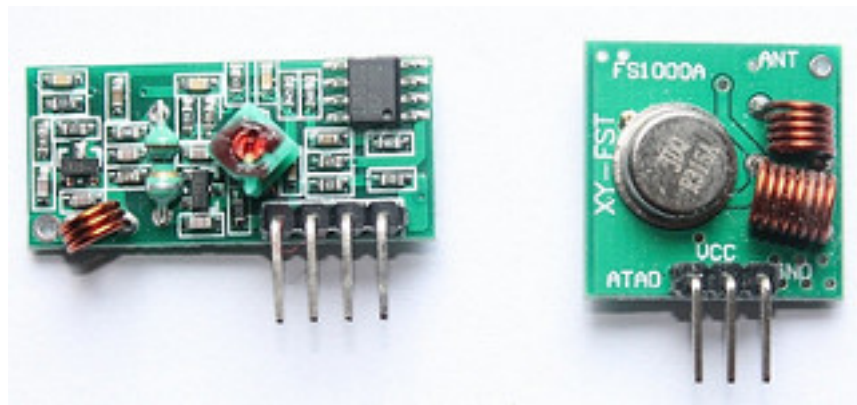


# HOW TO USE 315MHZ RF TRANSMITTER AND RECEIVER MODULES WITH ARDUINO

By **Sagar Sapkota** - June 14, 2014

**Description:** This wireless transmitter and receiver pair operate at 315Mhz. They can easily fit into a breadboard and work well with microcontrollers to create a very simple wireless data link. Since these are only transmitters, they will only work communicating data one-way, you would need two pairs (of different frequencies) to act as a transmitter/receiver pair.

**Note:** These modules are indiscriminate and will receive a fair amount of noise. Both the transmitter and receiver work at common frequencies and don't have IDs. Therefore, a method of filtering this noise and pairing transmitter and receiver will be necessary. The example code below shows such an example for basic operation. Please refer to the example code and links below for ways to accomplish a robust wireless data link.



*Module on the left: Receiver*

*Module on the right: Transmitter*

[Buy Now at BuildCircuit](#)

## Application environment

Remote control switch, receiver module, motorcycles, automobile anti-theft products, home security products, electric doors, shutter doors, windows, remote control socket, remote control LED, remote audio remote control electric doors, garage door remote control, remote control retractable doors, remote volume gate, pan doors, remote control door opener, door closing device control system, remote control curtains, alarm host, alarm, remote control motorcycle remote control electric cars, remote control MP3.

## Receiver module parameters



1. Product Model: MX-05V
2. Operating voltage: DC5V
3. Quiescent Current: 4mA
4. Receiving frequency: 315Mhz
5. Receiver sensitivity: -105DB
6. Size: 30 \* 14 \* 7mm

## Technical parameters of the transmitter module

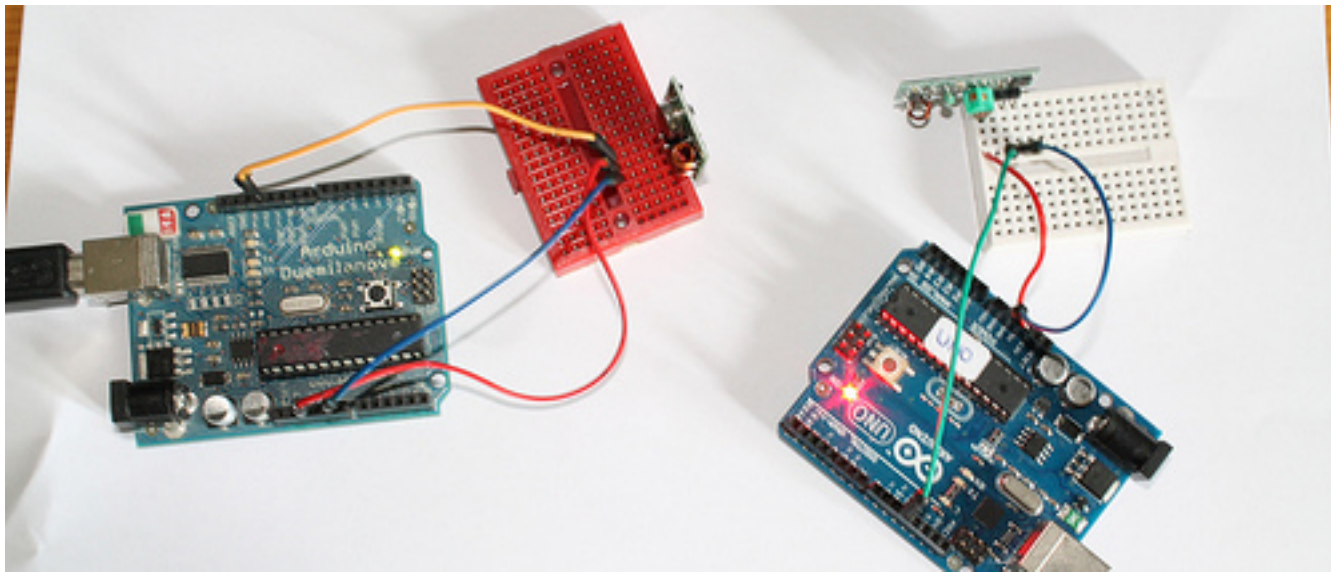


1. Product Model: MX-FS-03V
2. Launch distance :20-200 meters (different voltage, different results)
3. Operating voltage :3.5-12V
4. Dimensions: 19 \* 19mm
5. Operating mode: AM
6. Transfer rate: 4KB / S
7. Transmitting power: 10mW
8. Transmitting frequency: 315Mhz
9. An external antenna: 25cm ordinary multi-core or single-core line
10. Pinout from left to right: (DATA; VCC; GND)

# Buy Now at BuildCircuit

## An example:

In this example, receiver and transmitter modules are connected separately to two Arduino boards. The transmitter data pin is connected to Pin 12 of Arduino and the receiver data pin is connected to Pin 11 of Arduino.

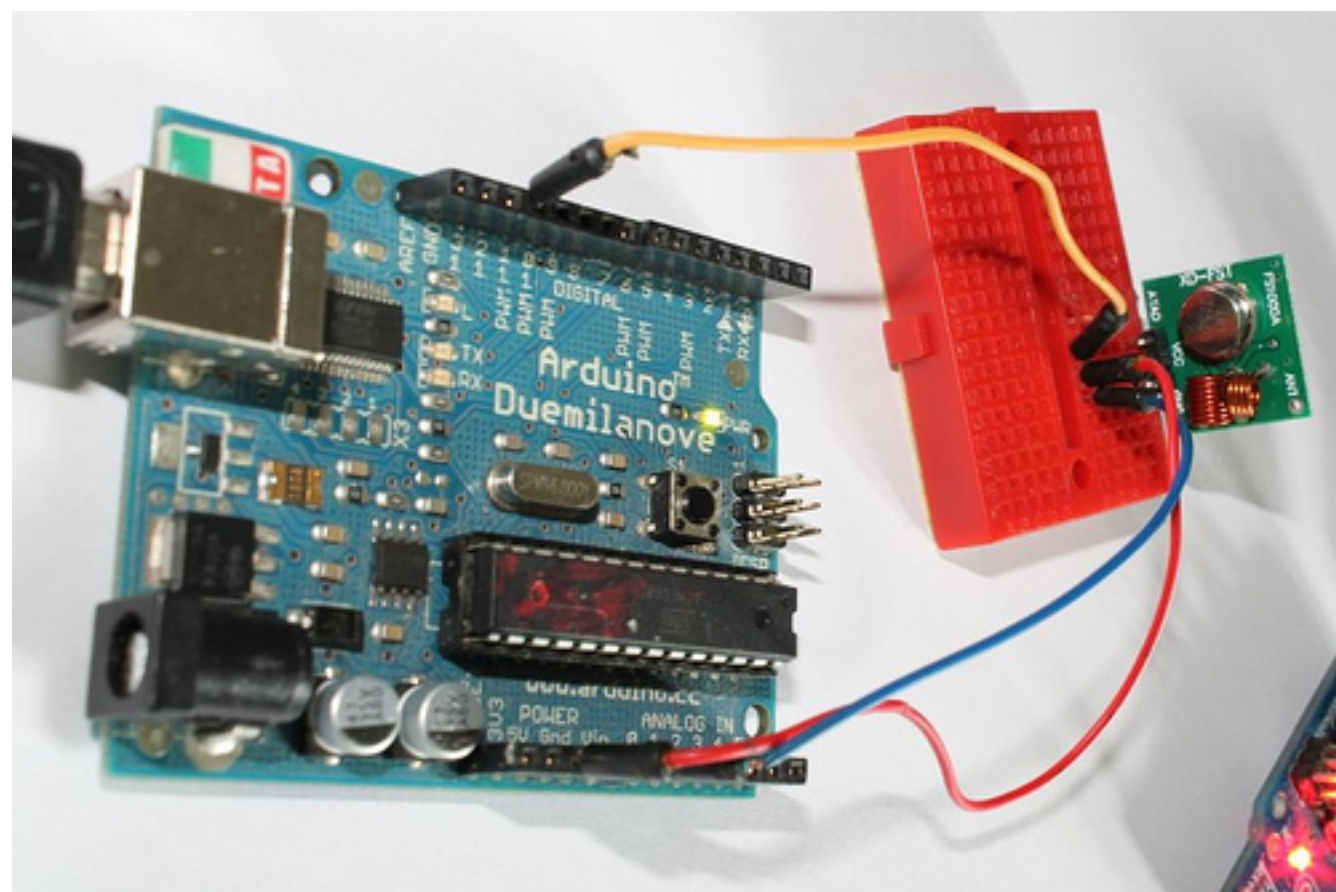


*Data pin of transmitter module to Pin 12 of Arduino.*

*Data pin of receiver module to Pin 11 of Arduino.*

*Please note that there are two separate Arduinos for each module.*

## Connecting transmitter module to Arduino:



See all the images on flickr

Sketch:

Download library(Virtual wire)

1	/*
2	SimpleSend
3	This sketch transmits a short text message using the VirtualWire library
4	connect the Transmitter data pin to Arduino pin 12
5	*/
6	#include <VirtualWire.h>
7	void setup()
8	{
9	// Initialize the IO and ISR
10	vw_setup(2000); // Bits per sec
11	}
12	void loop()
13	{
14	send("Hello there");
15	delay(1000);
16	}
17	void send (char *message)
18	{
19	vw_send((uint8_t *)message, strlen(message));
20	vw_wait_tx(); // Wait until the whole message is gone
21	}
22	

gistfile1.ino hosted with ❤ by GitHub

view raw

1	/*
2	SimpleSend
3	This sketch transmits a short text message using the VirtualWire library
4	connect the Transmitter data pin to Arduino pin 12
5	*/
6	#include <VirtualWire.h>
7	void setup()
8	{
9	// Initialize the IO and ISR
10	vw_setup(2000); // Bits per sec
11	}



```

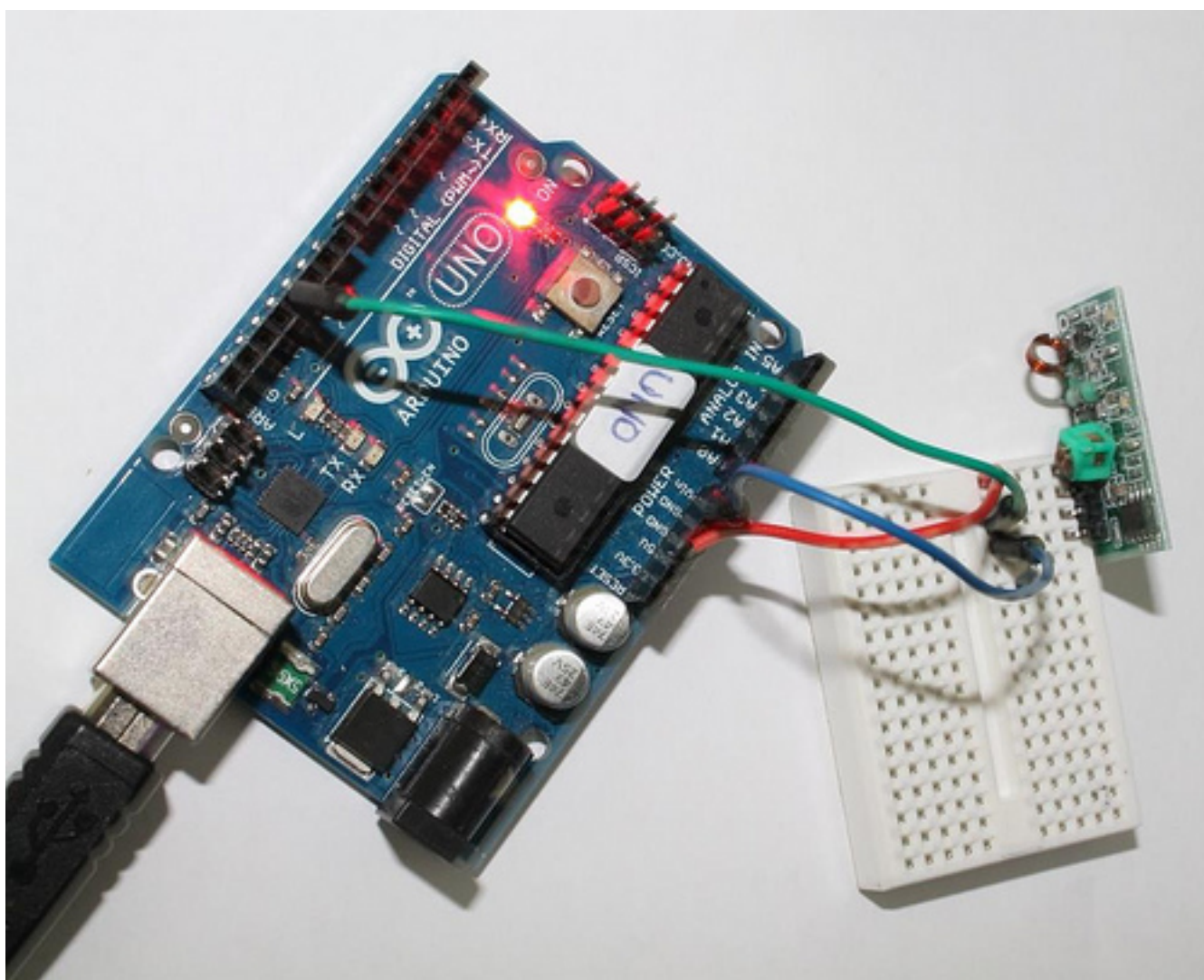
12 void loop()
13 {
14   send("Hello there");
15   delay(1000);
16 }
17 void send (char *message)
18 {
19   vw_send((uint8_t *)message, strlen(message));
20   vw_wait_tx(); // Wait until the whole message is gone
21 }
22

```

gistfile1.ino hosted with ❤ by GitHub

[view raw](#)

## Connecting receiver module to Arduino:



See all the images on flickr **Sketch:**

```

1  /*
2   SimpleReceive
3   This sketch displays text strings received using VirtualWire
4   Connect the Receiver data pin to Arduino pin 11
5   */
6   #include <VirtualWire.h>
7   byte message[VW_MAX_MESSAGE_LEN]; // a buffer to store the incoming messages
8   byte messageLength = VW_MAX_MESSAGE_LEN; // the size of the message

```

```
9 void setup()
10 {
11   Serial.begin(9600);
12   Serial.println("Device is ready");
13   // Initialize the IO and ISR
14   vw_setup(2000); // Bits per sec
15   vw_rx_start(); // Start the receiver
16 }
17 void loop()
18 {
19   if (vw_get_message(message, &messageLength)) // Non-blocking
20   {
21     Serial.print("Received: ");
22     for (int i = 0; i < messageLength; i++)
23     {
24       Serial.write(message[i]);
25     }
26     Serial.println();
27   }
28 }
```

gistfile1.ino hosted with ❤ by GitHub

[view raw](#)

```
1  /*
2  SimpleReceive
3  This sketch displays text strings received using VirtualWire
4  Connect the Receiver data pin to Arduino pin 11
5  */
6  #include <VirtualWire.h>
7  byte message[VW_MAX_MESSAGE_LEN]; // a buffer to store the incoming messages
8  byte messageLength = VW_MAX_MESSAGE_LEN; // the size of the message
9  void setup()
10 {
11   Serial.begin(9600);
12   Serial.println("Device is ready");
13   // Initialize the IO and ISR
14   vw_setup(2000); // Bits per sec
15   vw_rx_start(); // Start the receiver
16 }
17 void loop()
18 {
19   if (vw_get_message(message, &messageLength)) // Non-blocking
20   {
```

21	Serial. <a href="#">print</a> ("Received: ");
22	<a href="#">for</a> ( <a href="#">int</a> i = 0; i < messageLength; i++)
23	{
24	Serial. <a href="#">write</a> (message[i]);
25	}
26	Serial. <a href="#">println</a> ();
27	}
28	}
gistfile1.ino hosted with ❤️ by <a href="#">GitHub</a>	
<a href="#">view raw</a>	

[Download library\(Virtual wire\)](#)

## Output:

The transmitter sends a string “Hello there” and the receiver receives it and displays on serial monitor.



## Related links:

1. [Virtual wire- Arduino Library](#)
2. [Buy 315Mhz receiver and transmitter modules on www.buildcircuit.net](#)
3. [Video](#)
4. [Flickr Images](#)



- 
- **JLCPCB - Prototype PCBs for \$2 + Free Shipping on First Order**
  - China's Largest PCB Prototype Manufacturer, 290,000+ Customers & 8000+ Online Orders Per Day
  - 10 PCBs Price: \$2 for 2-layer, \$15 for 4-layer, \$74 for 6-layer