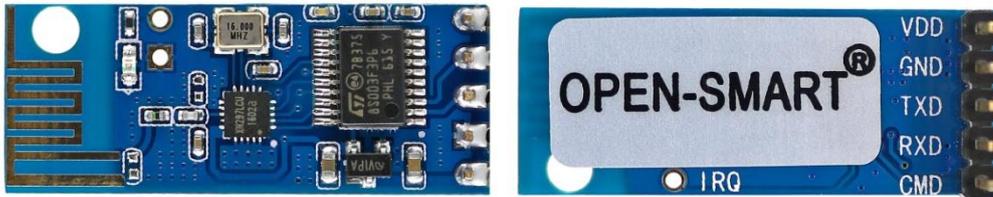


# OPEN-SMART



**Name: Blue 2.4G Wireless Serial manual**

**Version: v1.0**

**Date: 2017/7/12**

©OPEN-SMART

# CONTENT

1. Overview .....	- 3 -
1.1 Overview .....	- 3 -
1.2 Pins Description .....	- 3 -
1.3 Note .....	- 4 -
2 AT Commands .....	- 5 -
2.1 Summery .....	- 5 -
2.2 AT Commands List.....	- 6 -
2.3 Set Baud Rate and Query baud rate .....	- 7 -
2.4 Set Frequency and Query frequency .....	- 9 -
2.5 Set number of retransmissions and Query number of retransmissions .....	- 10 -
2.6 Set whether to return and Query whether to return .....	- 11 -
2.6.2 Query whether to return .....	- 11 -
2.7 Set TX ID and Query TX ID .....	- 12 -
2.7.2 Query TX ID .....	- 12 -
2.8 Set RX ID and Query RX ID .....	- 13 -
2.8.2 Query RX ID.....	- 13 -
2.9 Set Transmit Power and Query Transmit Power .....	- 14 -
2.9.2 Query Transmit Power .....	- 14 -
2.10 Restore Defaults.....	- 15 -
2.11 Query firmware version.....	- 16 -
3 Link .....	- 16 -
2.3.2 Query baud rate.....	错误! 未定义书签。

# 1. Overview

## 1.1 Overview

This is wireless serial transceiver module. And the frequency is 2.4GHz. There is the MCU on board with built-in encoding and decoding code and it is easy to implement remote control without pairing. You can set the ID number, working frequency and baud rate of the module through AT commands, and the modules who have the same ID number and operating frequency can communicate directly with each other. It is very popular for remote control systems, such as wireless doorbell, remote control rolling gates, smart car, smart home, etc. It can also used to make Arduino wireless programmer.

- Working voltage: 3.2 - 5.2VDC
- Working Current: 21mA(MAX)
- Working Frequency: 2478MHz (default), 2400MHz - 2527MHz programmable
- Transmission distance: 50 meters (open area)
- Logic Interface Type: UART TTL serial port
- Communication format: 8N1
- AT command mode baud rate: 9600bps(Fixed value)
- Transparent mode baud rate: 9600 (default), can be modified to other values by command
- Operating mode: **Before power up**, when CMD pin is low, it is AT command mode and it can be used to set the parameters of the module; When CMD is high, it is transparent mode, and it can communicate with the other one which has the same ID number and working frequency.
- Antenna: Built-in PCB antenna
- Great for DIY

## 1.2 Pins Description

Pin Name	Decription
<b>VDD</b>	Power supply positive, voltage range: 3.2 - 5.2V
<b>GND</b>	Power supply ground
<b>TXD</b>	Serial transmit pin, should connect to RXD of MCU
<b>RXD</b>	Serial receive pin, should connect to TXD of MCU
<b>CMD</b>	Command control pin. <b>Before power up</b> , when CMD pin is low, it is AT command mode (Baud rate must be 9600bps) and it can be used to set the parameters of the module; When CMD is high, it is transparent mode, and it can communicate with the other one which has the same ID number and working frequency.

## 1.3 Note

- This module is a serial data transparent transmission module, the same module can receive, but also can send. When it receives the data from serial port, the module will send the data to the target module, when it completely sends out all the data, the module automatically enters the receiving state.
- Power up, onboard LED flashes once, indicating module starts.
- Each time a packet of data is received, the LED on the module flashes once.
- When sending a packet of data, if the transmission is successful, the LED on the module will flash once.
- Usually data packet length should be 1 ~ 32 bytes, the minimum transmission interval is 40ms.
- A packet of data length is limited to be 200, the module will automatically divide it into several packets.

## 2 AT Commands

### 2.1 Summary

- AT commands for this module do not have end character so that it just only recognizes the valid field, the extra characters are automatically masked.

Example: Set the frequency of the module to 2450MHz, you can send (AT + FREQ = 32), then even if sent (AT + FREQ = 32ABCD) is also possible, the command is not like the GPRS module, which needs to add a new line at the end of the command to determine whether it is the correct.

- All instructions except for numbers and symbols, only recognize capital letters. That is, if the letter contains the words, only can including capital letters, such as AT + BAUD ?, AT + FREQ = 3E

- When you send commands to set the parameters, it will return only 'Y' or 'N', so it is easy to know whether the command is sent successfully.

## 2.2 AT Commands List

Function	Command	Function	Command
<b>Set Baud Rate</b>	AT+BAUD=*	Query baud rate	AT+BAUD?
<b>Set Frequency</b>	AT+FREQ=**	Query frequency	AT+FREQ?
<b>Set number of retransmissions</b>	AT+RETRY=**	Query number of retransmissions	AT+RETRY?
<b>Set whether to return</b>	AT+BACK=*	Query whether to return	AT+BACK?
<b>Set TX ID</b>	AT+TID=*...*	Query TX ID	AT+TID?
<b>Set RX ID</b>	AT+RID=*...*	Query RX ID	AT+RID?
<b>Set Transmit Power</b>	AT+POWER=**	Query Transmit Power	AT+POWER?
<b>Restore Defaults</b>	AT+RESET	Query firmware version	AT+INF

## 2.3 Set Baud Rate and Query baud rate

### 2.3.1 Set Baud Rate

**Instruction format: AT+BAUD=\***

**Parameters \* Range: 0 ~ 9**

Through this instruction, the module can be configured with different data transmission baud rate to suit different situations.

Set the different baud rate as shown in the table below. If the setting is successful after sending the AT command, the module returns the character 'Y', and if the setting fails, the character 'N' is returned.

Baud Rate	AT Command	Baud Rate	AT Command
2400	AT+BAUD=0	38400	AT+BAUD=5
4800	AT+BAUD=1	57600	AT+BAUD=6
9600	AT+BAUD=2	115200	AT+BAUD=7
14400	AT+BAUD=3	128000	AT+BAUD=8
19200	AT+BAUD=4	256000	AT+BAUD=9

*Program macro definition:*

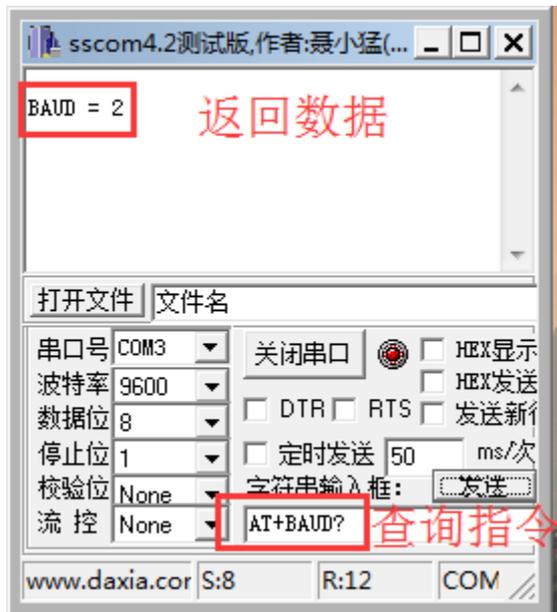
```
#define UART_BAUD_2400 '0'  
#define UART_BAUD_4800 '1'  
#define UART_BAUD_9600 '2'  
#define UART_BAUD_14400 '3'  
#define UART_BAUD_19200 '4'  
#define UART_BAUD_38400 '5'  
#define UART_BAUD_57600 '6'  
#define UART_BAUD_115200 '7'  
#define UART_BAUD_128000 '8'  
#define UART_BAUD_256000 '9'
```

## 2.3.2 Query baud rate

### Instruction format: AT+BAUD?

Return data: BAUD=2 corresponds to the baud rate is 9600

Return data: BAUD=7 corresponds to the baud rate is 115200



## 2.4 Set Frequency and Query frequency

### 2.4.1 Set Frequency

**Instruction format:** *AT+FREQ=\*\**

**Parameters \*\*:** *0x00 ~ 0x7F*

**(*AT+FREQ=00 ~ AT+FREQ=7F*)**

In order to set the offset frequency with other 2.4G radios such as WIFI, the module supports setting different transmission frequencies (Note: WIFI frequency is 2412 2417 2422 2407 2432 2437 2442 2452 2457 2462 2467 2472 2484). When two modules transmit data, they need to set the same frequency to match, otherwise they can not communicate.

If the setting is successful after sending the AT command, the module returns the character 'Y', and if the setting fails, the character 'N' is returned.

AT + FREQ = \*\*, \*\* range of the value can be 0x00 ~ 0x7F any value, the actual set frequency:

$F = (2400 + **) \text{ MHz}$

For example:

How to set the frequency of the module to be 2437MHz?

Calculate (2437-2400 = 37), hexadecimal of 37 is 0x25, so the command is

*AT+FREQ=25*

How to set the frequency of the module to be 2463MHz?

Calculate (2463-2400 = 63), hexadecimal of 63 is 0x3f, so the command is

*AT+FREQ=3F*

### 2.4.2 Query frequency

**Instruction format:** *AT+FREQ?*

Send the command to get the current frequency of the module, the calculation method corresponding to setting frequency principle.

## 2.5 Set number of retransmissions and Query number of retransmissions

### 2.5.1 Set number of retransmissions

**Instruction format:** `AT+RETRY=**`

**Parameters \*\*:** `0x02 ~ 0x0F`, the number of retransmissions is twice at least, up to 15 times.  
(`AT+RETRY=02 ~ AT+RETRY=0F`)

Set the number of repeated transmission of data per frame, set the higher, the more reliable data transmission, but the data will be delayed longer. When you use it as a remote control, try to set a larger value to ensure reliability.

When module sends data successfully, it does not resend again, even if it is set to 10 times in the transmission process, it may only transfer 3 times succeeded. The retransmissions is the maximum number of retransmissions **after a failed transmission**.

If the setting is successful after sending the AT command, the module returns the character 'Y', and if the setting fails, the character 'N' is returned.

For example:

How to set the number of retransmissions of the module to 10 times?

Hexadecimal of 10 is 0x0A, so the command is

`AT+RETRY=0A`

### 2.5.2 Query number of retransmissions

**Instruction format:** `AT+RETRY?`

Return data: `RETRY=5` corresponds to the retransmissions is 5 times

Return data: `RETRY=0D` corresponds to the retransmissions is 13 times

## 2.6 Set whether to return and Query whether to return

### 2.6.1 Set whether to return

**Instruction format: AT+BACK=\***

**Parameters \*: 0 is not return, 1 is return**

**(AT+BACK=0 AT+BACK=1)**

This instruction is to set to configure whether the serial module reply for each time to send data, so if you set enable the return function, each time send data successfully, the module returns back 'S' which is SUCCESS and 'F' is FAILURE.

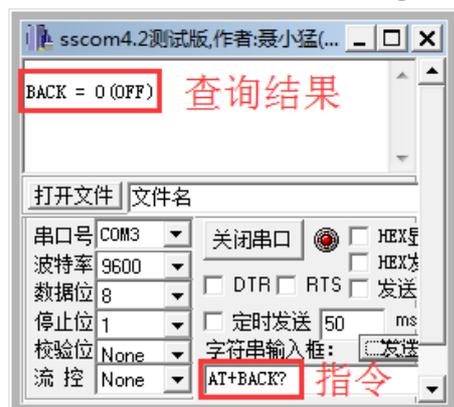
If the setting is successful after sending the AT command, the module returns the character 'Y', and if the setting fails, the character 'N' is returned.

### 2.6.2 Query whether to return

**Instruction format: AT+BACK?**

Return data: **BACK=0(OFF)** corresponds to return function is disable

Return data: **BACK=1(ON)** corresponds to return function is enable



## 2.7 Set TX ID and Query TX ID

### 2.7.1 Set TX ID

**Instruction format:** AT+TID=\*\*\*\*\*

**Parameters \* Range:** 0x000000000 ~ 0xFFFFFFFF, 5bytes

(AT+TID=000000000 ~ AT+TID=FFFFFFFF)

The transmission ID is the identifier of the radio data. In the case of the same frequency, only when the transmission ID and reception ID are the same, then can communicate with each other. That is to say all the modules whose reception ID are the same with the transmission ID of one module can receive the data it sends at the same time.

Transmission ID has a total of 5 bytes.

The transmission ID and reception ID of the same module can be the same or different.

If the setting is successful after sending the AT command, the module returns the character 'Y', and if the setting fails, the character 'N' is returned.

For example:

Set the sending ID of the module to be 0x1234567890:

AT+TID=1234567890

### 2.7.2 Query TX ID

**Instruction format:** AT+ TID?

Return data: TID = 0x555555555 corresponds to the transmission ID is 0x555555555



## 2.8 Set RX ID and Query RX ID

### 2.8.1 Set RX ID

**Instruction format:** AT+RID=\*\*\*\*\*

**Parameters \* Range:** 0x000000000 ~ 0xFFFFFFFF, 5bytes

**(AT+RID=000000000 ~ AT+RID=FFFFFFFF)**

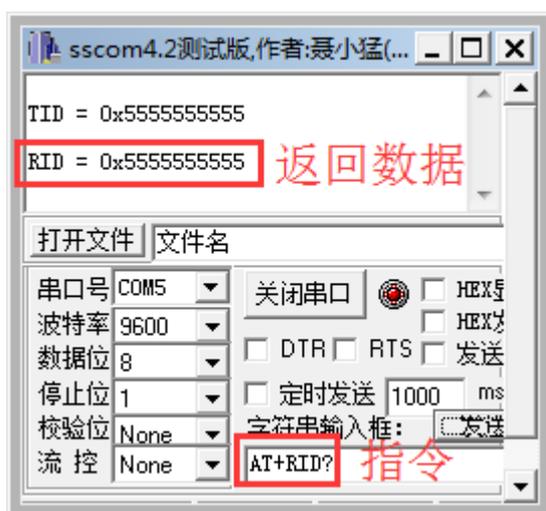
It is the same as TX ID setting. Please refer to chapter 2.7.1.

### 2.8.2 Query RX ID

**Instruction format:** AT+RID?

Return data: RID = 0x555555555 corresponds to the reception ID is 0x555555555

Each module at the factory, is set the same reception ID and transmission ID, which is 0x555555555



## 2.9 Set Transmit Power and Query Transmit Power

### 2.9.1 Set Transmit Power

**Instruction format: AT+POWER=\*\***

**Parameters \* Range: 0x00~0x0D (AT+POWER=00 ~ AT+POWER=0D)**

This command can be set up to 14 kinds of transmission power, each transmission power, the module consumes the current is not the same, the greater the transmission power, the greater the current consumption, otherwise smaller.

If the setting is successful after sending the AT command, the module returns the character 'Y', and if the setting fails, the character 'N' is returned.

AT Command	Power	AT Command	Power
<b>AT+POWER=00</b>	+13dBm	<b>AT+POWER=07</b>	0dBm
<b>AT+POWER=01</b>	+10dBm	<b>AT+POWER=08</b>	-3dBm
<b>AT+POWER=02</b>	+8dBm	<b>AT+POWER=09</b>	-6dBm
<b>AT+POWER=03</b>	+7dBm	<b>AT+POWER=0A</b>	-10dBm
<b>AT+POWER=04</b>	+5dBm	<b>AT+POWER=0B</b>	-18dBm
<b>AT+POWER=05</b>	+4dBm	<b>AT+POWER=0C</b>	-30dBm
<b>AT+POWER=06</b>	+2dBm		

### 2.9.2 Query Transmit Power

**Instruction format: AT+POWER?**

Return data: *POWER = 00* corresponds to the Transmit Power is +13dBm

Module at the factory default is the maximum power, if the distance is not high, may be appropriate to reduce the transmission power.



## 2.10 Restore Defaults

**Instruction format: AT+RESET**

**Return:**

**Module reseted**

**BAUD = 9600bps**

**FREQ = 2478MHz**

**TID = 0x5555555555**

**RID = 0x5555555555**

**RETRY = 0x0A**

**BACK = 0(OFF)**

**ACK = 1(ON)**

**POWER = 0(+13dBm)**

This instruction restore factory settings. Factory settings of all modules are the same.

## 2.11 Query firmware version

**Instruction format:** AT+INF

**Return:** about the firmware version

## 3 Link

Documents download link:

<https://drive.google.com/drive/folders/0B6uNNXJ2z4CxLUIEMTQzRTNMU0U?usp=sharing>

**OPEN-SMART Official Store:**

<https://open-smart.aliexpress.com/store/1199788>

**OPEN-SMART Youtube:** we will update more tutorial to it

[https://www.youtube.com/channel/UCM\\_mzBFIDyMUnItB88c6LWQ](https://www.youtube.com/channel/UCM_mzBFIDyMUnItB88c6LWQ)

Technical Support: [catalex\\_inc@163.com](mailto:catalex_inc@163.com)