Bangladesh University of Engineering and Technology

Department of Electrical and Electronics Engineering

Course No: **EEE-402** Group: **02**

Course Name: **Control System I Laboratory**

Project On: **Accelerometer Based Hand Gesture Controlled Robot Using Arduino**

Student ID: **1306139 (Tahmid Ibne Mannan)**

1306142 (Nowshin Nawal)

1306154 (Hasan Mahmud Prottoy)

1306159 (Abhijit Baul)

1306160 (Salauddin Omar Sifat)

1306162 (Md. Abu Bakar Siddik)

Submitted to: Dr. Mohammad Ariful Haque
Professor
Dept. of EEE, BUET

Md. Mukhlasur Rahman Tanvir
Lecturer
Dept. of EEE, BUET

Accelerometer Based Hand Gesture Controlled Wheel Chair Using Arduino

Introduction:

We all wish we could control everything with our hands! Sitting in my chair and controlling things like a BOSS. So we tried to built a robot which is controlled by hand gesture.

Robots are playing an important role in automation across all the sectors like construction, military, medical, manufacturing, etc. After making some basic robots like line follower robot, computer controlled robot, etc, we have developed this **accelerometer based gesture controlled robot** by using arduino uno. In this project we have used hand motion to drive the robot. For this purpose we have used accelerometer which works on acceleration.

Here the robot is divided into two parts, transmitter and receiver. The receiver circuit consists of the receiver portion of the RF pair and we needed to program the transmitter circuit so that it can transmit signals to the receiver part. We used Arduino as the programming platform. To recognize the gestures made we have used an accelerometer sensor.

Required Components:

1. Arduino UNO
2. DC Motors
3. Accelerometer
4. HT12D
5. HT12E
6. RF Pair
7. Motor Driver L293D
8. 9 Volt battery
9. Battery Connector
10. USB Cable
11. Robot Chasis
12. Resistors
13. Wires

Description of Components

A) Accelerometer (ADXL335)

An accelerometer is a three-axis acceleration measuring device with +-3g range. The accelerometer used here is ADXL335 and it has 3 axes (X Y Z). This device is made by using polysilicon surface sensor and signal conditioning circuit to measure acceleration. The output of this device is analog in nature and proportional to the acceleration. This device measures the static acceleration of gravity when we tilt it and gives a result in form of motion or vibration.

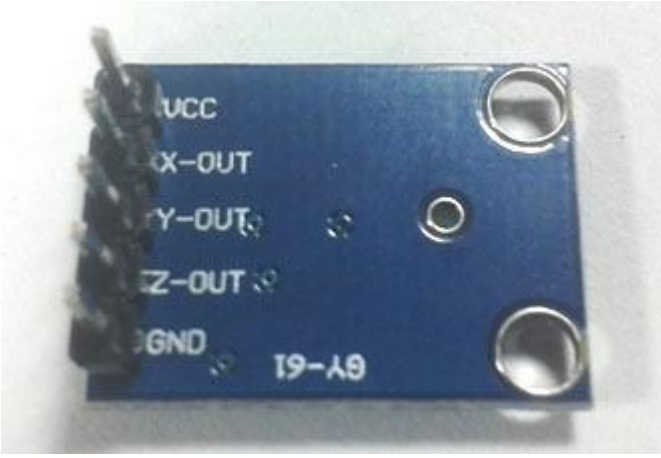


Figure -1: ADXL335

According to the datasheet of ADXL335 polysilicon surface-micromachined structure placed on top of silicon wafer. Polysilicon springs suspend the structure over the surface of the wafer and provide a resistance against acceleration forces. Deflection of the structure is measured using a differential capacitor which incorporates independent fixed plates and plates attached to the moving mass. The 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.

Pin Description of accelerometer:

1) Vcc : 5 Volt supply should connect at this pin.
2) X-OUT: This pin gives an analog output in x direction.
3) Y-OUT: This pin gives an analog output in y direction.
4) Z-OUT: This pin gives an analog output in z direction.
5) GND: Ground
6) ST: This pin used for set sensitivity of sensor.

B) RF pair

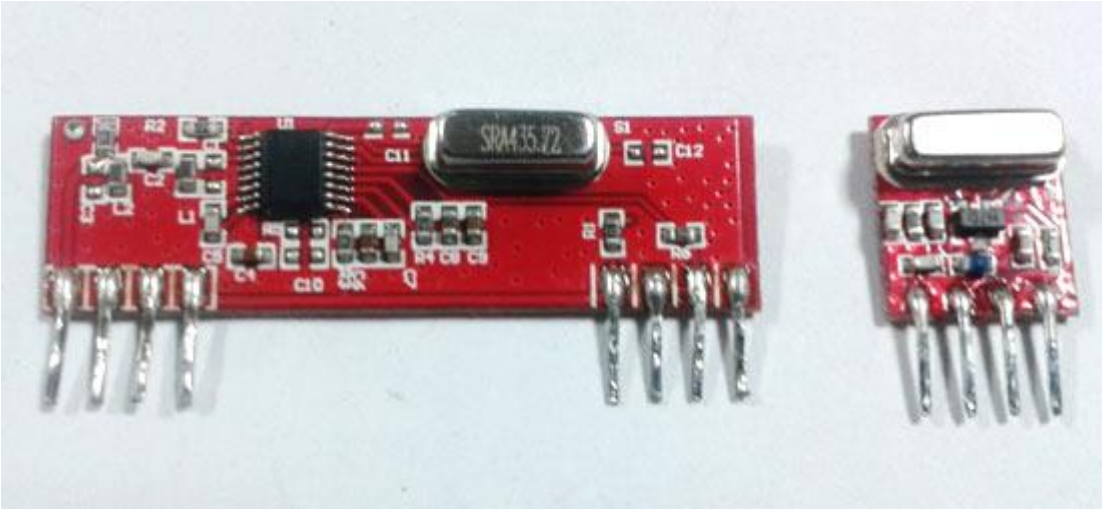


Figure 2: RF pair module

A **gesture controlled robot** is controlled by using hand in place of any other method like buttons or joystick. Here one only needs to move hand to control the robot. A transmitting device is used in your hand which contains RF transmitter and accelerometer. This will transmit command to robot so that it can do the required task like moving forward, reverse, turning left, turning right and stop. All these tasks will be performed by using hand gesture.

C) Encoder

HT12E is an **encoder integrated circuit** of 2^{12} series of encoders. They are paired with 2^{12} series of decoders for use in remote control system applications. It is mainly used in interfacing RF and infrared circuits. The chosen pair of encoder/decoder should have same number of addresses and data format. Simply put, HT12E converts the parallel inputs into serial output. It encodes the 12 bit parallel data into serial for transmission through an RF transmitter. These 12 bits are divided into 8 address bits and 4 data bits.

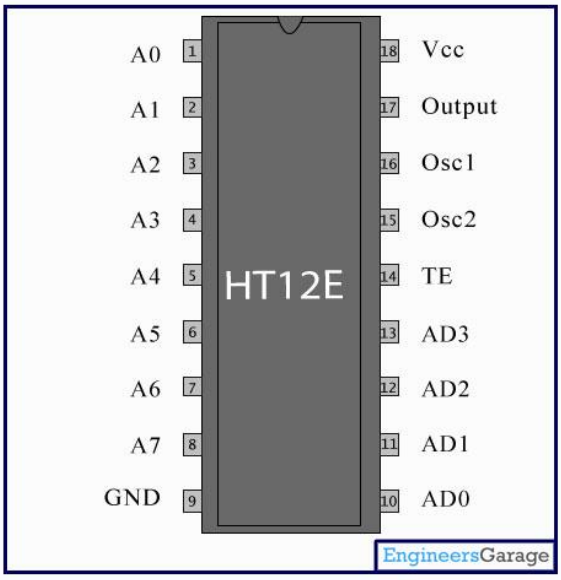


Figure-3: HT12E

HT12E has a transmission enable pin which is active low. When a trigger signal is received on TE pin, the programmed addresses/data are transmitted together with the header bits via an RF or an infrared transmission medium. HT12E begins a 4-word transmission cycle upon receipt of a transmission enable. This cycle is repeated as long as TE is kept low. As soon as TE returns to high, the encoder output completes its final cycle and then stops.

D) Decoder

HT12D is a **decoder integrated circuit** that belongs to 2^{12} series of decoders. This series

of decoders are mainly used for remote control system applications, like burglar alarm, car door controller, security system etc. It is mainly provided to interface RF and infrared circuits. They are paired with 2^{12} series of encoders. The chosen pair of encoder/decoder should have same number of addresses and data format. In simple terms, HT12D converts the serial input into parallel outputs. It decodes the serial addresses and data received by, say, an RF receiver, into parallel data and sends them to output data pins. The serial input data is compared with the local addresses three times continuously. The input data code is decoded when no error or unmatched codes are found. A valid transmission in indicated by a high signal at VT pin.

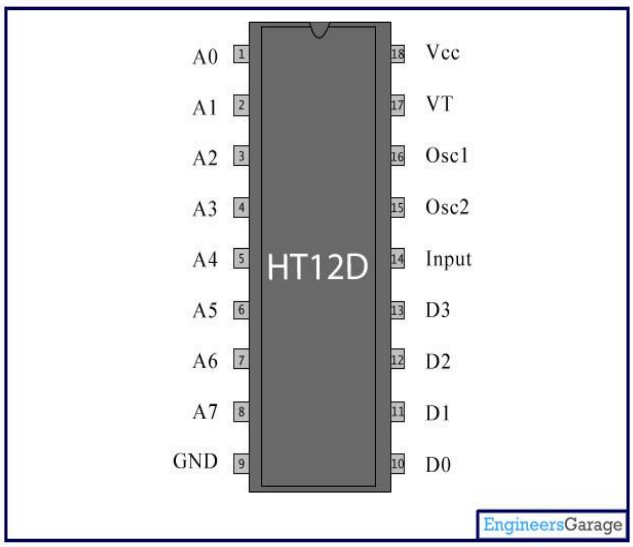
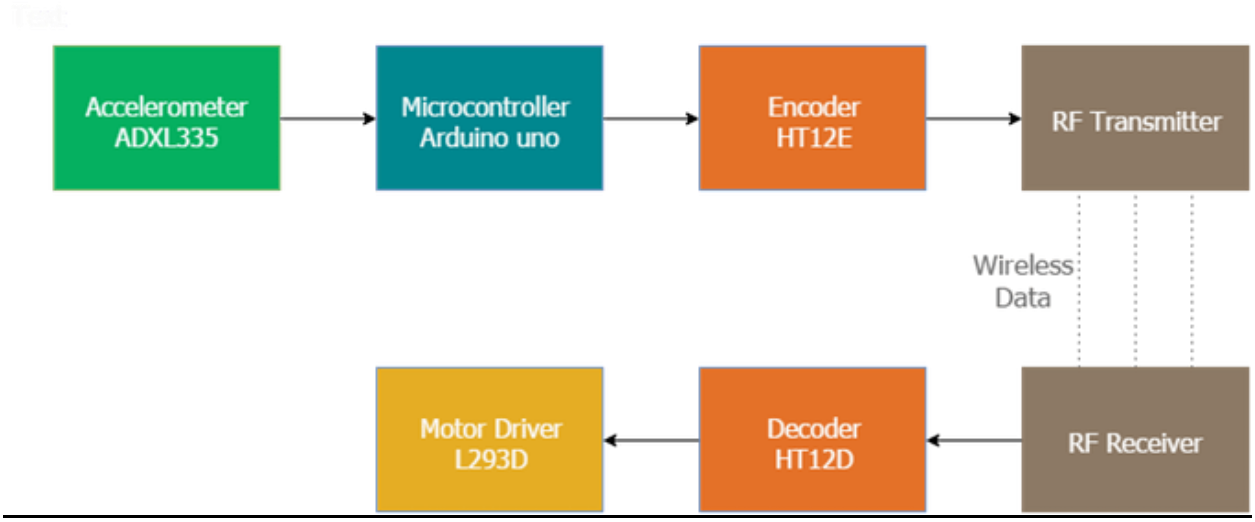


Figure-4: HT12D

HT12D is capable of decoding 12 bits, of which 8 are address bits and 4 are data bits. The data on 4 bit latch type output pins remain unchanged until new is received.

Working Principle:

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 (ADXL335).



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.

Making the transmitter circuit:

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. The given circuit is wired up as the transmitter circuit.

In transmitter part an accelerometer and a RF transmitter unit is used. As we have already discussed that accelerometer gives an analog output so here we need to convert this analog data in to digital. For this purpose we have used 4 channel comparator circuit in place of any ADC. By setting reference voltage we gets a digital signal and then apply this signal to HT12E encoder to encode data or converting it into serial form and then send this data by

using RF transmitter into the environment.

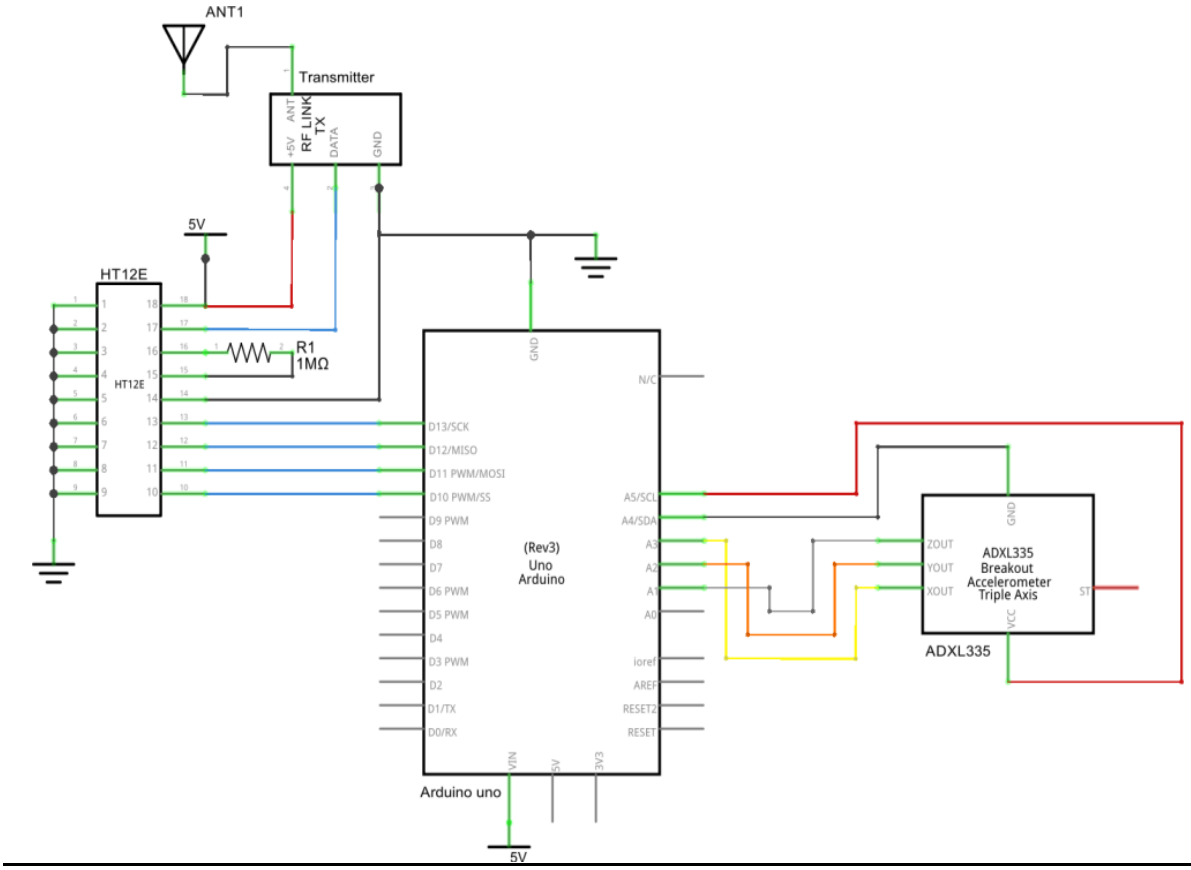


Figure-5: Transmitter Circuit

Code of the Transmitter:

```
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=7,Q2=8,Q3=9,Q4=10; //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
  Serial.begin(9600);
}
void loop()
{
  x = analogRead(xPin); //Reads X coordinates
  y = analogRead(yPin); //Reads Y coordinates
  z = analogRead(zPin); //Reads Z coordinates (Not Used)



  Serial.print(analogRead(x));
  // print a tab between values:
  Serial.print("\t");
  Serial.print(analogRead(y));
  // print a tab between values:
```

```
  Serial.print("\t");
  Serial.print(analogRead(z));
  Serial.println();
  // delay before next reading:
  delay(1000);



   if(x<320)     // Change the value for adjusting sensitivity
     forward();
   else if(x>350) // Change the value for adjusting sensitivity
     backward();
   else if(y>350) // Change the value for adjusting sensitivity
     right();
   else if(y<320) // 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);
}
```

Making the receiver circuit:

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

At the receiver end we have used RF receiver to receive data and then applied to HT12D decoder. This decoder IC converts received serial data to parallel and then read by using arduino. According to received data we drive robot by using two DC motor in forward, reverse, left, right and stop direction.

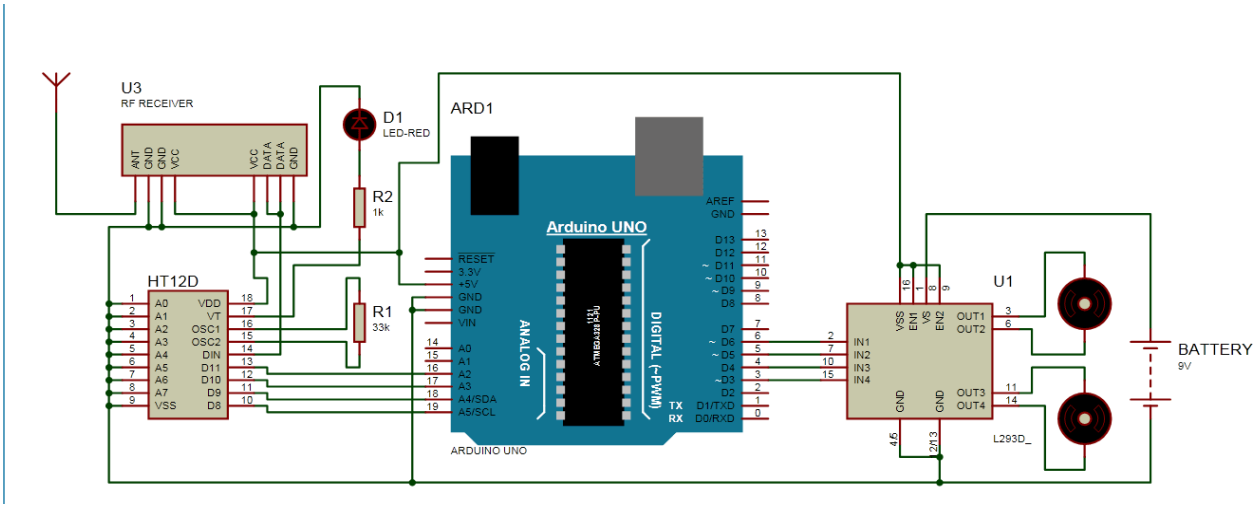


Figure -6: Receiver Circuit

Code of the Receiver:

```
#define FD 16

#define BD 17

#define LD 18

#define RD 19


#define m11 3

#define m12 11

#define m21 5

#define m22 6



#define a  9

#define b  10


void forward()
```

```
{
  analogWrite(m11, 100);
  analogWrite(m12, 0);
  analogWrite(m21, 100);
  analogWrite(m22, 0);
}

void backward()
{
  analogWrite(m11, 0);
  analogWrite(m12, 100);
  analogWrite(m21, 0);
  analogWrite(m22, 100);
}

void left()
{
  analogWrite(m11, 0);
  analogWrite(m12, 0);
  analogWrite(m21,100);
  analogWrite(m22, 0);
}

void right()
{
```

```
  analogWrite(m11, 100);
  analogWrite(m12, 0);
  analogWrite(m21, 0);
  analogWrite(m22, 0);
}

void Stop()
{
  digitalWrite(m11, LOW);
  digitalWrite(m12, LOW);
  digitalWrite(m21, LOW);
  digitalWrite(m22, LOW);
}

void setup()
{
 pinMode(FD, INPUT);
 pinMode(BD, INPUT);
 pinMode(LD, INPUT);
 pinMode(RD, INPUT);

 pinMode(m11, OUTPUT);
 pinMode(m12, OUTPUT);
 pinMode(m21, OUTPUT);
 pinMode(m22, OUTPUT);
```

```
  Serial.begin(9600);


 pinMode(a, OUTPUT);
 pinMode(b, OUTPUT);
 digitalWrite(a,HIGH);
 digitalWrite(b,HIGH);
}

void loop()
{

 int temp1=digitalRead(FD);
 int temp2=digitalRead(BD);
 int temp3=digitalRead(LD);
 int temp4=digitalRead(RD);
 //digitalWrite(a,LOW);
 //digitalWrite(b,LOW);
 Serial.print(digitalRead(FD));
 // print a tab between values:
 Serial.print("\t");
 Serial.print(digitalRead(BD));
 // print a tab between values:
 Serial.print("\t");
 Serial.print(digitalRead(LD));
```

```
 Serial.print("\t");
 Serial.print(digitalRead(RD));
 Serial.println();
 // delay before next reading:
 //delay(1000);



 if(temp1==0 && temp2==1 && temp3==0 && temp4==1)
 backward();

 else if(temp1==1 && temp2==0 && temp3==1 && temp4==0)
 forward();

 else if(temp1==0 && temp2==1 && temp3==1 && temp4==0)
 left();

 else if(temp1==1 && temp2==0 && temp3==0 && temp4==1)
 right();

 else
 Stop();
}
```

Cost Calculation:

Equipment	Quantity	Price
Arduino UNO	1	420
HT12D	2	100
HT12E	2	100
RF Module	1	350
Batteries	3	75
LM 324	2	20
LED	10	10
Jumper Wires		120
Chasis	1	500
Motor Driver 293	1	250
ADXL 335	1	300
Diode		5
Resistors		50
Total		**2300**

Conclusion:

This whole system can be modified and used for humanitarian work. As for example, this system can be used by paralyzed patients for controlling their own wheel chairs.Another use of this system can be in controlling one's own car or other moveable devices. This system lessens human effort and has great potential and can be used in various field according to need.