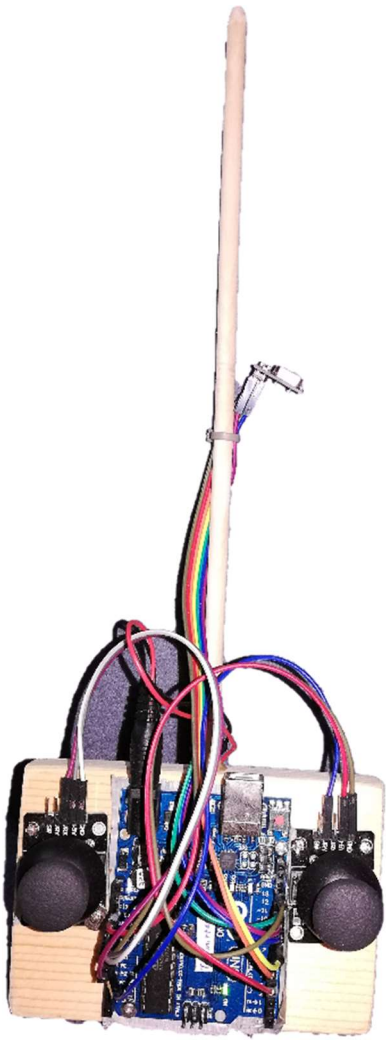
May 22, 2019

How to Build a Remote Control (R/C) Car



Materials Needed

- A used chassis with working motors (can find one for $1-$2 at DI's)
- x1: 5pack 9v Battery Clip with 2.1mm X 5.5mm Male DC Plug for Arduino by Corpco
- x1: 2 of SMAKN Fr4 Ky-023 Joystick Breakout Module Sensor Shield for Arduino Uno/arduino UNO R3/arduino 2560/arduino 2560 R
- x1: Qunqi 2Packs L298N Motor Drive Controller Board Module Dual H Bridge DC Stepper For Arduino
- x1: Multicolored Breadboard Dupont Jumper Wires – ALLUS J7011 120Pcs 3in1 Ribbon Cables Kit, Male to Male (M/M), Female to Female (F/F), Male to Female (M/F) for Arduino and Raspberry Pi
- x2: Arduino Uno R3 Development Board, Kit Microcontroller Based on ATmega328 and ATMEGA16U2 with USB Cable for Arduino, Original
- x1: 10pcs Arduino NRF24L01+ 2.4GHz Wireless RF Transceiver Module New Sold by: Deegoo-FPV
- x1: **6 AA Battery Holder With 2.1mm x 5.5mm Connector 9V Output 2 Pack by Corpco**
- **3/8" dowel**

- **3''x5''x1'' block of wood, with back of block recessed for a 9 volt battery**
- **3''x5''x1/4'' wood panel (for the receiver)**
- **~30 small machine screws to affix circuit boards to block of wood**
- **Rubber bands/velcro to attach receiver assembly to the chassis**

Tools Needed

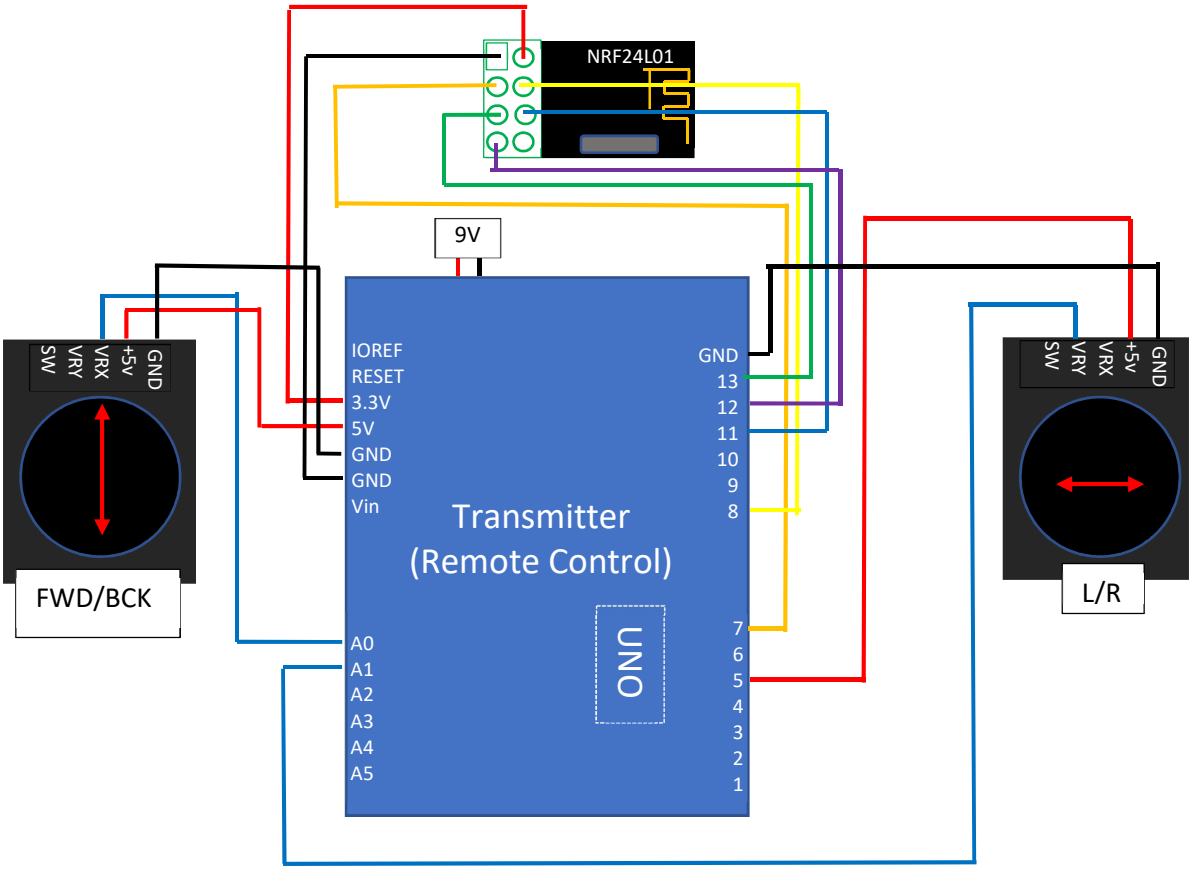
- Arduino App (found in the Windows App Store)
- A computer with a USB port
- Basic voltmeter to check for correct voltages and electrical connections
- The NRF24L01 libraries found at https://github.com/nRF24/RF24
- Saw for cutting wood

Instructions

- Go to the Windows Store and download the Arduino app: https://www.microsoft.com/en-us/p/arduino-ide/9nblggh4rsd8#activetab=pivot:overviewtab
- Download the NRF24L01 libraries and place in the default Arduino libraries path that was created when the Arduino app was installed: \Documents\Arduino\libraries\RF24-master
- Copy the code for the RC_Car_Transmitter and RC_Car_Receiver (see following sections)

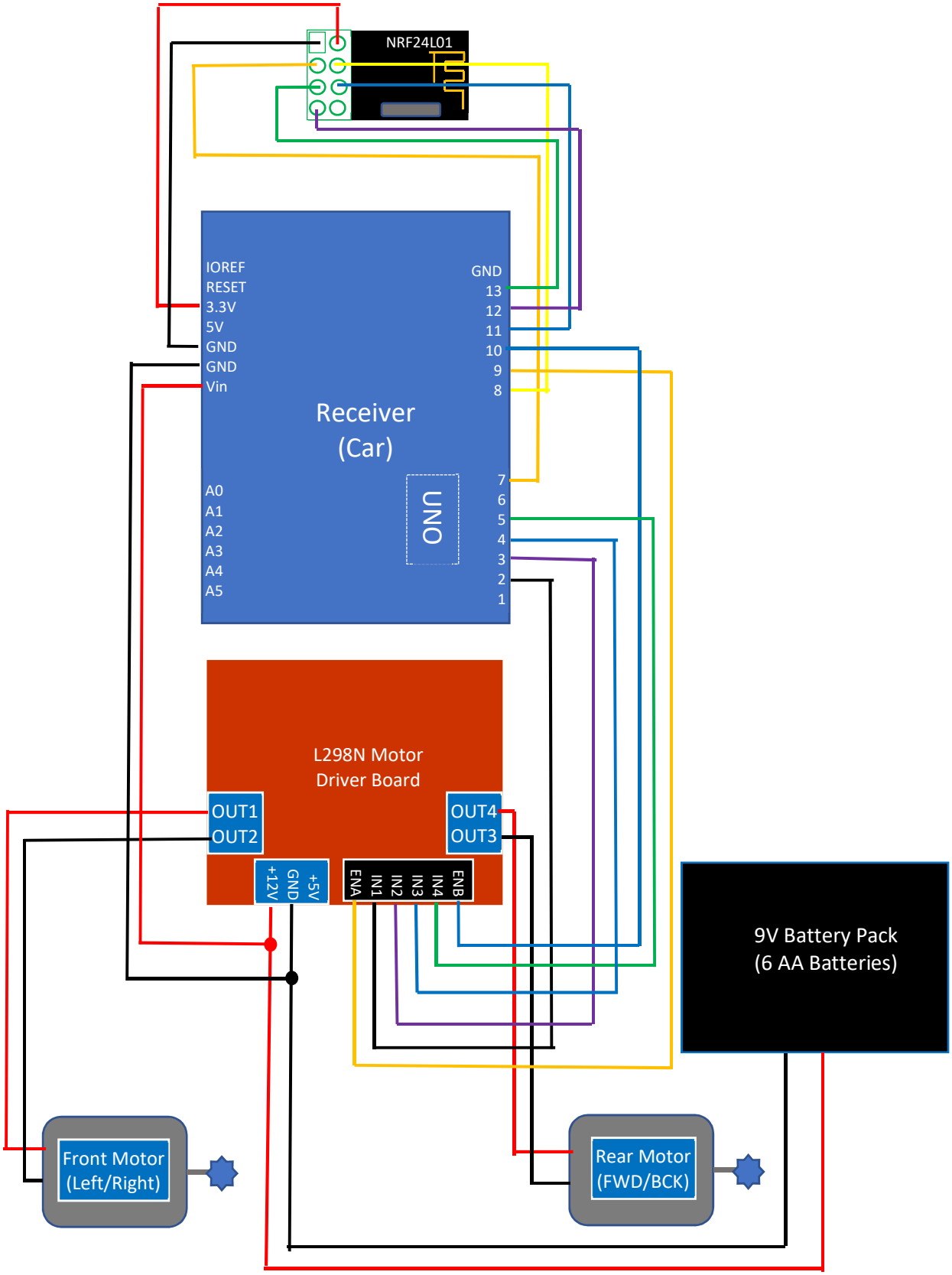
May 22, 2019

Transmitter Wiring



May 22, 2019

Receiver Wiring



May 22, 2019

RC_Car_Transmitter Code

```
 // define the radio
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
#define CE_PIN   7
#define CSN_PIN 8
const byte slaveAddress[5] = {'R','x','A','A','A'};
RF24 radio(CE_PIN, CSN_PIN); // Create a Radio

// joystick paremeters: define the data that will be sent
//char dataToSend[10] = "Message 0";
int joystick_bk_fwd = A0; // forward backward movement
int joystick_L_R = A1; // left right movement
int bk_fwd_val = 0;
int L_R_val = 0;
int input_cmd[2];

int l_r_power = 4;



void setup() {

   Serial.begin(9600);

   radio.begin();
   radio.setDataRate( RF24_250KBPS );
   radio.setRetries(3,5); // delay, count
   radio.openWritingPipe(slaveAddress);
```

```
   digitalWrite(l_r_power,HIGH);
}

void loop() {

 bk_fwd_val = analogRead(joystick_bk_fwd);
 L_R_val = analogRead(joystick_L_R);

 input_cmd[0]=bk_fwd_val;
 input_cmd[1]=L_R_val*10;

 //delay(400);



 //Serial.print("backward-forward value: ");
 //Serial.println(input_cmd[0]);

 Serial.print("left-right value: ");
 Serial.println(input_cmd[1]);



 radio.write( &input_cmd, sizeof(input_cmd) );
 //radio.write( &L_R_val, sizeof(L_R_val) );

}
```

RC_Car_Receiver Code

```
// define the radio
#include <SPI.h>
```

```
#include <nRF24L01.h>

#include <RF24.h>

#define CE_PIN   7

#define CSN_PIN 8

const byte thisSlaveAddress[5] = {'R','x','A','A','A'};

RF24 radio(CE_PIN, CSN_PIN);


// output pin assigments

int enA = 9; // goes to the motor circuit

int IN1 = 2; // goes to the motor circuit

int IN2 = 3; // goes to the motor circuit

int enB = 10; // goes to the motor circuit

int IN3 = 4; // goes to the motor circuit

int IN4 = 5; // goes to the motor circuit


// joystick parameters

int bk_fwd_val = 500;

int L_R_val = 500;

int input_cmds[2];

boolean ONE_PRESS=false;


// motor parameters

int motorSpeedA = 0;

int motorSpeedB = 0;


void setup() {

  // put your setup code here, to run once:


  Serial.begin(9600);
```

```
  // define the radio
  radio.begin();
  radio.setDataRate( RF24_250KBPS );
  radio.openReadingPipe(1, thisSlaveAddress);
  radio.startListening();

  pinMode(enA, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(enB, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);

}

void loop() {

  //bk_fwd_val = analogRead(joystick_bk_fwd);
  //radio.read( &bk_fwd_val, sizeof(bk_fwd_val) );
  radio.read( &input_cmds, sizeof(input_cmds) );

  L_R_val = input_cmds[1];
  bk_fwd_val = input_cmds[0];

  // had to add this in, or else the receiver gets updated too fast and the car turns on/off/on/off
  delay(70);
  //delay(400);

  //Serial.print("bk_fwd_val: ");
  //Serial.println(bk_fwd_val);
```

```
 //Serial.print("L_R_val: ");
 //Serial.println(L_R_val);

 // left/right logic
 if (L_R_val < 400) {
  Serial.println("<<<<<<<<<<<<<<<turn the car left");
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  motorSpeedA = 250;
  analogWrite(enA,motorSpeedA);
 }
 else if (L_R_val > 600) {
  Serial.println("turn the car right>>>>>>>>>>>>>");
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  motorSpeedA = 250;
  analogWrite(enA,motorSpeedA);
 }
 else {
  //Serial.println("the car shouldn't turn at all: ");
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  motorSpeedA = 0;
  analogWrite(enA,motorSpeedA);
 }

 // backward/forward logic
 if (bk_fwd_val < 450) {
  Serial.println("make the car go backward---VVVVVVV");
  digitalWrite(IN3, HIGH);
```

```
    digitalWrite(IN4, LOW);
    motorSpeedB=250;
    analogWrite(enB,motorSpeedB);

  }
  else if (bk_fwd_val > 550) {
    Serial.println("make the car go forward---^^^^^^^^");

    digitalWrite(IN3, LOW);
    digitalWrite(IN4, HIGH);

    //motorSpeedB=bk_fwd_val/4.1;
    motorSpeedB=250;
    analogWrite(enB,motorSpeedB);
  }
  else {
    //Serial.println("[[[[[[[[stop the car]]]]]]]]]]");

    //digitalWrite(IN1, LOW);
    //digitalWrite(IN2, LOW);
    //motorSpeedA = 0;
    //analogWrite(enA,motorSpeedA);

    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);
    motorSpeedB = 0;
    analogWrite(enB,motorSpeedB);
  }

}
```

Support

- Feel Free to call/text Jacey @ 385-288-2605, email: codiac06@gmail.com
- Visit the forum that tracks the history of making this car: https://forum.arduino.cc/index.php?topic=594674.new#new