Wireless Gesture Controlled Car

NAME	SECTION	ENROLLMENT NUMBER
GAURI SINHA	4	U101116FCS 038
GEHNA AHUJA	5	U101116FCS 040
ADYA MAHLAWAT	6	U101116FCS 004
CHIRAG JAIN	4	U101116FCS 027

Abstract

Nowadays, robotics are becoming one of the most advanced in the field of technology. The applications of robotics mainly involve in automobiles, medical, construction, defence and also used as a fire fighting robot to help the people from the fire accident. But, controlling the robot with a remote or a switch is quite complicated. So, a new project is developed that is, an accelerometer based gesture control robot. The main goal of this project is to control the movement of the robot with hand gesture using accelerometer.

In this wireless gesture controlled car project we are going to control a robot using hand gestures. This is an easy, user-friendly way to interact with robotic systems and robots. An accelerometer is used to detect the tilting position of your hand, and a microcontroller gets different analogue values and generates command signals to control the robot.

Introduction

Everyone has imagined being able to move an object with a flick of their wrist at their whim at some point or the other. A very natural tendency to move objects with the movement of ones hand is something of great convenience, and to be able to control a car, its speed, direction with the same motions is every child's dream. In our project we have

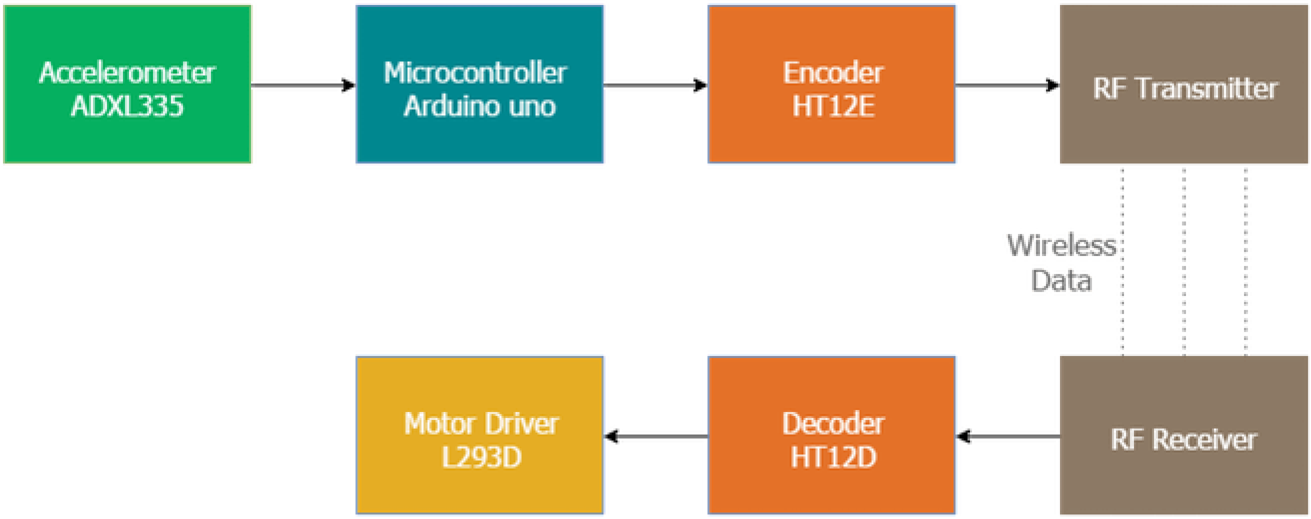
put together a car, which is a model of the cars we drive today, with servo motors controlling the direction of the car's motion, which is controlled by a glove. The glove is worn by the user, is mounted by accelerometers and it transmits data wirelessly to the car.

History

New intelligent algorithms could help robots to quickly recognize and respond to human gestures. Researchers in past have created a computer program which recognises human gestures quickly and accurately, and requires very little training. Interface improvements more than anything else has triggered explosive growth in robotics. Teitelman in 1964 developed the first trainable gesture recognizer. It was quite common in light-penbased systems to include some gesture recognition, for example in the AMBIT/G system (1968 -- ARPA funded). A gesture-based text editor using proof-reading symbols was developed at CMU by Michael Coleman in 1969. Gesture recognition has been used in commercial CAD systems since the 1970s. Furthermore, the research that will lead to the user interfaces for the computers of tomorrow is happening at universities and a few corporate research labs.

Circuit and working

Here the brain of the robot is Arduino Uno (Atmega32) it is fed with some set of code. The gestures/motion made by hand are recognized by a acceleration measuring device called accelerometer (ADXL345).



Here the accelerometer reads the X Y Z coordinates when we make gestures by hand and send the X Y Z coordinates to the Arduino (here we don't need the Z axis we need only two coordinated X and Y So neglect the Z coordinate). The Arduino checks the values of coordinates and sends a 4 bit code to the Encoder IC. The Encoder passes the data to RF transmitter and the transmitted data is received by the RF receiver. The receiver sends the 4 bit code to the Decoder IC and the decoder passes it to Motor Driver IC. Later the motor driver makes the decision to turn the two motors in the required direction.

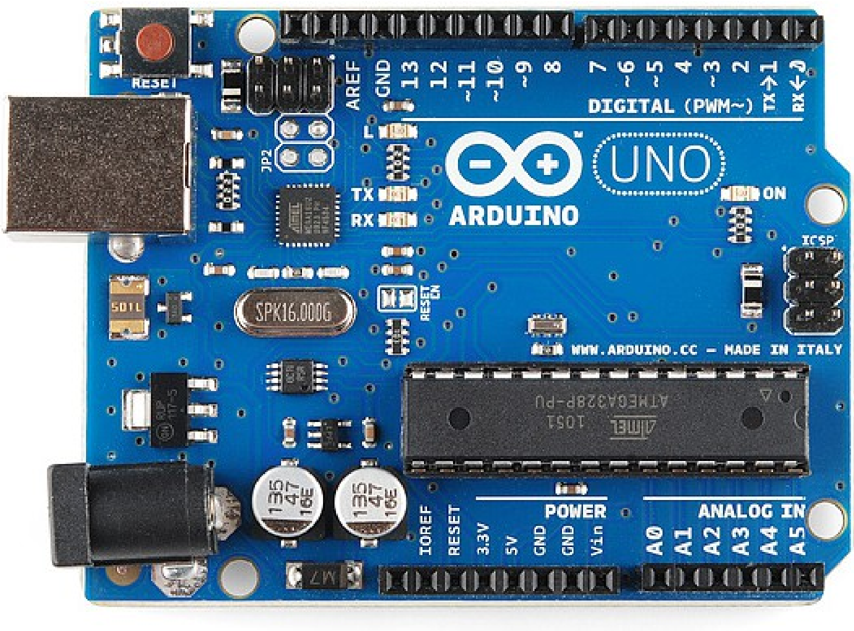
ATmega32

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load new code onto the board – you can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Finally, Arduino provides a standard form factor that breaks out the functions of the micro-controller into a more accessible package.

Arduino/Genuino Uno is a microcontroller board based on the ATmega32. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, 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.. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.



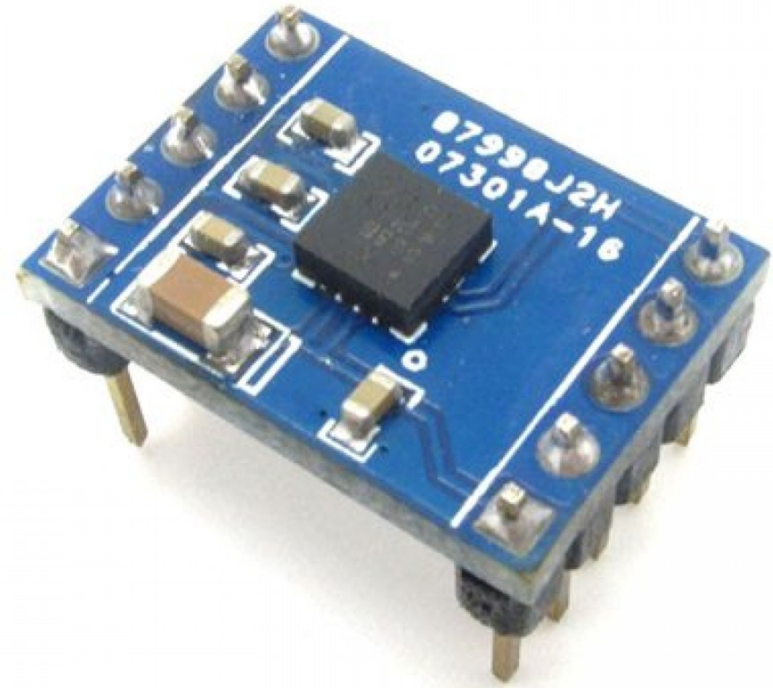
Accelerometer (ADXL345)

This is a complete three-axis acceleration measurement system. ADXL335 has a minimum measurement range of ±3g. It contains a poly-silicon-surface micro-machined sensor and signal-conditioning circuitry to implement open-loop acceleration measurement architecture. Output signals are analogue voltages that are proportional to acceleration.

The accelerometer can measure the static acceleration of gravity in tilt-sensing applications as well as dynamic acceleration resulting from motion, shock or vibration.

The sensor is a poly-silicon-surface micro-machined structure built on top of a silicon wafer. Poly-silicon springs suspend the structure over the surface of the wafer and provide resistance against acceleration forces. Deflection of the structure is measured using a differential capacitor that consists of independent fixed plates and plates attached to the moving mass.

Fixed plates are driven by 180° out-of-phase square waves. Acceleration deflects the moving mass and unbalances the differential capacitor, resulting in a sensor output whose amplitude is proportional to acceleration. Phase-sensitive demodulation techniques are then used to determine the magnitude and direction of the acceleration.



Motor Driver (L293D)

The motor driver is a device which gives the movement to do a task like a motor. So we require motor driver to run them through the controller. The interface between motor & microcontroller can be done using an L293D motor driver IC in this circuit.

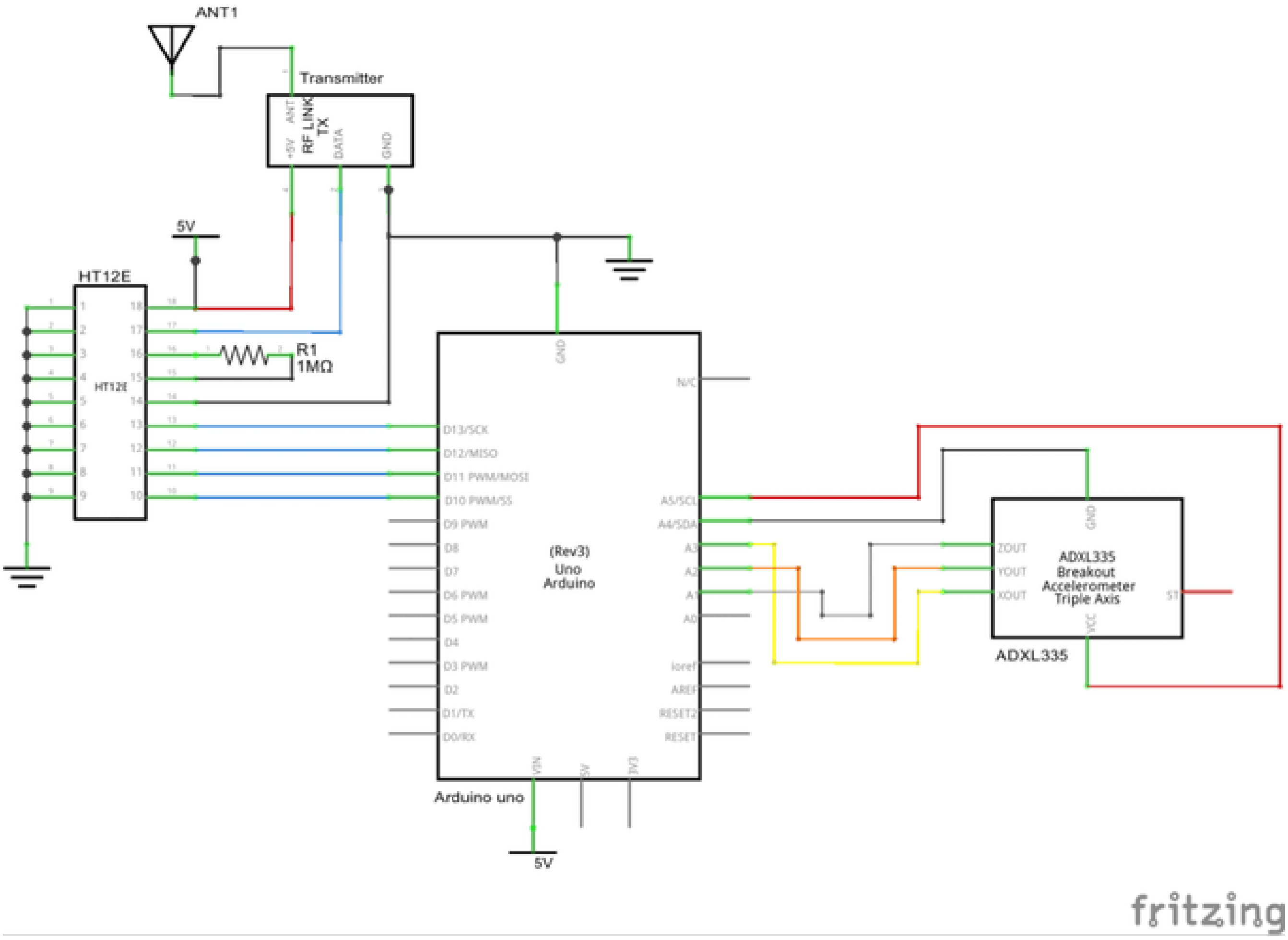
At the receiver section, an RF receiver module receives the data from the transmitter. The received data can be decoded by an IC HT12D. The received data can be

processed by AT89S51 microcontroller and motor driver is used to control the motor.

This is a 16-pin DIP package motor driver IC (IC6) having four input pins and four output pins. All four input pins are connected to output pins of the decoder IC (IC5) and the four output pins are connected to DC motors of the robot. Enable pins are used to enable input/output pins on both sides of IC6.

Transmitter (Remote)

The transmitter section consists of an accelerometer which detects the hand gesture and sends the data to the Arduino. Later Arduino sends data to the Encoder IC in accordance to the data received from accelerometer and the data is transmitted to the receiver. Wire up as per the below circuit:



Software program

The software program is written in Arduino programming language.

/*

*** This program lets you to control your robot with gesture made by your hand**

***/**

```
int GNDPin=A4; //Set Analog pin 4 as GND

int VccPin=A5; //Set Analog pin 5 as VCC

int xPin=A3; //X axis input

int yPin=A2; //Y axis input

int zPin=A1; //Z axis input(not used)

int Q1=10,Q2=11,Q3=12,Q4=13; //Output pins to be
connected to 10, 11, 12, 13 of Decoder IC

long x; //Variabe for storing X coordinates

long y; //Variabe for storing Y coordinates

long z; //Variabe for storing Z coordinates

void setup()
```

```
{
  Serial.begin(9600);
pinMode(Q1,OUTPUT);
  pinMode(Q2,OUTPUT);
  pinMode(Q3,OUTPUT);
  pinMode(Q4,OUTPUT);
  pinMode(GNDPin, OUTPUT);
  pinMode(VccPin, OUTPUT);
  digitalWrite(GNDPin, LOW); //Set A4 pin LOW
  digitalWrite(VccPin, HIGH); //Set A5 pin HIGH
}
```

```
void loop()

{

 x = analogRead(xPin); //Reads X coordinates

 y = analogRead(yPin); //Reads Y coordinates

 z = analogRead(zPin); //Reads Z coordinates (Not Used)

  if(x<340)     // Change the value for adjusting sensitivity

   forward();

  else if(x>400) // Change the value for adjusting sensitivity

   backward();

  else if(y>400) // Change the value for adjusting sensitivity

   right();
```

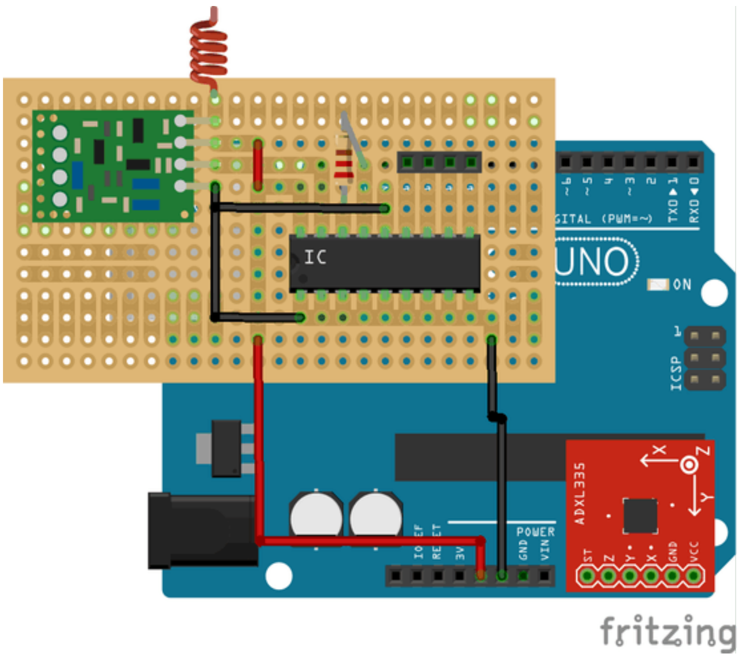
```
  else if(y<340) // Change the value for adjusting sensitivity
    left();
  else
    stop_();
}
void stop_()
{
 Serial.println("");
 Serial.println("STOP");
 digitalWrite(Q1,LOW);
 digitalWrite(Q2,LOW);
```

```
  digitalWrite(Q3,LOW);

  digitalWrite(Q4,LOW);

}

void forward()

{

  Serial.println("");

  Serial.println("Forward");

  digitalWrite(Q1,HIGH);

  digitalWrite(Q2,LOW);

  digitalWrite(Q3,HIGH);

  digitalWrite(Q4,LOW);
```

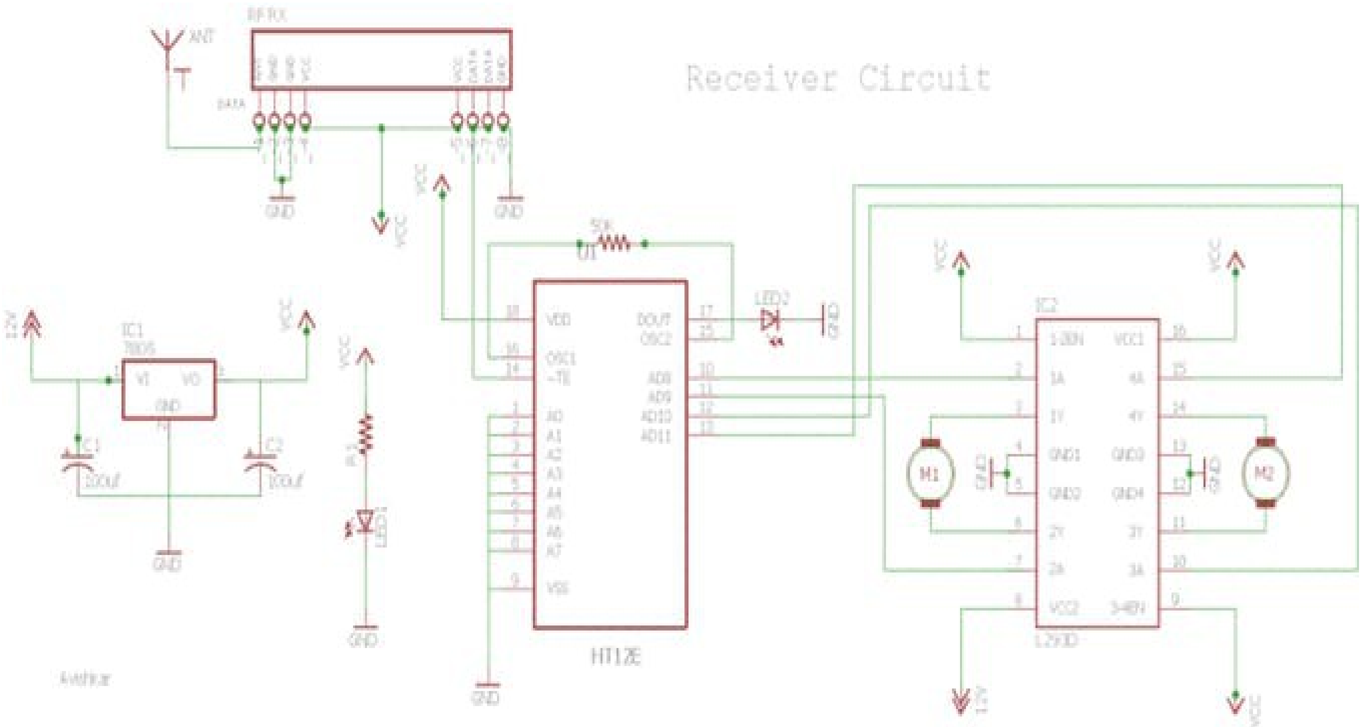
```
}

void backward()

{

  Serial.println("");

  Serial.println("Backward");

  digitalWrite(Q1,LOW);

  digitalWrite(Q2,HIGH);

  digitalWrite(Q3,LOW);

  digitalWrite(Q4,HIGH);

}

void left()
```

```
{
  Serial.println("");
  Serial.println("Left");
  digitalWrite(Q1,LOW);
  digitalWrite(Q2,HIGH);
  digitalWrite(Q3,HIGH);
  digitalWrite(Q4,LOW);
}
void right()
{
  Serial.println("");
```

```
  Serial.println("Right");

  digitalWrite(Q1,HIGH);

  digitalWrite(Q2,LOW);

  digitalWrite(Q3,LOW);

  digitalWrite(Q4,HIGH);

}
```



Receiver



The receiver circuit consists of 2 IC (HT12D decoder, L293D motor driver), RF receiver module.

Wire the circuit as per the above receiver schematic. There are 2 LEDs in the receiver board, one lights up when the power supply is given to the receiver and the other when power supply is given to the transmitter circuit. The LED near the IC HT12D should light up and this provides you a valid transmission (VT) when power is given at the transmitter if not there is something wrong with your connection or your RF-TX-RX module.

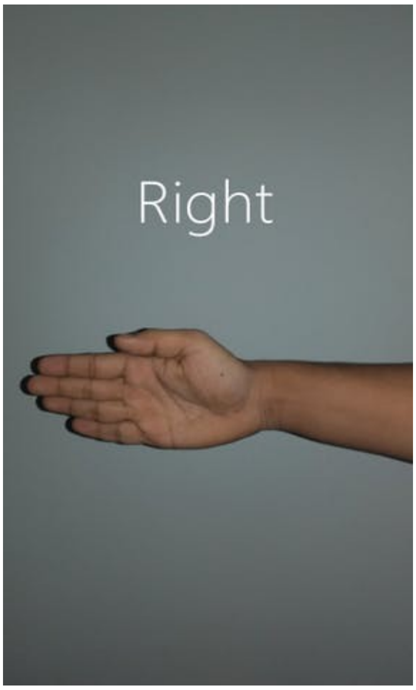
Encoder (HT12E) and decoder (HT12D) ICs

The 212 encoders are a series of CMOS LSIs for remote-control system applications. These are capable of encoding information that consists of N address bits and 12 N data bits. Each address/data input can be set to one of two logic states. Programmed addresses/data are transmitted together with header bits via an RF or infra-red transmission medium upon receipt of a trigger signal. The capability to select a TE trigger on HT12E or a data (DIN) trigger on HT12D decoder further enhances the application flexibility of 212 series of encoders. The HT12D also provides a 38kHz carrier for infra-red systems.

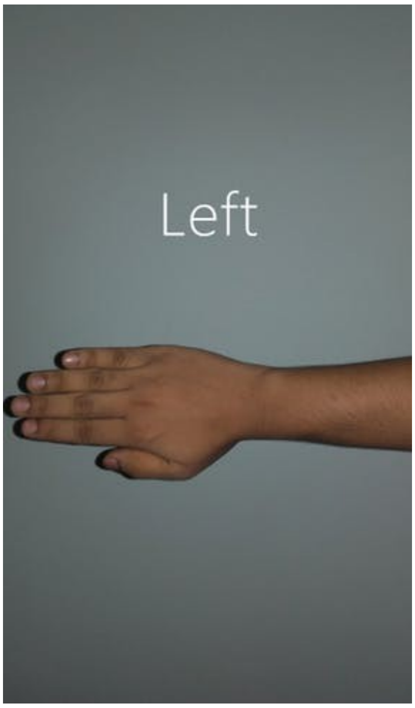
Gestures will the robot recognize

This robot is designed for recognizing five sets of gestures: forward, backward, left, right and stop. You will get a better idea if you check the photos of the gestures given below.

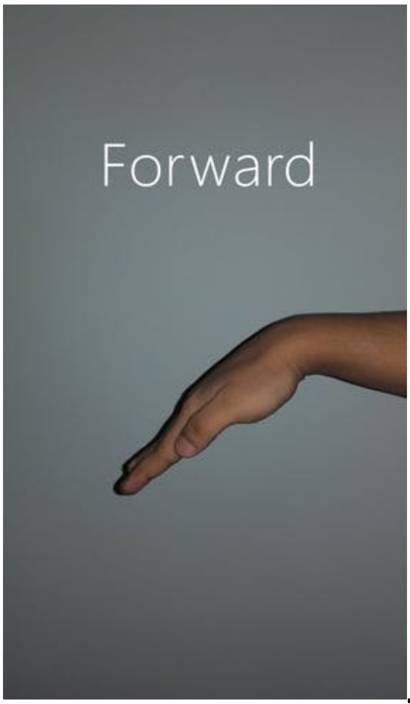
Stop_flat



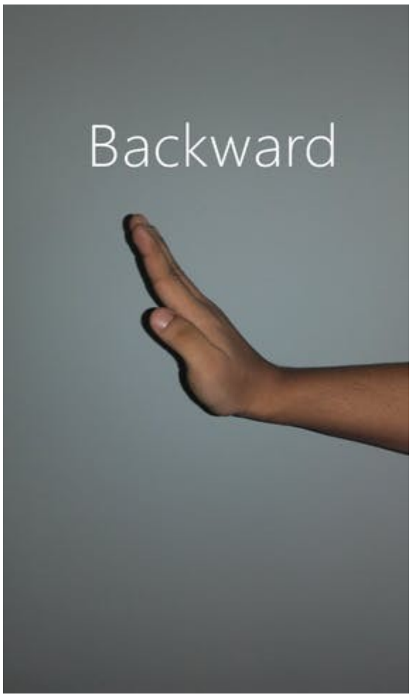
right_flat



left_flat



forward_flat



backward_flat

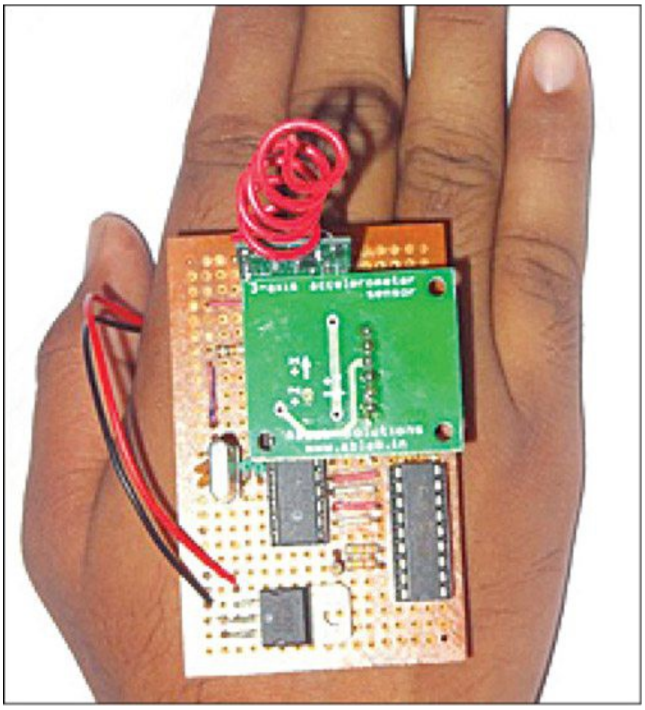
TABLE I
Movement of Robot and Decoder Outputs

Robot (accelerometer)	Input 1 (D11)	Input 2 (D10)	Input 3 (D9)	Input 4 (D8)
Forward (-Y)	0	1	0	1
Backward (+Y)	1	0	1	0
Right (+X)	1	0	0	1
Left (-X)	0	1	1	0

Construction and testing

An actual-size, single-side PCB layout of the transmitter circuit and its component layout . An actual-size, single-side PCB layout of the receiver circuit and its component layout .

The transmitter section can be held in your palm or on the other side . The receiver module is mounted on the robot.



Transmitter module of the gesture controlled robot

Mount all components on the PCBs shown here to minimise assembly errors. Fix the receiver PCB and 4.5V battery on the chassis of the robot. Fix two motors, along with wheels, at the rear side of the robot and a castor wheel on the front. After uploading the main code into the microcontroller, remove it from the Arduino Uno board and insert it into the populated transmitter PCB.

Now, switch-on the power supplies in the transmitter as well as receiver circuits. Attach the transmitter circuit to your hand and move your hand forwards, backwards and sideways. Directions of the robot movement are given in Table I. The robot will stop if you keep your palm horizontal, parallel to the Earth's surface.

Problems we faced

Soldering the IC was very difficult because of the package. After soldering, the accelerometer was not found to work. AT the X,Y and Z outputs, the voltage observed was zero.

Possible cause of damage

Soldering may not be proper, accelerometer may have been overheated while soldering, while doing the continuity test, the currents entering the device may have damaged it. Another board and IC was used and worked. We took a few readings of the accelerometer at in various tilts. The second IC on further testing also stopped working. Cause: spike in the IC probably.

Application

- **Gesture Driven systems are simpler and easier to operate. Any common man with basic knowledge can operate it.**

- **This system tends to penetrate deeper in everyday use rather than being confined in research labs.**

• It will open a new era in Home automation and remotely operated utility devices at households. Even physically disabled can gain benefit.

• With proper wireless connectivity, gesture recognition can provide a simple and reliable solution to household activities.

• Complex and realistic physical interactions in animation Industry.

• Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans than primitive text user interfaces or even GUIs (graphical user interfaces), which still limit the majority of input to keyboard and mouse

• Gesture recognition enables humans to interface with the machine (HMI) and interact naturally without any mechanical devices

Deficiency

• The user has a huge device on his hand which obstruct the user do normal hand movement.

• Since there is no force feedback, the user won't know what he is working on. But we can add feedback system in newer projects.

• Fine movement is difficult to achieve when working with bigger objects or controlling machines are bigger in size.

Future Scope

• Wireless modules consume very low power and is best suited for wireless, battery driven devices. • Advanced robotic arms that are designed like the human hand itself can easily controlled using hand gestures only.

• Proposed utility in fields of Construction, Hazardous waste disposal, Medical science.

• VR simulation may prove to be crucial for Military, LAW enforcement and Medical Surgeries.

References

- https://electronicsforu.com/electronics-projects/hardware-diy/wireless-gesture-controlled-robot/3

- https://create.arduino.cc/projecthub/user206876468/hand-gesture-controlled-robot-4d7587

- https://www.arduino.cc/

- https://www.youtube.com/results?search_query=hand+gesture+controlled+robot

- https://www.elprocus.com/accelerometer-based-gesture-control-robot/

Last but not the least God bless this encyclopaedic

Mr.Google .