EN: This Datasheet/Document is presented by the manufacturer.

Please visit our website for pricing and availability at www.hestore.hu.

User Guide

433MHz RF Transmitter/Receiver Module

These RF Modules are designed to serve as a tool for electronic design engineers, developers, hobbyists and students to perform wireless application. These 433MHz Receiver/Transmitter pair is an Arduino-compatible expansion board that gives your Arduino the ability to receive and decode radio signals transmitted on the unlicensed 433MHz band using ASK (Amplitude Shift Keying) or OOK (On-Off Keying) modulation. ASK signals at 433MHz are very commonly used by consumer electronics devices including weather stations, power consumption meters, security systems, and home automation controllers.



Brief Data:

Receiver:

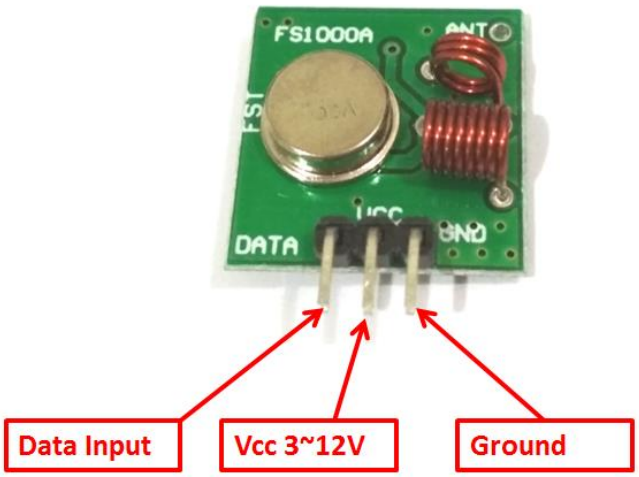
- Model: MX-05V.
- Voltage: 5Vdc
- Current: 4mA.
- Frequency: 433.92MHz.
- Sensitivity: -105 dB.
- Size: 30x14x7mm.

Transmitter:

- Model: MX-FS-03V.
- Transmitter distance: 20~200Meters.
- Voltage: 3.5~12V.
- Size: 19x19mm.
- Working Mode: Amplitude Modulation (AM).
- Transmitter Speed: 4KB/S.
- Transmitter power: 10mW.

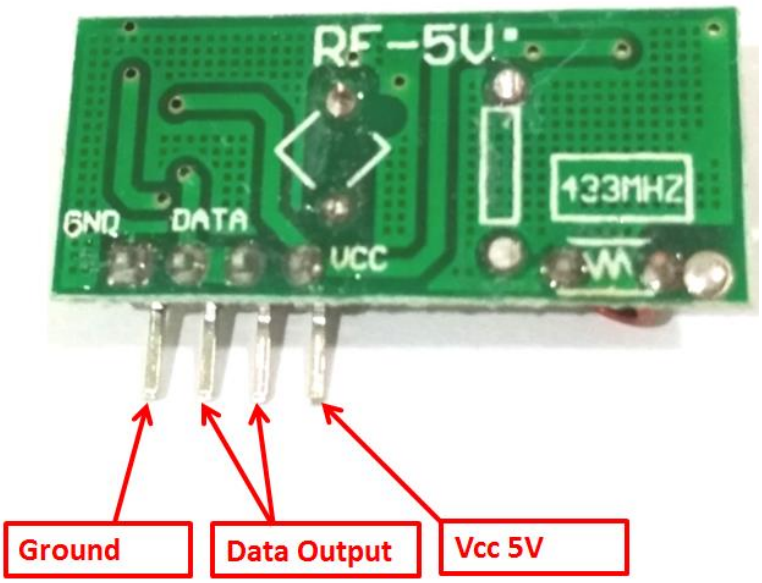
Pin Assignment:

Transmitter:



- VCC – This can be any positive DC voltage from 3 to 12 volts. In this experiment we will get 5 volts from our Arduino.
- GND – The ground connection, connected to one of the Arduino ground terminals.
- DATA IN – This is connected to pin 12 of the Arduino. You should try and use pin 12 as by default the library we'll be using in our sketch uses this pin for data output.

Receiver:



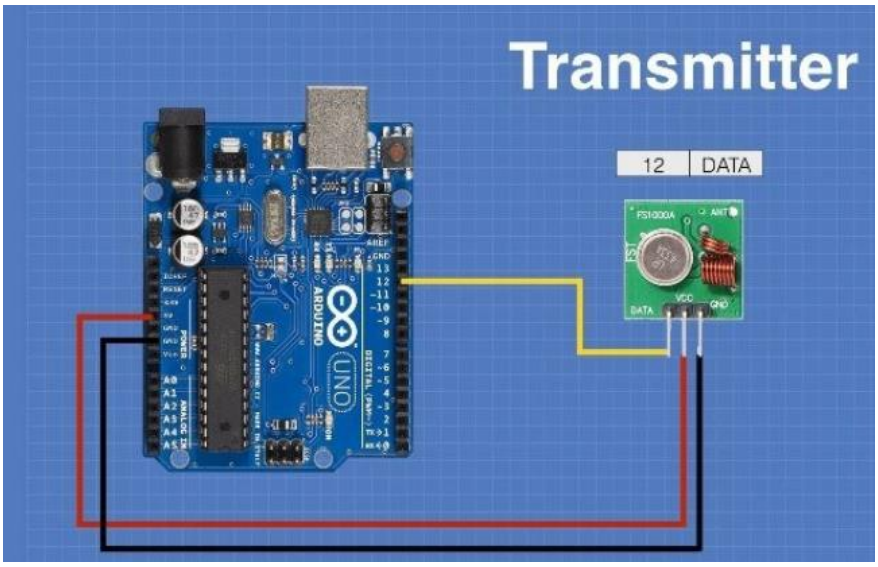
- VCC – Unlike the transmitter this needs to be 5 volts. We will use the 5 volt output from the Arduino.
- GND – Again a ground that is connected to any Arduino ground pin.
- DATA OUT – This needs to be connected to digital pin 11 on the Arduino.

Application with Arduino Board

In our first experiment we will just send a message from the transmitter to the receiver. While this isn't extremely useful it does serve to illustrate how to use the modules and can serve as the basis for more practical experiments and projects. It's also a great way to see if our antenna design is sound. As we will be sending data between two Arduino we will need two Arduino board. This guide shown on this experiment using an Arduino Uno but any Arduino will work.

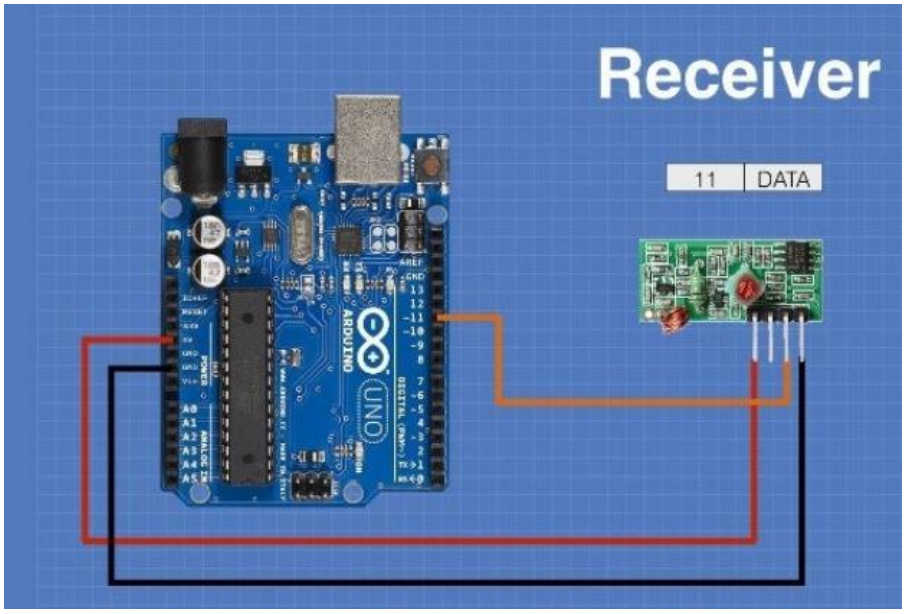
Hooking up the Transmitter:

Wire up the transmitter module to Arduino Uno board as below. Data pin of the transmitter connected to Arduino digital Pin 12 (D12).



Hooking up the Receiver:

Wire up the receiver module to Arduino Uno board as below. Data pin of the receiver connected to Arduino digital Pin 11 (D11).



Now that both the transmitter and receiver are wired up we will need to write some code and send it to the respective Arduino boards. Since you probably have only one computer we will start with the transmitter. Once the code has been loaded there we'll move on to the receiver. The transmitter can then be powered using a power supply or battery.

But before we start coding there is a library "RadioHead Library" need to install into our Arduino IDE that will make writing the code a lot simpler. This is a very complex library, we will only be using the Amplitude Shift Keying code from it.

You will need to download the library from the Airspayce website.

Look for a link to the library at the top of the page and download the ZIP file to a location on your computer that you will remember – your *downloads* folder is probably as good a place as any. Now open your Arduino IDE. Go to the *Sketch* pulldown menu and open the *Include Library* submenu. Choose Add ZIP library. A dialog box will open, allowing you to navigate to the location you put your RadioHead ZIP file. Find the file and select it. The library will be installed into your Arduino IDE. You may need to restart the IDE after doing this. Now that RadioHead is part of your Arduino IDE you are ready to code.

Transmitter Code Example:

Upload the below sketch to the Arduino board with transmitter module attached:

```
/*
  433 MHz RF Module Transmitter Demonstration 1
  ask_transmitter.ino
  Demonstrates 433 MHz RF Transmitter Module
  www.handsontec.com

*/

// Include RadioHead Amplitude Shift Keying Library
#include <RH_ASK.h>
// Include dependant SPI Library
#include <SPI.h>

// Create Amplitude Shift Keying Object
RH_ASK rf_driver;

void setup()
{
    // Initialize ASK Object
    rf_driver.init();
}

void loop()
{
    const char *msg = "Hello, from Handson Technology!";
    rf_driver.send((uint8_t *)msg, strlen(msg));
    rf_driver.waitPacketSent();
    delay(1000);
}
```

It's a pretty short sketch but it's all you need to get a signal transmitted.

Now disconnect your computer from the Arduino, find another power supply for the Arduino and power it back up. It will transmit continuously while we code our receiver.

Receiver Code Example:

Upload the below sketch to the Arduino board with receiver module attached:

```
/*
  433 MHz RF Module Receiver Demonstration 1
  ask_receiver.ino
  Demonstrates 433MHz RF Receiver Module
  www.handsontec.com
*/

// Include RadioHead Amplitude Shift Keying Library
#include <RH_ASK.h>
// Include dependant SPI Library
#include <SPI.h>

// Create Amplitude Shift Keying Object
RH_ASK rf_driver;

void setup()
{
    // Initialize ASK Object
    rf_driver.init();
    // Setup Serial Monitor
    Serial.begin(9600);
}

void loop()
{
    // Set buffer to size of expected message
    uint8_t buf[24];
    uint8_t buflen = sizeof(buf);
    // Check if received packet is correct size
    if (rf_driver.recv(buf, &buflen))
    {

      // Message received with valid checksum
      Serial.print("Message Received: ");
      Serial.println((char*)buf);
    }
}
```

After loading the code open your serial monitor. If all is working you should see your message.

