GOVT. POLYTECHNIC SUNDERNAGAR, MANDI (H.P.)

राजकीय बहुतकनीकी सुन्दरनगर, मण्डी (हि.प्र.)

MAJOR - PROJECT REPORT

ON

MOTION GESTURE ROBO CAR

Submitted in partial fulfillment of the requirements for the award of the Diploma in Computer Application

GUIDE:

Er. (Chander Shekhar) & Er. (Himani Vaidya)

Name: Aditya Soni

(SUBMITTED TO)

राजकीय बहुतकनीकी सुन्दरनगर, मण्डी (हि.प्र.)

ACKNOWLEDGEMENT

I am grateful to our faculty guide **Er. Chander Shekhar & Er. Himani Vaidya,** who has guided and helped me throughout the development of the initial stages of software.

I also express my thanks to **all our well-wishers** who acknowledge me for their valuable help provide to me.

Last but not the least, I would like to express our heartfelt thanks to my Parents who's blessing have made this possible, through discussion, giving suggestion and in many other ways.

Thanking you,
Aditya Soni

INDEX

OBJECTIVES OF PROJECT

Main objective of the "MOTION GESTURE CAR" is to develop a robotic vehicle using Arduino, to control the vehicle with the help of gyroscope & to provide user the front view of the car. It helps us by providing the view of short/smaller place, where we are not able to go.

Following are the main objectives:-

- Helpful for surveillance purpose.
- Small weapons can be attached for the combat.
- User friendly.
- Helpful in hospitals for providing food to the patients in this pandemic by maintaining social-distancing.

INTRODUCTION TO C++

C++ is a general-purpose programming language that was developed as an enhancement of the C language to include object-oriented paradigm. It is an imperative and a compiled language.

C++ is a middle-level language rendering it the advantage of programming low-level (drivers, kernels) and even higher-level applications (games, GUI, desktop apps etc.). The basic syntax and code structure of both C and C++ are the same.

ARDUINO

Arduino is an open-source hardware and software company, project and user community hat designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices. It's hardware products are licensed under a CC-BY-SA license, while software is licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially from the official website or through authorized distributors.

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards ('shields') or breadboards (for prototyping) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs. The micro-controllers can be programmed using the C and C++ programming languages, using a standard API which is also known as the "Arduino language". In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) and a command line tool developed in Go.

The Arduino project began in 2005 as a tool for students at the Interaction Design Institute Ivrea in Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors.

SOME DRAWBACKS OF C++ (ARDUINO)

1. Processing Power is weaker than the other micro-controllers.

Arduino has a decent processing power, when it comes to high-end power modules & power circuits it lags behind.

2. Special kind of C++ (Arduino).

As Arduino is a special kind of C++ language so it need some expert training to use and operate the software which make it incapable of providing a smooth experience for the user.

3. Expensive

Arduino hardware are very expensive. So everybody cannot afford it.

DRAWBACKS OF THE EXISTING SYSTEM

- COMPLEX CONTROLLERS ARE REQUIRED.
- LESS SENSITIVE SYSTEM.
- COSTLY.
- LESS ACCURATE.
- LESS PRECISED.
- CANNOT BE CONTROLLED EASILY BY A LAYMAN.
- NO CAMERA RECORDING.

ADVANTAGES OF THE PROPOSED SYSTEM

The proposed system that is "**MOTION GESTURE CAR**" will provide the efficient way to control the car using hand gesture and provides video coverage of the location.

- EASY AND QUICK RESPONSE.
- MORE PRECISE.
- MORE ACCURATE.
- HIGHLY SENSITIVE.
- ENABLE/DISABLE CAMERA.
- VIDEO RECORDING IS POSSIBLE.
- INTERCHANGABLE BATTERY IS POSSIBLE.

TOOLS ENVIRONMENT USED

C++ is a general-purpose programming language that was developed as an enhancement of the C language to include object-oriented paradigm. It is an imperative and a compiled language.

C++ is a middle-level language rendering it the advantage of programming low-level (drivers, kernels) and even higher-level applications (games, GUI, desktop apps etc.). The basic syntax and code structure of both C and C++ are the same.

Arduino is the sub part of C++ which I have used in this project, due to the fact that Arduino is more capable of handling the electronic circuits.

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards ('shields') or breadboards (for prototyping) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs. The micro-controllers can be programmed using the C and C++ programming languages, using a standard API which is also known as the "Arduino language". In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) and a command line tool developed in Go.

HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements:

- 2 x Arduino NANO.
- ADXL335 Module (Gyroscope).
- L298N Motor Driver Controller.
- TT Gear Motor x 4.
- Rubber Wheel x 4.
- Acrylic Sheet (For making a transparent chassis).
- NRF 24L01 + RF Module x 2
- Camera module.
- Connecting Wires.
- Soldering Kit.

Software Requirements:

- Window XP Professional.
- Arduino.

HARDWARE DESCRIPTION

1. Arduino Nano:

It is a small, compatible, flexible and breadboard friendly Microcontroller board developed by Arduino.cc in Italy, based on ATmega328p. Current models consist a USB interface, 6 analog input pins and 14 digital I/O pins that allows users to attach various extension boards.

2. ADXL335 Module:

It is a small, thin, low power, a complete 3-axis accelerometer with signal conditioned voltage outputs. An accelerometer can figure out the angle it is titled at with respect to earth. This module is also known as Gyroscope sensor (motion gesture sensor).

3. L298N Motor Driver Controller:

This is a high power motor driver module used for driving DC and Stepper motors.

4. TT gear motor:

The plastic gearbox motors (also known as TT motors). This is a TT DC gearbox motor with a gear ratio of 1:48, and it comes with 2 X 200mm wires with breadboard friendly 0.1'' male connectors.

5. NRF 24L01 + RF module:

These are the wireless transceiver module, meaning each module can both send as well as receive data in the form of radio signal.

6. Camera Module:

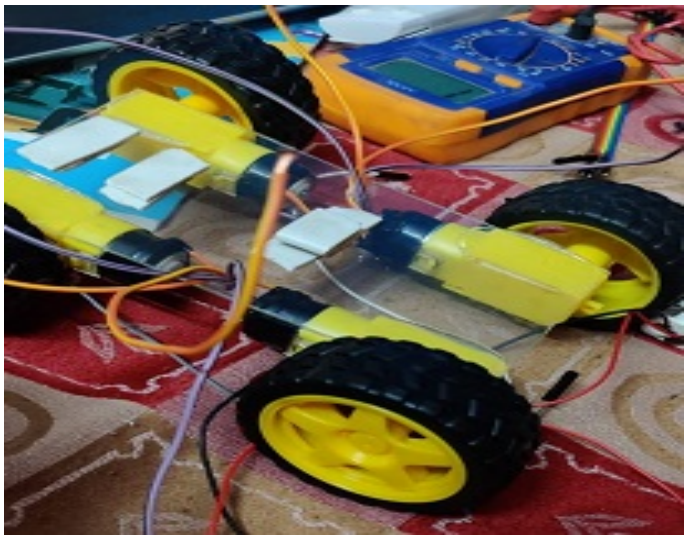
A camera module is an image sensor integrated with a lens, control electronics, and an interface like Ethernet or CSI. This sensor coverts the light from the lens into an electrical signal, and then converts it into a digital signal.

7. Connecting wires:

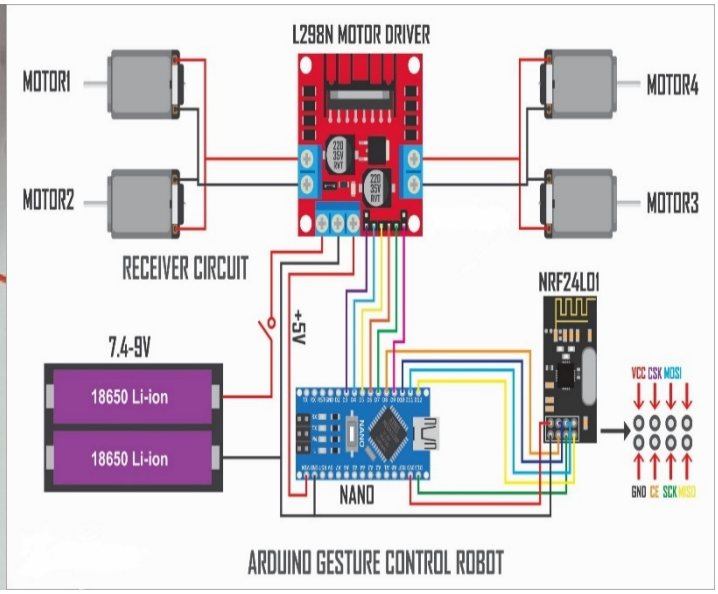
These are the wires which allows an electrical current to travel from one point on a circuit to another because the electricity needs a medium through which it can move. In Arduino these connecting wires are also known as Jumping wires. These Jump wires normally used to interconnect the components of a breadboard or other prototype or test circuit.

ASSEMBLY & CODES

1. First of all we make a chassis with the help of Acrylic sheet & attach 4 TT gear motor with 4 rubber wheel.

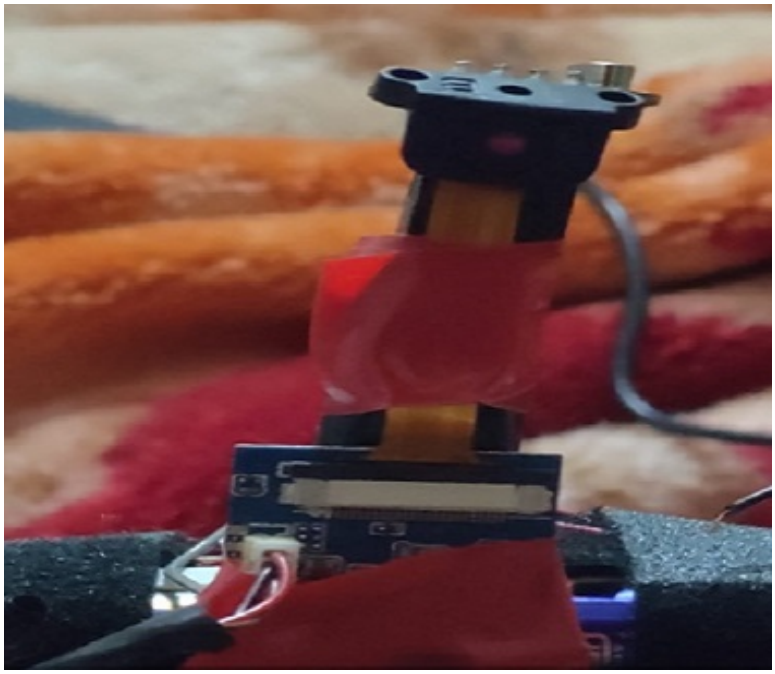


2. Then assemble the circuit which consists Arduino Nano, L298N Motor Driver Module, NRF 24L01 + RF module & battery with the help of given circuit diagram.

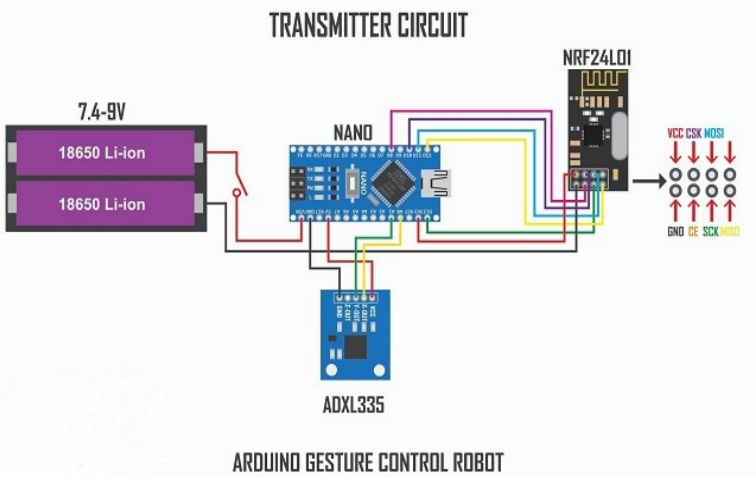


3. And then we installed the circuit on the chassis of the car with the help of double sided tape.

4. And then we will install the camera module on it.

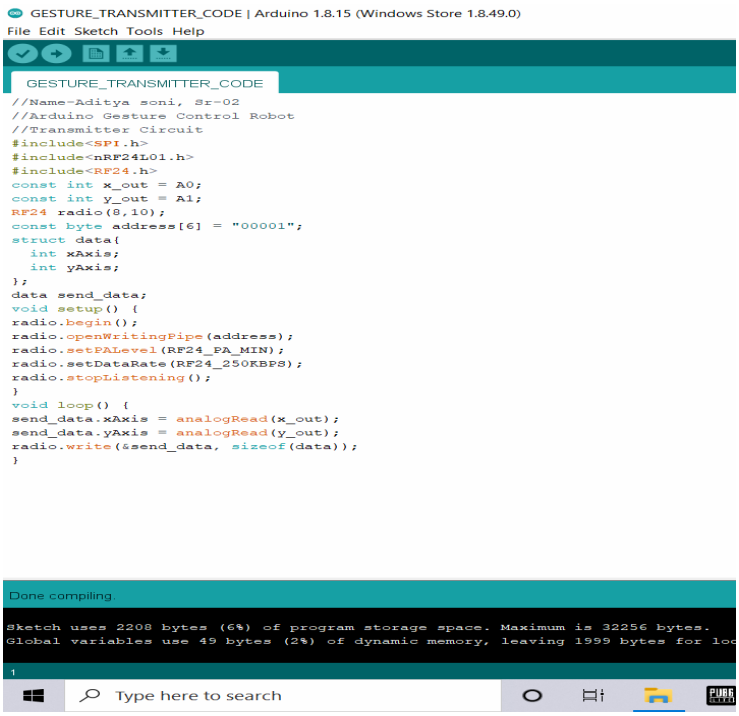


5. Then we made the controller band with help of circuit diagram given below. In this we are using an ADXL335 module (gyroscope sensor) coupled with Arduino Nano board & RFID circuit for data transmission which formed a transmitter.



6. After completing the hardware assembly, we upload the codes on Arduino boards and tested them. Codes are given below:

I. Transmitter Code**:**



II. Receiver Code:

GESTURE_RECEIVER_CODE | Arduino 1.8.13 (Win

File Edit Sketch Tools Help

GESTURE_RECEIVER_CODE

```
//Name-Aditya soni, SR-02
//Arduino Gesture Control Robot
//Reciver Circuit
#include<SPI.h>
#include<nRF24L01.h>
#include<RF24.h>
int ENA = 3;
int ENB = 9;
int MotorA1 = 4;
int MotorA2 = 5;
int MotorB1 = 6;
int MotorB2 = 7;
RF24 radio(8, 10);
const byte address[6] = "00001";
struct data {
  int xAxis;
  int yAxis;
};
data receive_data;
void setup() {
Serial.begin(9600);
radio.begin();
radio.openReadingPipe(0,address);
radio.setPALevel(RF24_PA_MIN);
radio.setDataRate(RF24_250KBPS);
radio.startListening();
pinMode(ENA, OUTPUT);
pinMode(ENB, OUTPUT);
pinMode(MotorA1, OUTPUT);
pinMode(MotorA2, OUTPUT);
pinMode(MotorB1, OUTPUT);
pinMode(MotorB2, OUTPUT);
}
void loop() {
 while(radio.available()) {
```

Done compiling.

```
Sketch uses 3496 bytes (10%) of prograr
Global variables use 224 bytes (10%) o:
```

25

GESTURE_RECEIVER_CODE | Arduino 1.8.13 (Windows Store 1.8.4

File Edit Sketch Tools Help

GESTURE_RECEIVER_CODE

```
void loop() {
 while(radio.available()) {
    radio.read(&receive_data, sizeof(data));
 if(receive_data.yAxis > 400) {
  digitalWrite(MotorA1, LOW);
  digitalWrite(MotorA2, HIGH);
  digitalWrite(MotorB1, HIGH);
  digitalWrite(MotorB2, LOW);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
}
else if(receive_data.yAxis < 320) {
  digitalWrite(MotorA1, HIGH);
  digitalWrite(MotorA2, LOW);
  digitalWrite(MotorB1, LOW);
  digitalWrite(MotorB2, HIGH);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
}
else if(receive_data.xAxis < 320){
  digitalWrite(MotorA1, HIGH);
  digitalWrite(MotorA2, LOW);
  digitalWrite(MotorB1, HIGH);
  digitalWrite(MotorB2, LOW);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
}
else if(receive_data.xAxis > 400){
  digitalWrite(MotorA1, LOW);
  digitalWrite(MotorA2, HIGH);
  digitalWrite(MotorB1, LOW);
  digitalWrite(MotorB2, HIGH);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
}
```

Done compiling.

```
Sketch uses 3496 bytes (10%) of program storage sp
Global variables use 224 bytes (10%) of dynamic me
```

25

GESTURE_RECEIVER_CODE | Arduino 1.8.13 (Windows Store 1

File Edit Sketch Tools Help

GESTURE_RECEIVER_CODE

```
}
else if(receive_data.yAxis < 320) {
  digitalWrite(MotorA1, HIGH);
  digitalWrite(MotorA2, LOW);
  digitalWrite(MotorB1, LOW);
  digitalWrite(MotorB2, HIGH);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
}
else if(receive_data.xAxis < 320){
  digitalWrite(MotorA1, HIGH);
  digitalWrite(MotorA2, LOW);
  digitalWrite(MotorB1, HIGH);
  digitalWrite(MotorB2, LOW);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
}
else if(receive_data.xAxis > 400){
  digitalWrite(MotorA1, LOW);
  digitalWrite(MotorA2, HIGH);
  digitalWrite(MotorB1, LOW);
  digitalWrite(MotorB2, HIGH);
  analogWrite(ENA, 150);
  analogWrite(ENB, 150);
}
else {
  digitalWrite(MotorA1, LOW);
  digitalWrite(MotorA2, LOW);
  digitalWrite(MotorB1, LOW);
  digitalWrite(MotorB2, LOW);
  analogWrite(ENA, 0);
  analogWrite(ENB, 0);
  }
 }
}
```

Done compiling.

```
Sketch uses 3496 bytes (10%) of program storage
Global variables use 224 bytes (10%) of dynamic
```

25

WORKING-PRINCIPLE OF CAR

- There will be a band in our hand, which will have a Gyroscope sensor, RF module and Arduino Nano chip mounted on a band. So when we move our hand gyroscope will sense the motion and give some value to the Arduino (and those value are based on the coding), because the gyroscope will work on the concept of x-axis, y-axis and z-axis. Then the transmitter, that is connected to Arduino will send the radio signal to the receiver end of the car.

- When receiver will get the value ,then 2nd Arduino mounted on car check that value and give instructions to the motor driver and then motor driver will perform the operation (because all the tires were connected to the motor driver so motor driver will control all the tires).

- EXAMPLE: When we want to move our car on forward direction so the all the tires have high power. And when we want to drive the car backwards, then the power of front 2 tires should off and the previous tire should be have high power. When we want to turn our car on right direction so the power of right 2 tires should be high or left tire should low, so the right side tires spin and left side tire on rest then the body of car will turn on right direction. And when we want to move our car on left direction then the power of left two tire should have high or right tire low so the left side tires spin and right side tire on rest then the body of car will turn on left direction.

- And there is a camera mounted on car which give us live video coverage of the location in our phone.

CONCLUSION

Today, technology is developing in the same direction in line with rapidly increasing human needs. The work done to meet these needs makes life easier every day, and these studies are concentrated in Motion Gesture Robo Car studies. This car is very easy to use. This car is very helpful for surveillance purpose which is very useful in army and also very helpful in hospitals for providing food facilities to the patients in this pandemic by maintaining the social distancing.