

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



پروژه ریزپردازنده

ربات کنترلی با فرستنده - گیرنده دوربرد  
(NRF)

ویانا رستمی  
مهدیه کاتبی  
الیاس سفردوست

استاد مربوطه:  
محسن عباسی جنت آباد

آذر 1403

این پروژه با هدف طراحی و ساخت یک ماشین متحرک کنترلی بی سیم انجام شده است که از طریق جوی استیک و ماژول NRF24L01 کنترل می شود. سیستم طراحی شده شامل یک بخش فرستنده (جوی استیک متصل به میکروکنترلر) و یک بخش گیرنده (ماشین مجهز به میکروکنترلر و موتورهای DC) است. ارتباط بین این دو بخش از طریق فناوری بی سیم برقرار می شود. مراحل پروژه شامل طراحی شماتیک مدار، مونتاژ سخت افزار، برنامه نویسی میکروکنترلر، و تست های عملکردی بوده است. آزمایش ها نشان داده اند که سیستم از دقت و پایداری مطلوبی در ارتباط و اجرای دستورات حرکتی برخوردار است. نتایج این پروژه می تواند به عنوان پایه ای برای توسعه ربات های پیشرفته تر در حوزه های آموزشی، تحقیقاتی و صنعتی مورد استفاده قرار گیرد.

## فهرست مطالب

صفحه	عنوان
5.....	هدف پروژه.....
6.....	1- تکنولوژی های مورد استفاده.....
7.....	2- معرفی اجزای مکانیکی ربات.....
8.....	3- شماتیک مدار:.....
9.....	4- مراحل مونتاژ ربات:.....
11.....	5- مراحل برنامه نویسی و تست برنامه:.....
14.....	6- آزمایش های عملکردی.....
15.....	7- محدودیت ها.....
16.....	8- تحلیل نتایج.....
17.....	9- چالش ها و راهکارها.....
18.....	10- نتیجه گیری و پیشنهادات آینده.....
19.....	11- منابع.....

## هدف پروژه

هدف این پروژه طراحی و ساخت یک سیستم کنترل از راه دور برای یک ماشین متحرک است که از طریق جوی استیک و ماژول ارتباطی بی‌سیم NRF24L01 کنترل می‌شود. این پروژه به منظور توسعه یک پلتفرم آموزشی و کاربردی برای یادگیری و پیاده‌سازی مفاهیم الکترونیک، برنامه‌نویسی میکروکنترلرها، و فناوری ارتباط بی‌سیم طراحی شده است. از اهداف کلیدی این پروژه می‌توان به موارد زیر اشاره کرد:

1. ایجاد ارتباط بی‌سیم پایدار و کم‌مصرف بین فرستنده (جوی استیک) و گیرنده (ماشین).
2. کنترل دقیق حرکات ماشین شامل حرکت در جهتهای مختلف و توقف به موقع.
3. طراحی و پیاده‌سازی یک سیستم مقرون به صرفه که برای پروژه‌های تحقیقاتی و کاربردهای عملی قابل توسعه باشد.
4. بررسی و رفع چالش‌های ارتباط بی‌سیم در محیط‌های واقعی و آزمایشگاهی.

## تکنولوژی های مورد استفاده

این بخش به اجزای الکترونیکی و فناوری های مرتبط می پردازد، مانند ابزارهایی که برای انتقال داده ها، کنترل ربات، و تامین انرژی به کار رفته اند:

- Arduino 2 عدد: میکروکنترلر برای پردازش و کنترل حرکات ربات.
- جوی استیک (2 عدد): برای ارسال دستورات حرکتی به ماشین.
- ماژول NRF24L01 2 عدد: برای ارتباط بی سیم بین جوی استیک و ماشین.
- درایور موتور: برای کنترل سرعت و جهت موتورهای DC.
- رگولاتورهای ولتاژ (3 عدد): برای تامین ولتاژ پایدار (3.3 ولت) برای ماژول ها.
- خازن های 16 آمپر 4  $22\mu F$  عدد: برای تثبیت ولتاژ در مدار.

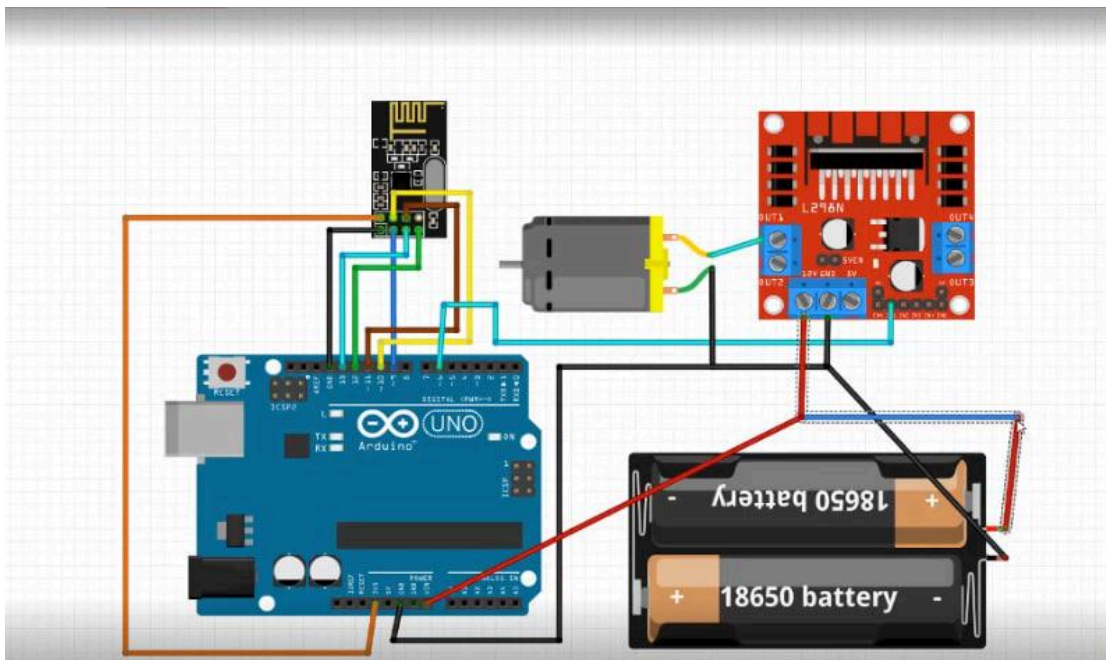
## معرفی اجزای مکانیکی ربات

این بخش به بخش‌های فیزیکی و مکانیکی ربات می‌پردازد، شامل بدنه و اجزایی که به حرکت و استحکام ربات کمک می‌کنند:

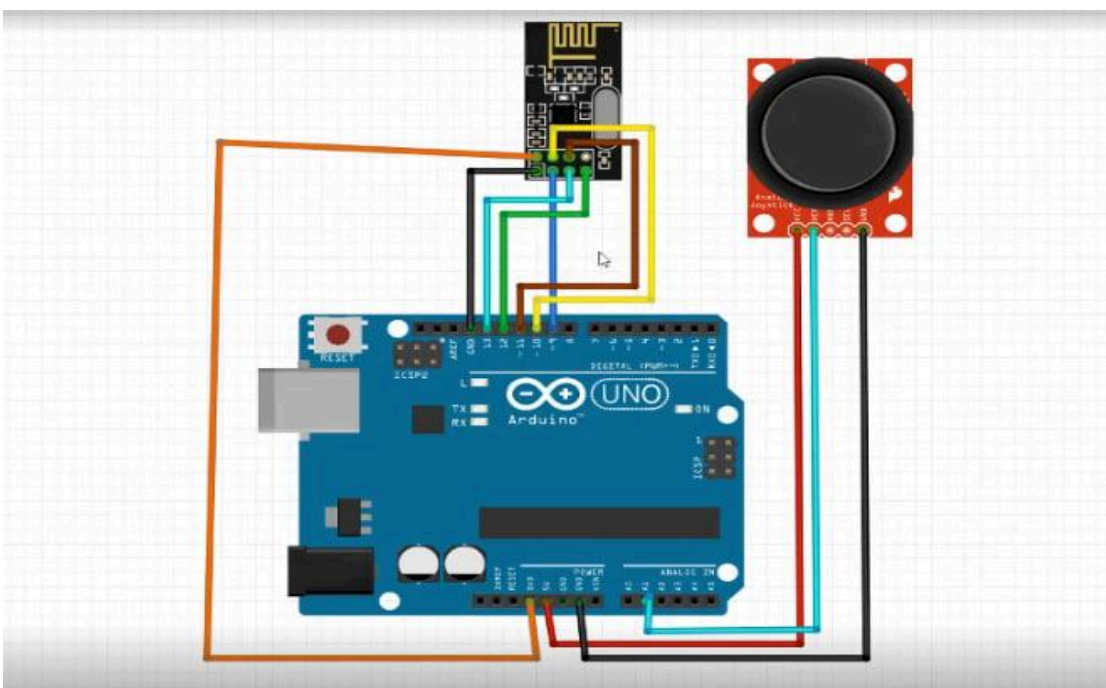
- موتور (4 عدد): نیروی حرکتی برای چرخ‌های ربات.
- چرخ (4 عدد): برای حرکت روی سطوح مختلف.
- استند: نگهدارنده برای پایداری ماشین و نصب قطعات.
- باتری‌های 3.6 ولت (4 عدد): منبع انرژی برای موتور و مدار.
- سیم جامپر و برد مورد: برای اتصال اجزای مدار الکترونیکی.

## شماتیک مدار

گیرنده



فرستنده





## مراحل مونتاژ ربات

. مونتاژ اجزای مکانیکی:

1. نصب موتورها:
  - چهار موتور DC را به پایه‌های استند یا شاسی ربات متصل کنید.
  - از پیچ و مهره برای محکم کردن موتورها استفاده کنید.
2. اتصال چرخ‌ها به موتورها:
  - چرخ‌ها را به شفت موتورها متصل کنید. اطمینان حاصل کنید که به خوبی جا افتاده‌اند و لغزش ندارند.
3. نصب باتری‌ها روی شاسی:
  - باتری‌ها را در محل مناسب روی شاسی نصب کنید. می‌توانید از نگهدارنده‌های باتری یا چسب دوطرفه استفاده کنید.
4. محکم کردن استند:
  - اجزای ربات (مثل شاسی) را به هم وصل کنید تا ساختار پایداری برای ربات شکل گیرد.

## اتصال اجزای الکترونیکی:

1. نصب برد آردوینو: برد آردوینو را روی شاسی ربات نصب کنید. از فاصله‌دهنده‌های پلاستیکی برای جلوگیری از اتصال کوتاه استفاده کنید.
2. اتصالات موتور و درایور:
  - سیم‌های هر موتور را به خروجی‌های مناسب درایور موتور متصل کنید.
  - درایور موتور را به برد آردوینو متصل کنید (پین‌های دیجیتال کنترل جهت و سرعت).
3. اتصال ماژول NRF24L01:
  - طبق تنظیمات زیر، ماژول NRF24L01 را به برد آردوینو وصل کنید:
    - GND: به GND آردوینو.
    - VCC: به 3.3 ولت آردوینو یا رگولاتور ولتاژ.
    - CE: به پین دیجیتال 9 آردوینو.
    - CSN: به پین دیجیتال 10 آردوینو.
    - SCK: به پین دیجیتال 13 آردوینو.
    - MOSI: به پین دیجیتال 11 آردوینو.
    - MISO: به پین دیجیتال 12 آردوینو.
4. اتصال جوی‌استیک‌ها:
  - سیم‌های هر جوی‌استیک را به پین‌های آردوینو متصل کنید:
    - GND: به GND آردوینو.
    - VCC: به 5 ولت آردوینو.
    - VRx: به پین آنالوگ A0.
    - VRy: به پین آنالوگ A1.
    - SW (دکمه): به پین دیجیتال D2 (اختیاری).
5. نصب خازن‌ها برای پایداری تغذیه:
  - خازن‌های  $10\mu F$  و  $0.1\mu F$  را بین پایه‌های VCC و GND ماژول NRF24L01 قرار دهید.
6. اتصال منبع تغذیه:
  - سیم‌های باتری را به درایور موتور و ماژول‌ها متصل کنید.
  - از یک منبع تغذیه جداگانه برای ماژول NRF24L01 استفاده کنید (در صورت وجود نویز).
  - اطمینان حاصل کنید که تمامی سیم‌ها محکم و اتصال به‌درستی انجام شده‌اند.

## مراحل برنامه‌نویسی و تست برنامه

1. نصب نرم‌افزار Arduino IDE:
  - Arduino IDE را از [سایت رسمی آردوینو](#) دانلود و نصب کنید.
2. نصب کتابخانه‌های موردنیاز:
  - کتابخانه‌های مرتبط با ماژول NRF24L01 و جوی‌استیک را نصب کنید:
    - RF24: برای کار با ماژول NRF24L01.
    - از منوی Sketch → Include Library → Manage Libraries استفاده کنید و عبارت "RF24" را جستجو و نصب کنید.
3. اتصال آردوینو به کامپیوتر:
  - برد آردوینو را از طریق کابل USB به کامپیوتر متصل کنید.
  - در Arduino IDE از مسیر Board → Tools → Port و Tools → Board برد و پورت صحیح را انتخاب کنید.

برنامه پایه برای ارتباط جوی‌استیک و NRF24L01:

1. تعریف کتابخانه‌ها و متغیرها:

```
1  #include <SPI.h>
2  #include <nRF24L01.h>
3  #include <RF24.h>
4
5  RF24 radio(9, 10); // CE, CSN
6  const byte address[6] = "000001";
7  int joystickX, joystickY, buttonState;
8  |
```

2. راه‌اندازی ماژول در بخش setup:

```
void setup() {
    Serial.begin(9600);
    radio.begin();
    radio.openWritingPipe(address);
    radio.setPALevel(RF24_PA_LOW);
    radio.stopListening();
}
```

### 3. خواندن جویاستیک و ارسال داده:

```
void loop() {  
    joystickX = analogRead(A0); // مقدار محور X  
    joystickY = analogRead(A1); // مقدار محور Y  
    buttonState = digitalRead(2); // وضعیت دکمه  
  
    int dataToSend[3] = {joystickX, joystickY, buttonState};  
    radio.write(&dataToSend, sizeof(dataToSend)); // ارسال داده ها  
  
    delay(100); // تاخیر برای پایداری ارسال  
}
```

### برنامه ربات گیرنده:

#### 1. تعریف متغیرها و تنظیمات:

```
#include <SPI.h>  
#include <nRF24L01.h>  
#include <RF24.h>  
  
RF24 radio(9, 10); // CE, CSN  
const byte address[6] = "00001";  
int receivedData[3]; // داده های دریافتی: X، Y، دکمه
```

#### 2. راه اندازی ماژول:

```
void setup() {  
    Serial.begin(9600);  
    radio.begin();  
    radio.openReadingPipe(0, address);  
    radio.setPALevel(RF24_PA_LOW);  
    radio.startListening();  
}
```

#### 3. دریافت داده و کنترل موتور:

```

void loop() {
    if (radio.available()) {
        radio.read(&receivedData, sizeof(receivedData));
        int joystickX = receivedData[0];
        int joystickY = receivedData[1];
        int buttonState = receivedData[2];

        // کنترل موتورها بر اساس داده ها
        if (joystickY > 600) {
            // حرکت به جلو
        } else if (joystickY < 400) {
            // حرکت به عقب
        } else if (joystickX > 600) {
            // چرخش به راست
        } else if (joystickX < 400) {
            // چرخش به چپ
        } else {
            // توقف
        }

        Serial.print("Joystick X: ");
        Serial.print(joystickX);
        Serial.print(" | Y: ");
        Serial.print(joystickY);
        Serial.print(" | Button: ");
        Serial.println(buttonState);
    }
}

```

### 3. تست برنامه:

1. آپلود کد در بردهای فرستنده و گیرنده:
  - ابتدا کد فرستنده را روی برد آردوینو متصل به جوی استیک آپلود کنید.
  - سپس کد گیرنده را روی برد آردوینو متصل به ربات آپلود کنید.
2. بررسی ارتباط بین ماژولها:
  - از Serial Monitor در Arduino IDE برای مشاهده داده‌های ارسال شده و دریافت شده استفاده کنید.
3. تست حرکات ربات:
  - جوی استیک را در جهات مختلف حرکت دهید و مطمئن شوید که موتورها بر اساس دستورالعمل‌ها حرکت می‌کنند.

## آزمایش‌های عملکردی

آزمایش‌های عملکردی برای ماژول NRF24L01:

1. آزمایش ارتباط بی‌سیم:

- بررسی صحت ارتباط بی‌سیم بین دو ماژول NRF24L01 (ارسال و دریافت داده‌ها).
- نتیجه انتظار: ارسال و دریافت داده‌ها با موفقیت انجام شود بدون قطعی یا تاخیر غیرمنتظره.
- کد نمونه: برای آزمایش ارتباط، می‌توانید از کدهای ارسال و دریافت ساده برای ارسال پیامی از یک ماژول و دریافت آن در ماژول دیگر استفاده کنید.

2. آزمایش پایداری تغذیه ماژول:

- بررسی عملکرد ماژول پس از اتصال خازن‌های  $10\mu F$  و  $0.1\mu F$  به پایه‌های VCC و GND.
- نتیجه انتظار: پایداری و کاهش نویز در اتصال و عملکرد بهتر در طول زمان.
- نتیجه مشاهده: پس از اتصال خازن‌ها، عدم وجود قطع شدن ارتباط یا مشکلات نویز مشاهده شد.

3. آزمایش اتصال به آردوینو:

- بررسی صحت اتصال پایه‌های ماژول NRF24L01 به پین‌های آردوینو طبق دستورالعمل.
- نتیجه انتظار: اتصال صحیح بدون خطا و عملکرد مناسب پس از انجام تنظیمات در نرم‌افزار.

آزمایش‌های عملکردی برای جوی‌استیک:

1. آزمایش حرکت محور X و Y:

- آزمایش اینکه آیا مقادیر خروجی محور X و Y به درستی تغییر می‌کنند با حرکت جوی‌استیک.
- نتیجه انتظار: مقادیر خوانده‌شده از ورودی‌های آنالوگ بین 0 تا 1023 تغییر کنند.
- کد نمونه: کد آزمایشی که مقادیر محور X و Y را در Serial Monitor چاپ می‌کند تا دقت حرکت جوی‌استیک را بررسی کنید.

2. آزمایش دکمه فشاری جوی‌استیک:

- بررسی عملکرد دکمه فشاری برای تشخیص وضعیت فشردن آن (0 یا 1).
- نتیجه انتظار: زمانی که دکمه فشرده می‌شود، مقدار خوانده‌شده برای دکمه باید 0 باشد و در حالت عادی 1 باشد.
- نتیجه مشاهده: وضعیت دکمه در Serial Monitor به درستی نمایش داده می‌شود.

## محدودیت ها

### محدودیت های ماژول NRF24L01:

#### 1. محدودیت تغذیه:

- ماژول NRF24L01 نیاز به تغذیه 3.3 ولت دارد و استفاده از 5 ولت ممکن است باعث خرابی ماژول شود.
- در صورت استفاده از آردوینو های قدیمی مانند UNO یا NANO، پین 3.3 ولت ممکن است توان کافی برای تامین انرژی ماژول را نداشته باشد، که ممکن است نیاز به استفاده از یک رگولاتور ولتاژ یا ماژول مبدل ولتاژ باشد.
- راه حل: استفاده از یک رگولاتور یا منبع تغذیه جداگانه برای تامین ولتاژ مناسب.

#### 2. محدودیت برد ارتباطی:

- برد ارتباطی ماژول NRF24L01 به شدت وابسته به نویز محیطی و کیفیت تغذیه است. در محیط هایی با نویز زیاد یا اتصالات ضعیف، ممکن است برد مفید کمتر از حد انتظار باشد.
- راه حل: استفاده از خازن های مناسب برای کاهش نویز و استفاده از سیم های کوتاه برای اتصال ماژول.

#### 3. محدودیت سرعت انتقال داده:

- در برخی شرایط، سرعت انتقال داده بین ماژول ها ممکن است تحت تاثیر نویز، فاصله، یا کیفیت سیگنال قرار گیرد.
- راه حل: کاهش فاصله بین ماژول ها یا استفاده از پروتکل های اصلاحی برای افزایش دقت انتقال داده.

### محدودیت های جوی استیک:

#### 1. محدودیت دقت:

- مقادیر خروجی از جوی استیک به صورت آنالوگ بین 0 تا 1023 است، و ممکن است در برخی شرایط دقت مورد نیاز برای کاربرد خاص فراهم نباشد. همچنین ممکن است برای جوی استیک های ارزان تر، دقت حرکات کمتر از حد مطلوب باشد.
- راه حل: انجام کالیبراسیون نرم افزاری برای افزایش دقت.

## تحلیل نتایج

. نتایج عملکرد ماژول NRF24L01

- پس از اتصال ماژول NRF24L01 به آردوینو و اجرای کد، عملکرد ماژول در ارسال و دریافت داده‌ها بررسی شد. در آزمایش‌ها، موفقیت‌های انتقال داده به‌طور کلی رضایت‌بخش بود و داده‌ها به‌درستی بین آردوینوها منتقل شدند.
- نتیجه مطلوب: عملکرد ماژول در شرایط تغذیه مناسب (3.3V) و استفاده از خازن‌های اضافی به‌خوبی ارزیابی شد و برد ارتباطی در فاصله‌های کوتاه (حدود 5 متر) عملکرد خوبی داشت

2. نتایج عملکرد جوی‌استیک

- جوی‌استیک به‌خوبی مقادیر آنالوگ محور X و Y را خوانده و داده‌های مربوط به موقعیت حرکت و فشردن دکمه به‌درستی در Serial Monitor نمایش داده شد.
- نتیجه مطلوب: دقت مقادیر خوانده‌شده برای محورهای X و Y به‌طور عمومی در محدوده مورد انتظار قرار داشت. مقادیر محور X و Y در مرکز جوی‌استیک به 512 نزدیک بودند، که نشان‌دهنده عملکرد درست و دقیق است.



## چالش‌ها و راهکارها

- در برخی شرایط با نویز بالا یا فاصله زیاد، سرعت انتقال داده کاهش یافت.
  - مشکلاتی در ارتباط در فواصل طولانی‌تر یا محیط‌های دارای تداخل سیگنال مشاهده شد.
- راه‌حل: استفاده از قطعات اضافی برای بهبود تغذیه و کاهش نویز، کاهش فاصله و انتخاب محیط‌های با نویز کمتر به عنوان راهکارهای عملی برای حل این مشکلات.
- در برخی از جوی‌استیک‌ها، تنظیمات مرکز دقیق نبوده و نیاز به کالیبراسیون بیشتری برای به دست آوردن دقت بیشتر در حرکت‌های جوی‌استیک بود.
  - دکمه فشاری در بعضی از مدل‌ها به درستی شناسایی نشد که موجب کاهش دقت در کنترل‌های اضافی شد.
- راه‌حل: استفاده از کالیبراسیون نرم‌افزاری برای دقت بیشتر و نصب مقاومت‌های pull-up برای بهبود تشخیص وضعیت دکمه.

## نتیجه گیری

در این پروژه، هدف اصلی طراحی و پیاده‌سازی یک سیستم ارتباطی بی‌سیم با استفاده از ماژول NRF24L01 و تعامل با جوی‌استیک برای دریافت داده‌های آنالوگ و دیجیتال بود. آزمایش‌های انجام‌شده نشان داد که این سیستم در شرایط عادی و در فواصل کوتاه (حدود ۵ متر) عملکرد قابل قبولی داشته است. انتقال داده‌ها بین ماژول‌ها به‌خوبی انجام شد و جوی‌استیک نیز مقادیر مورد انتظار از محورهای X و Y و وضعیت دکمه را به درستی شبیه‌سازی کرد.

## پیشنهادهای برای آینده

1. بهبود پایداری ارتباطات بی‌سیم:
  - استفاده از تقویت‌کننده سیگنال یا آنتن‌های خارجی می‌تواند باعث افزایش برد ارتباطی ماژول NRF24L01 شود.
  - در صورت نیاز به استفاده در فواصل بلندتر یا محیط‌های با نویز زیاد، می‌توان از ماژول‌های ارتباطی با برد بالاتر یا تکنولوژی‌های دیگر مانند LoRa استفاده کرد.
2. کالیبراسیون دقیق‌تر جوی‌استیک:
  - برای افزایش دقت و بهبود عملکرد جوی‌استیک، می‌توان از الگوریتم‌های نرم‌افزاری برای کالیبراسیون خودکار محورهای X و Y استفاده کرد.
  - استفاده از جوی‌استیک‌های با کیفیت بالاتر و دقت بیشتر نیز می‌تواند عملکرد سیستم را ارتقا دهد.
3. افزایش قابلیت اطمینان سیستم:
  - برای جلوگیری از قطع‌های ناگهانی در ارتباطات، می‌توان مکانیزم‌های تصحیح خطا را برای انتقال داده‌ها بین ماژول‌ها پیاده‌سازی کرد.
  - همچنین، استفاده از پروتکل‌های ارتباطی پایدارتر مانند I2C یا SPI برای مواقعی که به ارتباطات پایدارتر نیاز است، توصیه می‌شود.
4. استفاده از منابع تغذیه مستقل برای ماژول‌ها:
  - به منظور افزایش پایداری سیستم و کاهش تاثیر نویز در عملکرد ماژول‌ها، پیشنهاد می‌شود که منابع تغذیه مجزا برای ماژول‌ها به کار گرفته شوند، به‌ویژه اگر سیستم در شرایط محیطی با تداخل الکترومغناطیسی یا نیاز به مصرف توان بالا قرار گیرد.

## منابع

<https://wokwi.com/projects/new/arduino-uno?template=arduino-uno>  
<https://youtu.be/4GYGwT5vJ-M?si=uyJfMfSGKK0G9ITW>  
<https://www.aparat.com/v/o47k67x>  
<https://www.aparat.com/v/q4188ev>  
<https://irenx.ir/arduino/rc-control-car-with-arduino>  
<https://thecaferobot.com/learn/interfacing-nrf24l01-wireless-2-4-ghz-transceiver/-module-with-arduino>  
<https://www.aparat.com/v/j7019rb>