Eng. Clinics Project Report
Fall Sem - 2019

RF CONTROLLED CAR USING ARDUINO

Guided by: Prof. Balu Laxman Parne

(Batch:190196)

Team:

1) .Athrwa Deshmukh [19BCE7381]
2). Abhay Chaudhary [19BCE7290]
3).Nachane Hrishikesh Sudhir[19BCE7116]
4).Shreepad Gangadhar Labshetwar [19BCE7648]
5).Fachara Prashant Govindbhai[19BCE7271]
6).Arbinndo Sinha[19BME7025]

ABSTRACT

- A robot is a machine which is designed to perform several tasks simultaneously and frequently with precision and speed. It is a virtual artificial mediator, usually an electro-mechanical machine that is being controlled by a computer program to perform different tasks based on electronic circuits and usage. Robots can replace humans in performing repetitive and dangerous tasks which humans do not prefer to do because of various constraints. Robots are being designed to do the operations in extreme environments such as the bottom of the sea, outer space etc. This paper gives a broad idea about how to design a remotely controlled robotic car over a Radio Frequency network by using an Arduino Uno connected to an ASK(Amplitude Shift Keying) RF module.

- The advantage of using a robot-controlled car is it can be applied to various purposes like attempts can be made to use solar cells instead of the regular lithium-ion battery for the project and this project can be modified quite easily to include a spy camera as well that can stream the videos to the user over Radio Frequency which can be beneficial for our armed forces. This robot car can also be used to push various objects from one place to another. This project can be enhanced with better Radio Frequency , which would enable long-distance communication.

1. Introduction:

- The first technology that can be used by army and journalists
- Uses Radio Frequency for the transmission and reception of signal
- Can be controlled by using webpage

Radio Frequency controlled car using Arduino is an innovation which helps the journalists and armed force to navigate the regions where it is physically impossible to go and helps in detecting the nearby objects using the help of GO-PRO which then records the video for further review. They only need to operate it using a webpage.

There are many systems which are controlled either by Radio Frequency transmission or by creating intelligence. Robots called Non-autonomous robots have the programming logic to do the desired task but the decision power lies in the hand of the controller (human) handling the robot. Here the interface can be made using two methods: Wired and Wireless. The Radio Frequency controlled car uses transmission of signals wireless in air by transmitter which is captured by the receiver and sent to the microcontroller on robot to carry out the task.

The Radio Frequency controlled car minimizes the human efforts and it can be deployed in a lot of fields like military, surveillance application and Industrial use. This project uses the Arduino Uno connected to Radio Frequency Receiver which in turn is connected to it's decoder and controlled using a transmitter connected to a encoder especially designed for this purpose.

2. Background:

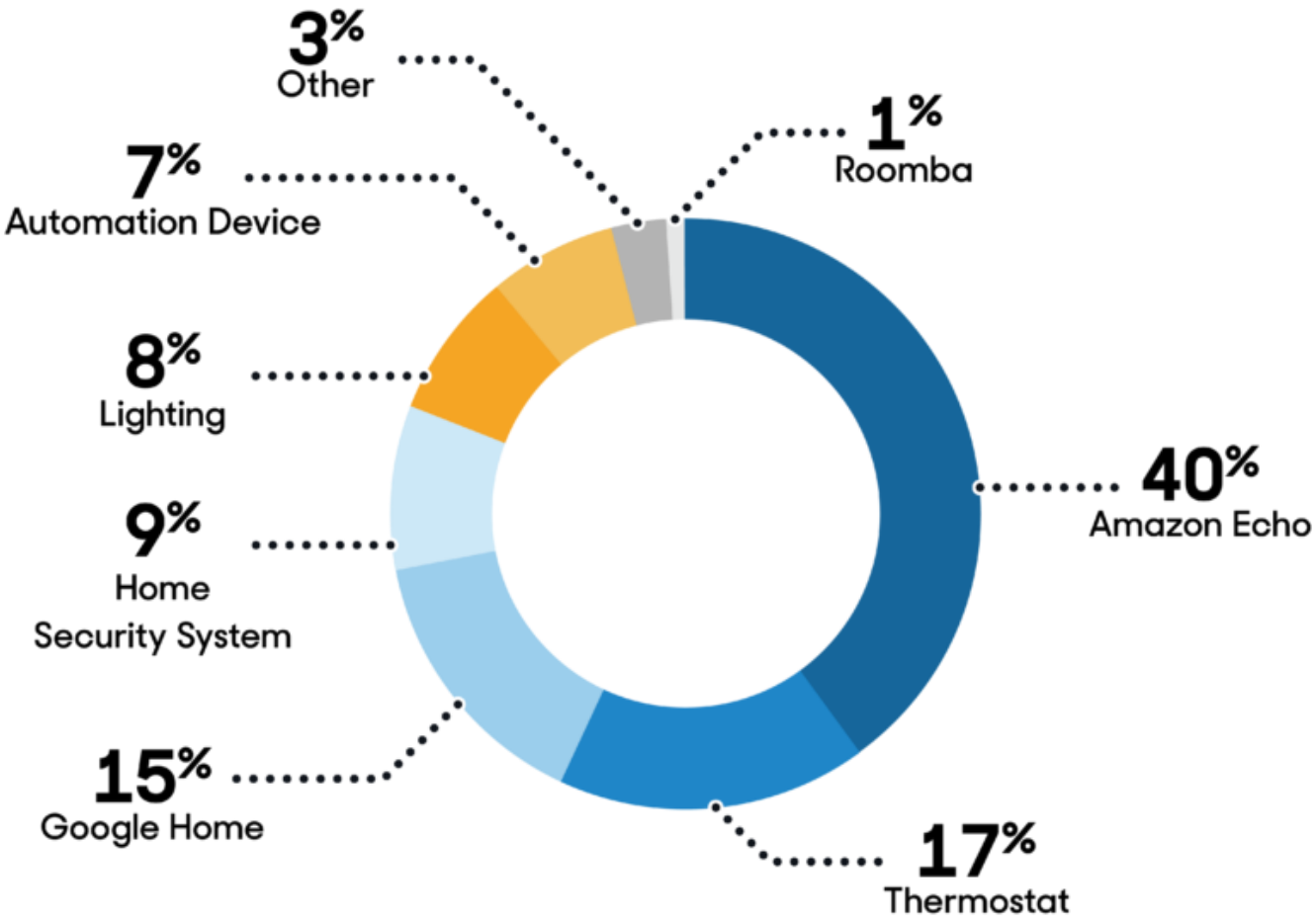
Gone are the days of seeing home automation as a novelty born from "The Jetsons" and "2001: A Space Odyssey." With major tech companies racing against one another to be the leader in the industry, it's clear gadgets like thermostats that learn your preferred room temperatures, and doorbells that show you who's on your porch from the comfort of your couch aren't just a passing trend. Innovation is key in this space, and companies like Amazon and Google want to ensure their tech is integral to the future of home automation.

So, how do these new smart devices fit into your daily life? As technology evolves at breakneck speed, it's easy to feel confused over where to even start. After weeks of research and a survey of smart device owners, we created a guide to help you find the right gadgets for you.

3. Problem Definition:

To get a better idea of which smart devices are popular and how they're used in a home, we conducted a survey of over 1,200 people. When asked how often they use smart devices, over 43% of our respondents said they use them multiple times per day with close to half saying they were motivated to automate their homes for convenience and better organization. Nearly 45% own between two and five smart devices, smart hubs/speakers taking the top spot as the most frequently used device.

We also asked how people got their first device and, though we weren't shocked to see that many people bought their devices online, it was interesting that 27.3% of respondents received smart devices as a gift. Though seeing as smart speakers like the Amazon Echo Plus and Google Home are still considered somewhat of a novelty by most, people beginning their home automation journeys with a gifted smart device shouldn't come as a surprise.



The best way to begin automating your home is by purchasing a smart hub or smart speaker (if you haven't already been gifted one) and researching devices that integrate with it that you believe will help simplify your life. We'll help you get started.

Smart hubs and speakers

The brains of your smart home setup, hubs connect your devices by integrating them in a single interface. While smart speakers like the Amazon Echo and Google Home aren't traditional hubs — they don't include all the necessary protocols to automatically detect and pair your devices — they're considered simplified hubs because their skills and actions still make it possible to sync and manage devices. With a hub, you can control your devices with the same app and, if you opt for one that comes with a voice assistant, you will also be able to manage your devices via voice commands. Hubs are crucial for those looking to build a large-scale smart home and who want convenience when managing devices.

Problem of the Existing Systems:

- The Radio Frequency transmission and reception of signal can be interfered easily and results in increasing the bit error rate due to increased EMF.
- Full project can cost up to 10,000 Rs.
- Common Disadvantages (Including the the smart devices) range may vary from place to place and should be handled carefully.

4. Objective:

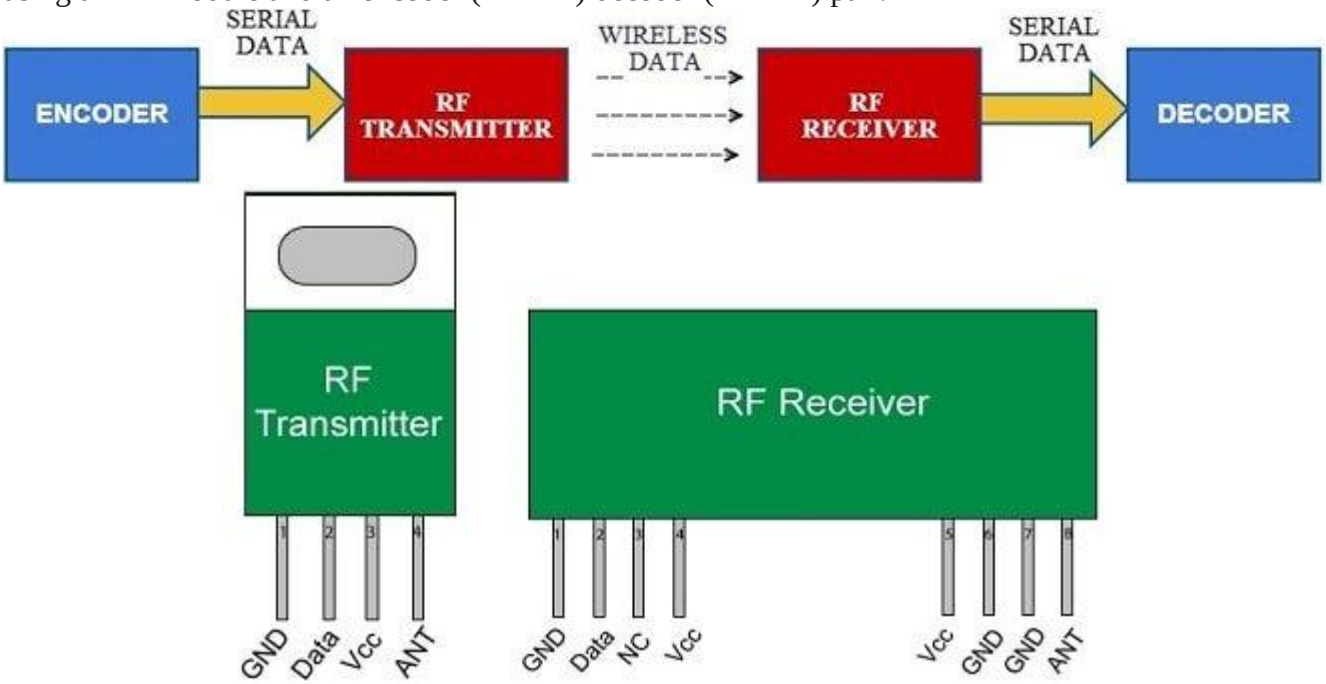
This is the first RF based car using Arduino for various types of people which resolves all the problems of existing technologies. Now a days there are so many instruments and smart devices for navigation but most of them have certain problems for carrying and the major drawbacks is those need a lot of training to use.

The one of the main peculiarity of this innovation is, it is affordable for everyone, the total cost being less than 10,000 Rs. There are no such devices available in the market that have such a great extent of usability having such a low cost and simplicity. When used on a large scale, with improvements in the prototype, it will drastically benefit the community.

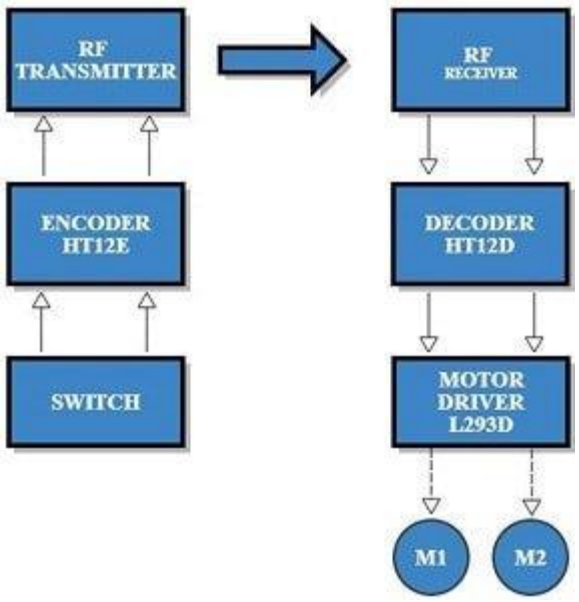
This device will help the user to navigate without even complex technology usage which would be easy for them. They can simply use it as a navigator which can be used to view the horizon in various aspects.

5.Methodology/Procedure:

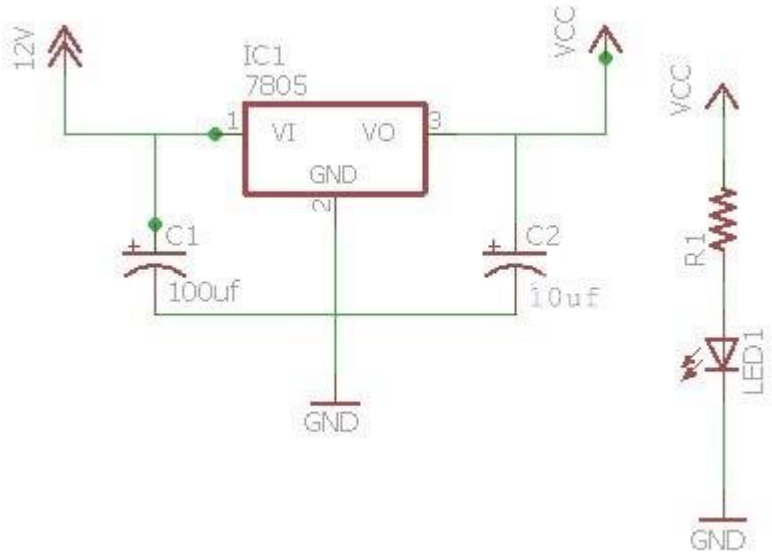
Here we will use a couple of ICs and a motor fixed to a chassis to make a remote control car. The brief idea is to transmit control signals through radio frequency and receive it through a receiver module in the car. We will have two switches in our remote control to power each motor of the car. The state of the switches (ON/OFF) is the control data. This data from the remote control is encoded before transmission, received back, and decoded again to be sent to the motor drivers. This is achieved using an RF module and an encoder (HT12E) decoder (HT12D) pair.



Using the combination of different states of the two switches, you can control the direction of motion of your remote control car. If both switches are off, both motors will be off, and the car will not move. If both are on, the car will move straight ahead. And to turn the car, switch on only the motor on the side you want the car to turn to.

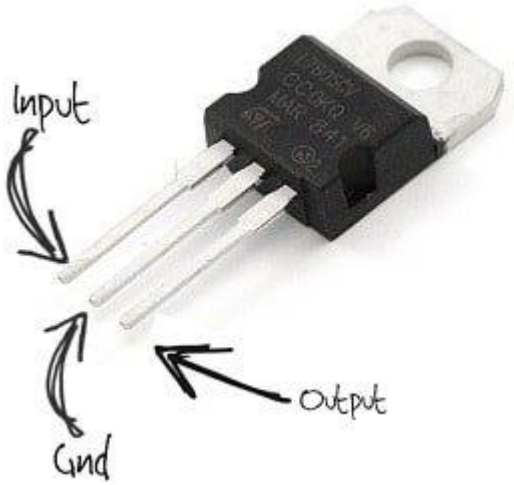


Making Power Supplies for the Remote Control Car



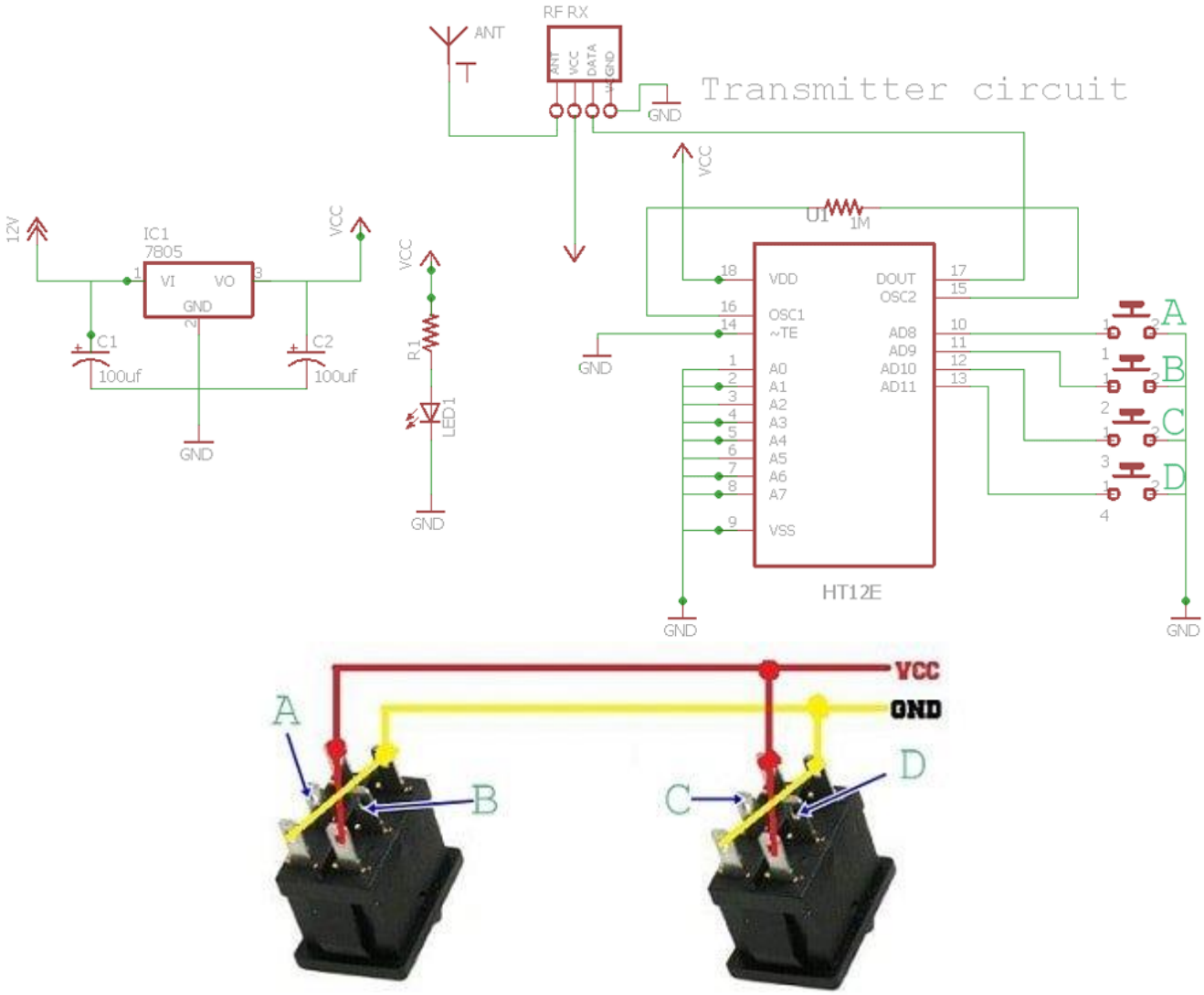
First, we will start with the power supply circuits. Both the RF transmitter and receiver circuits need separate power supplies. The receiver circuit needs to be powered with a 12V supply and transmitter circuit with a 9V battery.

You can see the circuit for the receiver power supply on the right. Using this diagram, wire up the supply circuit. You can also add an LED via a 1K resistor to indicate the state of the power supply.



- IC 7805 regulates the 12V supply to 5V (you can also use a 9V supply here).
- You can also use 0.1uF and 470uF capacitors in the circuit and 1K resistor for status LED.
  NOTE: Use heat sink for 7805 because we are dropping 7V (12-5), so a lot of heat will be produced to burn the regulator.

Making the Transmitter (Remote Control)



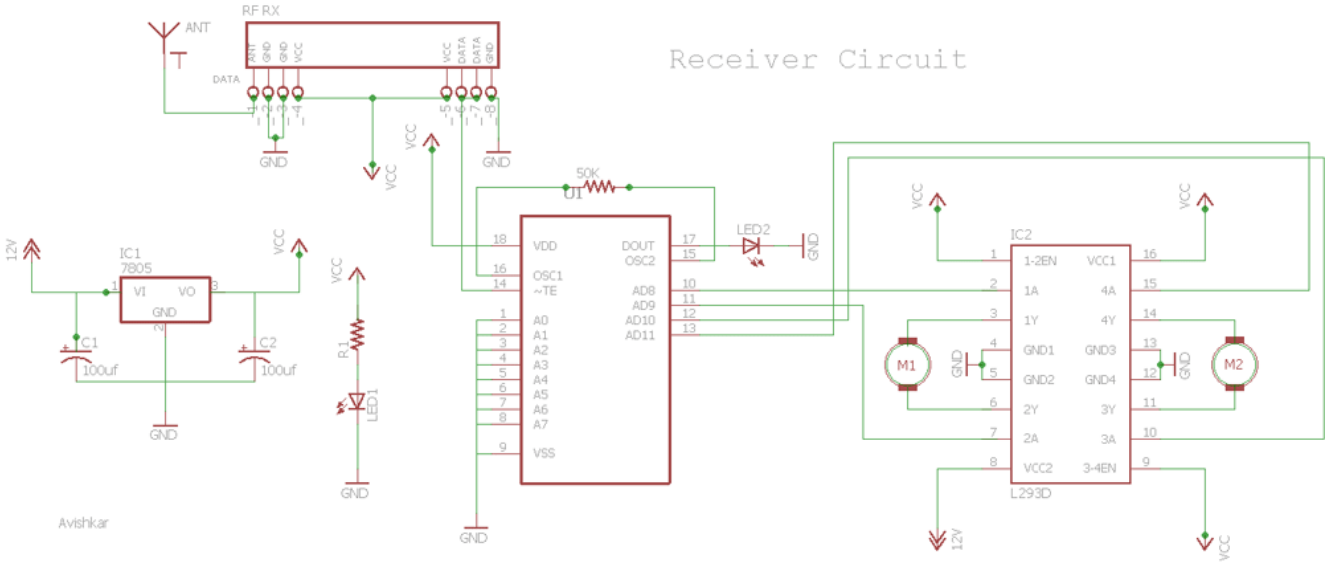
DPDT Switches for Remote Control

The transmitter circuit consists of ...

1. HT12E encoder (Pin Out)
2. RF transmitter module (Pin Out)
3. 2 DPDT switches
4. Power supply circuit
5. 1M resistor

You can see I have marked A, B, C, D in the transmitter circuit after the switch. The same has been marked on the DPDT switch diagram. Connect the A,B,C,D on the transmitter circuit to the A,B,C,D on the two DPDT switches.

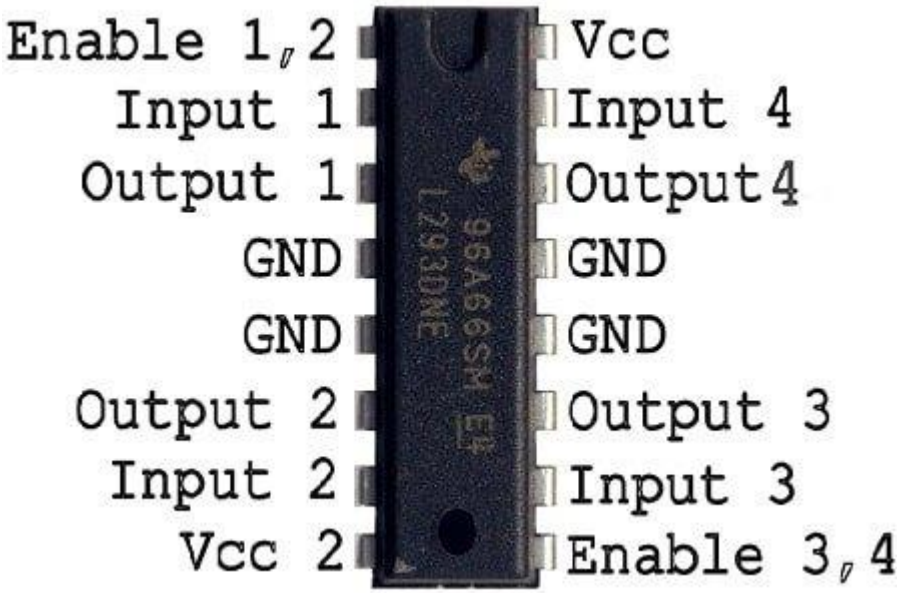
Making the Receiver Circuit



The receiver circuit consists of three ICs:

1. HT12D decoder (Pin Out)
2. L293D motor driver (Pin Out)
3. RF receiver module (Pin Out)

Wire the circuit as per the above receiver schematic. There are two LEDs in the receiver board. One lights up when power supply is given to the receiver. The other one near the IC HT12D should light up when power supply is given to transmitter circuit. This provides you with 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.



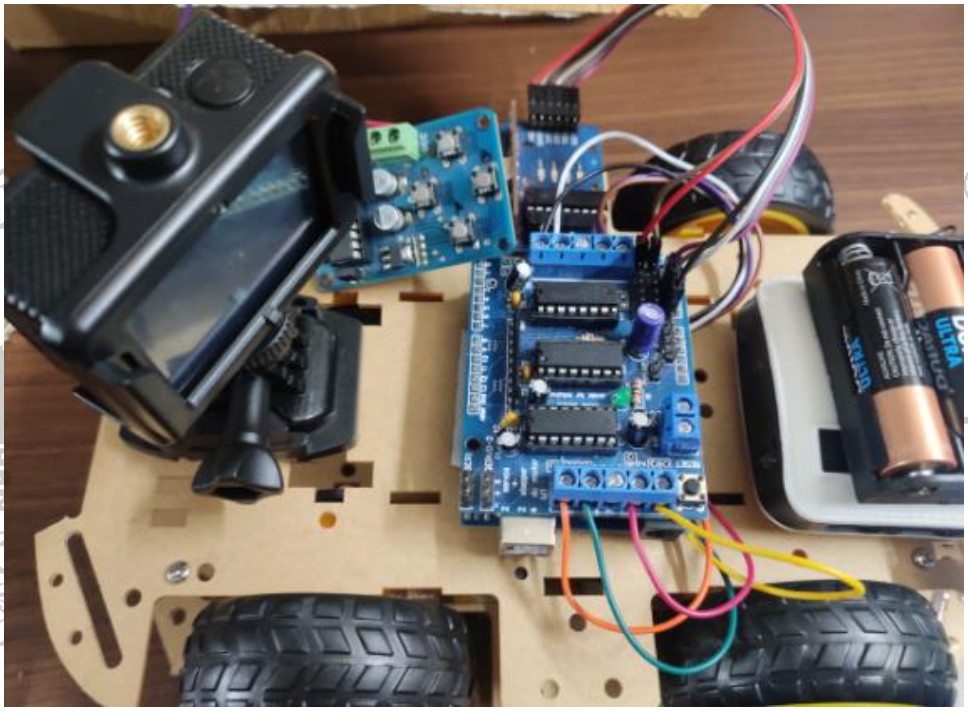
NOTE: Use red wires for positive and black for negative. If there are any problems, it will be easier to debug the circuit.

Choosing the Right Motor

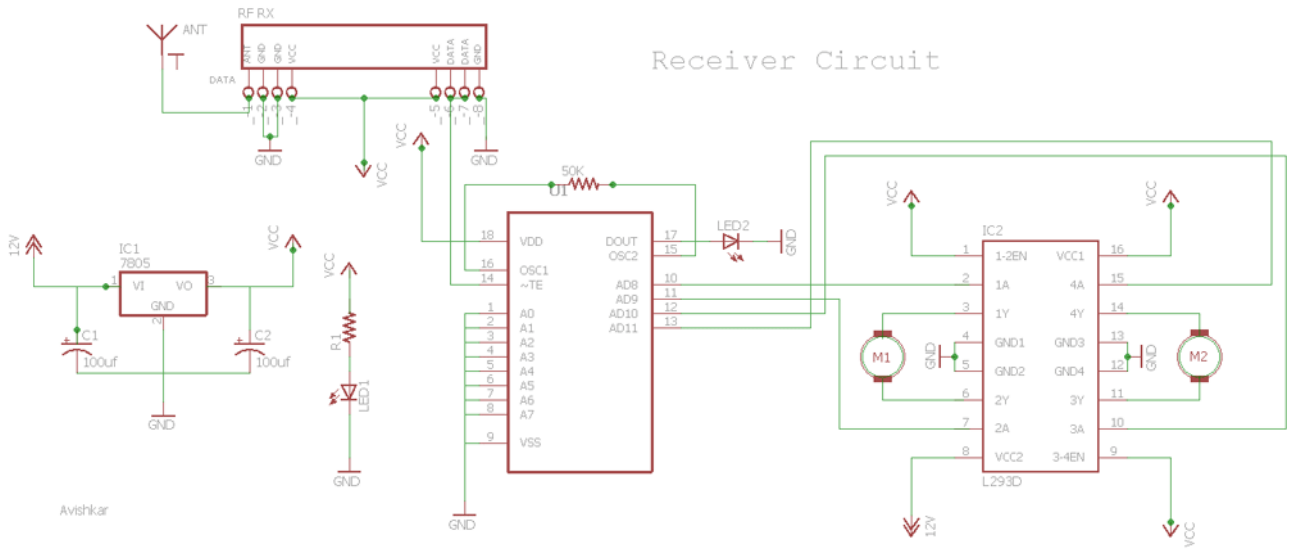
Choosing a motor is very important, and totally depends on the type of robot (car) you are making. If you are making a smaller one, use 6V Bo motor. If you are making a larger one, which will need to carry heavy load, then use a 12V DC motor.

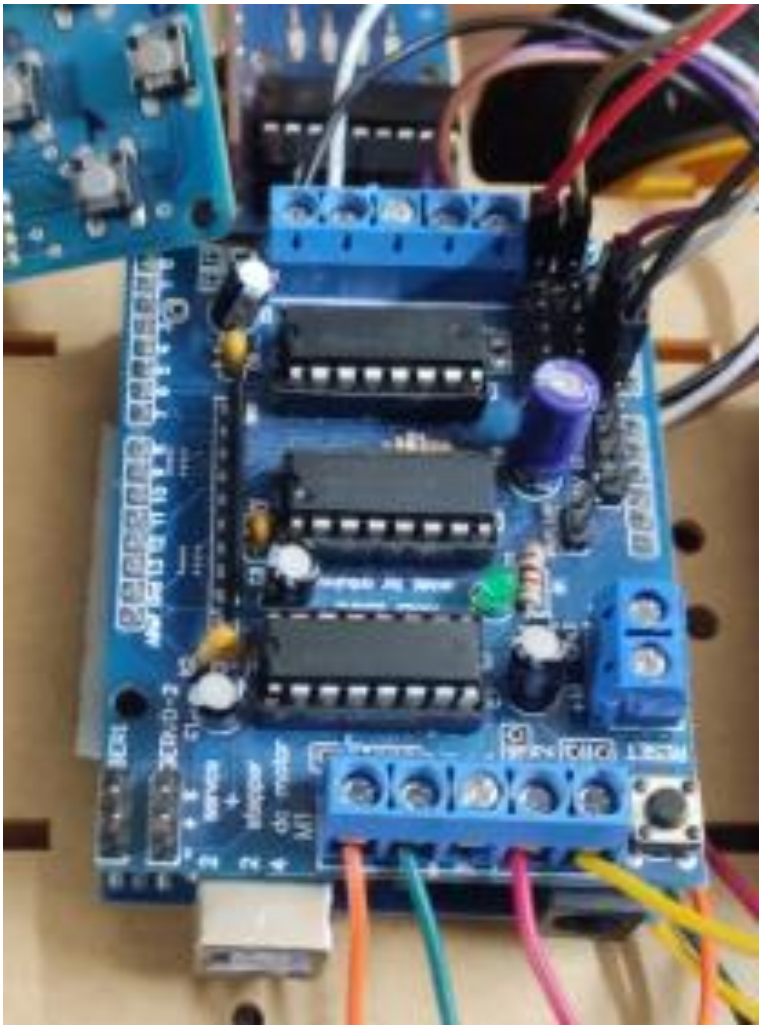
THINGS USED IN THIS PROJECT:

- Arduino UNO
- L293D motor driver
- Jumper wires
- Elastic band
- Snap cable for power supply from battery to arduino
- 9V battery
- 6V battery
- ASK Radio Frequency module
- Encoder
- Decoder



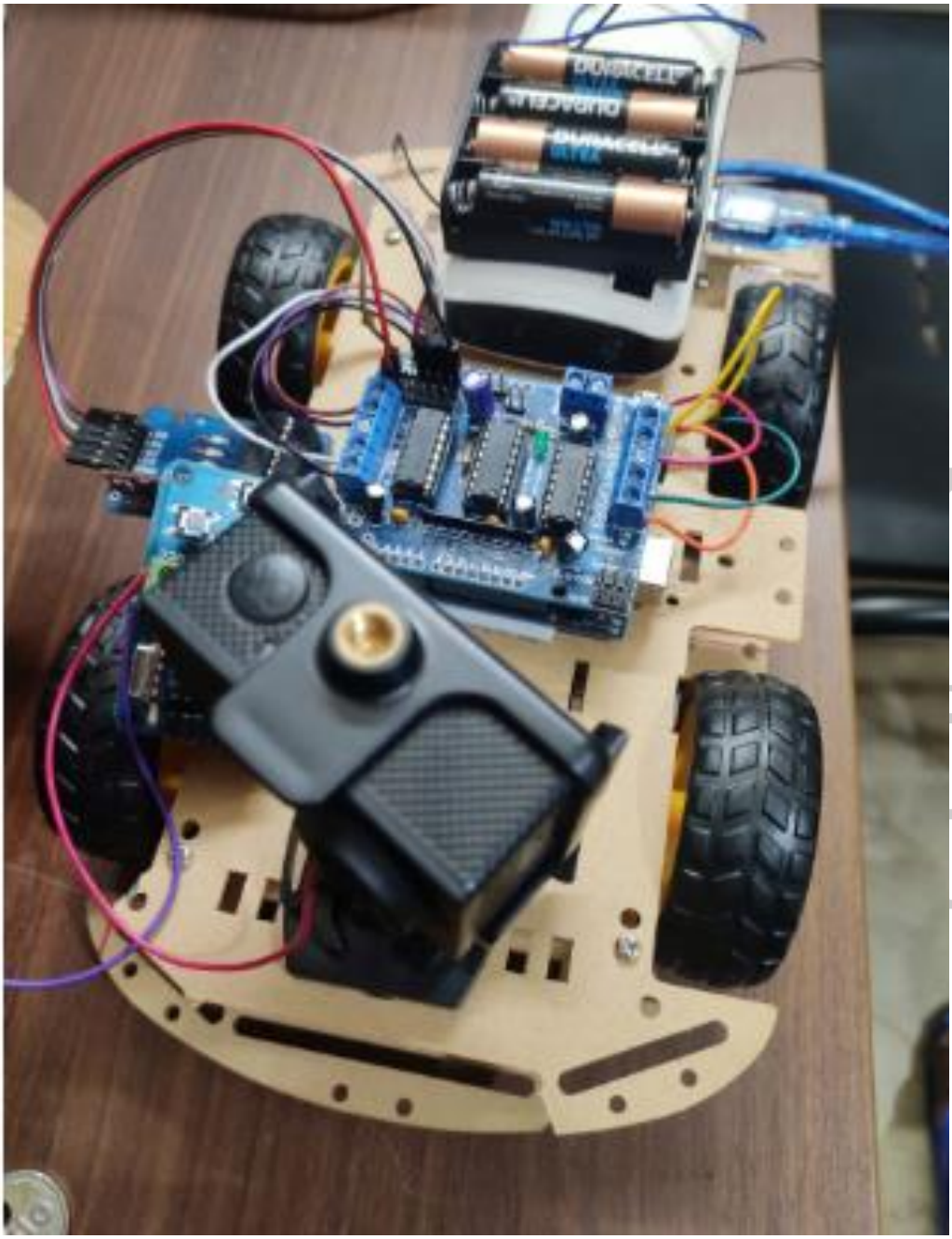
Basic* circuit diagram





6. Results and Discussions:

.



The project thus formed can be used for various purposes and the car can be controlled On the basis of RF Transmission with 433.93MHz.

7. Conclusion and Improvements:

- For now, working prototype can send the transmission signal and receive the signal and based on that it can be maneuvered.
- Now it records the video later it may send the live feed to the user.
- The accuracy of the model can be improvised from a small region to a big region
- The same concept can be used for special operations by the armed forces for scanning the horizon.

8. References:

- https://www.arduino.cc/en/Guide/ArduinoMega2560
- http://playground.arduino.cc
- https://learn.adafruit.com/category/learn-arduino
- www.c-shrpcorner.com
- https://msdn.microsoft.com
- www.codeproject.com
- Sams Teach Yourself " Arduino Programming" by Richard Blum ,Pearson Education 2015.

APPENDIX:

Code used in Arduino:

```
#include <AFMotor.h>
const int analogInPin0 =
  A0;  // Analog input pin
  that the potentiometer is
  attached to
const int analogInPin1 =
  A1;  // Analog input pin
  that the potentiometer is
  attached to
const int analogInPin2 =
  A2;  // Analog input pin
  that the potentiometer is
  attached to
const int analogInPin3 =
  A3;  // Analog input pin
  that the potentiometer is
  attached to

int sensorValue0 = 0;       //
  value read from the pot
int sensorValue1 = 0;
int sensorValue2 = 0;
int sensorValue3 = 0;

AF_DCMotor motor(4);
AF_DCMotor motor1(3);
AF_DCMotor motor2(2);
```

```
AF_DCMotor motor3(1);
void setup() {
  Serial.begin(9600);
  // set up Serial library at
  9600 bps
  Serial.println("Motor
  test!");

  // turn on motor
  motor.setSpeed(200);
 motor1.setSpeed(200);
 motor2.setSpeed(200);
 motor3.setSpeed(200);
  motor.run(RELEASE);
  motor1.run(RELEASE);
motor2.run(RELEASE);
  motor3.run(RELEASE);

}

void loop() {
  uint8_t i;
  sensorValue0 =
  analogRead(analogInPin0)
  ;
  sensorValue1 =
  analogRead(analogInPin1)
  ;
  sensorValue2 =
  analogRead(analogInPin2)
```

```
;
sensorValue3 =
analogRead(analogInPin3)
;

Serial.print("\n sensor =
");
Serial.print(sensorValue0);



while(sensorValue0<=80)
{
Serial.print("forward");
motor.run(FORWARD);
motor1.run(FORWARD);
motor2.run(FORWARD);
motor3.run(FORWARD);
motor.setSpeed(255);
motor1.setSpeed(255);
motor2.setSpeed(255);
motor3.setSpeed(255);
delay(10);
sensorValue0 =
analogRead(analogInPin0)
;
if(sensorValue0>80)
{
  motor.run(RELEASE);
  motor1.run(RELEASE);
  motor2.run(RELEASE);
```

```
  motor3.run(RELEASE);
}
}

while(sensorValue1<=80)
{
Serial.print("backward");
motor.run(BACKWARD);

motor1.run(BACKWARD
);

motor2.run(BACKWARD
);

motor3.run(BACKWARD
);
motor.setSpeed(255);
motor1.setSpeed(255);
motor2.setSpeed(255);
motor3.setSpeed(255);
delay(10);
sensorValue1 =
analogRead(analogInPin1)
;
if(sensorValue1>80)
{
  motor.run(RELEASE);
  motor1.run(RELEASE);
  motor2.run(RELEASE);
```

```
  motor3.run(RELEASE);
}
}

while(sensorValue2<=80)
{
Serial.print("left");
motor.run(FORWARD);
motor1.run(FORWARD);

motor2.run(BACKWARD
);

motor3.run(BACKWARD
);
motor.setSpeed(255);
motor1.setSpeed(255);
motor2.setSpeed(255);
motor3.setSpeed(255);
delay(10);
sensorValue2 =
analogRead(analogInPin2)
;
if(sensorValue2>80)
{
  motor.run(RELEASE);
  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
}
```

```
  }

  while(sensorValue3<=80)
  {
  Serial.print("right");
  motor.run(BACKWARD);

  motor1.run(BACKWARD
  );
  motor2.run(FORWARD);
  motor3.run(FORWARD);
  motor.setSpeed(255);
  motor1.setSpeed(255);
  motor2.setSpeed(255);
  motor3.setSpeed(255);
  delay(10);

  sensorValue3 =
  analogRead(analogInPin3)
  ;
  if(sensorValue3>80)
  {
    motor.run(RELEASE);
    motor1.run(RELEASE);
    motor2.run(RELEASE);
    motor3.run(RELEASE);
  }
  }
}
```