



Grove - 433MHz Simple RF link kit User Manual

Release date: 2015/9/22

Version: 1.0

Wiki: http://www.seeedstudio.com/wiki/Grove_-

[433MHz Simple RF link kit](http://www.seeedstudio.com/wiki/Grove_-433MHz_Simple_RF_link_kit)

Bazaar: <http://www.seeedstudio.com/depot/Grove-433MHz-Simple-RF-link-kit-p-1062.html>

Document Revision History

Revision	Date	Author	Description
1.0	Sep 22, 2015	Jiankai.li	Create file

Contents

Document Revision History	2
1. Introduction	2
2. Features	3
3. Specification	4
3.1 Transmitter Module	4
3.2 Receiver Module	4
4. Application Ideas	5
5. Usage	6
6. Resources	9

Disclaimer

For physical injuries and possessions loss caused by those reasons which are not related to product quality, such as operating without following manual guide, natural disasters or force majeure, we take no responsibility for that.

Under the supervision of Seeed Technology Inc., this manual has been compiled and published which covered the latest product description and specification. The content of this manual is subject to change without notice.

Copyright

The design of this product (including software) and its accessories is under tutelage of laws. Any action to violate relevant right of our product will be penalized through law. Please consciously observe relevant local laws in the use of this product.

1. Introduction

This kit is used for one way wireless communication at a frequency of 433MHz and includes a transmitter module and a receiver module. The twig configuration of this kit allows for around 40 meters of transmitting distance indoors, or around 100 meters outside.



2. Features

- GROVE compatible interface.
- Uses ASK (Amplitude Shift Keying) Modulation.
- One way communication.

3. Specification

3.1 Transmitter Module

Item	Min	Typical	Max	Unit
Working Voltage	3.0	5.0	12.0	VDC
Current	3	/	10	mA
Work Mode	ASK			/
Transmit Power(Max)	15			mW
Working Distance	40	/	100	m

3.2 Receiver Module

Item	Typical	Unit
Working Voltage	5	VDC
Quiescent Current	5	mA
Receiver Sensitivity	-105	dBm
Operating frequency	433.92	MHz

4. Application Ideas

- Remote control
- Remote automation
- Alarm

5. Usage

The transmitter and receiver modules both rely on a single wire for communication. Though using the UART supplied by the Arduino platform can work, it is recommended, instead, to use the VirtualWire library which uses Amplitude Shift Keying for modulation which provides better communication.

Both the transmitter and receiver modules require three wires: Vcc, Ground, and signal. Pin 2 of both parts of the kit are not connected.

- Connect the Transmitter module to Digital I/O 2 of the [Grove - Base Shield](#) on the Arduino being used for transmission.

Error creating thumbnail: Invalid thumbnail parameters

- Connect the Receiver module to Digital I/O 2 of the [Grove - Base Shield](#) on the receiving Arduino.

Error creating thumbnail: Invalid thumbnail parameters

- Download the [File: VirtualWire library](#) and unzip it into the libraries file of Arduino IDE by the path: ..\arduino-1.0\libraries. Please reference [here](#).
- Upload the code below for transmitter module:

```
#include <VirtualWire.h>

//Grove - 315(433) RF link kit Demo v1.0
//by :http://www.seeedstudio.com/
//connect the sent module to D2 to use
#include <VirtualWire.h>

int RF_TX_PIN = 2;

void setup()
{
  vw_set_tx_pin(RF_TX_PIN); // Setup transmit pin
  vw_setup(2000); // Transmission speed in bits per second.
}

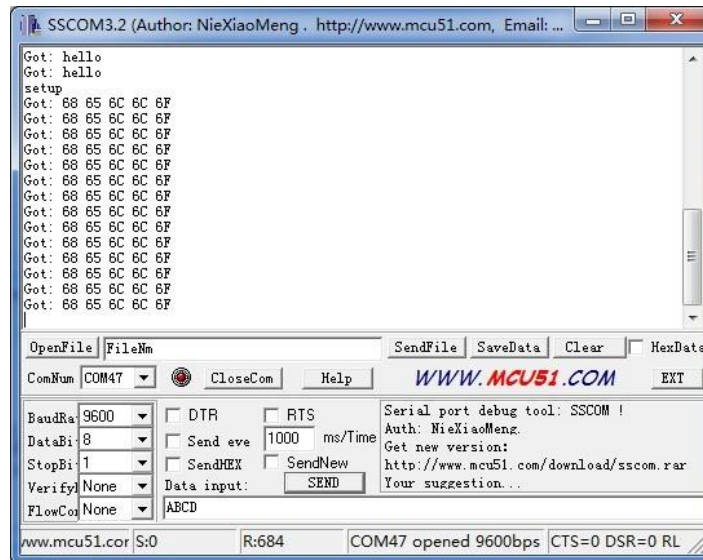
void loop()
```

```
{  
  const char *msg = "hello";  
  vw_send((uint8_t *)msg, strlen(msg)); // Send 'hello' every 400ms.  
  delay(400);  
}
```

- Upload the code below for receiver module:

```
//Grove - 315(433) RF link kit Demo v1.0  
//by :http://www.seeedstudio.com/  
//connect the receive module to D2 to use ..  
#include <VirtualWire.h>  
  
int RF_RX_PIN = 2;  
  
void setup()  
{  
  Serial.begin(9600);  
  Serial.println("setup");  
  vw_set_rx_pin(RF_RX_PIN); // Setup receive pin.  
  vw_setup(2000); // Transmission speed in bits per second.  
  vw_rx_start(); // Start the PLL receiver.  
}  
  
void loop()  
{  
  uint8_t buf[VW_MAX_MESSAGE_LEN];  
  uint8_t buflen = VW_MAX_MESSAGE_LEN;  
  if(vw_get_message(buf, &buflen) // non-blocking I/O  
  {  
    int i;  
    // Message with a good checksum received, dump HEX  
    Serial.print("Got: ");  
    for(i = 0; i < buflen; ++i)  
    {  
      Serial.print(buf[i], HEX);  
      Serial.print(" ");  
      //Serial.print(buf[i]);  
    }  
    Serial.println("");  
  }  
}
```

- Open the serial monitor of receiver module to see the result.



This is just a simple transmitter and receiver instance as a reference.