

# RADIO MODULE FOR NRF24L01

## I . MODULE FEATURES:

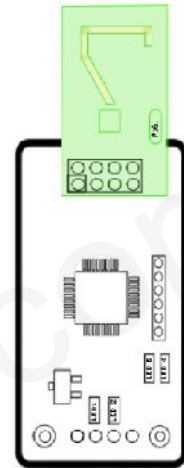
I2C (TWI) communication protocol is used, which simplifies the NRF24L01 communication process greatly, thus users no need to understand the tedious communication process of NRF24L01, it improves the efficiency of project development Immensely by this way.

- 2) Each module will be as a slave of I2C (TWI). (address 35)
- 3) Module is in a small frame with length 73mm, width 22mm.
- 4) Smooth bidirectional communication and no discernable the slave or master.
- 5) Maximum transmissive distance can reach 70m(Just for reference)
- 6) Wireless corresponding chip is reliable NRF24L01 communication protocol, it will reduce data lost.
- 7) I2C/TWI protocol simplifies the communication lines greatly.
- 8) Compatible with arduino perfectly.

## II . APPLICATIONS:

Robot control, remote control., Information collection and etc.

Note: The wireless communications may result in radiation, so it is not recommended for life protection area.



### III. RELEVANT PARAMETERS:

**Operating voltage:** 5V.(No short-circuit protection, so you must be careful with polarity connection when you use.)

**Weight:** 6g

**Hold time of module receiving data :** more than 2S

**Hold time of module transmitting data:** more than 300s

**Transfer rate:** 50Byte / S (theoretical value)



**I2C/TWI Address:** 0x47 (decimal 35)

**NRF24L01 Address:** 0x34, 0x43, 0x10, 0x10, 0x01 (from low to high)

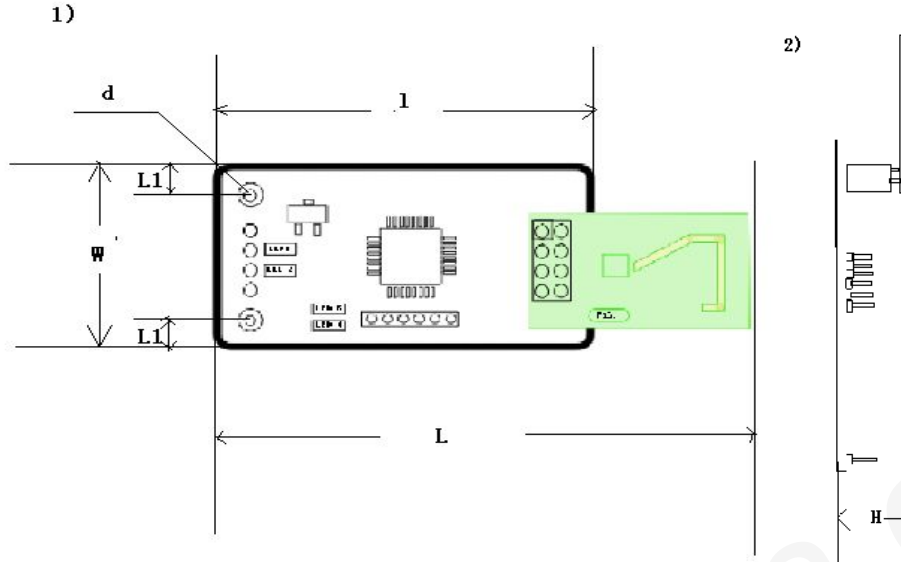
#### Related terms:

Hold time of module transmitting data means when the module (TWI slave) receives data sent from master, it will transmit the date to another module, and then it will wait to receive another responded signal of that module, if there is no responded signal after a while, the module will send the data again, repeated sending times multiply waiting time is the hold time.

### IV. TESTED COMPATIBLE DEVICES

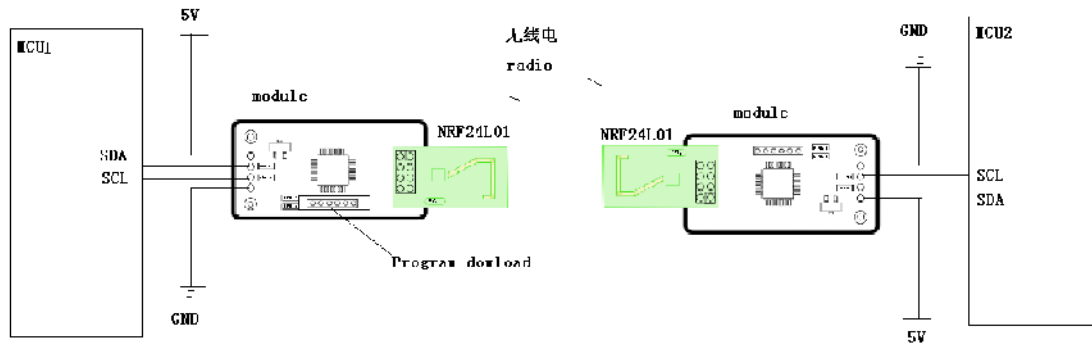
	Model 1	Model 2
Image		
Max transmitting distance (Open air)	70m	100m
Length		
Width		
Relevant link	<a href="http://www.icstation.com/ebay/IC/All%20data%20modules/1388.zip">http://www.icstation.com/ebay/IC/All%20data%20modules/1388.zip</a>	<a href="http://www.icstation.com/ebay/IC/All%20data%20modules/1438.zip">http://www.icstation.com/ebay/IC/All%20data%20modules/1438.zip</a>

## V. MECHANICAL DIAGRAM



Size information(尺寸信息)	
L	Max 73mm
l	45.5mm
L1	3mm
W	22.6mm
d	2mm
H	<20mm

## VI. OPERATIONS GUIDE



This module is used I2C/TWI communication, NRF wireless communication module used, it is safe and reliable. And it reduces the possibility of data lost greatly. In the communication process there are three stages: MCU reading data, Free Mode (MCU write data or NRF24L01 write data), Module busy mode(module sends data).

1) MCU reading data means a status that MCU reads the data of the module, after the module receives data which sent from NRF24L01 then will save it and wait for the master to read. (LED IN light of Module is on)

Note: Reading data status shield NRF24L01, namely NRF24L01 can not do any operation. While this status has a data reading hold time, if the master cannot read the data after it exceeds the holding time, the module will jump to the free mode automatically.

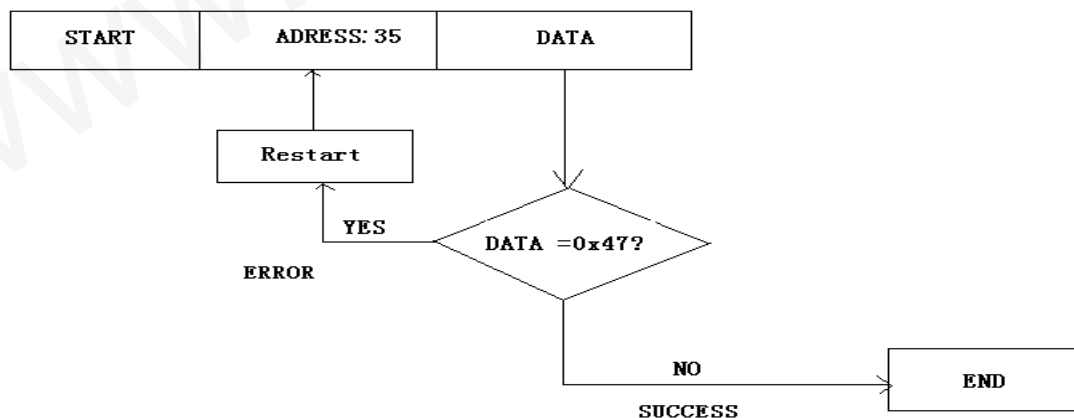
2) Free mode (MCU write data or NRF24L01 write data) means NRF24L01 module and the master (TWI/I2C) just can do the operation with it. (LED OUT / IN are off)

Several possibilities for modules Jump to Free Mode:

1. The times of module sending data reaches maximum.
2. Module sent data successfully.
3. MCU receives data from the module.
4. Free status is beyond the hold time.

3) Module busy mode: when module receives the data which sent from master then jumps into the "busy status", and sends a data to another module periodically while waiting for a responded signal from another module. If after a period of time (module transmit Data Hold Time) there is still no answer, module will jump to Free Mode automatically. If it has received the response signal, it can also jump to Free Mode. (LED OUT light of module is on)

#### 1..READING DATA FORMAT:



First MCU sends address to module, and module will response a signal if returned data is not "0x47", it indicates that reading data is successful.

Not that: MCU just can read the data with 1 byte size every time.

## 2..Writing Data Format:



MCU sends address and data to module, and every time just can send a data with 1 byte, after that there is a stop bit occur. If the module is in Busy Mode then it should resend the address and data again.

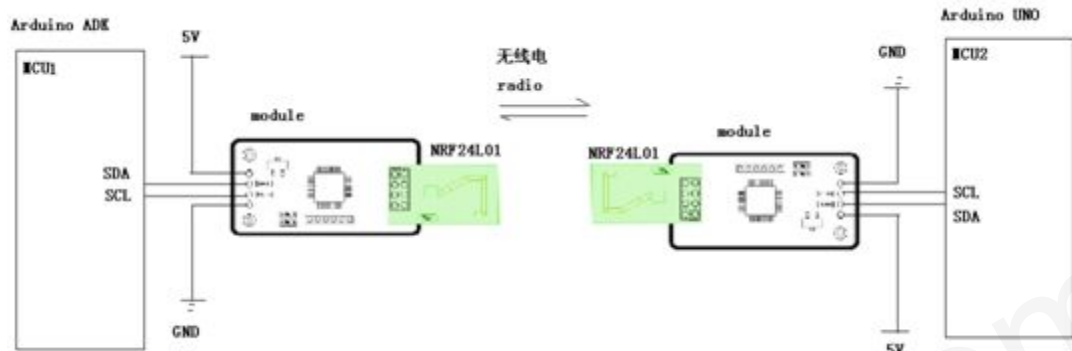
Note: When the module is in idle the data which is read by master is always "0x47".

When MCU receives data, you have to filter "0x47" (Decimal number 71), Otherwise the Program will encounter error.

## COMMUNICATION EXPERIMENT WITH MODULE:

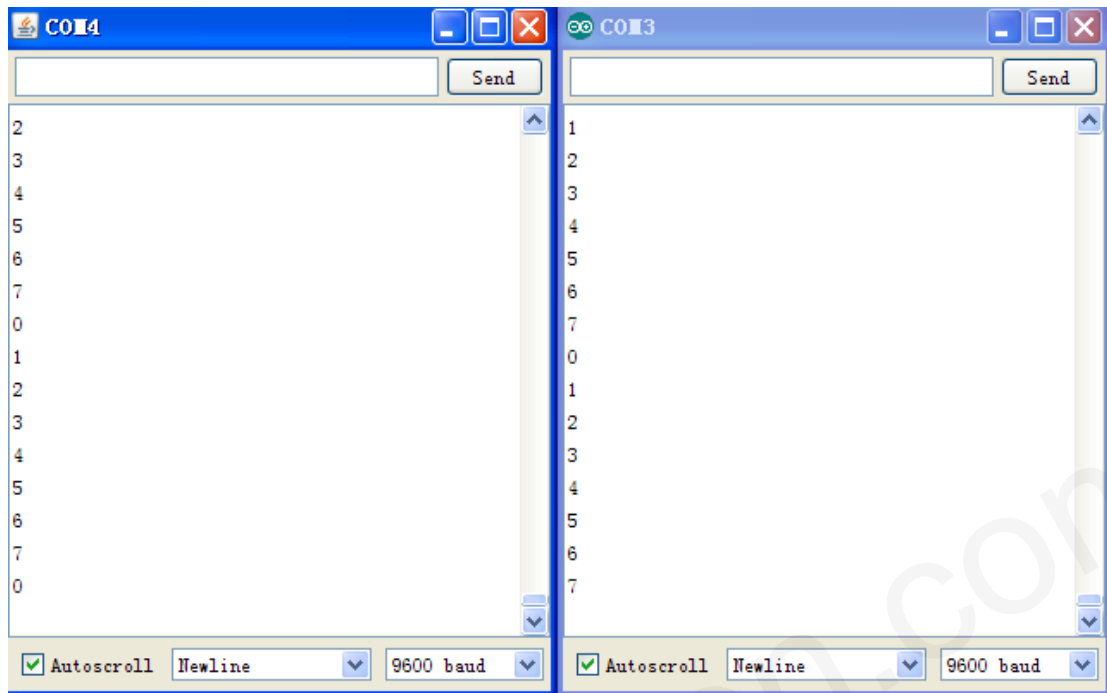
Experiment condition:

- Two Wireless modules, Arduino 328, Arduino ADK.



2. Procedure and results:

<p>I) 设备1 接收数据并且将接收的数据发送出去 Device 1 receiving data and will be sent the received data</p> <pre style="font-family: monospace; font-size: 0.8em;"> master_reader   Arduino 1.0.5 File Edit Sketch Tools Help master_reader\$ // Wire Master Reader // by Nicholas Zambetti &lt;http://www.zambetti.com&gt;  // Demonstrates use of the Wire library // Reads data from an I2C/TWI slave device // Refer to the "Wire Slave Sender" example for use with this  // Created 29 March 2006  // This example code is in the public domain. unsigned char ppt = 0;  #include &lt;Wire.h&gt;  void setup() {   Wire.begin(); // join i2c bus (address optional for master)   Serial.begin(9600); // start serial for output }  unsigned char judge = 0, good;  void loop() {   [ </pre> <p>Done uploading.</p> <p>Binary sketch size: 4,498 bytes (of a 30,720 byte maximum)</p> <p>38 Arduino Nano w/ ATmega328 on COM4</p>	<p>II) 设备2 首先发送数据,再将发送的数据接收回来 Device 2 To send data, and then will send back data reception</p> <pre style="font-family: monospace; font-size: 0.8em;"> master_writer   Arduino 1.0.5 File Edit Sketch Tools Help master_writer\$ // Wire Master Writer // by Nicholas Zambetti &lt;http://www.zambetti.com&gt;  // Demonstrates use of the Wire library // Writes data to an I2C/TWI slave device // Refer to the "Wire Slave Receiver" example for use with this  // Created 29 March 2006  // This example code is in the public domain.  #include &lt;Wire.h&gt;  void setup() {   Wire.begin(); // join i2c bus (address optional for master)   // Serial.begin(9600); // start serial for output }  [ </pre> <p>Done uploading.</p> <p>Binary sketch size: 6,358 bytes (of a 258,048 byte maximum)</p> <p>46 Arduino Mega 2560 or Mega ADK on COM3</p>
--	--



Description: This experiment is like this: Aduino ADK sends a data first , and then waits to receive the same data, if it has received data, then it will show the data as above picture COM3; Arduino UNO waits to receive data first, and then it will display the data after it receives every time, after that it will send back the data to module. So On Ad Inifinitum.

3. Code: See Attachment.