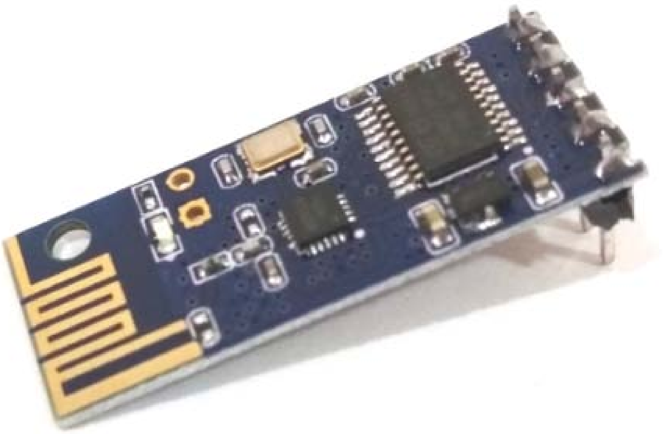
User Guide

2.4GHz RF Transceiver Module

This module is fully compatible with Arduino board and generally used in pairs for 2-ways RF communication. This module uses the AT-Command for configuration setting. Operate in unlicensed ISM RF band of 2.405GHz~2.485GHz, operating distance can up to 200 meter line of sight



SKU: MDU-1054

Specifications:

- Operating frequency: 2.4G (2400MHz ~ 2485MHz). User setting through the AT command.
- Operating voltage: 3.3-5.5V.
- Operating current: Transparent Mode@ 21mA.
- Operating current: Configuration mode @ 5mA.
- Data Format: Serial RxD, TxD.
- Transmit Power: -30dBm ~ + 13dBm (Configurable with AT command).
- Operating Distance: Open Space line-of-sight, 150 meters to 200 meters.
- Module Size: 32mmX12.7mmX2mm (length X width x height).

Module Pins Assignment:

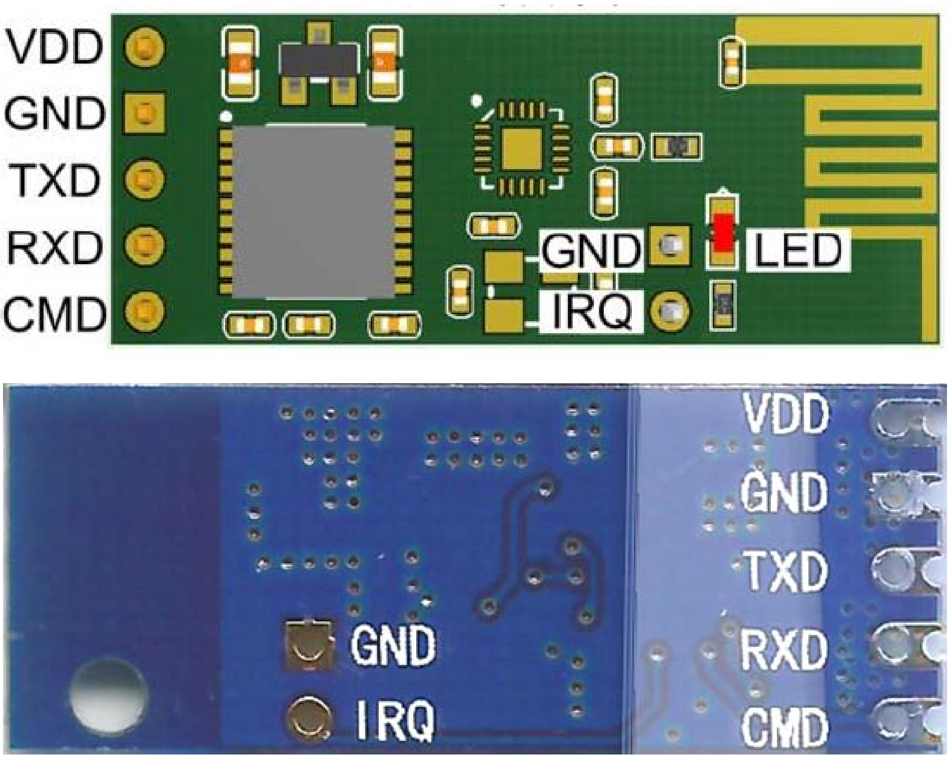


Figure-1

Pin Name	Description
VDD	Power Supply: 3.3Vdc ~ 5Vdc.
GND	Ground.
TXD	Serial Transmitting pin.
RXD	Serial Receiving pin.
CMD	Command Pin. Gound: Command Setting Mode. Vdd: Transparent Transmitting/Receiving Mode. Fixed at Baud-rate 9600.

Initial Setting:

In this section we will demonstrate the procedure to perform initial setup of this 2.5GHz RF Transceiver before it can be in RF communication link project. We will need two pieces of this module to complete the task.

First, we will perform some simple setting on the 2.4GHz module using AT-Command.

Parts needed in this task:

- 2.4GH RF Transceiver Module, 2-pieces.
- PL2303HX USB-to-UARTS/TTL Interface cable.

Do the wiring as the below diagram between USB-UARTS/TTL and the RF module:

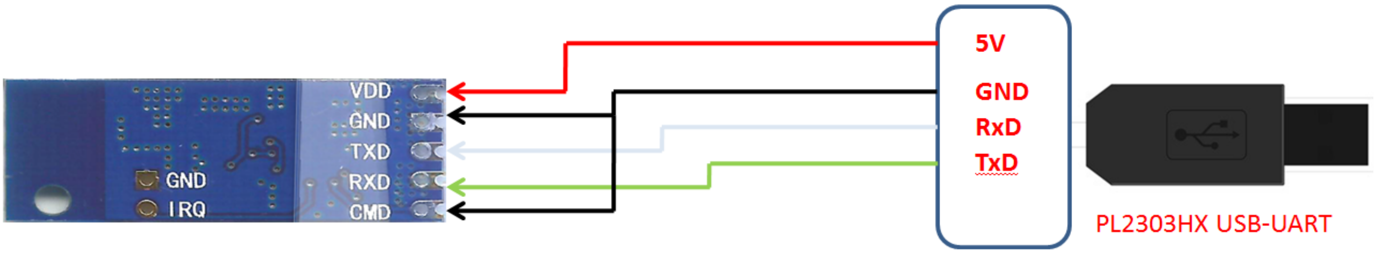
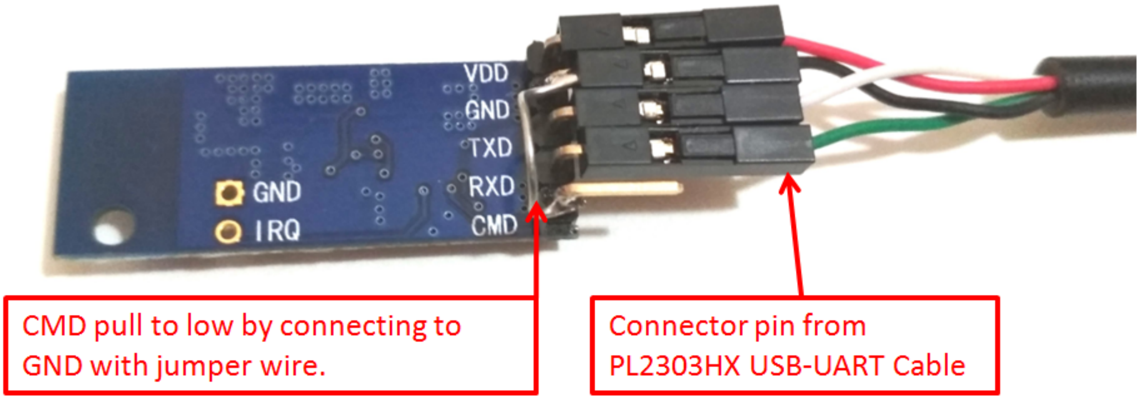


Figure-2

Table-1: Pin Assignment.

2.4GHz RF Module	USB-to-UARTS/TTL Note1
VDD	5V (Red Wire)
GND	GND (Black Wire)
TXD	RxD (White Wire)
RXD	TxD (Green Wire)
CMD	GND



CMD pull to low by connecting to GND with jumper wire.

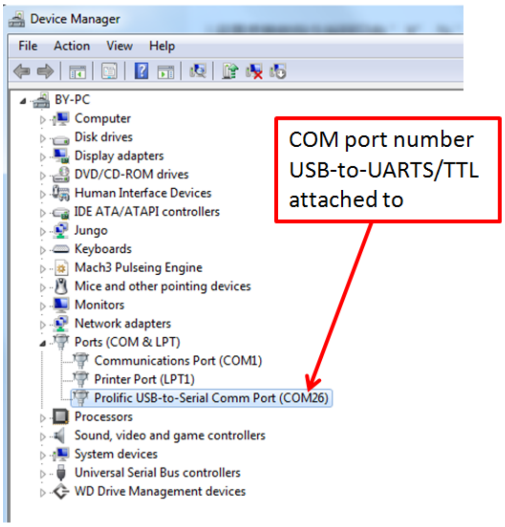
Connector pin from PL2303HX USB-UART Cable

Figure-3

Note1: Refer to user guide on USB-to-UARTS/TTL for detail.

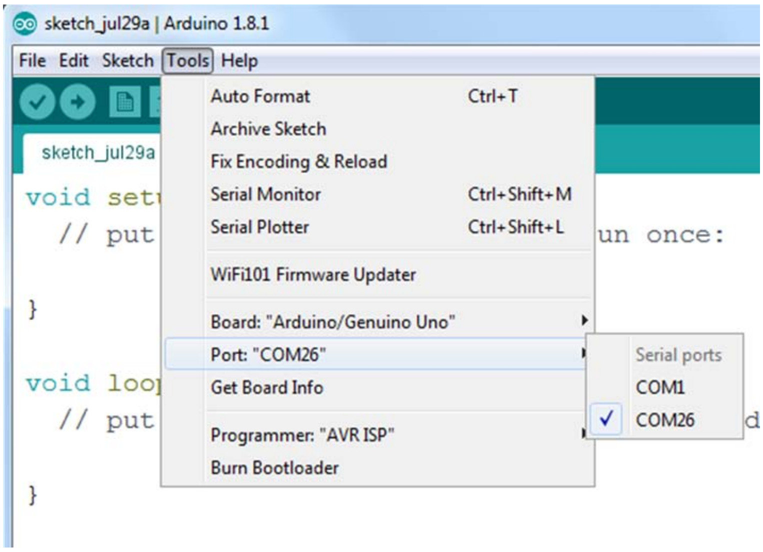
The CMD pin need to pull low (connect to ground) in order to initiate the AT-Command communication. We will use the Serial Monitor from Arduino IDE to perform the setting.

1. Plug in the USB-to-UARTS/TTL cable to your computer USB port and find out the COM port number from Windows Device Manager panel.



COM port number USB-to-UARTS/TTL attached to

2. Open up Arduino IDE and select the COM port the USB-to-UARTS/TTL cable attached to.



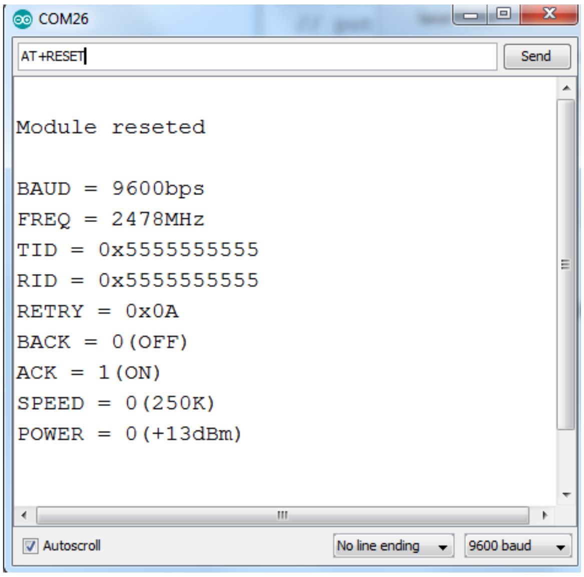
```
void setu
  // put                    un once:

}

void loop
  // put                                    d

}
```

3. Launch the Serial Monitor with baud rate of 9600. Type ‘AT+RESET” command in the text box and click send. RF Module will echo back “Module reseted” indicate successful communication.

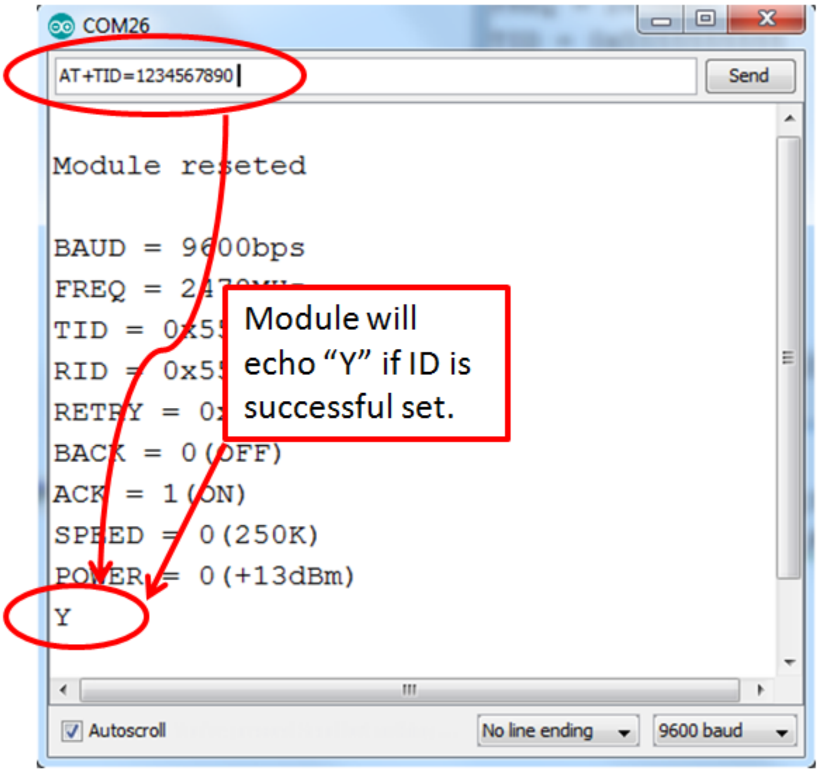


AT+RESET

```
Module reseted

BAUD = 9600bps
FREQ = 2478MHz
TID = 0x5555555555
RID = 0x5555555555
RETRY = 0x0A
BACK = 0(OFF)
ACK = 1(ON)
SPEED = 0(250K)
POWER = 0(+13dBm)
```

4. Now we need to pair Transmitting and Receiving module with same ID in order to communicate to each other. For the transmitting module, key in “ AT+TID=1234567890 “ into the text back and hit ‘send’ button on the right. Module will echo “Y” if AT command is successful sent, “N” if command is not success full send. You can change the ID to any character and number after the equal sign.



```
AT+TID=1234567890
```

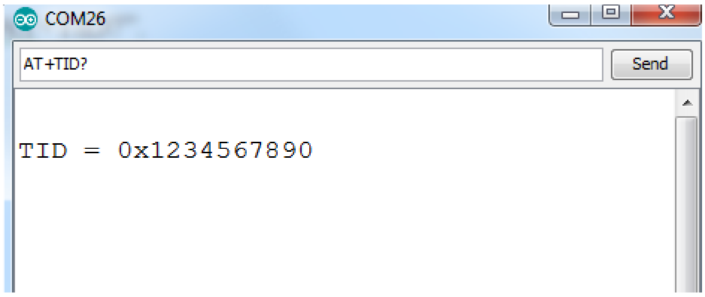
```
Module reseted

BAUD = 9600bps
FREQ = 2470MHz
TID = 0x55
RID = 0x55
RETRY = 0x
BACK = 0(OFF)
ACK = 1(ON)
SPEED = 0(250K)
POWER = 0(+13dBm)
Y
```

Module will echo “Y” if ID is successful set.

☑ Autoscroll   No line ending   9600 baud

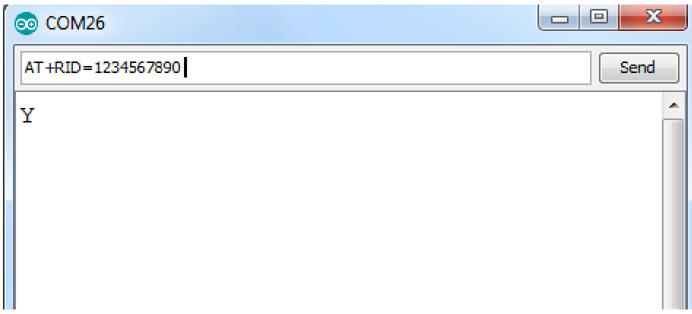
5. We can verify the ID by issue this command “ AT+TID?”.



```
AT+TID?
```

```
TID = 0x1234567890
```

6. As we need a receiving portion with the same ID, repeat the step 4 above with another RF Module and issue this command “ AT+RID=1234567890 “.



```
AT+RID=1234567890
```

```
Y
```

7. Now we have two RF Module, one for transmitting and one for receiving, configured with same ID. Now we can proceed to the next step for data and message transmitting and receiving in transparent mode.

Operating in Transparent Mode for Data and Message Transmitting and Receiving

In this demo section, we will use two Arduino board to perform the task for data or message transmitting and receiving. One Arduino board will act as data/message transmitter, another Arduino board act as receiver to receive the data/message and display on Serial Monitor. For the RF module to operated in transparent data/message transmitting and receiving, the " CMD " pin need to pull high by connecting to VDD pin.

Parts Needed in this Demo:

- Arduino Uno Board, 2-pcs.
- 2.4GHz RF Transceiver Module, 2-pcs.

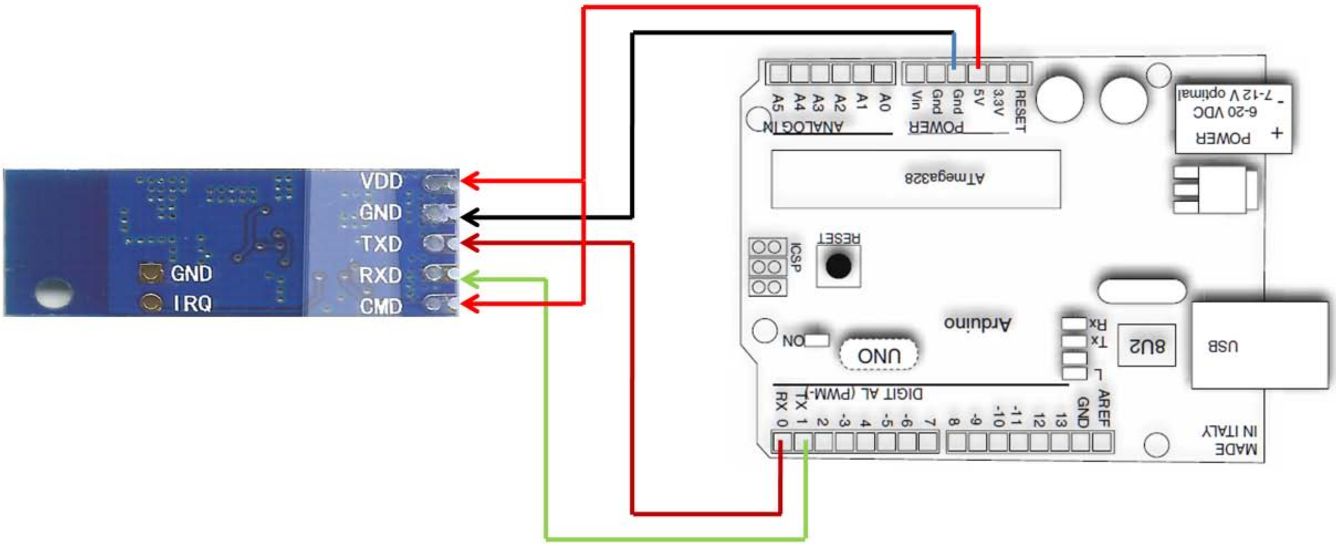


Figure-4

We will mark this Arduino Board as "Board-TX" for transmitting board.

Upload the following sketch into this Arduino marked as "Board-TX":

```
/*============================================================================
//  Author      : Handson Technology
//  Project     : Arduino Uno
//  Description : 2.4GHz RF Transceiver Module Demo
//  Source-Code : RF2.4GHZ-TX.ino
//============================================================================
*/

void setup()
{
  Serial.begin(9600);
}
void loop()
{
   Serial.write("A"); // transmitting character "A" thru 2.4GHz RF Module
}
```

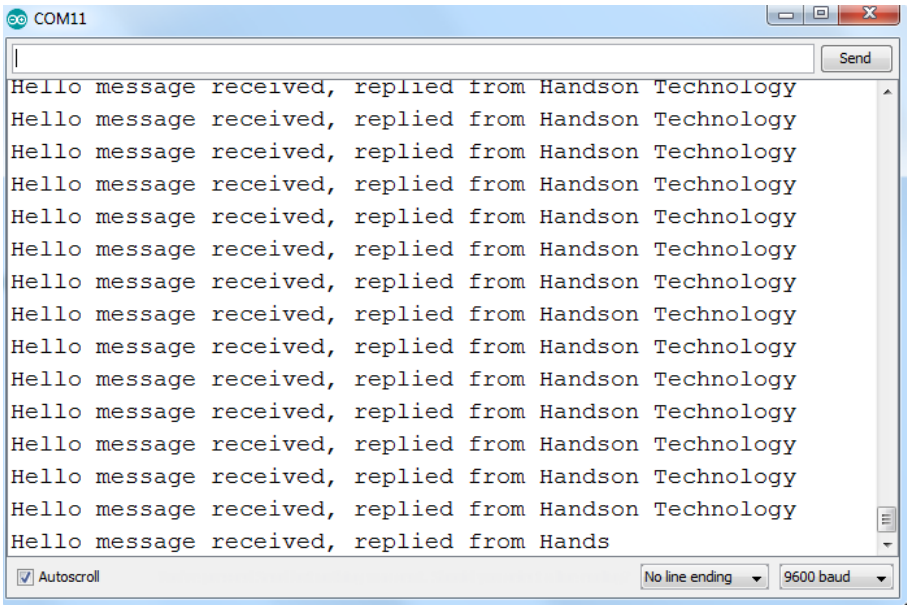
Connect another Arduino Board and 2.4GHz module as in Figure-4 and marked as “Board-RX” for receiving the RF message and display in Serial Monitor.

Upload the following sketch to this 2nd board marked as “Board-RX”:

```
/*============================================================================
//  Author      : Handson Technology
//  Project     : Arduino Uno
//  Description : 2.4GHz RF Transceiver Module Demo
//  Source-Code : RF2.4GHZ-RX.ino
//============================================================================
*/

void setup()
{
  Serial.begin(9600);
}
void loop()
{
   while(Serial.available())
    {
      char c=Serial.read();
       if(c=='A')
         {
           Serial.println("Hello message received, replied from Handson Technology");
         }
    }
}
```

Place the 1st Arduino board marked as “Board-TX” some distance away from 2nd board marked as “Board-RX”. For the 1st board “Board-TX”, you can use USB power bank as power supply. For the 2nd board marked as “Board-RX”. Open up the Serial Monitor from Arduino IDE, baud rate set to 9600. The following screen message should show up in Serial Monitor. Once the Board-RX received character “A” transmitted by Board-TX, it will display: “`Hello message received, replied from Handson Technology”.`



```
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Handson Technology
Hello message received, replied from Hands
```

Congratulation: You have successfully setup RF communication link between the 2 RF Modules !

AT Command Set Introduction:

Coming…