



## Getting Started With NRF24L01 Wireless Transceiver Module



by Smart Technology

In this project, we are going to learn how to use NRF24L01 Transceiver Module in order to make a wireless communication between two Arduino boards where using the PushButton at the first Arduino we will control the LED at the second Arduino.

The purpose is to discover the perfect way to communicate wirelessly between Arduino's by using the NRF24L01 Wireless Transceiver Module. It allows us to remote things wirelessly.

Hardware components :

1. Arduino Uno × 2

2. NRF24L01 Transceiver Module × 2

3. Base module with voltage regulator × 2

4. PushButton

5. LED

6. Breadboard

7. Wires

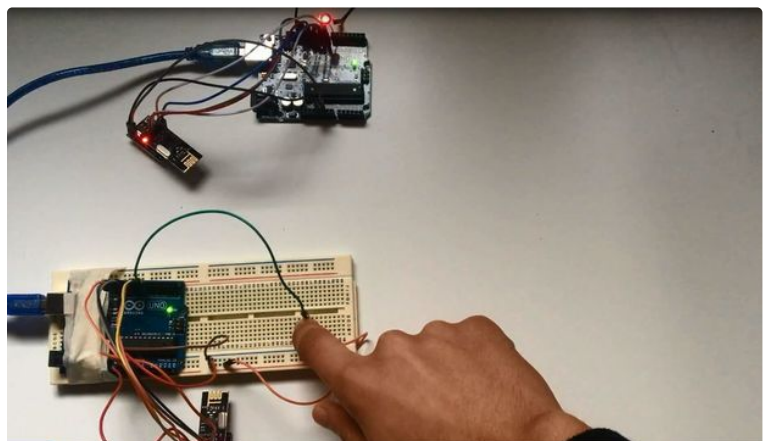
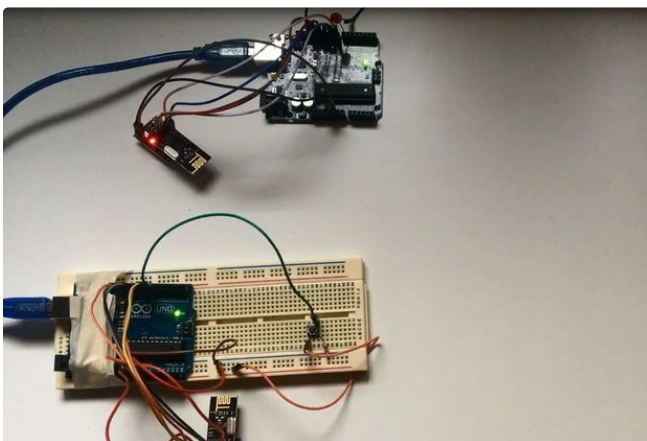
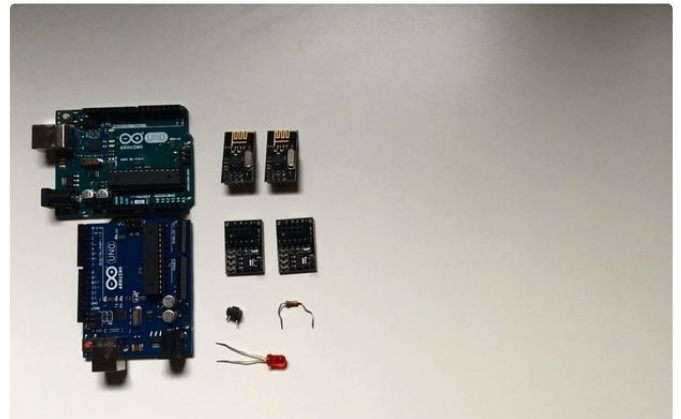
Software apps :

1. Arduino IDE

### WIRELESS REMOTE USING 2.4 GHZ NRF24L01



### SIMPLE TUTORIAL USING NRF24L01 & ARDUINO



---

## Step 1: NRF24L01 Features and Specifications

### Features of the NRF24L01 Wireless Transceiver Module :

- Cost (between 1\$ to 3\$ regarding to the type version)
- Range
- Availability
- Easy to use

### Types of the NRF24L01 Wireless Transceiver Module :

There is two types of NRF24LO1 :

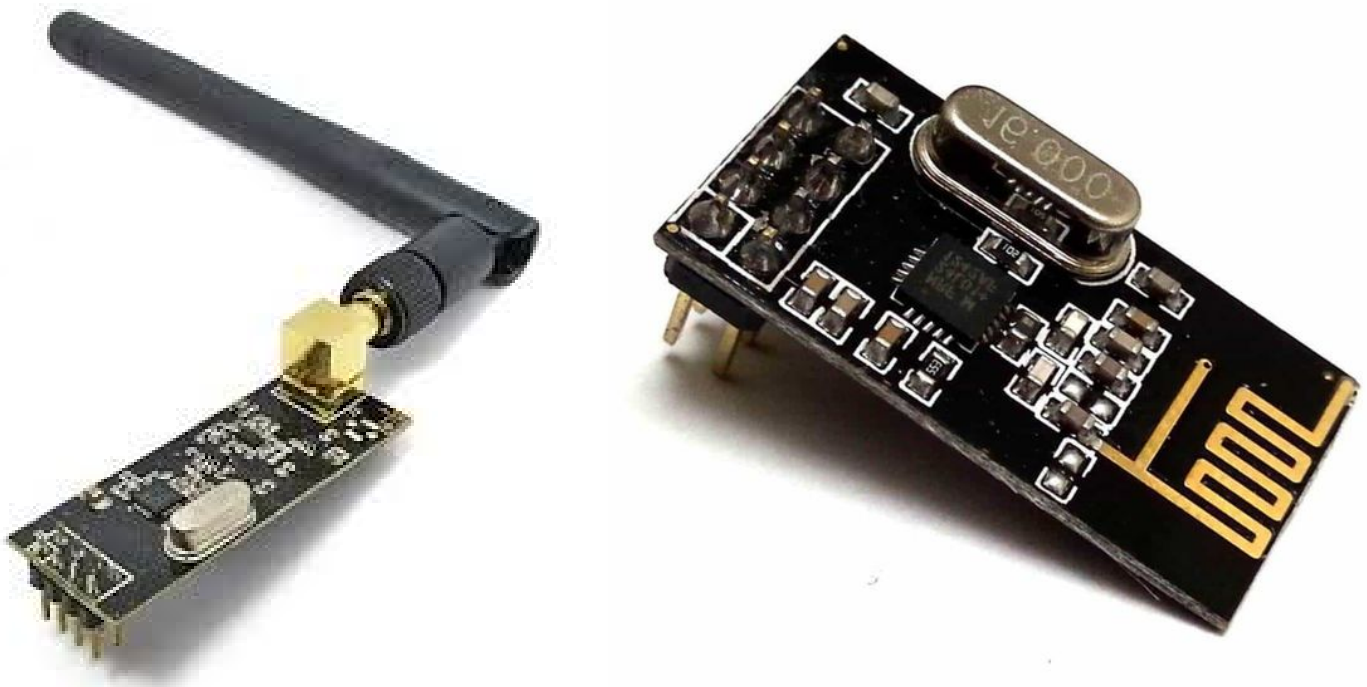
- NRF24L01 with antenna (About a Kilometer in openspace)
- NRF24L01 without antenna (About 50 meters in openspace)

### Applications of the NRF24L01 Wireless Transceiver Module :

- Wireless home automation
- Drones remote control
- Remote sensors for temperature, pressure, alarms, much more
- Robot control and monitoring

### The power consumption of the NRF24L01 Wireless Transceiver Module :

- The power consumption of this module is just around 12mA during transmission
- The operating voltage of the module is from 1.9 to 3.6V



## Step 2: Watch the Video for More Details

[https://youtu.be/W8a1T\\_Ornos](https://youtu.be/W8a1T_Ornos)

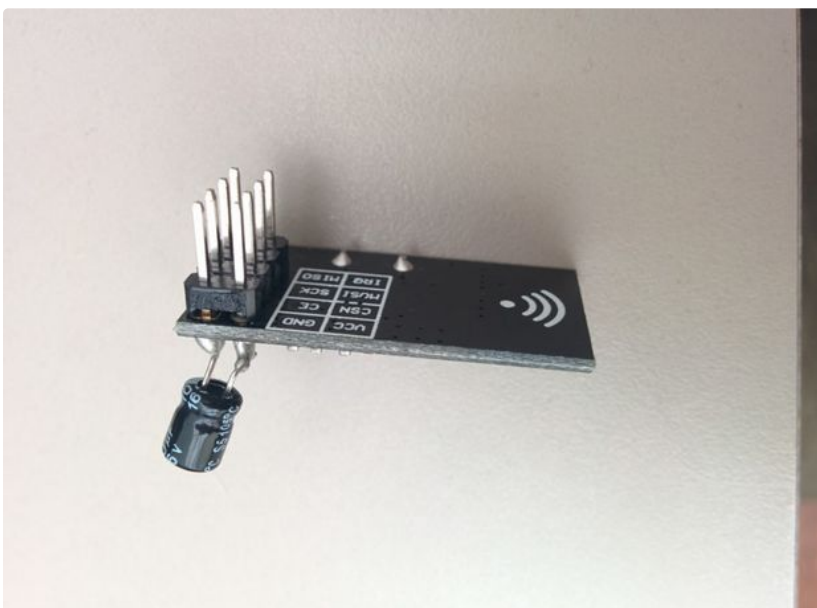
## Step 3: Problem to Be Overcome

Many users confront issues with NRF24L01 module. the source of these issues often come from the 3.3v Power. due to the module does not have enough current capability. this is why I recommend to either use:

- Base module with voltage regulator and bypass capacitors for stability

OR

- 3.3 uF to 10 uF (MicroFarad) capacitor directly on the module from +3.3V to Gnd



## Step 4: Schematic

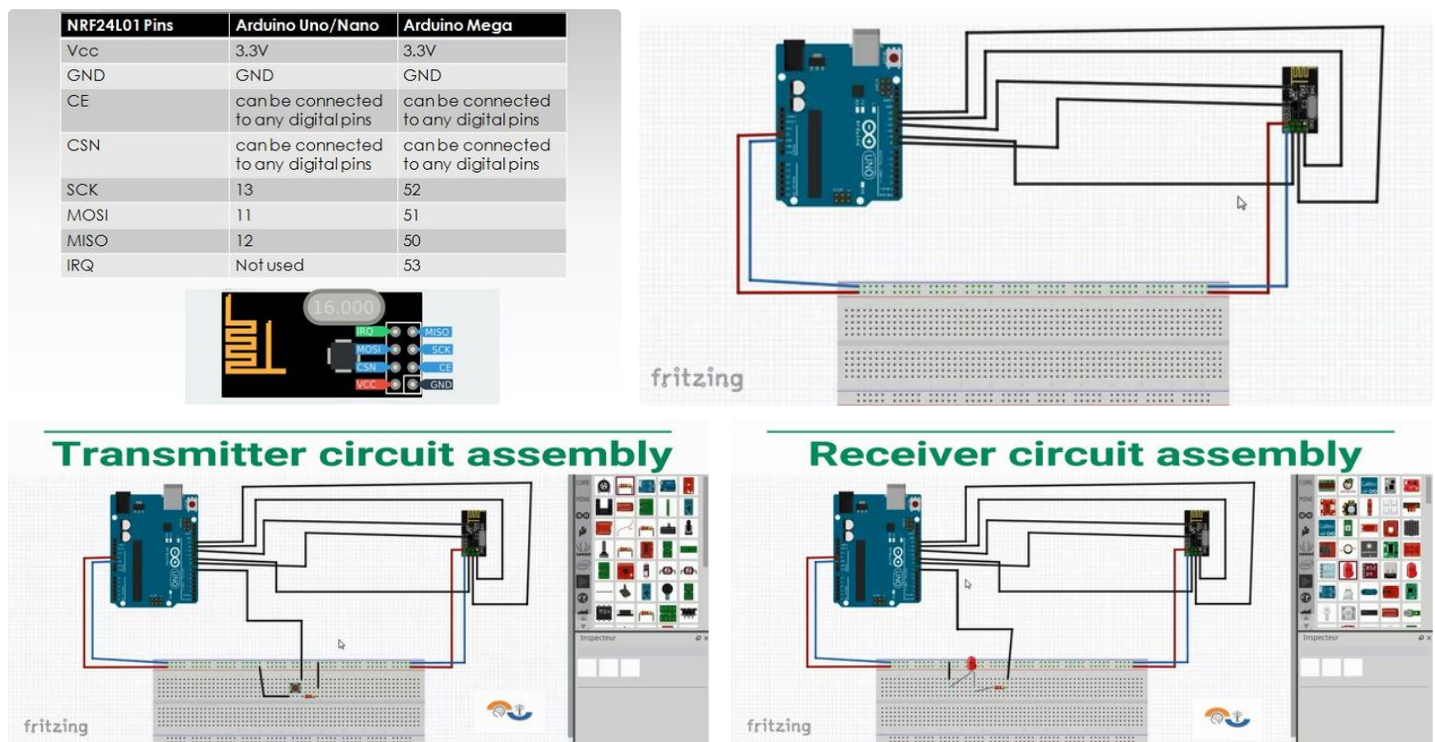
CE and CSN pins can be connected to any digital pins. Then in RF24 library, you can specify which pins you used. I chose pins 8 and 9 because I will use them in the examples.

In our case we will use the Arduino Uno this is why you should follow this instructions (or follow the picture above) :

- MOSI is connected to the digital pin 11
- MISO is connected to the digital pin 12
- SCK is connected to the digital pin 13
- SS (not used)

Note: While using the NRF24L01 you have to remember that these digital pins won't be available.

Kindly watch the video which explained all steps clearly.



## Step 5: Download RF24 Library

In this project we used RF24 library, which can be downloaded on Github: [RF24 library](#)

1. First you need to click on “Download ZIP” button
2. Extract the zip file to your Arduino home directory: Arduino/libraries on Linux or Documents/ Arduino/libraries in Windows.

---

## Step 6: Transmitter Arduino Code

Transmitter sketch will look like:

```
/** Arduino Wireless Communication Tutorial
 *   Transmitter Code
 *
 * by Smart Technology, <a href="https://makesmarttech.blogspot.com/" rel="nofollow"> https://makesmarttech.blogspot.com/
</a>
 *
 * Library: TMRh20/RF24,<a href="https://tmrh20.github.io/RF24" rel="nofollow">https://tmrh20.github.io/RF24</a>
 */
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
#define button 7
RF24 radio(8, 9); // CE, CSN
const byte address[6] = "00001";
boolean buttonState = 0;
void setup() {
  pinMode(button, INPUT);
  radio.begin();
  radio.openWritingPipe(address);
  radio.setPALevel(RF24_PA_MAX);
  radio.stopListening();
}
void loop() {
  delay(5);
  radio.stopListening();
  buttonState = digitalRead(button);
  radio.write(&buttonState, sizeof(buttonState));
}<br>
```



<http://www.instructable...>

Download



## Step 7: Receiver Arduino Code

Receiver sketch code will look like :

```
/** Arduino Wireless Communication Tutorial
 *   Receiver Code
 *
 * by Smart Technology, https://makesmarttech.blogspot.com/
 *
 * Library: TMRh20/RF24, https://tmrh20.github.io/RF24
 */
#include <SPI.h>

#include <nrf24l01.h>
#include <RF24.h>
#define led 7
RF24 radio(8, 9); // CE, CSN
const byte address[6] = "00001";
boolean buttonState = 0;

void setup() {
  pinMode(7, OUTPUT);
  Serial.begin(9600);
  radio.begin();
  radio.openReadingPipe(0, address);
  radio.setPALevel(RF24_PA_MAX);
  radio.startListening();
}

void loop() {
  delay(5);
  radio.startListening();
  while (!radio.available());
  radio.read(&buttonState, sizeof(buttonState));
  if (buttonState == HIGH) {
    digitalWrite(led, HIGH);
  }
  else {
    digitalWrite(led, LOW);
  }
}
```

<http://www.instructable...>

Download

## Step 8: For Support

You can subscribe to the my YouTube channel for more tutorials and projects.

Subscribe for support. Thank you. Go to my YouTube Channel -link <https://goo.gl/EtQ2mp>

