Complete Guide for nRF24L01 – 2.4GHz RF Transceiver Module With Arduino

This post aims to be a complete guide for nRF24L01 – 2.4GHz RF Transceiver modul explain what it does, show its specs and share an Arduino project example that you take and apply to your own projects.

I have more complete guides for other popular sensors, check them below:

» Complete Guide for Ultrasonic Sensor HC-SR04

» Complete Guide for DHT11/DHT22 Humidity and Temperature Sensor

» Complete Guide for RF 433MHz Transmitter/Receiver Module

» Arduino with PIR Motion Sensor

Description

These RF modules are very popular among the Arduino tinkerers. The nRF24L01 is on a wide variety of applications that require wireless control. They are transceivers which this means that each module can transmit and receive data.

These modules are very cheap and you can use them with any microcontroller (MC

Specifications nRF24L01 – 2.4GHz RF Transceiver

» Low cost single-chip 2.4GHz GFSK RF transceiver IC

» Range with Antenna: 250Kb rate (Open area) >1000 meter

» Power: Ultra low power consumption

» Input Voltage: 3.3V

» Pins: 5V tolerant

» Price: $2

Where to buy?

You can purchase these modules from eBay for just a few dollars. Click here to see nRF24L01 module on eBay. They come in two versions with external antenna (more range) or built-in antenna (less range).

Arduino with nRF24L01

You need the following components to make this example:

» 2x Arduino (eBay)

» 2x nRF24L01 – 2.4GHz RF Transceiver (eBay)

» Breadboard (eBay)

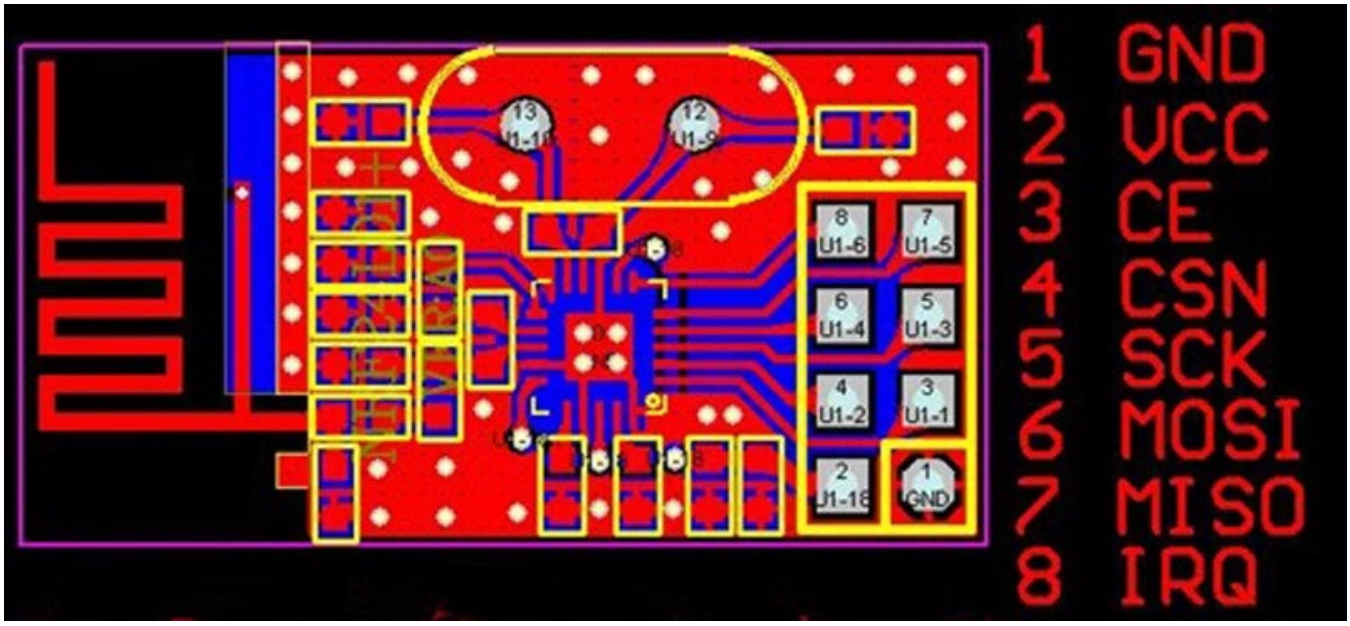
Library download

Here's the library you need for this project:

1. Download the **RadioHead library**
2. Unzip the RadioHead library
3. Install the RadioHead library in your Arduino IDE
4. Restart your Arduino IDE

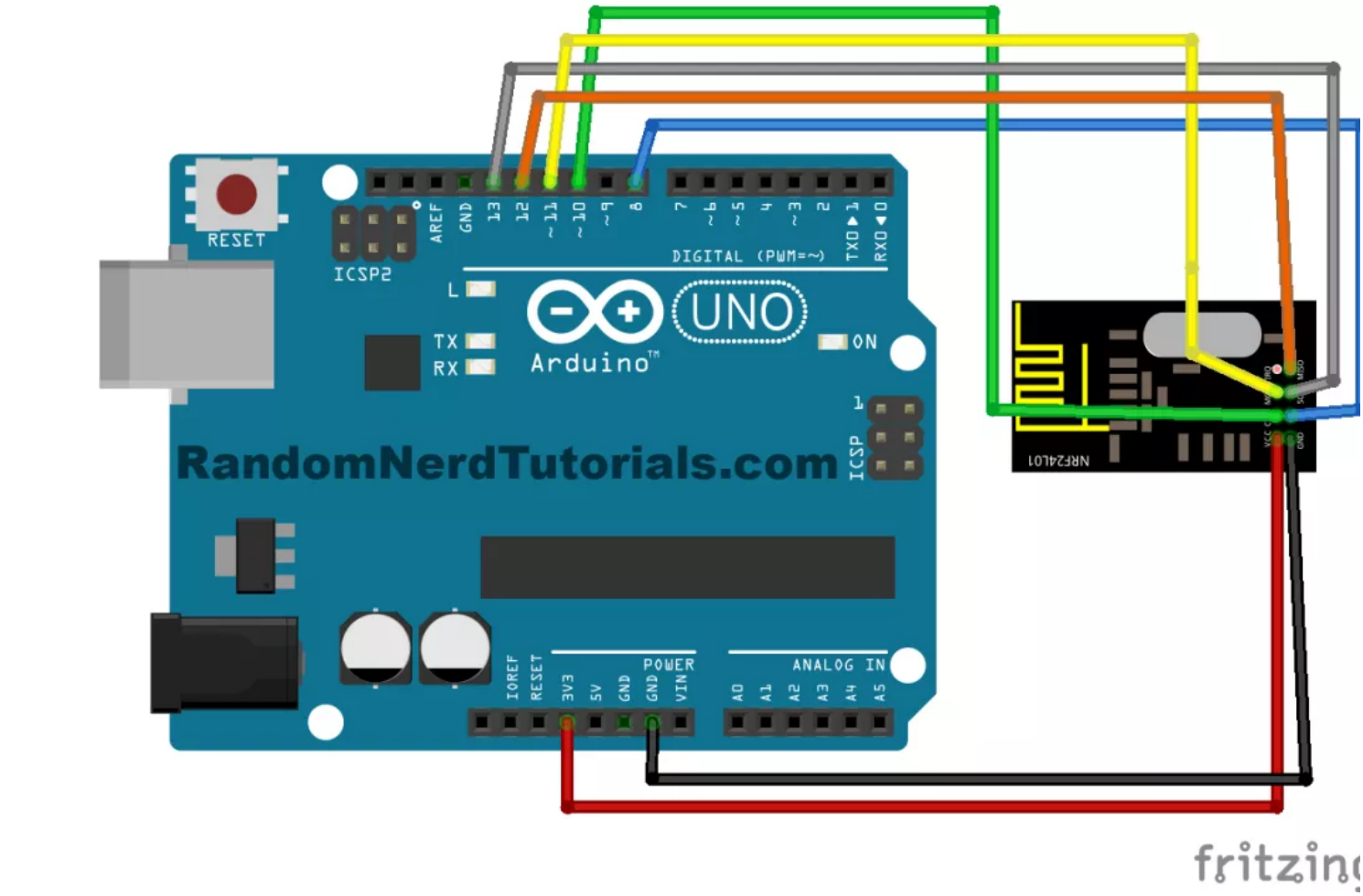
The RadioHead library is great and it works with almost all RF modules in the marke
can read more about this project here.

Pinout



Top view of NRF24L01

Client Circuit



Important: Input voltage is of 1.9V~3.6V, do not exceed this voltage, otherwise it wil your module.

Follow the circuit above for your client. Then upload the code below which can be f in your Arduino IDE (after installing the RadioHead library).

Go to File > Examples > RadioHead > nrf24 > nrf24_client.

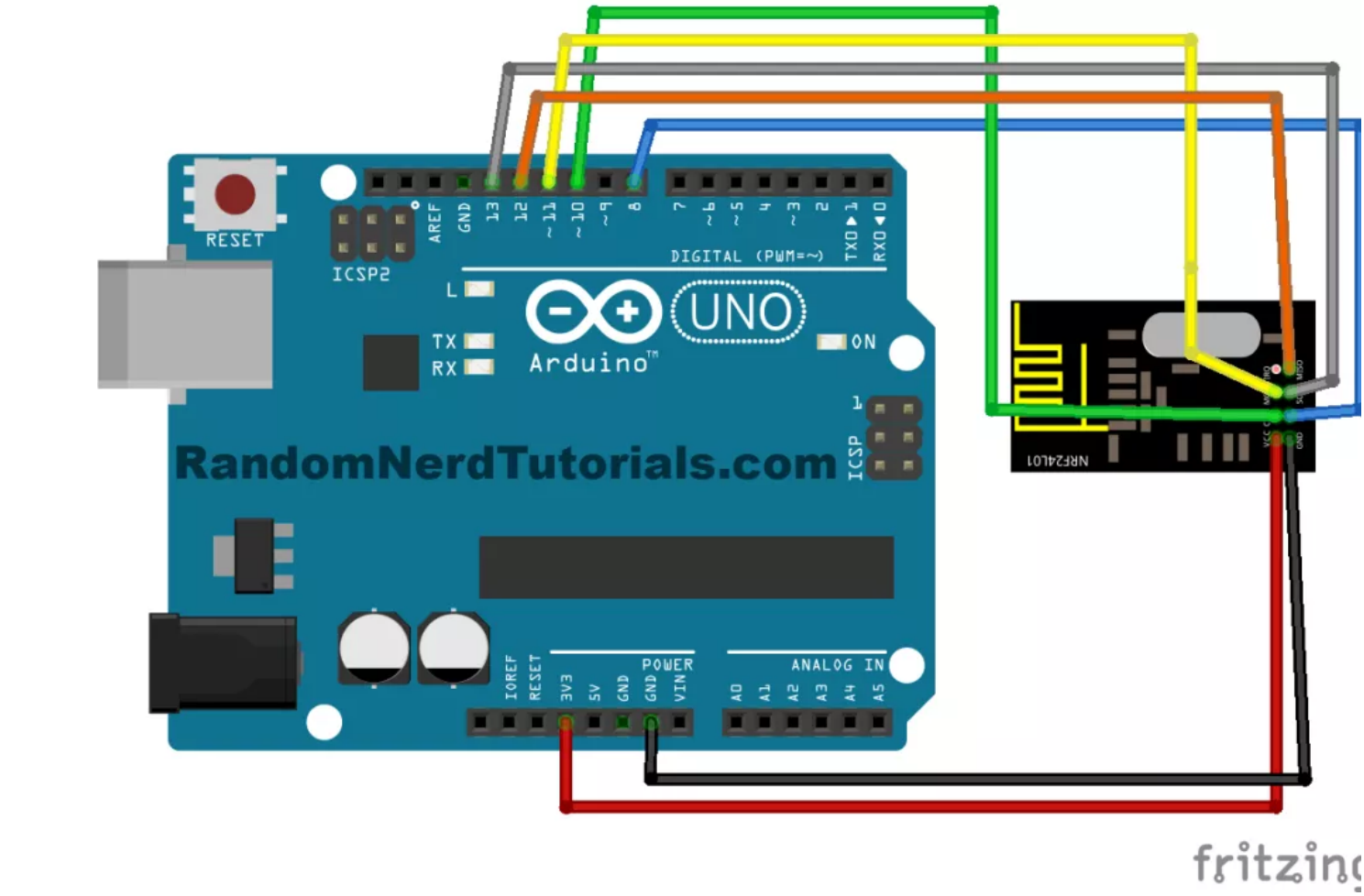
```
// nrf24_client

#include <SPI.h>
#include <RH_NRF24.h>

// Singleton instance of the radio driver
RH_NRF24 nrf24;
```

```
// RH_NRF24 nrf24(8, 7); // use this to be electrically compatible with Mirf
// RH_NRF24 nrf24(8, 10);// For Leonardo, need explicit SS pin
// RH_NRF24 nrf24(8, 7); // For RFM73 on Anarduino Mini

void setup()
{
  Serial.begin(9600);
  while (!Serial)
    ; // wait for serial port to connect. Needed for Leonardo only
  if (!nrf24.init())
    Serial.println("init failed");
  // Defaults after init are 2.402 GHz (channel 2), 2Mbps, 0dBm
  if (!nrf24.setChannel(1))
    Serial.println("setChannel failed");
  if (!nrf24.setRF(RH_NRF24::DataRate2Mbps, RH_NRF24::TransmitPower0dBm))
    Serial.println("setRF failed");
}


void loop()
{
  Serial.println("Sending to nrf24_server");
  // Send a message to nrf24_server
  uint8_t data[] = "Hello World!";
  nrf24.send(data, sizeof(data));

  nrf24.waitPacketSent();
  // Now wait for a reply
  uint8_t buf[RH_NRF24_MAX_MESSAGE_LEN];
  uint8_t len = sizeof(buf);
```

```
  if (nrf24.waitAvailableTimeout(500))
  {
    // Should be a reply message for us now
    if (nrf24.recv(buf, &len))
    {
      Serial.print("got reply: ");
      Serial.println((char*)buf);
    }
    else
    {
      Serial.println("recv failed");
    }
  }
  else
  {
    Serial.println("No reply, is nrf24_server running?");
  }
  delay(400);
}
```

Projects/nrf24_client.ino vie

Server Circuit



Important: Input voltage is of 1.9V~3.6V, do not exceed this voltage, otherwise it wil your module.

Follow the circuit above for your server. Then upload the code below which can be in your Arduino IDE (after installing the RadioHead library).

Go to File > Examples > RadioHead > nrf24 > nrf24_server.

```
// nrf24_server

#include <SPI.h>
#include <RH_NRF24.h>

// Singleton instance of the radio driver
RH_NRF24 nrf24;
```

```
// RH_NRF24 nrf24(8, 7); // use this to be electrically compatible with Mirf
// RH_NRF24 nrf24(8, 10);// For Leonardo, need explicit SS pin
// RH_NRF24 nrf24(8, 7); // For RFM73 on Anarduino Mini

void setup()
{
  Serial.begin(9600);
  while (!Serial)
    ; // wait for serial port to connect. Needed for Leonardo only
  if (!nrf24.init())
    Serial.println("init failed");
  // Defaults after init are 2.402 GHz (channel 2), 2Mbps, 0dBm
  if (!nrf24.setChannel(1))
    Serial.println("setChannel failed");
  if (!nrf24.setRF(RH_NRF24::DataRate2Mbps, RH_NRF24::TransmitPower0dBm))
    Serial.println("setRF failed");
}

void loop()
{
  if (nrf24.available())
  {
    // Should be a message for us now
    uint8_t buf[RH_NRF24_MAX_MESSAGE_LEN];
    uint8_t len = sizeof(buf);
    if (nrf24.recv(buf, &len))
    {
//      NRF24::printBuffer("request: ", buf, len);
      Serial.print("got request: ");
      Serial.println((char*)buf);
```

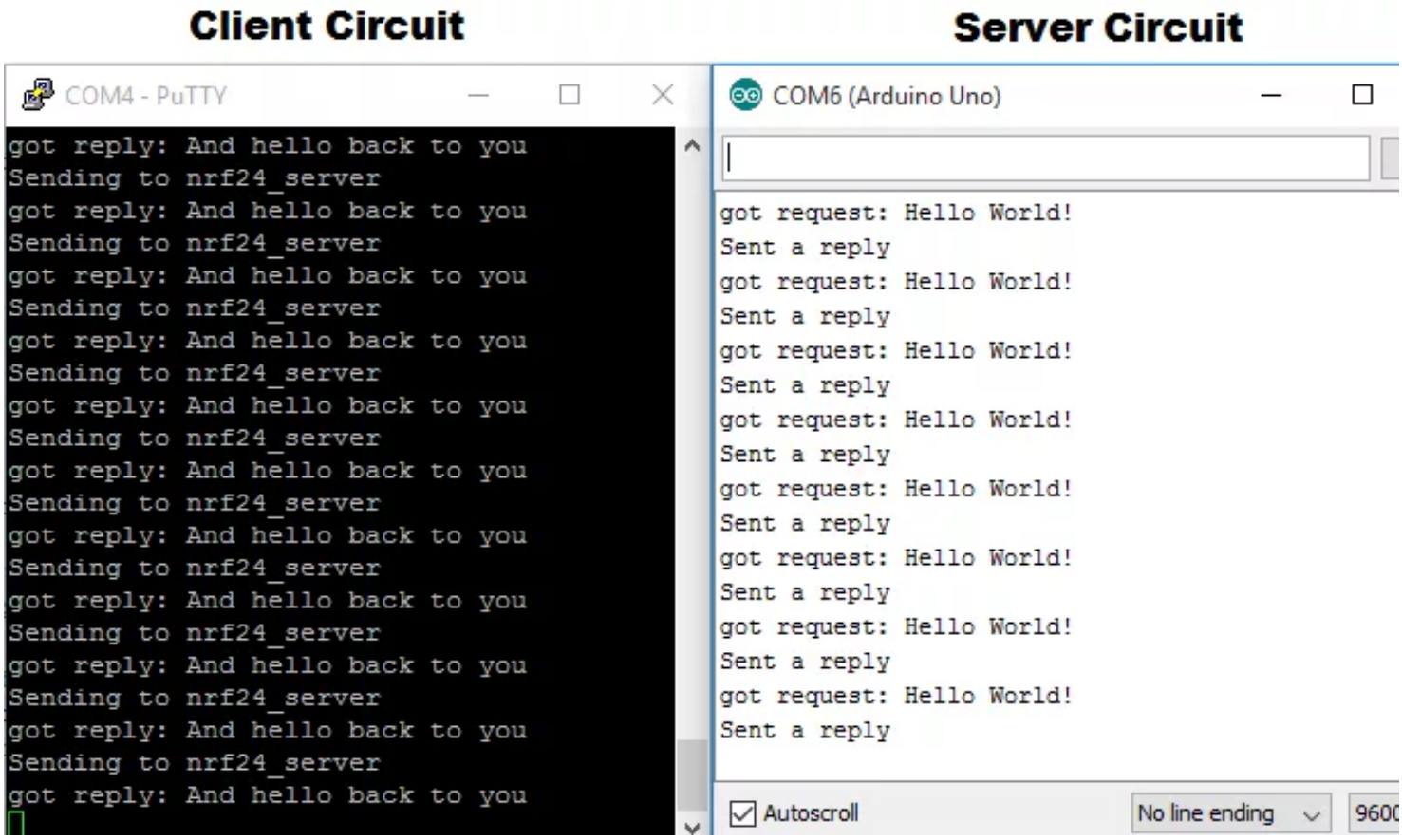
```
      // Send a reply
      uint8_t data[] = "And hello back to you";
      nrf24.send(data, sizeof(data));
      nrf24.waitPacketSent();
      Serial.println("Sent a reply");
    }
    else
    {
      Serial.println("recv failed");
    }
  }
}
```

Projects/nrf24_server.ino vie

Demonstration

RELATED CONTENT: Like ESP8266? Check out Home Automation Using ESP82

In this project the client is sending a message “Hello World!” to the server via RF an server is sending back the following message “And hello back to you”. Those messages are being displayed in the serial monitor. Here’s what you should see in your serial terminal windows (see Figure below).



Note: On the left window, I'm establishing a serial communication with PuTTY.org. C right window, I'm using the Arduino IDE Serial Monitor.

Conclusion

You need to have some realistic expectations when using this module. They work v well when the receiver and transmitter are quite close to each other. If you separat them too far you'll loose the communication.

The communication range will vary. It depends on how much noise in your environ if there's any obstacles and if you're using an external antenna.

I hope you found this guide useful.

Share this post with a friend that also likes electronics!

You can contact me by leaving a comment. If you like this post probably you might my next ones, so please support me by subscribing my blog and my Facebook Page

Thanks for reading,

-Rui Santos

What to Read Next...

21 Apr, 2016

19 Awesome Robots You Can Build With an Ardu

23 Mar, 2016

Security Access using MFRC522 RFID Reader with Arduino

12 Feb, 2016

ESP8266 Web Server with Arduino IDE

29 Sep, 2015

Complete Guide for Nokia 5110 LCD with Arduin

01 Jul, 2015

Complete Guide for RF 433MHz Transmitter/Rec
Module With Arduino

Leave a Comment:

Name *

E-Mail *

Website

Comment

Post Comment

Notify me of follow-up comments by email.

Notify me of new posts by email.

Biplab Roy says
September 2, 2015

Hello,

I've been following your posts from quite a few days. The nRF tutorial is really helpful. I have a query. Can we use this library for wireless sensor nodes. How to communicate with different clients with different device addresses?
Suppose we have 2 clients & 1 server. Then how to ask for data from a specific client at a time? Please help.

Reply

Rui Santos says
September 13, 2015

Hi Roy,
Thank you for trying my projects! Yes you can have multiple noc
and you can set different device addresses to allow multiple clie
The library comes with an example that will help you use that concept.
Having the library installed in your Arduino IDE. Go to File > Examples > Radiohead > nrf24
And use these examples "nrf24_reliable_datagram_client" and "nrf24_reliable_datagram_server".
If you look at the code you can set different addresses to each device

➔ Reply

Emil Mashiah says
September 3, 2015

Great job, as always! I hadn't tried it yet, but I definitely will. I hope I'll succeed.

Thank you Rui

➔ Reply

Rui Santos says
September 13, 2015

Thank you Emil for your continuous support! Have a great day,
-Rui

➔ Reply

arun says
September 7, 2015

when i insert code of nrf24_server into ardiuno uno, at serial monitor it shows init failed .

nrf24_client working fine, sending data but nothing is happening from server site , it shows only INIT FAILED.
I created server on both uno and mega 2560 but getting same response init failed,
Please help me in these case, its urgent.
Thanks

Reply

Rui Santos says
September 13, 2015

Have you followed my exact schematics?
Are you using the proper nrf example for the Arduino IDE?
Check carefully the COM ports while uploading the Arduino cod

Reply

luke says
February 9, 2016

see this page:
http://www.airspayce.com/mikem/arduino/RadioHead/classRH_RF24.html

you must swap 10>53, 13>52, 11>51, 12>50 on the arduino meg and instantiate (line 7) using the following instead:

RH_NRF24 nrf24(8, 53); //mega

having said that, I can get the mega to act as client but not as server.

Reply

Eduardo says
September 20, 2015

Hello, this module is popular, common, is that transmits audio, which is nRF24L01z has little information on his internet, you want to do some tutorial?

Reply

Rui Santos says
September 24, 2015

Hi,
I don't know any tutorial on that subject... Sorry
Thanks for asking!

Reply

Add Your Reply

Popular posts

ESP8266 Web Server Tutorial

ESP8266 ESP8266 Project Web Server Feb 18, 2015

Arduino – Control 2 DC Motors Via Bluetooth (Perfect To Build a Robot)

Android Arduino Arduino Project Tutorials Sep 25, 2013

Arduino –

Arduino Tutorials A