user acceptability.

**Chapter 2**

**Review of Literature**

Tee et al. [1] Introduces telepresence robot with motion-based consideration course to situate the

robot towards consideration focuses as per human deictic motions. Motion based consideration

course is acknowledged by consolidating Localist Attractor Network (LAN) and Short-Term

Memory (STM). likewise propose various media combination dependent on setting subordinate

prioritization among the 3 kinds of broad media signs (motion, discourse source area, head area).

The technology was based on mostly image recognition and high-speed neural processing.

Rishab et al. [2] in the Circuits Communication Control and Computing conference paper studied

on Real-time dynamic gesture recognition system based on depth perception for robot

navigation. The trajectories are then classified to one of six commands for navigation a car-robot.

A 3-axis accelerometer is adopted to record a user’s hand trajectories. Simulation results show

that the classifier could achieve 92.02% correct rate. With this technological progress of

industrial production, accelerometers can be implemented in small scale vehicles where the

priority of motion is based on rapid changes in small vehicles.

SK et.al [3] ISCO contained research about various intelligent systems that can used for Object

Recognition systems using MATLAB. Most of the studies in this paper was based on use of

OpenCV and various algorithms within the OpenCV environment. Here a new segmentation

method was used to analyze the color structure in an image to identify various objects. Color

structure descriptor (CSD) was used to capture the spatial distribution of color of images. Then

the identification was carried out with keeping histogram values to a constant.

Rajesh et al. [4] The research was based on Wireless Robotic Arm Control and various small

scaled robotic mechanism that can be implemented in low powered sensors.  The implementation

is done in two versions. The first one uses the accelerometer from a smart watch and based on

the operator hand movements the robotic arm is controlled. The second one uses the

accelerometer from a smart phone and moves the robotic arm accordingly. With the smart phone

the robotic arm can be controlled in two ways by finger gestures on the touch screen or by

moving the phone in the air. The smart phone connects to a computer which controls the robotic

arm by sending commands on the serial interface. The smart phone can connect to the computer

via Wi-Fi or Bluetooth. The smart watch connects to the computer via a special wireless protocol

using a USB wireless stick, shipped with the smart watch.

Ali et al. [5] In this paper, a six-hub robot which can be constrained by motions is created. The

robot had six degrees of opportunity and is effectively open since the robot doesn’t require higher

endeavors of controlling. A human hand glove is incorporated with handset and flexes sensors;

the opposition variety of the flex sensor is transmitted to the robot hub. In view of the variety the

robot can pivot rakish or in a direct way about its hub. A basic handset is utilized so as to control

the sign which are sent and got between the automated arm and the human hand. Arduino writing

computer programs is utilized for controlling the framework.

**Chapter 3**

**Study of Various Target Boards**

**Arduino Uno: -** Arduino Uno is an open-source based on daily used hardware and software technologies. These boards are able to read inputs like light on a sensor, finger on a button, or a Twitter message which in turn gives the output like activating a motor, turning on an LED, publishing something online. Following are the characteristics of Arduino Board:

1. Microcontroller: ATmega328
2. Operating Voltage: 5V
3. Input Voltage (recommended): 7-12V and Input Voltage (limits): 6-20V
4. Digital I/O Pins: 14 (of which 6 provide PWM output)
5. Analog Input Pins: 6
6. DC Current per I/O Pin: 40 mA
7. DC Current for 3.3V Pin: 50 mA
8. Flash Memory: 32 KB of which 0.5 KB used by bootloader
9. SRAM: 2 KB (ATmega328)
10. EEPROM: 1 KB (ATmega328)
11. Clock Speed: 16 MHz

**Programming: -** Arduino uses Arduino IDE to compile and upload its programs. Arduino uses C/C++ programming language to perform its functions on board. All standard C/C++ functions works on Arduino.

1. First, setup the port and connector of the board.
2. In Arduino IDE, there are three parts of programming: -
3. Initialization
4. Void setup()
5. Void loop()
6. Then, void setup() function is used to declare the port number used on board.
7. Then, void loop() function is where the actions of the code is written.
8. Now, compile the program and upload in Arduino board.

**Interfacing of Sensors: -**

1. Temperature Sensor Module
2. LDR and IR sensor module
3. Alcohol Gas Detector sensor module
4. Ultrasonic Distance sensor
5. Bluetooth sensor
6. WIFI sensor
7. Servo motor

**Intel Galileo: -**

Intel Galileo is an [Arduino](https://en.wikipedia.org/wiki/Arduino)-certified development board based on [Intel](https://en.wikipedia.org/wiki/Intel) x86 architecture.

Intel has two versions of Galileo, which are Gen 1 and Gen 2. These boards are also called "Breakout boards". It combines Intel technology with support for Arduino ready-made hardware expansion cards (shields) and the Arduino software development environment and libraries.

**Characteristics: -**

* Shield Compatibility - The expansion header on the top of Galileo should look familiar since it’s compatible with 5V and 3.3V. Arduino shields designed for the Uno R3 .
* Familiar IDE - The Intel-provided integrated development environment for the Galileo looks exactly like the Arduino IDE on the surface.
* Real Time Clock - Most Linux boards rely on a connection to the Internet to get the current date and time. But with Galileo’s on-board RTC (real time clock), you’ll be able to track time even when the board is powered off.
* Serial Connectivity - Not only is there the typical serial port for your sketches on pins 0 and 1 of the Arduino pinout, but there’s also a separate serial port for connecting to the Linux command line from your computer.

**Interfacing of Sensors: -**

* 16x2 LCD RGB backlight: -

It is almost similar to other 16x2 character LCDs with one different feature i.e. the backlight used is an RGB LED. The backlight of the display can be changed to any color by controlling the three backlight levels. It uses HD44780 parallel interface.

The backlight is controlled by pins 18, 17, and 16 with a common Cathode on pin 15.

A circuit board

Description automatically generated

Fig.4.1: Interfacing of LCD with Intel Galileo

* Sound sensor
* LED socket kit (Red, Green, Blue)
* Touch sensor
* Light sensor
* Buzzer
* Rotary angle sensor
* Button
* Temperature sensor
* Relay
* Servo motor with accessories

**Raspberry Pi: -**

A Raspberry Pi is a credit card-sized computer originally designed for education, inspired by the 1981 BBC Micro. The Raspberry Pi is slower than a modern laptop or desktop but is still a complete Linux computer and can provide all the expected abilities that implies, at a low-power consumption level. The Raspberry Pi is open hardware, with the exception of the primary chip on the Raspberry Pi, the [Broadcom SoC](http://www.raspberrypi.org/documentation/hardware/raspberrypi/) (System on a Chip), which runs many of the main components of the board–CPU, graphics, memory, the USB controller, etc.

Some of the characteristics of Raspberry Pi 3 are as follows: -

* CPU: Quad-core 64-bit ARM Cortex A53 clocked at 1.2 GHz
* GPU: 400MHz VideoCore IV multimedia
* Memory: 1GB LPDDR2-900 SDRAM (i.e. 900MHz)
* USB ports: 4
* Video outputs: HDMI, composite video (PAL and NTSC) via 3.5 mm jack
* Network: 10/100Mbps Ethernet and 802.11n Wireless LAN
* Peripherals: 17 GPIO plus specific functions, and HAT ID bus
* Bluetooth: 4.1
* Power source: 5V via MicroUSB or GPIO header
* Size: 85.60mm × 56.5mm

A circuit board

Description automatically generated

Fig. 4.2: Raspberry Pi

**Chapter 5**

**Report on Proposed System and its implementation**

As per our system, below is the required Block diagram along with circuit diagram.

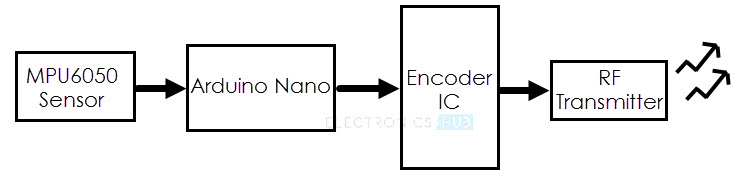


Fig 5.1 **Transmitter Block Diagram**

The MPU6050 sensor consists of 3 Axis Gyroscope and Accelerometer, it collects the data in real time and provides it to the Arduino Nano where its processed and sent to the Encoder IC which then transmits real time data to the RF Receiver.

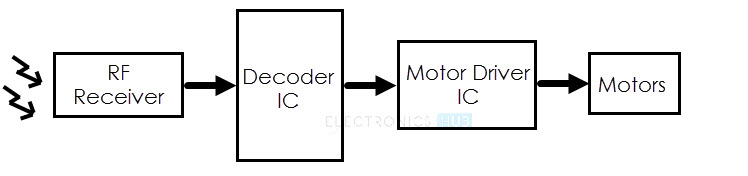


Fig 5.2 **Receiver Block Diagram**

Ref-Electrohub.com

**RF Receiver Receives real time data from the Transmitter, then it forwards this data to Decoder IC where the data is decoded and fed to the Motor Driver IC with appropriate movements for the motors to operate on.**

**Circuit Diagram of the Transmitter Section**

The following image shows the circuit diagram of the Transmitter part of the Hand Gesture Controlled robot.

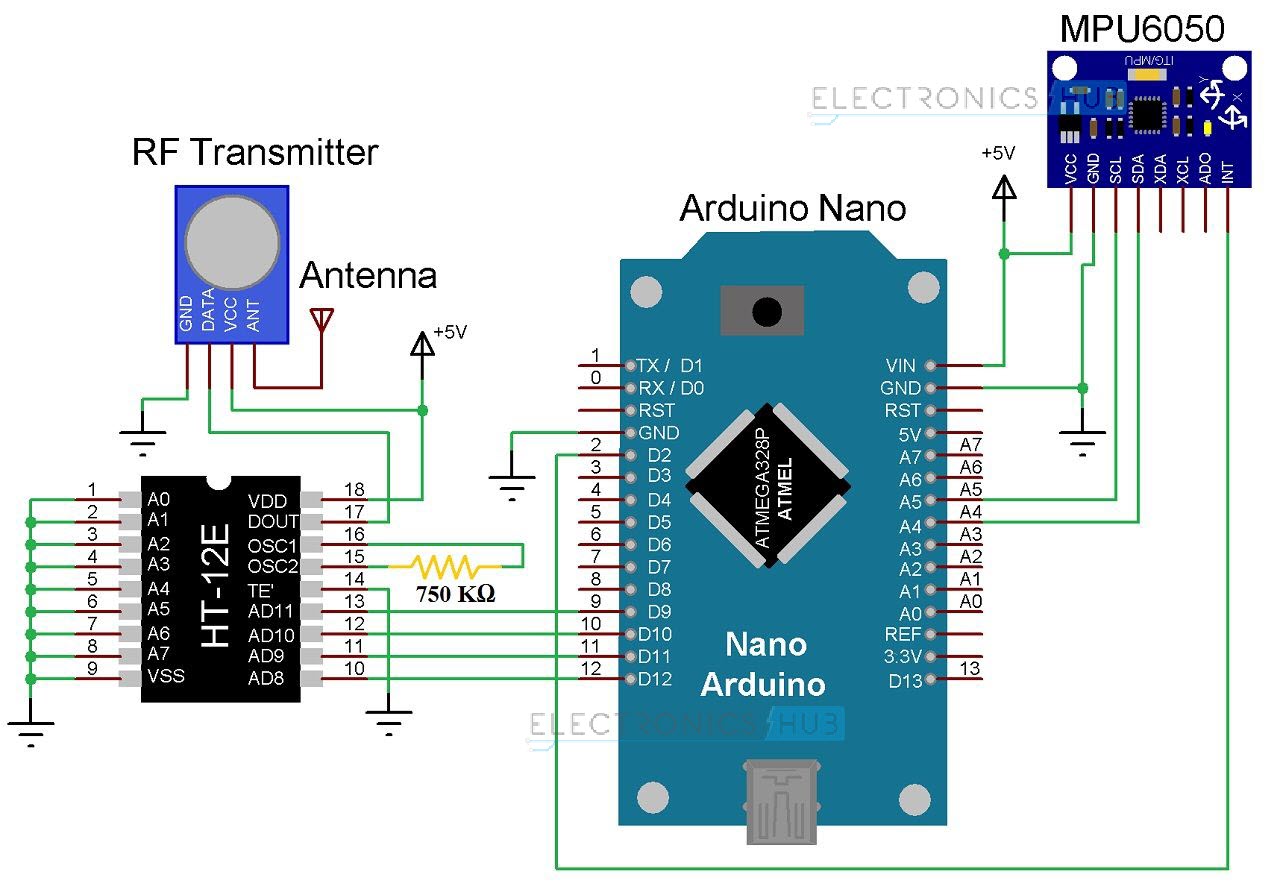


Fig 5.3 Transmitter Circuit

Ref-Electrohub.com

There are totally 14 digital Pins and 8 Analog pins on your Nano board. The digital pins can be used to interface sensors by using them as input pins or drive loads by using them as output pins. A simple function like pinMode() and digitalWrite() can be used to control their operation. The operating voltage is 0V and 5V for digital pins. The analog pins can measure analog voltage from 0V to 5V using any of the 8 Analog pins using a simple function liken analogRead()

These pins apart from serving their purpose can also be used for special purposes which are discussed below:

**Serial Pins 0 (Rx) and 1 (Tx):** Rx and Tx pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB to TTL serial chip.

**External Interrupt Pins 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.

**PWM Pins 3, 5, 6, 9 and 11:** These pins provide an 8-bit PWM output by using analogWrite() function.

**SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK):** These pins are used for SPI communication.

**In-built LED Pin 13:** This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.

**I2C A4 (SDA) and A5 (SCA):** Used for IIC communication using Wire library.

**AREF:** Used to provide reference voltage for analogue inputs with analogReference() function.

**Reset Pin:** Making this pin LOW, resets the microcontroller.

**MPU6050-**

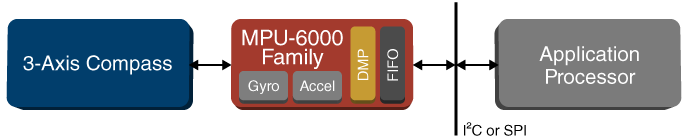


Fig: 5.4 MPU6050 Function

The MPU-6050 devices combine a 3-axis gyroscope and a 3-axis accelerometer on the same silicon die, together with an onboard Digital Motion Processor™ (DMP™), which processes complex 6-axis Motion Fusion algorithms. The device can access external magnetometers or other sensors through an auxiliary master I²C bus, allowing the devices to gather a full set of sensor data without intervention from the system processor. The devices are offered in a 4 mm x 4 mm x 0.9 mm QFN package

**HT12E-(ENCODER)**

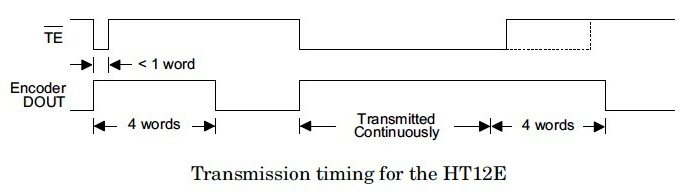


Fig-5.5 Transmission Timing for HT12E

The HT12E 212 series encoder starts a 4 word transmission cycle upon receiving transmission enable signal on TE input. This output cycle will repeats as long as the transmission is enabled. When the transmission enable (TE) signal switches to HIGH, the encoder output completes the current cycle and stops as shown above. The encoder will be in the Standby mode when the transmission is disabled.

**RF Transmitter-**

This RF module comprises of an RF Transmitter and an RF Receiver. The transmitter/receiver (Tx/Rx) pair operates at a frequency of 434 MHz’s An RF transmitter receives serial data and transmits it wirelessly through RF through its antenna connected at pin4. The transmission occurs at the rate of 1Kbps – 10Kbps.The transmitted data is received by an RF receiver operating at the same frequency as that of the transmitter.

**Components for Transmitter Section**

* Arduino Nano
* 434MHz RF Transmitter
* HT-12E Encoder IC
* MPU6050 Accelerometer/Gyroscope Sensor
* 750KΩ Resistor

**Circuit Diagram of the Receiver Section**

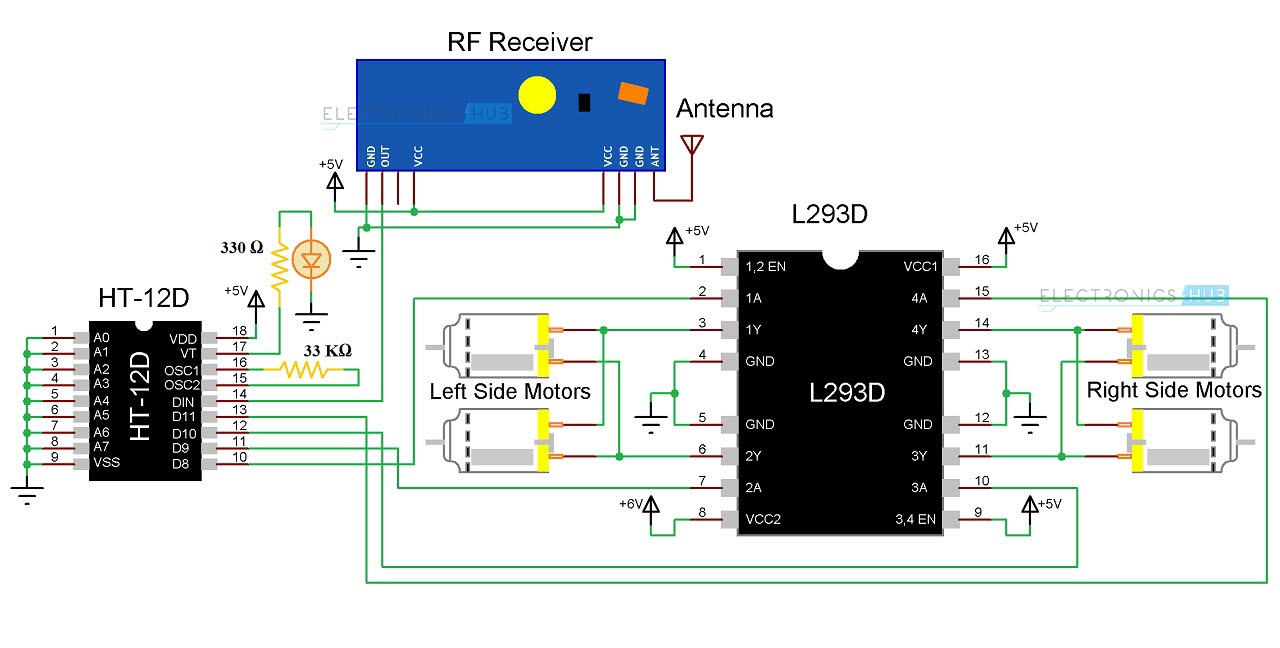


Fig 5.6 Receiver Section

*Ref-Electrohub.com*

**Components for Receiver Section**

* L293D Motor Driver IC
* HT-12D Decoder IC
* 434 MHz RF Receiver
* 33KΩ Resistor
* 330Ω Resistor
* LED
* 4 Geared Motors with Wheels
* Robot Chassis

**L293D Motor Driver IC -**

It works on the concept of H-bridge. H-bridge is a circuit which allows the voltage to be flown in either direction. As you know voltage need to change its direction for being able to rotate the motor in clockwise or anticlockwise direction, Hence H-bridge IC are ideal for driving a DC motor.

In a single L293D chip there are two h-Bridge circuit inside the IC which can rotate two dc motor independently. Due its size it is very much used in robotic application for controlling DC motors. Given below is the pin diagram of a L293D motor controller**.**

There are two Enable pins on l293d. Pin 1 and pin 9, for being able to drive the motor, the pin 1 and 9 need to be high. For driving the motor with left H-bridge you need to enable pin 1 to high. And for right H-Bridge you need to make the pin 9 to high. If anyone of the either pin1 or pin9 goes low then the motor in the corresponding section will suspend working. It’s like a switch.

**HT-12D Decoder IC-**

HT12D decoder will be in standby mode initially ie, oscillator is disabled and a HIGH on DIN pin activates the oscillator. Thus the oscillator will be active when the decoder receives data transmitted by an encoder. The device starts decoding the input address and data. The decoder matches the received address three times continuously with the local address given to pin A0 – A7. If all matches, data bits are decoded and output pins D8 – D11 are activated. This valid data is indicated by making the pin VT (Valid Transmission) HIGH. This will continue till the address code becomes incorrect or no signal is received.

**Chapter 6**

**Code**

#include <SPI.h> //SPI library for communicate with the nRF24L01+

#include "Wire.h" //For communicate

#include "I2Cdev.h" //For communicate with MPU6050

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

//Define the object to access and cotrol the Gyro and Accelerometer (We don't use the Gyro data)

MPU6050 mpu;

int16\_t ax, ay, az;

int16\_t gx, gy, gz;

const int IN1 = 8; //Right Motor (-)

const int IN2 = 9; //Right Motor (+)

const int IN3 = 7; //Left Motor (+)

const int IN4 = 6; //Right Motor (-)

//Define packet for the direction (X axis and Y axis)

int data[2];

void setup(void){

//Define the motor pins as OUTPUT

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

pinMode(IN3, OUTPUT);

pinMode(IN4, OUTPUT);

Serial.begin(9600);

Wire.begin();

mpu.initialize(); //Initialize the MPU object

}

void loop(void){

//With this function, the acceleration and gyro values of the axes are taken.

//If you want to control the car axis differently, you can change the axis name in the map command.

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

//In two-way control, the X axis (data [0]) of the MPU6050 allows the robot to move forward and backward.

//Y axis (data [0]) allows the robot to right and left turn.

data[0] = map(ax, -17000, 17000, 300, 400 ); //Send X axis data

Serial.println(data[0]);

data[1] = map(ay, -17000, 17000, 100, 200); //Send Y axis data

if(data[0] > 375){

//forward

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

if(data[0] < 330){

//backward

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

}

if(data[1] > 180){

//left

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

}

if(data[1] < 110){

//right

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

if(data[0] > 330 && data[0] < 360 && data[1] > 130 && data[1] < 160){

//stop car

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, LOW);

}

}

**Chapter 7**

**Test and Procedures**

Because of transmitter device wearing on hand and receiver on the robot, the robot starts moving according to the movement of hand gestures. In this paper, we have explained about the 5 different hand gesture or movement positions i.e. stop condition, forward movement, backward movement, moves towards right and moves towards left.

**4.1. Stop Condition**

The robot can be stopped by making the accelerometer parallel to the horizontal plane; this makes all the output pins of decoder (13, 12, 11, 10) set to high.

**4.2 Forward Movement**

The robot starts moving in forward direction, by making accelerometer tilted to forward direction, this condition sets the two output pin of decoder (13, 11) to low and set high on the other two output pin of decoder (12, 10).

**4.3 Backward Movement**

The robot starts moving in forward direction, by making accelerometer tilted to forward direction (upwards), this condition sets the two output pin of decoder (13, 11) to high and set low on the other two output pin of decoder (12, 10).

**4.4 Moves towards Right**

The robot starts move towards right side by tilting the accelerometer towards right, and this makes the two output pin of decoder (12, 11) low and other two output pin of decoder (13, 10) high.

**4.5 Moves towards Left**

The robot starts move towards left side by tilting the accelerometer towards left, and this makes the two output pin of decoder (12, 11) high and other two output pin of decoder (13, 10) low.

**Chapter 8**

**Final Result**

**A picture containing ground, floor, outdoor, toy

Description automatically generated**

Fig 8.1 Robot Vehicle

Final working model of the Vehicle in which all the components are Assembled together on the Chassis along with the 4 Wheels attachment and Battery Pack.

**A picture containing person, man, building, wall

Description automatically generated**

Fig 8.2 Hand Gesture Controller

Hand Glove Controller for the vehicle which contains the RF Transmitter and a small battery backup for the controller.

**Chapter 9**

**Conclusion**

In our project we have added special features by which our robot can overcome so many problems in industry. If it is further developed then it can be used for military application. A Gesture Controlled robot with Arduino Uno microcontroller has been designed in this work, which can be controlled by human hand gestures. This requires to wear a small transmitting device on our hand included an accelerometer, which transmits particular commands to the robot to move according to the users hand gesture and one receiver at the robot.

**9.1 Future Scope**

This project can be enhanced using voice circuit in this for deaf and dumb people. Voice circuit converts gestures into voice. With voice circuit implemented this will be useful for Animal Planet, Discovery people for their studies on animals by playing different sounds & for their exploration. Further we can add GPRS and GPS modules for place location. We can add video camera for live streaming.

**References**

1. Althoff, F., McGlaun, G., Lang, M., Rigoll, G. “Comparing an Innovative 3D and a Standard 2D User Interface for Automotive Infotainment Applications”, Paper. Munich: Technical University of Munich, (2016)
2. Alpern, M., Minardo, K. “Developing a Car Gesture Interface For Use as a Secondary Task”, Paper PA15213-3891 USA Pittsburgh, USA: Carnegie Mellon University, (n.d.)
3. Baudel, T., Beaudouin-Lafon, M. “CHARADE: Remote control of Objects using Free-Hand Gestures”, Paper. Paris: Universite de Paris, (2017)
4. Fels, S. S., Hinton, G. E. “Glove Talk II - a neural network interface which maps gestures to parallel formant speech synthesizer controls”, Neural Networks, IEEE Transactions. 8, Issue: 5, 977-984, (2018)
5. Harish Kumar Kaura, Vipul Honrao, Sayali Patil, Pravish Shetty, “Gesture Controlled Robot using Image Processing”, International Journal of
6. Advanced Research in Artificial Intelligence (IJARECE), PP.-69-77, Vol-2, No.-5[2016].
7. Aditya Purkayastha, Akhil Devi Prasad, Arunav Bora, Akshay kumar Gupta, Pankaj Singh, “Hand Gestures Controlled Robotic Arm”, Journal of International
8. Academic Research for Multidisciplinary, Vol-2, Issue-4, PP.-234-240, May 2018.
9. Love Aggarwal, Varnika Gaur, Puneet Verma, “Design and Implementation of Wireless Gesture Controlled Robotic Arm with Vision”, International Journal of Computer Application, Vol-79, No.-13, PP.-39-43, October 2015.
10. Vicky Somkuwar, Roshan Gabhane, Sandeep kakde, “Design and Implementation of Gesture Controlled Robot using Flex sensor and Accelerometer”.
11. Ieeexplorer.com
12. Arduino.cc
13. ResearchGate.com