

# Implementasi Modul Nirkabel nRF24L01+ Sebagai Media Pengiriman dan Penerimaan Data Dengan Antarmuka NodeMCU

**Taryana Suryana**

081221480577

Teknik Informatika

Universitas Komputer Indonesia  
Jln.Dipatiukur 112-114 Bandung

[taryanarx@email.unikom.ac.id](mailto:taryanarx@email.unikom.ac.id) – [taryanarx@gmail.com](mailto:taryanarx@gmail.com)

[Jurnal Komputa Unikom 2021](#)

<http://iot.ciwaruga.com>

## **Pendahuluan**

Jika Anda Memiliki dua buah atau lebih perangkat Arduino kemudian ditambahkan dengan Modul nRF24L01 maka peralatan tersebut dapat dirangkau untuk berkomunikasi jarak jauh satu sama lain secara nirkabel, dan membuka banyak kemungkinan yang dapat anda lakukan misalnya memantau data sensor dari jarak jauh, mengendalikan robot, otomatisasi rumah, dan banyak lagi.

Modul transceiver nRF24L01+ (plus) dapat diperoleh secara online dan dapat dijadikan sebagai salah satu opsi komunikasi data paling murah yang bisa Anda dapatkan. Dan yang terbaik dari semuanya, modul ini sangat kecil, memungkinkan Anda untuk menggabungkan antarmuka nirkabel ke hampir semua proyek.

<https://iot.ciwaruga.com>

# Pembahasan

## Frekuensi Radio

Modul transceiver nRF24L01+ dirancang untuk beroperasi pada pita frekuensi ISM 2,4 GHz di seluruh dunia dan menggunakan [modulasi GFSK](#) untuk transmisi data. Kecepatan transfer data dapat berupa 250kbps, 1Mbps, dan 2Mbps.

### Apa itu pita ISM 2,4 GHz?

Pita 2,4 GHz adalah salah satu pita [Industri, Ilmiah, dan Medis \(ISM\)](#) yang dicadangkan secara internasional untuk penggunaan perangkat berdaya rendah tanpa izin. Contohnya adalah telepon tanpa kabel, perangkat Bluetooth, perangkat komunikasi jarak dekat (NFC), dan jaringan komputer nirkabel (WiFi) semuanya menggunakan frekuensi ISM.

## Konsumsi daya

Tegangan operasi modul nRF24L01 adalah dari 1,9 hingga 3,6V , tetapi kabar baiknya adalah bahwa pin logika toleran 5 volt , sehingga kita dapat dengan mudah menghubungkannya ke Arduino atau mikrokontroler logika 5V tanpa menggunakan konverter level logika apa pun.

Modul ini mendukung daya keluaran yang dapat diprogram yaitu. 0 dBm, -6 dBm, -12 dBm atau -18 dBm dan Luar biasanya hanya mengkonsumsi sekitar 12 mA selama transmisi pada 0 dBm, yang bahkan lebih rendah dari satu LED.

Dan yang terbaik lagi adalah modul ini hanya mengkonsumsi 26  $\mu$ A waktu mode standby dan 900 nA pada mode power down. Itu sebabnya modul ini termasuk perangkat nirkabel berdaya rendah.

## Antarmuka SPI

Modul transceiver nRF24L01+ berkomunikasi melalui Serial Peripheral Interface (SPI) 4-pin dengan kecepatan data maksimum 10Mbps. Semua parameter seperti saluran frekuensi (125 saluran yang dapat dipilih), daya keluaran (0 dBm, -6 dBm, -12 dBm atau -18 dBm), dan kecepatan data (250kbps, 1Mbps, atau 2Mbps) dapat dikonfigurasi melalui antarmuka SPI.

Bus SPI menggunakan konsep Master dan Slave, dalam aplikasi yang paling umum Arduino adalah Master dan modul transceiver nRF24L01+ adalah Slave. Berbeda dengan bus I2C jumlah slave pada bus SPI terbatas, pada Arduino Uno Anda dapat menggunakan maksimal dua slave SPI yaitu dua modul transceiver nRF24L01+.

Berikut spesifikasi lengkapnya:

Tabel 1. Spesifikasi nRF24L01

Rentang frekuensi	Pita ISM 2,4 GHz
Kecepatan Data Udara Maksimum	2 Mb/dtk
Format Modulasi	GFSK
Maks. Daya Keluaran	0 dBm
Tegangan Pasokan Operasi	1,9 V hingga 3,6 V
Maks. Operasi Saat Ini	13.5mA
min. Saat Ini (Mode Siaga)	26 $\mu$ A
Masukan Logika	5V Toleran
Jangkauan Komunikasi	800+ m (garis pandang)

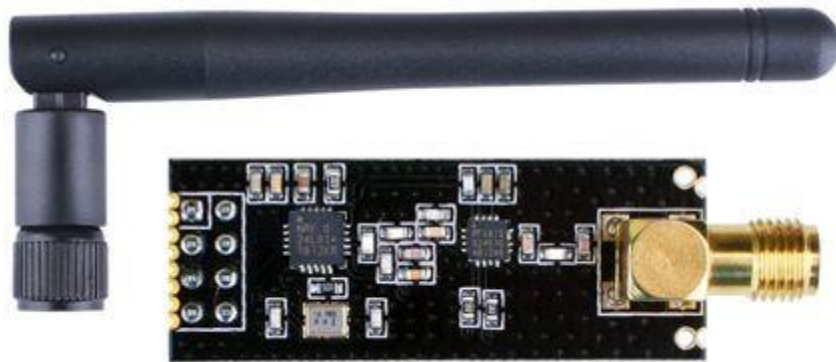
## nRF24L01+ modul Vs nRF24L01+ modul PA/LNA

Ada berbagai modul yang tersedia berdasarkan chip nRF24L01+. Di bawah ini adalah versi paling populer.



Gambar 1. nRF24L01+ Modul Nirkabel

Versi pertama menggunakan antena on-board. Ini memungkinkan versi breakout yang lebih ringkas. Namun, antena yang lebih kecil juga berarti jangkauan transmisi yang lebih rendah. Dengan versi ini, Anda akan dapat berkomunikasi melalui jarak 100 meter. Tentu saja itu di luar ruangan di ruang terbuka. Untuk jangkauan di dalam ruangan, terutama akan sedikit melemah apabila terhalang dinding atau tembok.

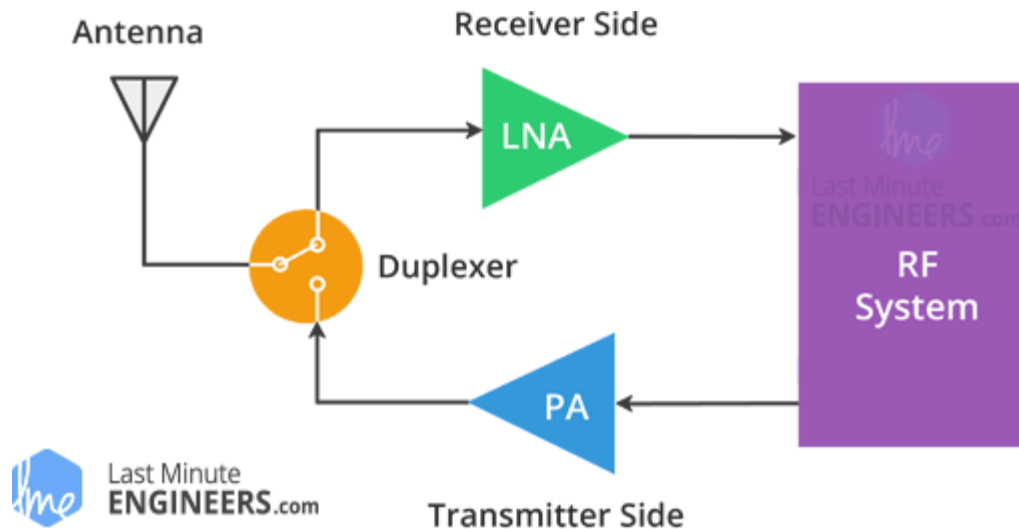


Gambar 2. Modul Transceiver nRF24L01+ PA LNA dengan Antena Eksternal

Versi kedua hadir dengan konektor SMA dan antena bebek tetapi bukan itu perbedaan sebenarnya. Perbedaan sebenarnya adalah ia hadir dengan [chip RFX2401C](#) khusus yang mengintegrasikan PA, LNA, dan sirkuit switching transmit-receive. Chip ini berfungsi untuk menguatkan power sehingga jangkauan tx-tx akan semakin jauh bisa lebih dari 1000m.

## Apa itu PA LNA?

PA adalah singkatan dari Power Amplifier . Ini hanya meningkatkan kekuatan sinyal yang ditransmisikan dari chip nRF24L01+. Padahal, LNA adalah singkatan dari Low-Noise Amplifier .



Gambar 3. Diagram Blok nRF24L01+ PA/LNA

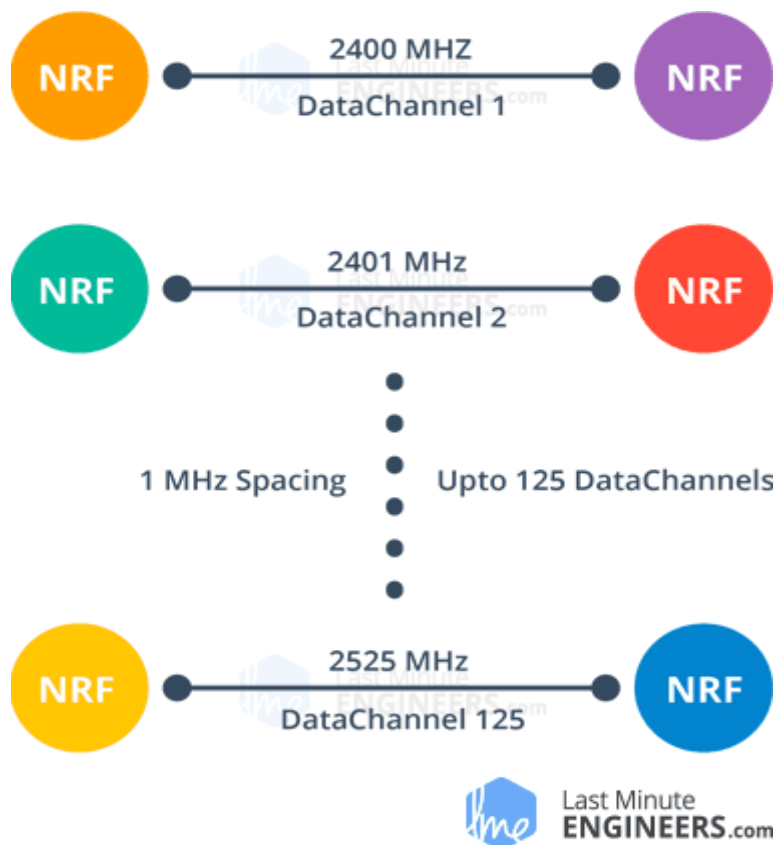
Fungsi LNA adalah untuk menangani sinyal yang sangat lemah dan tidak pasti dari antena (biasanya pada urutan mikrovolt atau di bawah -100 dBm) dan memperkuatnya ke tingkat yang lebih berguna (biasanya sekitar 0,5 hingga 1V)

Low-Noise Amplifier (LNA) dari jalur penerimaan dan penguat daya (PA) dari jalur transmisi terhubung ke antena melalui duplexer, yang memisahkan dua sinyal dan mencegah output PA yang relatif kuat dari kelebihan beban input LNA yang sensitif.

# Bagaimana modul transceiver nRF24L01+ bekerja?

Modul transceiver nRF24L01+ mentransmisikan dan menerima data pada frekuensi tertentu yang disebut Channel . Juga agar dua atau lebih modul transceiver dapat berkomunikasi satu sama lain, untuk dapat berkomunikasi harus berada di saluran yang sama. Channel ini bisa berupa frekuensi di pita ISM 2,4 GHz atau lebih tepatnya bisa antara 2.400 hingga 2.525 GHz (2400 hingga 2525 MHz).

Setiap Channel menempati bandwidth kurang dari 1MHz. Ini memberi kita sebanyak 125 saluran yang dapat dialokasikan dengan jarak 1MHz. Jadi, modul dapat menggunakan 125 saluran berbeda yang memberikan kemungkinan untuk memiliki jaringan 125 modem yang bekerja secara independen di satu tempat.



Gambar 4. Pembagian Channel Frekuensi

Setiap Channel menempati bandwidth kurang dari 1MHz pada kecepatan data 250kbps dan 1Mbps. Namun pada kecepatan data udara 2Mbps, bandwidth 2MHz ditempati (lebih lebar dari resolusi pengaturan frekuensi saluran RF). Jadi, untuk memastikan saluran yang tidak tumpang tindih dan mengurangi cross-talk dalam mode 2Mbps, Anda perlu menjaga jarak 2MHz di antara dua channel.

Frekuensi saluran RF dari saluran yang Anda pilih diatur sesuai dengan rumus berikut:

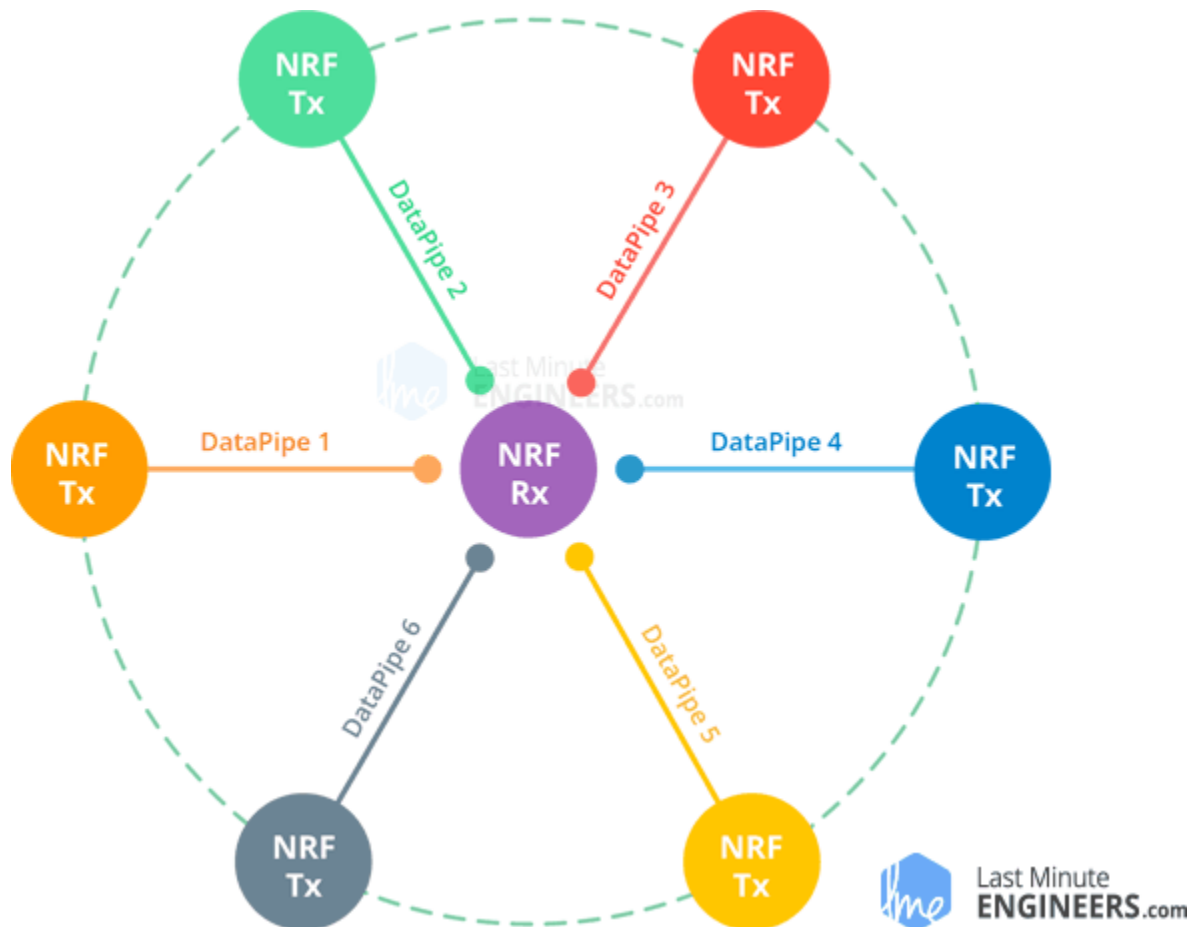
$$\text{Freq (Terpilih)} = 2400 + \text{CH (Terpilih)}$$

Misalnya, jika Anda memilih Channel 108 untuk transmisi data, maka channel frekuensi RF Anda akan menjadi 2508MHz (2400 + 108).

## nRF24L01+ Jaringan Multiceiver

nRF24L01+ menyediakan fitur yang disebut Multiceiver. Ini adalah singkatan dari Multiple Transmitters Single Receiver. Di mana setiap Channel RF secara logis dibagi menjadi 6 channel data paralel yang disebut Data Pipes. Dengan kata lain, pipa data adalah Channel logis dalam Channel RF fisik. Setiap pipa data memiliki alamat fisiknya sendiri (Alamat Pipa Data) dan dapat dikonfigurasi.

Hal ini dapat digambarkan seperti gambar di bawah ini.



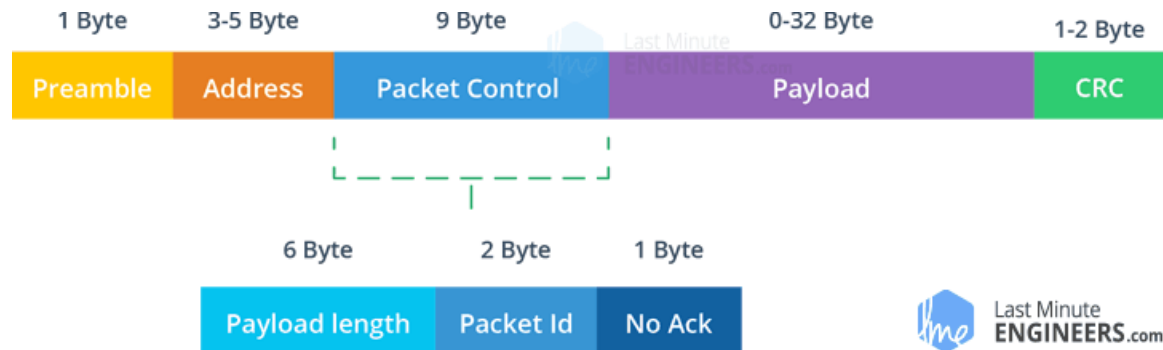
Gambar 5. nRF24L01+ Jaringan Multiceiver - Beberapa Pemancar Satu Penerima

Untuk menyederhanakan diagram di atas, bayangkan penerima utama bertindak sebagai penerima hub yang mengumpulkan informasi dari 6 node pemancar yang berbeda secara bersamaan. Penerima hub dapat berhenti mendengarkan kapan saja dan bertindak sebagai pemancar, tetapi ketika menjadi TX hanya bisa dilakukan satu pipa/simpul pada satu waktu.



# Protokol ShockBurst yang Ditingkatkan

Modul transceiver nRF24L01+ menggunakan struktur paket yang dikenal sebagai Enhanced ShockBurst. Struktur paket sederhana ini dipecah menjadi 5 field yang berbeda, yang diilustrasikan di bawah ini.



Gambar 6. nRF24L01+ Struktur Paket ShockBurst yang Ditingkatkan

Struktur ShockBurst asli hanya terdiri dari 5 Field yaitu Preamble, Address, Payload and the Cyclic Redundancy Check (CRC) fields. Enhanced ShockBurst menghadirkan fungsionalitas yang lebih besar untuk komunikasi yang lebih ditingkatkan menggunakan Packet Control Field (PCF) yang baru diperkenalkan .

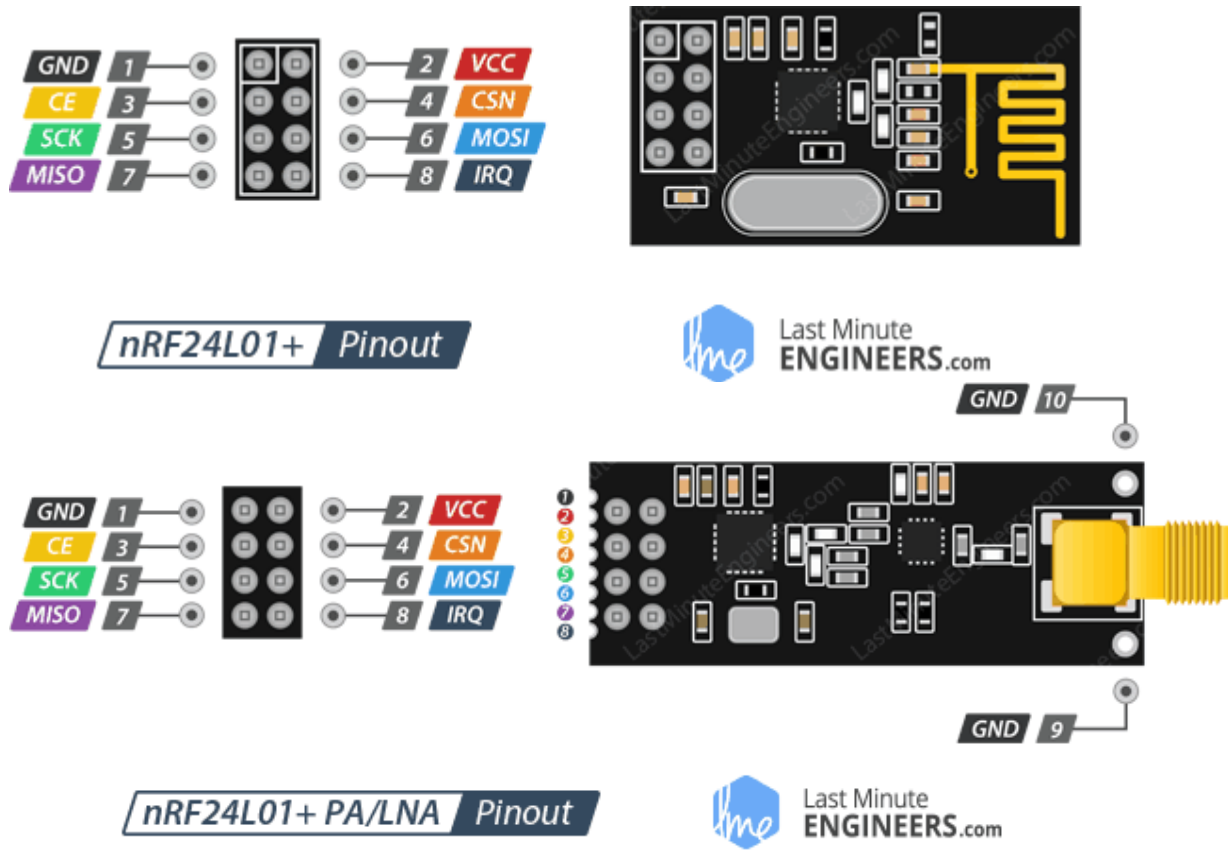
Struktur baru ini sangat bagus karena sejumlah alasan. Pertama, ini memungkinkan untuk muatan panjang variabel dengan penentu panjang muatan, yang berarti muatan dapat bervariasi dari 1 hingga 32 byte.

Kedua, ia menyediakan setiap paket yang dikirim dengan ID paket, yang memungkinkan perangkat penerima untuk menentukan apakah sebuah pesan baru atau telah dikirim ulang.

Terakhir, dan yang paling penting, setiap pesan dapat meminta pengakuan bahwa pesan yang dikirim telah diterima oleh perangkat lain.

# Pinout Modul Transceiver nRF24L01+

Mari kita lihat pinout dari kedua versi Modul transceiver nRF24L01+.



Gambar 7. Pinout nRF24L01

**CE (Chip Enable)** is an active-HIGH pin. When selected the nRF24L01 will either transmit or receive, depending upon which mode it is currently in.

**CSN (Chip Select Not)** is an active-LOW pin and is normally kept HIGH. When this pin goes low, the nRF24L01 begins listening on its SPI port for data and processes it accordingly.

**SCK (Serial Clock)** accepts clock pulses provided by the SPI bus Master.

**MOSI (Master Out Slave In)** is SPI input to the nRF24L01.

**MISO (Master In Slave Out)** is SPI output from the nRF24L01.

**IRQ** is an interrupt pin that can alert the master when new data is available to process.

## Menghubungkan modul transceiver nRF24L01+ ke Arduino UNO

Untuk memulainya, sambungkan pin VCC pada modul ke 3.3V pada Arduino dan pin GND ke ground. Pin CSN dan CE dapat dihubungkan ke pin digital apa pun di Arduino. Dalam kasus kami, masing-masing terhubung ke pin digital #8 dan #9. Sekarang tinggal pin yang digunakan untuk komunikasi SPI.

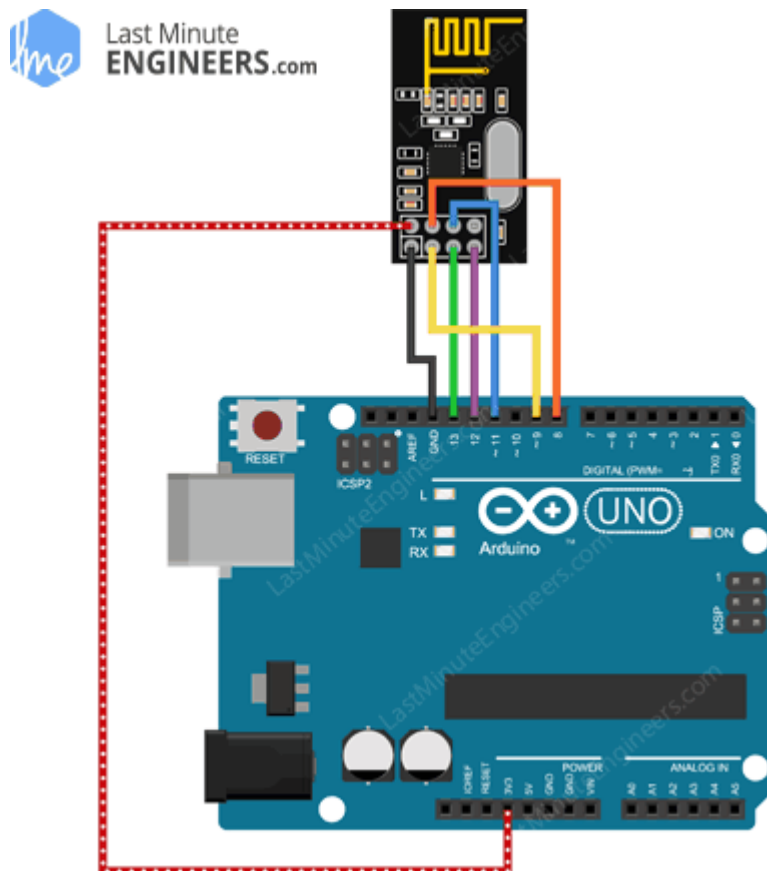
Karena modul transceiver nRF24L01+ memerlukan banyak transfer data, modul ini akan memberikan kinerja terbaik saat dihubungkan ke pin SPI perangkat keras pada mikrokontroler. Pin SPI perangkat keras jauh lebih cepat daripada menggunakan set pin lain.

Perhatikan bahwa setiap Papan Arduino memiliki pin SPI berbeda yang harus dihubungkan sesuai. Untuk papan Arduino seperti UNO/Nano V3.0 pin tersebut adalah digital 13 (SCK), 12 (MISO) dan 11 (MOSI).

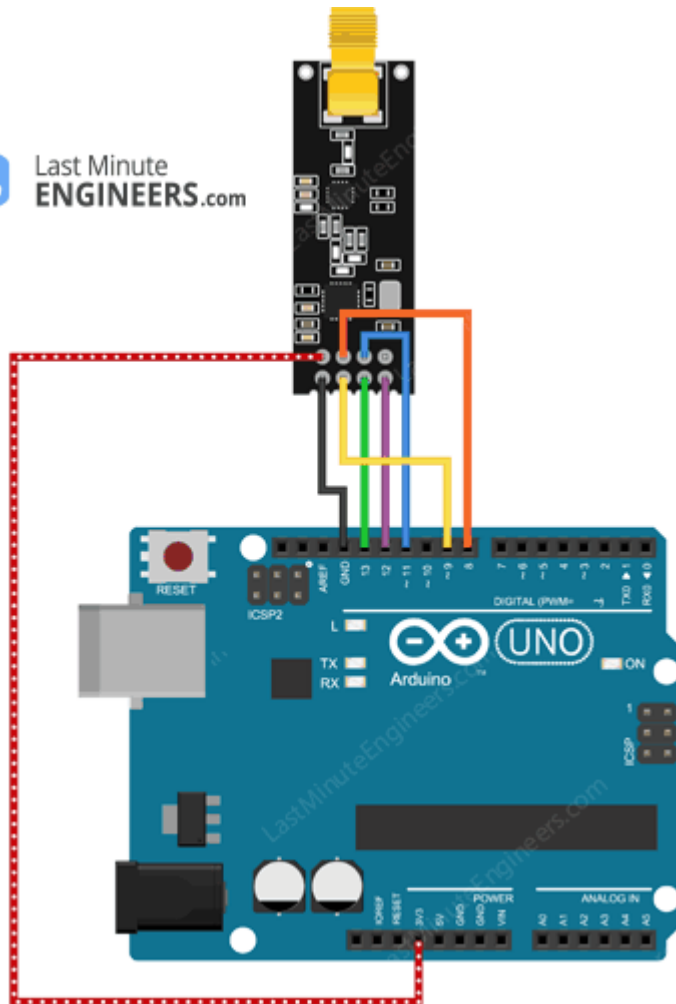
Jika Anda memiliki Mega, pinnya berbeda! Anda akan ingin menggunakan digital 50 (MISO), 51 (MOSI), 52 (SCK), dan 53 (SS). Lihat tabel di bawah ini untuk pemahaman cepat.

	MOSI	MISO	SCK
Arduino Uno	11	12	13
Arduino Nano	11	12	13
Arduino Mega	51	50	52

Gambar 8. Pin Arduino

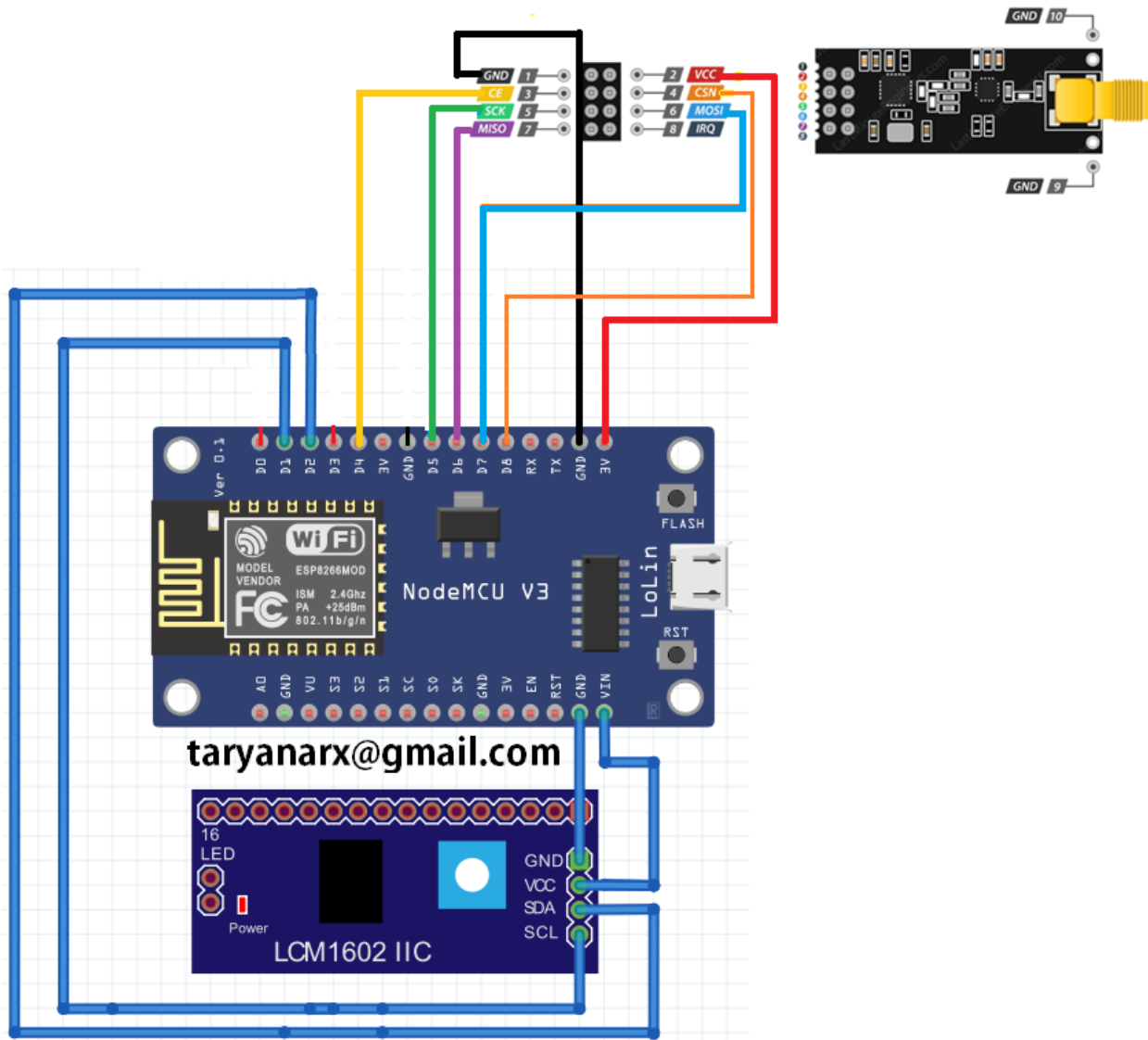


Gambar 9. Menghubungkan Modul Transceiver nRF24L01+ ke Arduino UNO

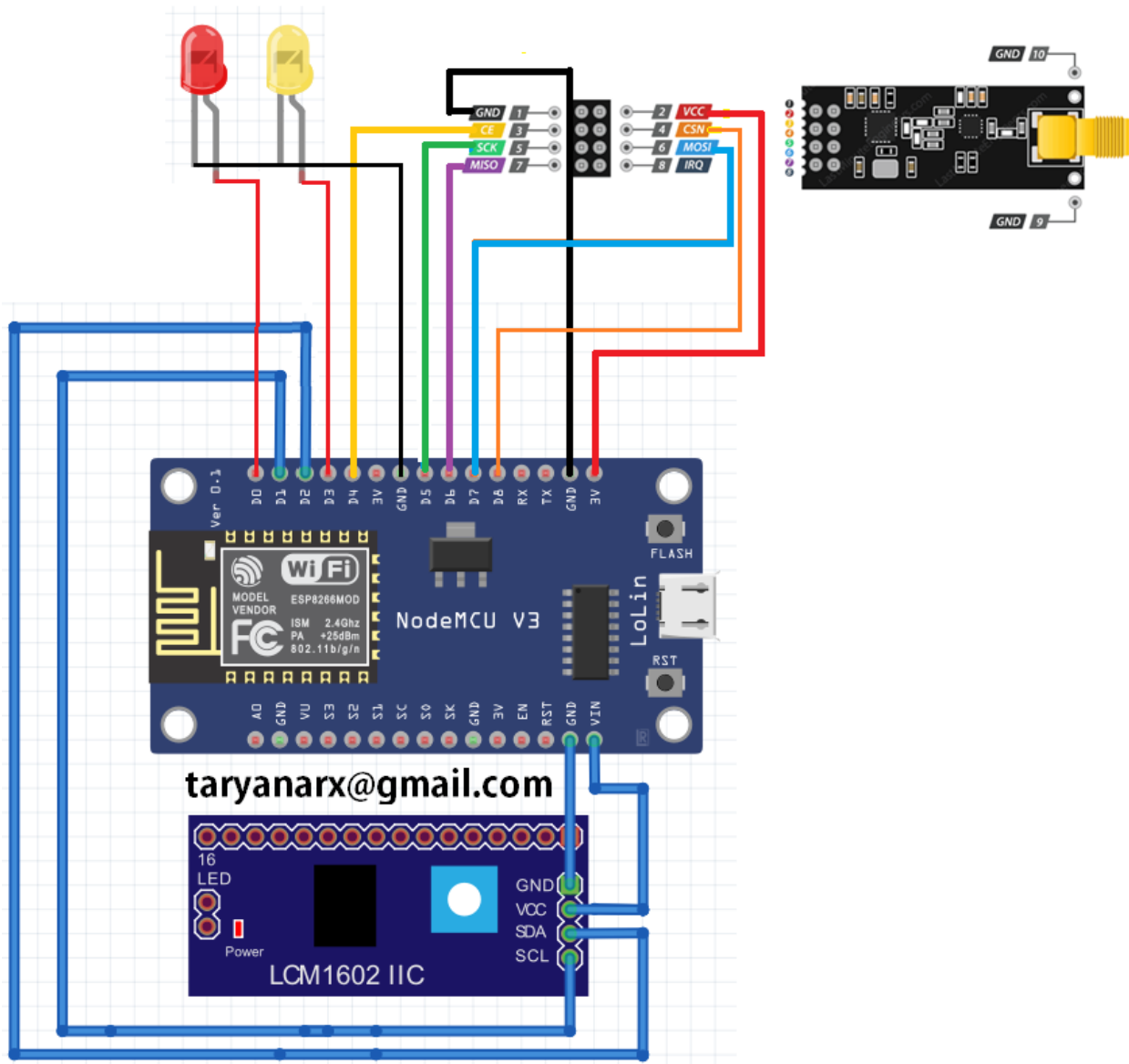


Gambar 10. Menghubungkan Modul nRF24L01+ PA LNA ke Arduino UNO

Ingat! Anda perlu membuat dua sirkuit ini. Yang satu bertindak sebagai pemancar dan yang lain sebagai penerima. Kabel untuk keduanya identik.



Gambar 11. Skema Pengirim (TX) Data nRF24L01+ PA LNA ke NodeMCU



Gambar 12. Skema Penerima (RX) nRF24L01+ PA LNA ke NodeMCU+LCD

Tabel 2. PIN Konektor NRF ke NODEMCU

NRF	NODEMCU
VCC	VV (5V)
GND	GND (G)
CE	D4 (GPIO2)
CSN	D2 (GPIO4) / D8
SCK	D5 (GPIO14)
MOSI (MO)	D7 (GPIO13)
MISO (MI)	D6 (GPIO12)
IRQ	NOT USED

## Library RF24 untuk Modul nRF24L01+

Salah satu library yang banyak digunakan adalah [RF24](#). Perpustakaan ini telah ada selama beberapa tahun. Ini mudah digunakan untuk pemula, tetapi menawarkan banyak hal untuk pengguna tingkat lanjut. Dalam percobaan kami, kami akan menggunakan perpustakaan yang sama.

Anda dapat mengunduh pustaka versi terbaru di [RF24 GitHub repository fork](#),

Untuk menginstalnya, buka Arduino IDE, buka Sketch > Include Library > Add .ZIP Library, lalu pilih file master RF24 yang baru saja Anda unduh



# Kode Program Untuk Pengirim (TX)

Dalam percobaan ini, akan dikirimkan 2 buah pesan, yaitu LED1ON dan LED2ON, jika pesan yang diterima adalah LED1ON maka nyalakan lampu LED1 dan matikan lampu LED2, begitupun sebaliknya, jika pesan yang diterima adalah LED2ON maka LED2 menyala dan LED1 dimatikan.

Berikut adalah sketch yang akan kita gunakan untuk pengiriman data:

```
/**
 * Pengirim Data TX
 * taryanarx@gmail.com
 */
//library
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

//create an RF24 object
RF24 radio(D4, D2); // CE, CSN D8

//address through which two modules communicate.
const byte address[6] = "00001";
void setup()
{
  radio.begin();
  Serial.begin(9600);
  //set the address
  radio.openWritingPipe(address);

  //Set module as transmitter
  radio.stopListening();
}

void loop()
{
  //Send message to receiver
  char text1[] = "LED1ON";
  radio.write(&text1, sizeof(text1));
  delay(5000);
  char text2[] = "LED2ON";
  radio.write(&text2, sizeof(text2));
}
```

Sketsa dimulai dengan memasukkan Library. Library SPI.h menangani komunikasi SPI sementara nRF24L01.h dan RF24.h mengontrol modul.

```
//Include Libraries
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
```

Selanjutnya, kita perlu membuat objek RF24. Objek mengambil dua nomor pin sebagai parameter yang menghubungkan sinyal CE dan CSN.

```
//create an RF24 object
RF24 radio(D4, D2); // CE, CSN
```

Selanjutnya kita perlu membuat array byte yang akan mewakili alamat pipa yang melalui dua modul nRF24L01+ untuk berkomunikasi.

```
//address through which two modules communicate.
const byte address[6] = "00001";
```

Kita dapat mengubah nilai alamat ini ke string 5 huruf apa pun seperti "node1". Alamat diperlukan jika Anda memiliki beberapa modul dalam jaringan. Berkat alamatnya, Anda dapat memilih modul tertentu yang ingin Anda komunikasikan, jadi dalam kasus kami, kami akan memiliki alamat yang sama untuk pemancar dan penerima.

Selanjutnya dalam fungsi setup: kita perlu menginisialisasi objek radio menggunakan `radio.begin()` dan menggunakan `radio.openWritingPipe()` fungsi kita mengatur alamat pemancar.

```
//set the address
radio.openWritingPipe(address);
```

Terakhir, kita akan menggunakan `radio.stopListening()` fungsi yang menetapkan modul sebagai pemancar.

```
//Set module as transmitter
radio.stopListening();
```

Kemudian pada bagian fungsi loop

Membuat Pesan atau karakter yang akan dikirim, misalnya adalah

Pesan = "LED1ON".

```
char text1[] = "LED1ON";
radio.write(&text1, sizeof(text1));
```

Melalui metode ini, Anda dapat mengirim hingga 32 byte sekaligus. Karena itu adalah ukuran maksimum dari satu paket yang dapat ditangani oleh nRF24L01+. Jika Anda memerlukan konfirmasi bahwa penerima menerima data, metode `radio.write()` mengembalikan `bool` nilai. Jika mengembalikan `TRUE`, data mencapai penerima. Jika mengembalikan `FALSE`, data telah hilang.

## Kode Program Untuk Penerima (RX)

Berikut ini adalah sketch yang digunakan untuk receiver atau penerima data

```
/**
 * *****
 * //Sketch Penerima RX
 * //taryanarx@gmail.com
 * *****
 * //Include Libraries
 * #include <SPI.h>
 * #include <nRF24L01.h>
 * #include <RF24.h>
 * #include <Wire.h>
 * #include <LiquidCrystal_I2C.h>
 * LiquidCrystal_I2C lcd(0x27, 20, 4);
 *
 * #define PIN_LED1 D3
 * #define PIN_LED2 D0
 * //create an RF24 object
 * RF24 radio(D4, D8); // CE, CSN
 *
 * //address through which two modules communicate.
 * const byte address[6] = "00001";
 *
 * void setLCD()
```

```

{
  lcd.init();
  lcd.clear();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("  SISTEM TELEMETRI");
  lcd.setCursor(0, 1);
  lcd.print("    NRF24-L01");
  lcd.setCursor(0, 2);
  lcd.print("  Data Recevice");
}

void setup()
{
  setLCD();
  pinMode(PIN_LED1, OUTPUT);
  pinMode(PIN_LED2, OUTPUT);
  while (!Serial);
  Serial.begin(9600);
  radio.begin();

  //set the address
  radio.openReadingPipe(0, address);

  //Set module as receiver
  radio.startListening();
}

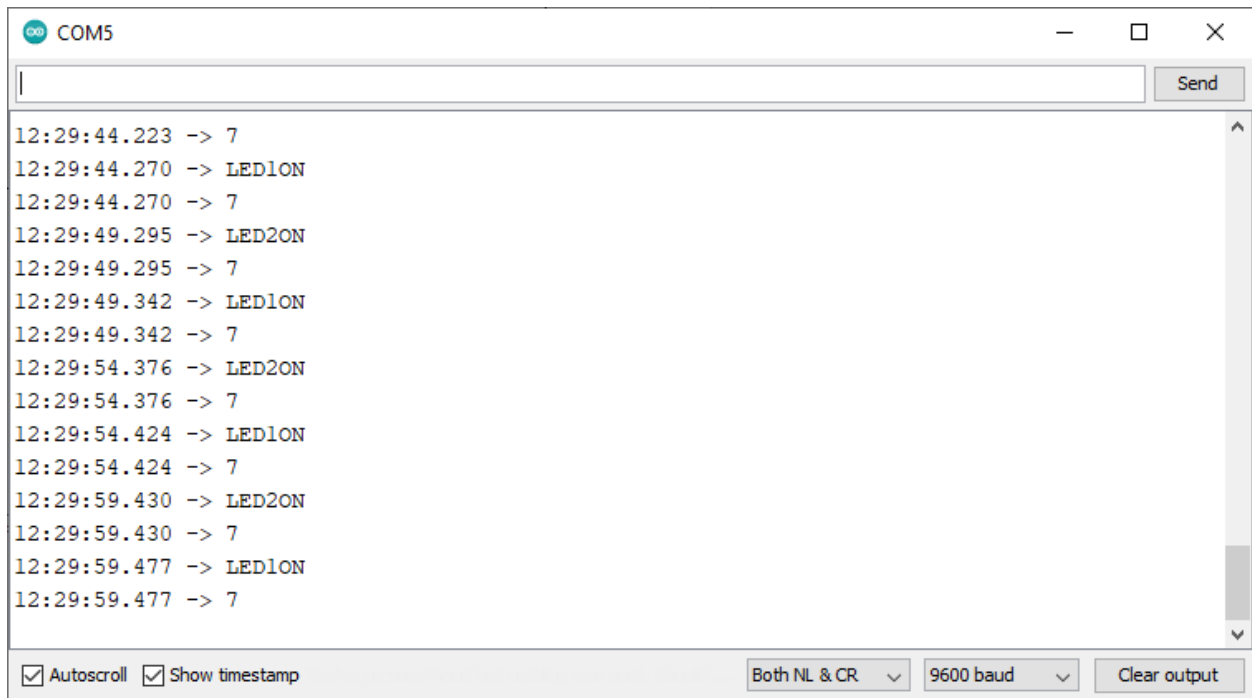
String cstring;
void loop()
{
  //Read the data if available in buffer
  cstring="";
  if (radio.available())
  {
    char text[32] = {0};
    radio.read(&text, sizeof(text));
    Serial.println(text);
    Serial.println(sizeof(text));
    lcd.setCursor(0, 3);
    lcd.print("                ");
    lcd.setCursor(0, 3);
    lcd.print(text);
    cstring = String (text);
    Serial.print("cstring:");
    Serial.println(cstring);
    Serial.print("panjang:");
    Serial.println(cstring.length());
    if(cstring=="LED1ON")
    {
      digitalWrite(PIN_LED1, HIGH);
    }
  }
}

```

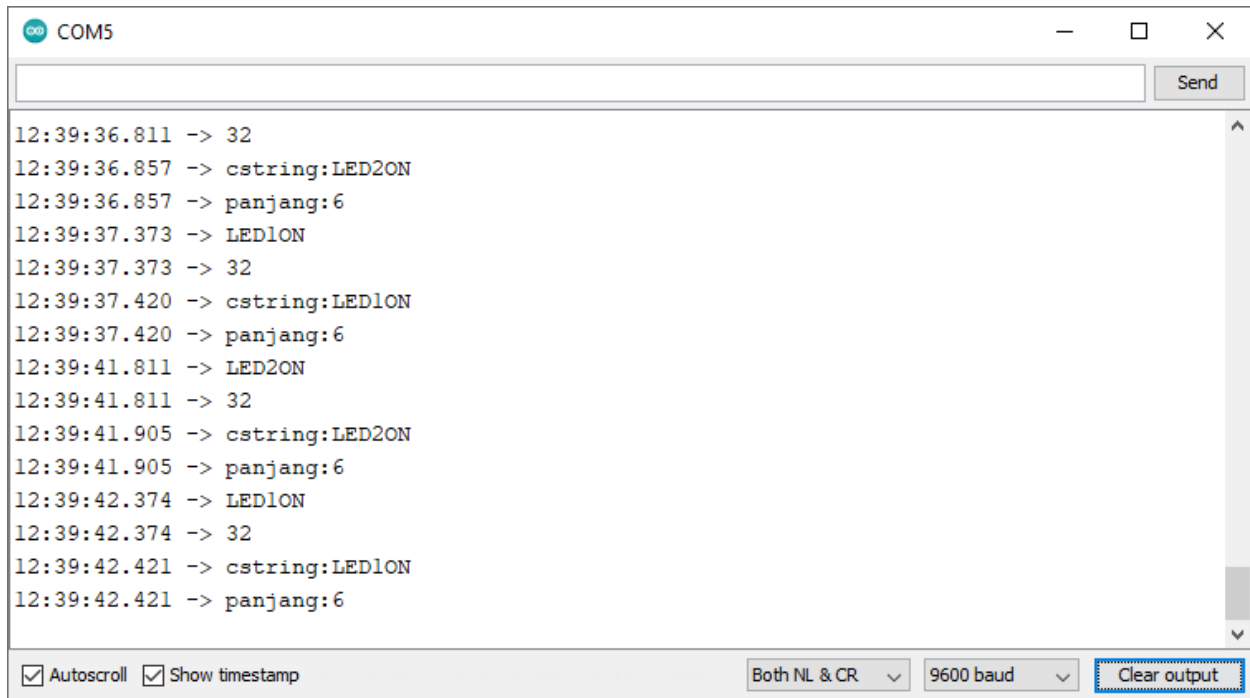
```
    digitalWrite(PIN_LED2, LOW);
}
else
if(cstring=="LED2ON")
{
    digitalWrite(PIN_LED2, HIGH);
    digitalWrite(PIN_LED1, LOW);
}

    delay(500);
}
}
```

Berikut adalah tampilan dari sisi Pengirim Data



Gambar 13. Serial Monitor untuk Pengirim Data



Gambar 14. Serial Monitor untuk Penerima Data



Gambar 15. Pengirim Data / Transmitter (TX)



Gambar 16. Penerima Data / Receiver (RX)





Gambar 17. Penerima Data / Receiver (RX)

# Kesimpulan

Setelah dilakukan pengujian dan pengiriman data, sepanjang 6 byte, pada jarak 100 Meter ditempat terbuka, data dapat diterima dengan sempurna, dan mengembalikan nilai true ke pengirim, untuk jangkauan lebih jauh misalnya untuk telemetri data yang dikirim dari tengah hutan, gunung, lautan atau dari udara maka dapat menambahkan Rf Amplifier dan untuk sumber daya bisa menggunakan solar cell (listrik tenaga surya).

# Daftar Pustaka

1. Suryana, Taryana (2021) [\*Impelementasi Kartu Tanda Penduduk Elektronik E-ktip Sebagai Single Identity Number, Dengan Rfid Reader Rc522 Pada Nodemcu Esp8266.\*](#) [Teaching Resource]
2. Suryana, Taryana (2021) [\*Implementasi Komunikasi Web Server Nodemcu Esp8266 Dan Web Server Apache Mysql Untuk Otomatisasi Dan Kontrol Peralatan Elektronik Jarak Jauh Via Internet.\*](#) [Teaching Resource]
3. Suryana, Taryana (2021) [\*Implementasi Kontrol Peralatan Elektronik Dengan Menggunakan Bot Telegram Dan Php Webhook.\*](#) [Teaching Resource]
4. Suryana, Taryana (2021) [\*Implementasi Raindrops Sensor Untuk Peringatan Terjadinya Hujan Dan Menutup Jemuran Otomatis.\*](#) [Teaching Resource]
5. Suryana, Taryana (2021) [\*Implementasi Sensor Photosensitive Pada Nodemcu Esp8266 Untuk Menyalakan Lampu.\*](#) [Teaching Resource]
6. Suryana, Taryana (2021) [\*Implementasi Web Server Nodemcu Esp8266 Untuk Kontrol Peralatan Elektronik Jarak Jauh Via Internet.\*](#) [Teaching Resource]
7. Suryana, Taryana (2021) [\*Implementation Ds18b20 1-wire Digital Temperature Sensor With Nodemcu Ideal Temperature For Brewing Coffee.\*](#) [Teaching Resource]
8. Suryana, Taryana (2021) [\*Membaca Masukan Dari Sensor Getar \(vibration Sensor\) Sw-420 Dengan Nodemcu Esp8266 Implementasi Sensor Untuk Peringatan Dini Apabila Terjadi Gempa Bumi.\*](#) [Teaching Resource]
9. Suryana, Taryana (2021) [\*Menghidupkan Lampu Dengan Menggunakan Sensor Ldr Pada Nodemcu Esp8266.\*](#) [Teaching Resource]
10. Suryana, Taryana (2021) [\*Mengirim Data Hasil Pengukuran Humidity Dan Temperature Sensor Dht11 Dengan Arduino Uno Wifi R3 Atmega328p Esp8266.\*](#) [Teaching Resource]
11. <https://lastminuteengineers.com/nrf24l01-arduino-wireless-communication/>
12. <https://www.nyebarilmu.com/cara-mengakses-module-wireless-nrf24l01-menggunakan-arduino/>
13. <https://www.esp8266.com/viewtopic.php?p=82379>
14. <https://acoptex.com/wp/basics-project-083c-esp8266-esp-12e-module-and-nrf24l01-2-4ghz-rf-transceiver-module-dht11-sensor-readings-on-lcd1602-i2c-module/>