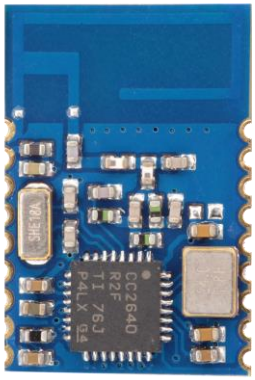
CC2640R2FRSM Bluetooth 4.2 Low Energy Master-Slave Module and Protocol

Version: V4.30u

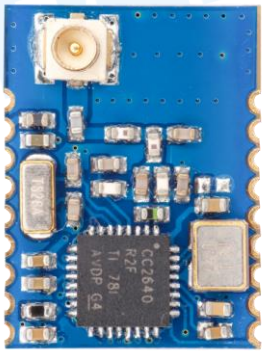
(Transparent Transmission)

Shenzhen RF-star Technology Co., Ltd.

Nov. 18th, 2019



RF-BM-4044B2



RF-BM-4044B3

RF-BM-4044B4

Table of Contents

Table of Figures

Table of Tables

1 Summary

The Bluetooth Low Energy module mentioned in this document can only work in bridge mode (transparent transmission mode).

After powered on, the module can broadcast automatically. Smart phone with specific APP running will scan and pair with it. After successful connection, the smart phone can monitor and control the module by Bluetooth protocol.

In bridge mode, user's CPU and the mobile device can communicate bi-directionally by the serial ports of module. Users can also manage and control certain communication parameters by specific serial port AT commands. The detailed definition of the user data is defined by the upper application. Mobile devices can write the module by APP. And the data written will be sent to the user CPU by serial ports. After the module received the data packet from user CPU, it will transmit it to the mobile devices automatically. Development based on this mode, the user needs to undertake the code design for master CPU and the APP for mobile devices.

2 Features

1. Easy to use, no need of any application experience of Bluetooth protocol stack.
2. Module with function of master-slave. Transmit commands to control the mode of master or slave. The module supports AT command to start to scan & connection and directed MAC connection.
3. UART design for user interface, full-duplex bi-directional communication, and the minimum baud rate of 4800 bps.
4. Default connection interval of 20 ms, which makes quick connection.
5. Through AT command, support module software reset and MAC address acquisition.
6. The serial data packet length can be any length below 1024 Byte (including 1024 Byte) (large packets are automatically distributed).
7. Through AT command, support the adjustment of Bluetooth connection interval and the control of different transmitting rates (dynamic power consumption adjustment).
8. Through AT command, support the adjustment of the transmit power and broadcast interval, the customization of broadcast data and product ID, and the modification of the serial port baud rate and module names.
9. High-speed transparent transmission rate is maximum to 6 K/s and the stable rate is 5.3 K/s.
10. Through APP, support the modification of module name, UART baud rate and product ID, support the customization of broadcast content and cycle.
11. Through APP, support the remote reset of module and the configuration of transmit power.
12. Through APP, support the adjustment of Bluetooth connection interval (dynamic power consumption adjustment).
13. Support the prompt pins of connection status and broadcasting status.
14. Support the remote recovery initialization of modules by mobile device APP.
15. Support string prompts of Bluetooth connection status from TX serial port (connection, normal disconnection and timeout disconnection).
16. Extremely low power in standby mode, sleeping current of 0.1 μA from TI official data for CC2640R2F, and the measured power consumption data is as follows:

Table 1. Power Consumption of SoC CC2640R2F

Event	Average Current (Silicon Labs EVB[1] by default)		Testing Condition
	EN Internal Pull-up Enabled	EN Internal Pull-up Disabled	
Sleeping	2.71 μA	2.71 μA	EN is in high level
Broadcasting	379.11 μA	160.54 μA	Broadcasting cycle: 200 ms (by default)
Broadcasting	243.56 μA	38.11 μA	Broadcasting cycle: 1000 ms
Broadcasting	224.44 μA	13.75 μA	Broadcasting cycle: 5000 ms

Connection	683.13 μA	457.13 μA	Connection interval: 20 ms
Connection	358.57 μA	188.72 μA	Connection interval: 100 ms

Notes:

1. Silicon Labs EVB test method: The Silicon Labs EVB is connected to the power in loop of module, open the official software Simplicity Studio to monitor the power waveform data, and automatically calculate the power consumption value.
2. Official test method: Connect a 10 Ω resistor in series in the power circuit, and get the voltage drop waveform with oscilloscope and conduct integrated computation.
3. Multi-meter test method: Connect a multi-meter (set at μA or mA level) in series between the battery and the module to check the value shown, with the test voltage of 3.3 V.

Above is the measured sampling data of module RF-BM-4044B2 and for reference only.

If lower power consumption is expected, connection interval or broadcast cycle can be appropriately increased, as shown in related chapters of “module parameter configuration” and “AT command”.

3 Schematic Diagram of Working Mode

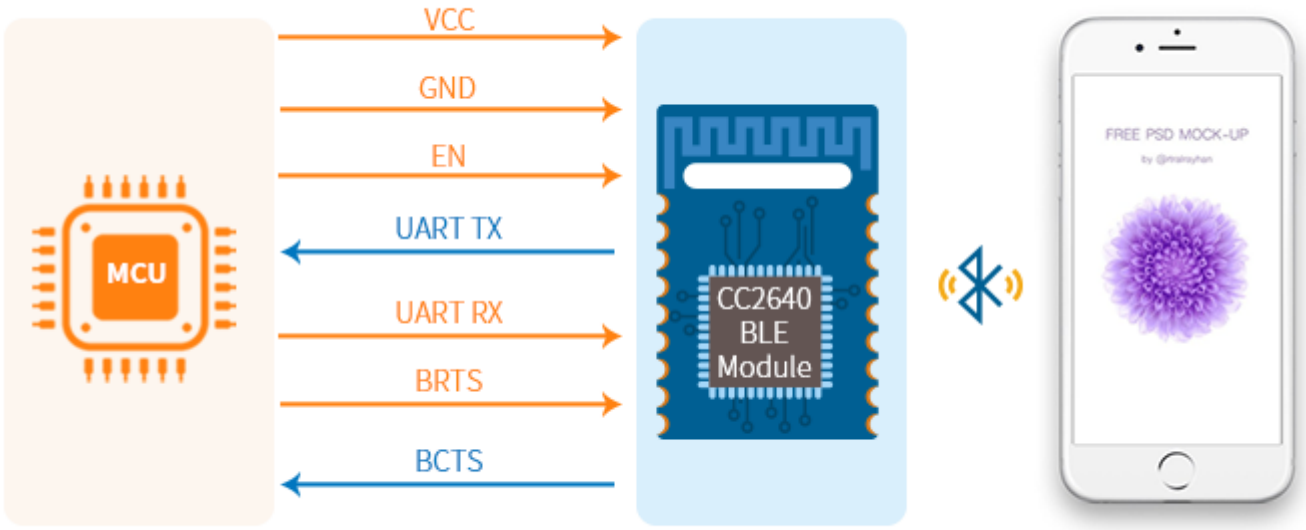


Figure 1. Schematic Diagram of Module (V4.30u) in Bridge Mode

Note:

In order to avoid the high current caused by the output level difference between user CPU IO and module IO, a small isolation resistor is suggested to be connected in series in the output signal line TX, BCTS.

4 Package and Pin Assignment

4.1 RF-BM-4044B2, RF-BM-4044B3

RF-BM-4044B2 is pin to pin compatible with and RF-BM-4044B3. The difference is that the antenna of RF-BM-4044B2 is PCB type, while RF-BM-4044B3 is IPEX type. And the dimension of antenna part of RF-BM-4044B2 is a little bit longer than RF-BM-4044B3.

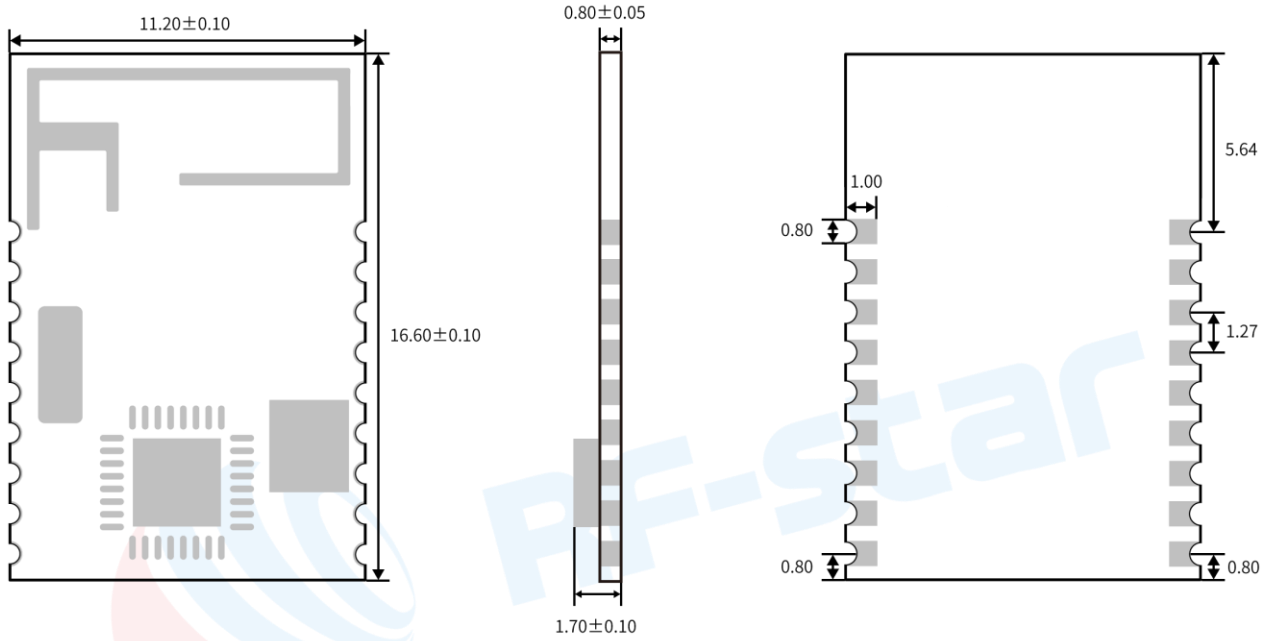


Figure 2. PCB Footprint of RF-BM-4044B2

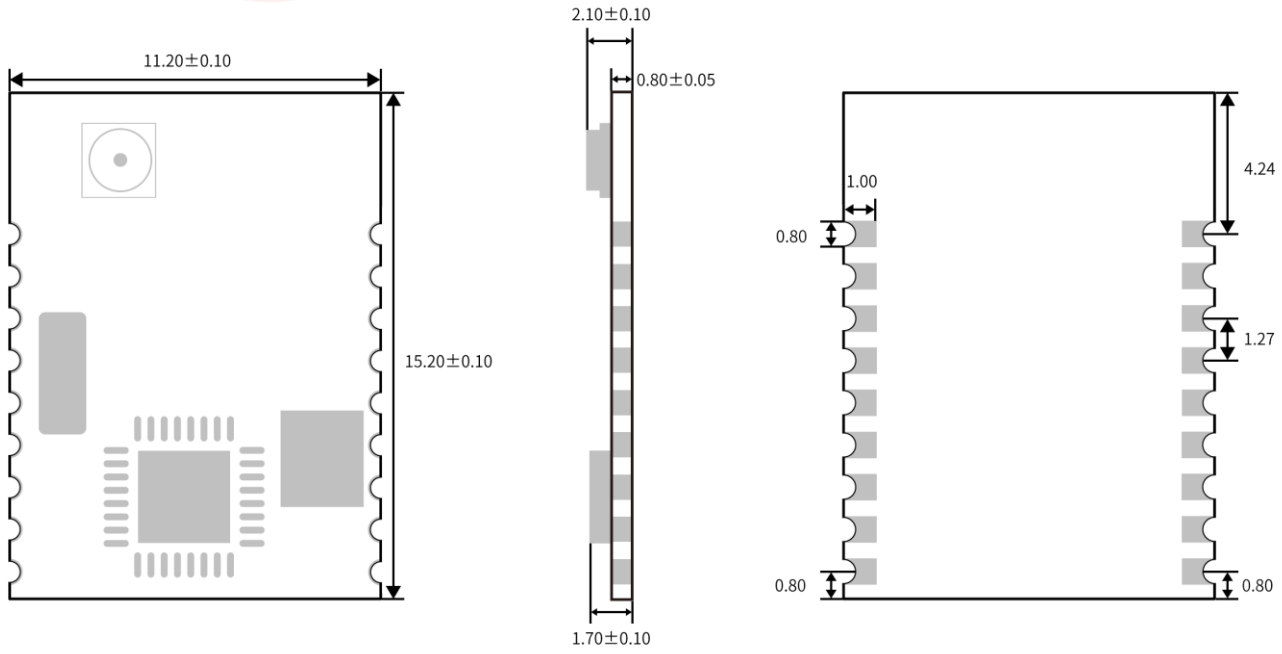


Figure 3. PCB Footprint of RF-BM-4044B3

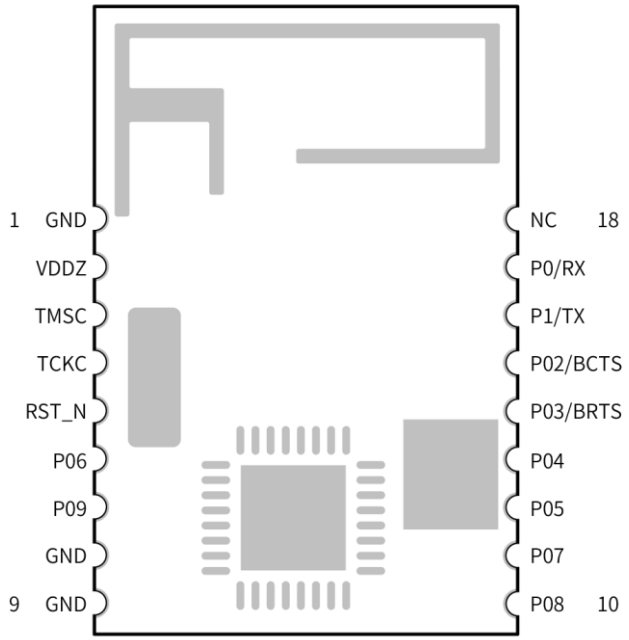


Figure 4. Pin Assignments of RF-BM-4044B2

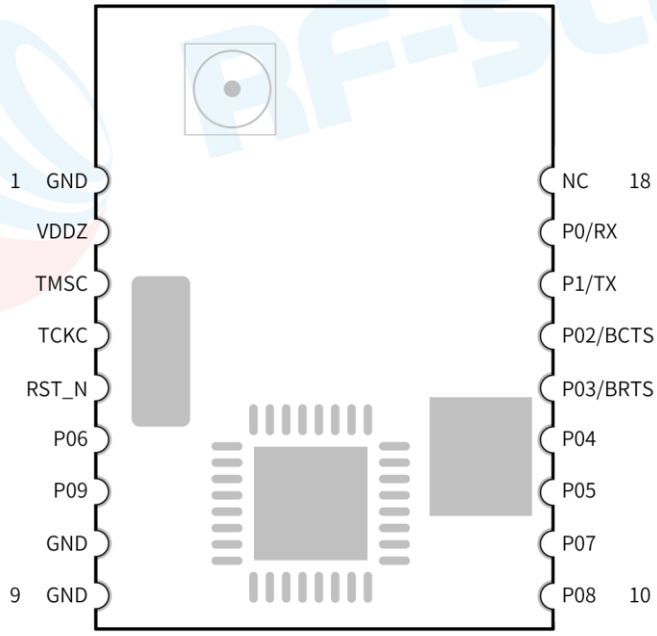


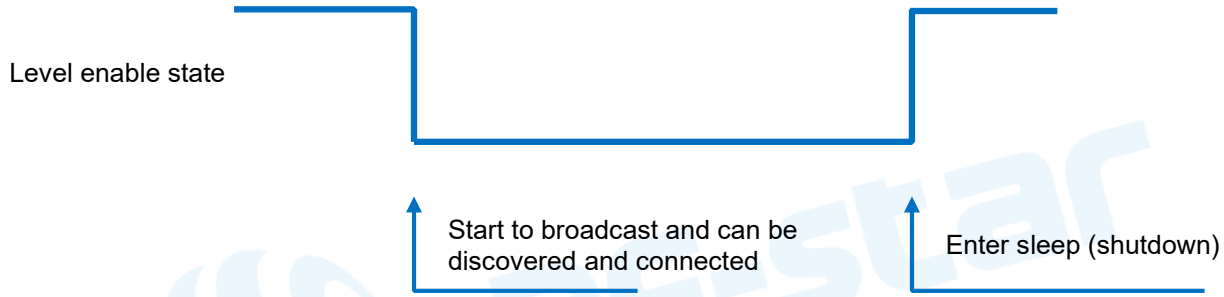
Figure 5. Pin Assignments of RF-BM-4044B3

Table 2. Pin Assignments of RF-BM-4044B2 and RF-BM-4044B3

Pin	Name	Chip Pin	I/O	Description
1	GND	-	-	Ground
2	VCC	-	-	Power supply 1.8 V ~ 3.8 V
3	TMS	-	I/O	JTAG: TMS
4	TCK	-	I/O	JTAG: TCK
5	RST	RST	-	RESET, active low, no internal pull-up
6	EN	P06	I	Module-enabled control circuit (active low) 0: Module starts to broadcast, until it connects to the mobile device. 1: Enter sleep mode immediately (0.1 μA), regardless of the current status.
7	-	P09	I/O	
8	GND	-	-	Ground
9	GND	-	-	Ground
10	-	P08	I/O	Factory reset trigger Keep this pin at low level for 5 s, the system can be partially reset (light recovery). If keep more than 20 s, the system can be completely reset (deep recovery). (See details in "System reset and recovery")
11	-	P07	I/O	
12	Link Indication	P05	O	Link indication 0: Bluetooth connected 1: Bluetooth disconnected
13	Broadcast Indication	P04	O	Broadcast indication 0: Turn-on broadcast 1: Turn-off broadcast
14	BRTS	P03	-	As the request of data sending (for module wake-up) 0: Master has data to send, and module will wait for data transmission from the master, so the module will not sleep. 1. Master has no data to send, or data has been sent. So, the value of the signal line should be set at "1".

15	BCTS	P02	O	Data input signal (for master wake-up, optional) 0: Module has data to send, and the master will receive the data from the module. 1: Module has no data to send, or data has been sent, and the value of the signal line should be set at “1”.
16	TX	P01	O	Serial port TX
17	RX	P00	I	Serial port RX
18	NC	-	-	None connect

Note 1:



In level-enabled mode, broadcast (can be discovered and connected) has the following features:

1. If EN pin (pin 6) is enabled (set low), the module will keep broadcasting, until it is connected or EN is set high.
2. Regularly disconnected or timeout disconnected, as long as EN is set low, the module will always keep broadcasting, until it is connected again.

Note 2:

In level-EN mode, P04 works as broadcast prompt pin, when the module starts broadcast, it will output low level; while the module turns off broadcast, it will output high level. P05 works as link indication pin, if Bluetooth is connected, it will output low level. If Bluetooth is disconnected (either timeout or active disconnecting by APP) and not re-connected and Bluetooth keeps disconnected status, it will output high level.

Table 3. P04 & P05 Prompts in Level-EN

Module Status	**Enabled but Not Connected**		**Connected**		**Normal Disconnected**		**Timeout Disconnected**	
	P04 Prompt Mode	P05 Prompt Mode	P04 Prompt Mode	P05 Prompt Mode	P04 Prompt Mode	P05 Prompt Mode	P04 Prompt Mode	P05 Prompt Mode
Level-EN Mode	Low level Broadcast	High level Disconnected	High level No broadcast	Low level Connected	Low level Broadcast	High level Disconnected	Low level Broadcast	High level Disconnected

4.2 RF-BM-4044B4

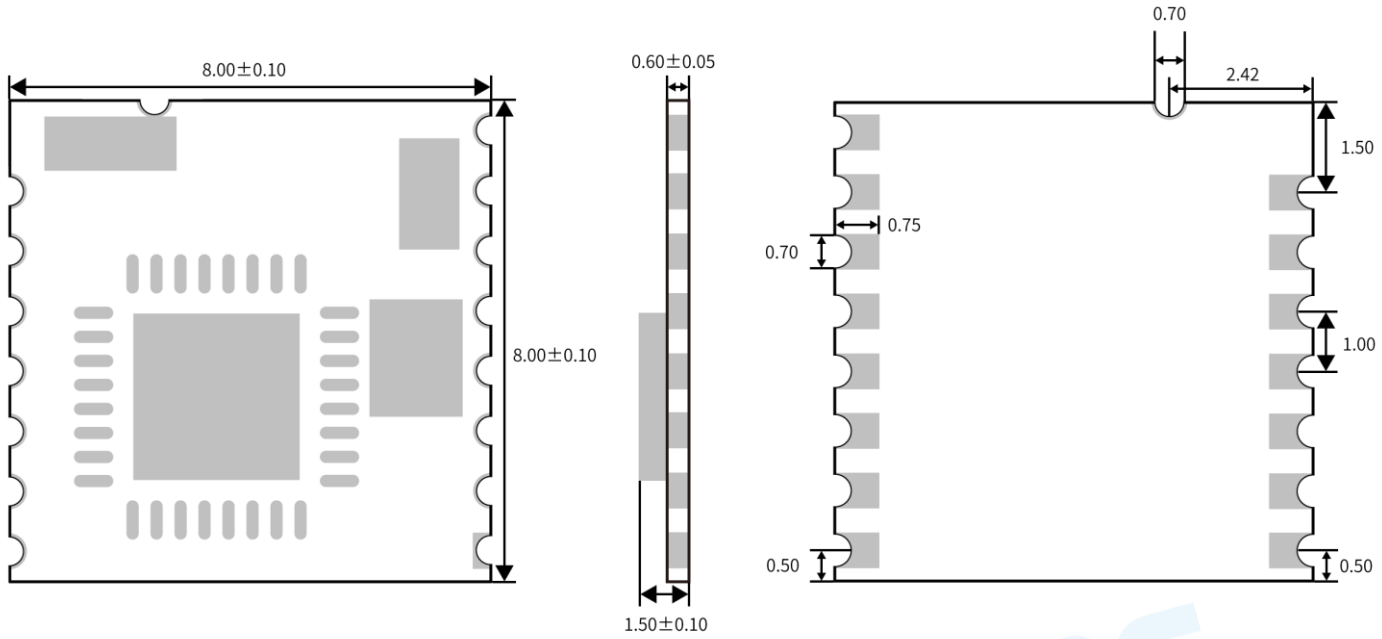


Figure 6. PCB Footprint of RF-BM-4044B4

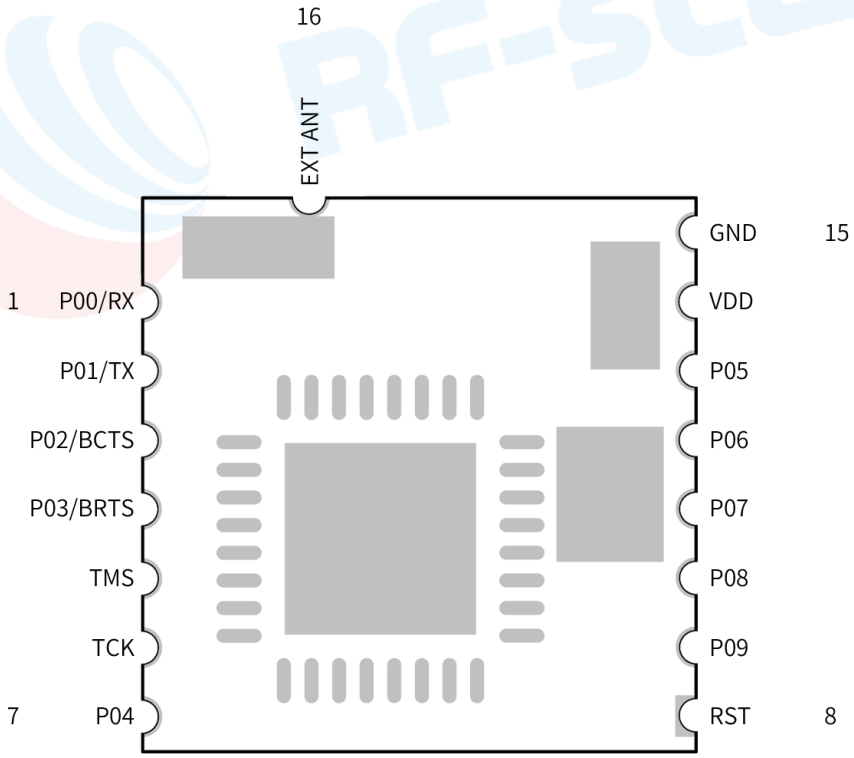


Figure 7. Pin Assignments of RF-BM-4044B4

Table 4. Pin Assignments of RF-BM-4044B4

Pin	Name	Chip Pin	I/O	Description
1	RX	P00	I	Serial port RX
2	TX	P01	O	Serial port TX
3	BCTS	P02	O	Data input signal (for master wake-up, optional) 0: Module has data to send, and the master will receive the data from the module. 1: Module has no data to send, or data has been sent, and the value of the signal line should be set at “1”.
4	BRTS	P03	I	As the request of data sending (for module wake-up) 0: Master has data to send, and module will wait for data transmission from the master, so the module will not sleep. 1. Master has no data to send, or data has been sent. So, the value of the signal line should be set at “1”.
5	TMS	-	I/O	JTAG: TMS
6	TCK	-	I/O	JTAG: TCK
7	P04	P04	O	Broadcast indication 0: Turn-on broadcast 1: Turn-off broadcast
8	RES	-	-	Reset pin, active low, no internal pull-up.
9	-	P09	I/O	
10	-	P08	I/O	Factory reset trigger Keep this pin at low level for 5 s, the system can be partially reset (light recovery). If keep more than 20 s, the system can be completely reset (deep recovery). (See details in “System reset and recovery”)
11	-	P07	I/O	
12	EN	P06	I	Module-enabled control circuit (active low) 0: Module starts to broadcast, until it connects to the mobile device. 1: Enter sleep mode immediately (0.1 μA), regardless of the current status.
13	Link	P05	O	Link indication

	indication			0: Bluetooth connected 1: Bluetooth disconnected
14	VCC	-	-	Power supply: 1.8 V ~ 3.8 V
15	GND	-	-	Ground
16	EXT ANT	-	-	External antenna output

5 UART Transparent Transmission Protocol

The bridge mode means to set up a bi-directional communication way between user CPU and mobile devices by the mutual connection between serial port and user CPU. Users can reset serial port baud rate and BLE connection interval by the specified AT commands (see the section "AT Command"). The module will have different data handling capability, as per different serial port baud rates and BLE connection intervals. The default baud rate is set at 115200 bps.

When the BLE connection interval is 20 ms and the serial port baud rate is at 115200 bps, the module has the highest transmit ability in theory (6.7 K/s). Given the configuration in the level-enabled mode as an example, UART transparent transmission protocol will be detailed introduced as below.

The module can customize the serial port packet according to the obtained MTU of the connected device. The module will automatically sub-pack and send packets according to the packet size. The maximum payload of each packet is 3 Bytes smaller than MTU (for example: the module and iPhone are connected, the obtained MTU is 185, so, the data send to iPhone by the module is with a maximum of 182 Bytes per packet). Packets sent to the module by the mobile device must be sent after sub-packed (3 Bytes/packet will be loss after one pass to MTU). After receiving the packet, the module will transmit it to the serial port receiving end of the host.

1. The hardware protocol of serial port: 115200 bps, 8, no parity, 1 stop bit.
2. When EN is set at high level, the Bluetooth module is in full sleep mode. When EN is set low, the module will start broadcasting at the internal of 200 ms, until it pairs with mobile devices. When EN jumps from low to high, the module will enter sleep mode immediately, regardless of current status.
3. After the module is connected, BRTS needs to be pulled low if the master (MCU) has data to send to the BLE module, and the data transmission can be started around 50 ms afterwards. BRTS should be pulled high by the master after transmission finished and make the module exit the serial RX mode. Pay attention to confirming that the data transmission has been completely finished before BRTS pulled-high. Otherwise there will be data truncation.
4. When there is data upload request, the module will set BCTS low and start to transmit, until data transmission finishes. BCTS will be set high by the module when data transmission is finished.
5. If the mater BRTS is being kept a low level, the Bluetooth module will always be in serial port receiving mode and the power consumption will be high.
6. After the module is connected, a string of "TTM:CONNECTED" will be prompted from TX. The string could be used to determine if the normal transmit operation can be done. Of course, the connection status prompt pin can be used instead. Also, the connection can be confirmed by sending a specific confirmation string to the module from mobile devices, when APP actively disconnect the module, there will be a prompt of string "TTM:DISCONNET" from TX. If the disconnection is abnormal, the string prompt will be "TTM:DISCONNET FOR TIMEOUT".

7. The default Bluetooth connection interval is 20 ms. If low-speed TX mode is needed for saving power, connection interval must be adjusted by AT command (longest connection interval: 2000 ms). In each interval, the maximum transmission is 248 Bytes. Set the connection interval as T (unit: ms), and highest transmit rate per second as V (Byte/s), then their relation is as follows:

   V = 248 * 1000 / T (V is only relevant with T)

   If the Bluetooth connection interval of the module is 10 ms, and in each interval it can transmit maximum 248 Bytes, the theoretical maximum transmission capacity (transmit rate) will be 248 * 1000 / 10 = 24.8K Byte/s. Tests have shown that the packet loss is very little when transmit rate under 16 K/s. For safety's sake, it is suggested to do check-sum and re-transmission processing in the up-layer, no matter for high or low speed transmit applications.

   Note: The MTU of Android is 251 Bytes, the iOS is 185 Bytes, and the size of each packet when sending is MTU-3 Bytes.

8. The size of the serial port data packets can be various and the length can be any value less than 1K Bytes, as long as the above conditions are met. But in order to utilize the communications payload in highest efficiency, while to avoid communication running in full capacity, it is recommended to use serial port data packets of 20, 80, or 248 Bytes in length, and interval between packets is made more than 20 ms.

Note:

Test shows that in iOS, the writing function of Characteristic with the parameter CBCharacteristicWriteWithResponse (writing mode with response) will reduce partially the transmission efficiency, but the correctness of a single packet will be ensured. While with the parameter CBCharacteristicWriteWithoutResponse (writing mode without response), the transmission efficiency will be increased, but the correctness of data packet needs to be checked by APP in upper layer.

6 AT Command

Strings starting with "TTM" will be regarded as AT commands to be parsed and executed, and will return exactly the same from the serial port. Afterwards the execution result will be output ("TTM:OK\r\n\0"or "TTM:ERP\r\n\0", etc.). **Data packets which do not start with "TTM" will be regarded as transparent transmission data.**

- **AT Command List**

- **AT Command List under Master-Slave Connection**

Table 5. AT Command List under Master-Slave Connection

AT Command	Saved After Power-off	Parameter and Description	Possible Response	Remarks
TTM:SCA-?	-	The module will scan the surrounding slave devices as a master, and will put the device information into the scanned slave devices list. (the module will turn off broadcast and enter master mode.)	TTM:OK\r\n\0	Effective Command
			TTM:ERP\r\n\0	Invalid Command
			Device:A Mac:B..... Rssi:C.... Adv:DDD....." For example: Device:1 Mac:0xBA036545C2A1 Rssi:-69 Adv:0201060502F0FFB0FF11FF52531851BA036545C2A1050500010000）	Wherein "A" means the number in the scanned device list, "B" means the MAC address of this device. "C" means its RSSI. "D" means its broadcast data. "E" means the scanned device number.
TTM:CON-x	-	Connect the specified slave module. Wherein x (x = 0 ~ 15) means the number in the scanned device list.	TTM:TCONNECTED MTU1 Size: X TTM:FINISH	X means the size of MTU of successful connected device.

TTM:CON-xxxxxxxxxxxx	-	Directionally connect slave device. Wherein x (x = 12 bit MAC address) means the MAC address of slave module.	TTM:TCONNECTED MTU1 Size: X TTM:FINISH	X means the size of MTU of successful connected device.
TTM:ROL-?	-	Inquiry the module role	TTM:ROL-PERIPHERAL	Module is in slave mode.
			TTM:ROL-CENTRAL	Module is in master mode.
TTM:SSC	-	Exit the master mode.	TTM:OK\r\n\0	Module exits the master mode and back to slave mode.

Note: “TTM:SCA-?” is sent to make the module turn off broadcast and enter master mode, while “TTM:SSC” make the module exit master mode.

- **AT Command List of Basic Parameters**

Table 6. AT Command List of Basic Parameters

AT Command	Saved After Power-off	Parameter and Description	Possible Response	Remarks
TTM:CIT-X ms (Valid only during successful connection)	No	Set BLE connection interval (in ms) X= **"20"**, ”30”, "50", "100", "200", "300", "400", "500", "1000", "1500", "2000".	TTM:TIMEOUT TTM:OK TTM:ERP	Timeout configuration. Successful operation. Invalid command.
TTM:REN-?	Yes	Acquire module name	TTM:NAM-xxxxxxxxxxxx\r\n\0 “xxxxxxxxxxxx” is the name of module.	Module name is returned.
TTM:REN-+Name	Yes	Set module name. “Name” is the new module name with any string of no less than 16-bit length.	TTM:OK TTM:ERP	Successful operation. Invalid command.
TTM:BPS-?	-	Acquire baud rate	TTM:BPS-X “X” is the baud rate	Baud rate is returned.
TTM:BPS-X	Yes	Set baud rate. X= 4800", "9600", "19200", "38400", "57600", **"115200"**.	TTM:BPS SET AFTER 2S ..." TTM:ERP\r\n\0	Successful operation with new baud rate in 2 s. Incorrect command format.
TTM:MAC-?	-	Acquire MAC address	TTM:MAC-xxxxxxxxxxxx “xxxxxxxxxxxx” is module MAC address	MAC address is returned.
TTM:MAC-X	Yes	Set MAC address. X is 1 12-bit MAC string, for example: 123456789ABC. (the reset of MAC address is only effective by reset after power-on.)	TTM:OK\r\n\0 TTM:ERP\r\n\0	Successful operation. Invalid command.
TTM:VER-?	-	Acquire version number.	TTM:VER-XXXXXX,	Version number is returned.

			“XXXXXX” means the version number.	
TTM:PWD-?	-	Acquire connection password.	TTM:PWD-xxxxxx, “xxxxxx” means the connection password.	Password is returned.
TTM-PWD-xxxxxx	Yes	Set connection password.	TTM:OK-xxxxxx\r\n\0 TTM:ERP\r\n\0	Successful operation. Invalid command.
TTM:RST-SYSTEMRESET	-	System reset.	Module is working!	Reset module
TTM-RST-RSTPWD	-	Light recovery.	Module is working!	Recovery password.
TTM-RST-RESET	-	Deep recovery.	Module is working!	Recovery all parameters.
TTM:EUP-ON	No	Enable pull-up of EN pin.	TTM:OK\r\n\0 TTM:ERP\r\n\0	Successful operation. Invalid command.
TTM:EUP-OFF	No	Disable pull-up of EN pin.	TTM:OK\r\n\0 TTM:ERP\r\n\0	Successful operation. Invalid command.
TTM:ADP-(X)	Yes	Set broadcast cycle. T = X* 100 ms. X = "20", "50", "100", "150", **"200"**, "500", “1000”, “1500”, “2000”, “2500”, “3000”, “4000”, “5000”	TTM:OK TTM:ERP	Successful operation. Invalid command.
TTM:ADD-+Data	Yes	Set customized broadcast content. “Data” for customized broadcast data, and length L < = 16	TTM:OK TTM:ERP	Successful operation. Invalid command.
TTM:PID-+Data	Yes	Set customized product ID. Data is the customized product ID. Data length = 2. **5246** is by default.	TTM:OK TTM:ERP	Successful operation. Invalid command.
TTM:TPL-(X)"	No	Set transmit power (in dBm).	TTM:OK	Successful operation.

		X = **"2"**, “1”, "0", “-3”, “-6”, “-9”, “-12”, “-15”, “-18”, “21.	TTM:ERP	Invalid command.
TTM:CDL-Xms	Yes	Set data delay. X = **"0"**, "2", "5", "10", "15", "20", or "25"	TTM:OK\r\n\0 TTM:ERP\r\n\0	Successful operation. Invalid command.

Note: Word in bold blue is by default.

- **Scan Slave Device (Enter Master Mode)**

Input the following string "TTM:SCA-?" to RX to scan the slave devices:

The device information in the broadcast device list which has a maximum of 8. The scanning duration is 8 second.

After the command is executed, the following confirmation will be got from TX:

“Device:A MAC:B… RSSI:C… Adv:DDD…”

……

“Scan End, Found Devices:E”

Wherein “A” means the number in the scanned device list, “B” means the MAC address of this device, “C” means its RSSI, “D” means its broadcast data, and “E” means the scanned device number.

- **Connect Specified Slave Device (Master Mode)**

Input the following string "TTM:CON-x" to RX to connect slave device:

Wherein x (x = 0 ~ 15) means the number in the scanned device list

After the command is executed, the following confirmation will be got from TX:

“TTM:TCONNECTED”

“MTU1 Size: X”

“TTM:FINISH”

X means the size of MTU of successful connected device.

- **Directionally Connect Slave Device (Master Mode)**

Input the following string "TTM:CON-x" to RX to connect slave device:

Wherein x (x = 12 bit MAC address) means the MAC address of the specified slave module.

After the command is executed, the following confirmation will be got from TX:

“TTM:TCONNECTED”

“MTU1 Size: X”

“TTM:FINISH”

X means the size of MTU of successful connected device.

- **Acquire Module Role (Master Mode)**

Input the following string "TTM:ROL-?" to RX to inquiry the module role:

Situation 1: After powered on, check the module role.

After the command is executed, the following confirmation will be got from TX:

“TTM:ROL-PERIPHERAL” means the module is worked as a slave.

Situation 2: After the scan command “TTM:SCA-?” is sent, the module will turn off broadcast, enter master mode and check the module role.

After the command is executed, the following confirmation will be got from TX:

“TTM:ROL-CENTRAL” means the module is worked as a master.

Situation 3: When the module asked for connection and get successful connection, check the module role.

After the command is executed, the following confirmation will be got from TX:

“TTM:ROL-CENTRAL” means the module is worked as a master.

- **Exit Master Mode (Enter Slave Mode)**

Input the following string "TTM:SSC" to RX to exit master mode:

When the module is worked in master mode (no broadcast for now), and it needs to change to slave mode which can broadcast, the module can send “TTM:SSC” to exit master mode and back into slave mode.

After the command is executed, the following confirmation will be got from TX:

"TTM:OK\r\n\0" means the module is enter slave mode.

Note: This command is only effective under master mode, once the command is sent, the module will enter slave mode.

- **Connection Interval Configuration**

Input the following string "TTM:CIT-Xms" to RX to set the BLE connection interval, wherein X = **"20"**, "30", “50”, "100", "200", "300", "400", "500", "1000", "1500", or "2000" (in ms) (all data format is in ASCII code).

For example: “TTM:CIT-30ms” means the BLE connection interval is set as 30 ms.

After the command is executed, the following confirmation will be got from TX:

"TTM:TIMEOUT": It means timeout and the failed modification.

"TTM:OK": It means the operation is successful and the new connection interval is applied.

The success of connection interval configuration depends on the limit of connection intervals by mobile devices. The maximum connection intervals are varied in different version.

Note: The modification of AT command is only effective when the connection is successful.

- **Acquire Module Name**

Input the following string "TTM:NAM-?" to RX to acquire the module name.

After the command is executed, the following confirmation will be got from TX:

“TTM:NAM-xxxxxxxxxxxx\r\n\0”, and “xxxxxxxxxxx” is the module name.

- **Module Rename**

Input the following string "TTM:REN-" + Name to RX to rename the module (length of name should not exceed 16 Bytes in ASCII code format).

For example: “TTM:REN-ABC123” means the module is renamed as “ABC123”.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK": It means the successful operation.

"TTM:ERP": It means the incorrect command format.

Test shows that the device name cannot be modified immediately in iOS, but can in Android. The user can set it through the PC, or through the BLE APP interface of the mobile device. See details in “Module Parameter Configuration”.

- **Acquire Baud Rate**

Input the following string "TTM:BPS-?" to RX to acquire the baud rate.

After the command is executed, the following confirmation will be got from TX:

“TTM:BPS-X”, and “X” is the baud rate. Wherein X = "4800", "9600", "19200", "38400", "57600", **"115200"** (all data

format is in ASCII code).

- **Baud Rate Configuration**

Input the following string "TTM:BPS-X" to RX to set the baud rate. Wherein X = "4800", "9600", "19200", "38400", "57600", **"115200"** (all data format is in ASCII code).

For example: “TTM:BPS-115200” means the baud rate is 115200 bps.

After the command is executed, the following confirmation will be got from TX:

“TTM:BPS SET AFTER 2S…”: It means the modification is successful.

"TTM:ERP\r\n\0": It means the incorrect command format.

- **Acquire MAC Address**

Input the following string "TTM:MAC-?" to RX.

After the command is executed, the following confirmation will be got from TX:

“TTM:MAC-xxxxxxxxxxxx", and "xxxxxxxxxxxx" is the Bluetooth module address in 6 Bytes.

- **MAC Address Configuration**

Input the following string "TTM:MAC-xxxxxxxxxxxx" to RX, wherein “xxxxxxxxxxxx” is the Bluetooth module address in 6 Bytes.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK\r\n\0": It means the successful operation.

"TTM:ERP\r\n\0": It means the incorrect command format.

This command can be saved after power-off. when reset, the module will operate with the new MAC address.

- **Version Number Acquisition**

Input the following string "TTM:VER-?" to RX to acquire module version number.

“TTM:VER-XXXXXX": It means the successful operation and XXXXXX is the module version number.

"TTM:ERP\r\n\0": It means the incorrect command format.

- **Connection Password Acquisition**

Input the following string "TTM:PWD-?" to RX to acquire connection password.

“TTM:PWD-XXXXXX": It means the successful operation and XXXXXX is the password in 6-bit.

"TTM:ERP\r\n\0": It means the incorrect command format.

- **Connection Password Configuration**

Input the following string "TTM:PWD-?" to RX to acquire connection password.

“TTM:OK-xxxxxx\r\n\0": It means the successful operation and XXXXXX is the password in 6-bit.

"TTM:ERP\r\n\0": It means the incorrect command format.

- **Module Reset**

Input the following string "TTM:RST-SYSTEMRESET" to RX to force the module system reset once.

After the command is executed, the following confirmation will be got from TX:

“Module is working!": It means the successful operation.

- **Module Reset - Light Recovery**

Input the following string "TTM:RST-RSTPWD" to RX to force the module light reset once and recovery the password parameters.

After the command is executed, the following confirmation will be got from TX:

“Module is working!": It means the successful operation.

- **Module Reset - Deep Recovery**

Input the following string "TTM:RST-RESET" to RX to force the module deep reset once and recovery all the modified parameters, which means the module factory reset.

After the command is executed, the following confirmation will be got from TX:

“Module is working!": It means the successful operation.

- **Internal Enable**

Input the following string "TTM:EUP-ON\r\n\0" to RX to enable internal pull-up of EN pin which is the default configuration. Input the following string "TTM:EUP-OFF\r\n\0" to RX to disable internal pull-up of EN pin. When broadcast during disabled internal pull-up, more than 120 μA current.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK\r\n\0": It means the successful operation.

"TTM:ERP\r\n\0": It means the incorrect command format.

Note: This configuration cannot be saved after power-off. The default configuration is internal pull-up enabled.

- **Broadcast Cycle Configuration**

Input the following string "TTM:ADP-(X)" to RX to set the broadcast cycle of the module.

Wherein X = "20", "50", "100",**"200",** "500", "1000", "1500", "2000", "2500", "3000", "4000", or "5000" (all data format is in ASCII code).

For example: “TTM:ADP-(200)” means the broadcast cycle is 200 ms.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK": It means the successful operation.

"TTM:ERP": It means the incorrect command format.

This command can be saved after power-off. when reset, the module will operate with new broadcast cycle.

- **Add Customized Broadcast Packet**

Input the following string "TTM:ADD-"+ Data to RX to customize broadcast packet. Wherein “Data” is the additional data ready to be broadcast (0 < Length < = 16 Bytes) (all data format is in ASCII code).

For example: “TTM:ADD-Advertisement!”.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK": It means the successful operation.

"TTM:ERP": It means the incorrect command format.

This AT command is effective after configuration, and realize some customized broadcast packets. The data can be saved after power-off. If 16-Byte data are set all set as 0, customized broadcast packets will not be used. Instead, the default broadcast contents are applied.

- **Product ID Definition**

Input the following string "TTM:PID-"+ Data to RX to define product ID, wherein “Data” is a 2-Byte product ID with the range from 0x0000 range to 0xFFFF (L = 2) (all data format is in ASCII code).

For example: “TTM:PID-RS” means the product ID is RS and RS is equal to 0x5253 in hexadecimal.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK": It means the successful operation.

"TTM:ERP": It means the incorrect command format.

This product ID can be saved after power-off. ID will show in the broadcast content and can be used to filter devices or to determine if it is a specific product.

- **Transmit Power Configuration**

Input the following string "TTM:TPL-(X)" to RX to set the corresponding transmit power (in dBm). Wherein X = **"2"**, "1", "0", "-3", "- 6", "-9", "-12", "-15", "-18" or "- 21" (all data format is in ASCII code).

For example: “TTM:TPL-(2)” means the transmit power is set as +2 dBm.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK": It means the successful operation.

"TTM:ERP": It means the incorrect command format.

Note: This configuration cannot be saved after power-off.

- **Data Delay Configuration**

Input the following string "TTM:CDL-Xms" to RX to set the delay time between low level output of BCTS and TX data output (in ms). Wherein X = "0", "2", "5", "10", "15", "20", or "25" (all data format is in ASCII code).

For example: “TTM:CDL-2ms” means the delay time is 2 ms.

After the command is executed, the following confirmation will be got from TX:

“TTM:OK\r\n\0": It means the successful operation.

"TTM:ERP\r\n\0": It means the incorrect command format.

To make the user CPU have enough time to wake-up from sleep mode and ready to receive data, the module is provided this delay (X) configuration. The BRTS will be set low before there is data to be sent through the serial port, while the delay time between low level output of BCTS and TX data output will be set by this parameter. The actual delay (T) will be T = (X + Y) ms, if the minimum delay is no less than X, while 500 μs < Y < 1 ms.

This configuration can be saved after power-off. The scheme of data delay configuration is as follows:

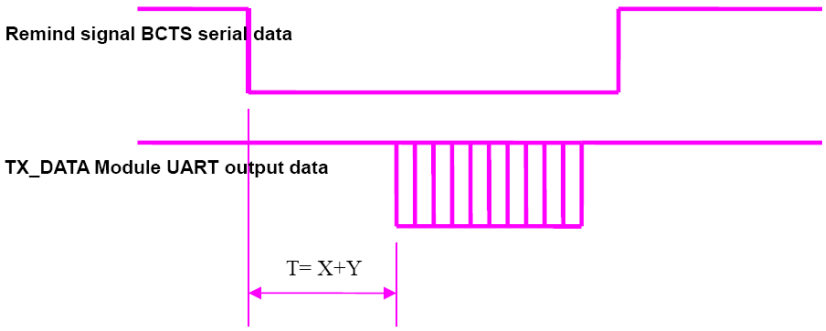


Figure 8. Scheme of Data Delay Configuration

7 BLE Protocol (APP Interface)

- Bluetooth Data Channel [Service UUID:0xFFE5]

Characteristic UUID	Operation	Bytes	Default Value	Remarks
FFE9	Write	MTU-3	None	Written data will be transmitted from serial port TX

Remark: Bluetooth input data will be transmitted to serial output. APP operates write in this channel by BLE API, and the data will be output from TX. See details in "UART Transparent Transmission Protocol".

- **Serial Port Data Channel [Service UUID:0xFFE0]**

Characteristic UUID	Operation	Bytes	Default Value	Remarks
FFE4	Notify	MTU-3	None	Data input from serial port RX will generate a notification which will be sent to mobile devices.

Remark: Serial input data will be transmitted to BLE output. If notification switch of FFE4 channel is enabled, (01 00 is needed to be written in 0x000E+1 = 0x000F by BTool), a notify event will be generated in this channel when the master CPU transmits legal data to the module RX through serial port, and APP can directly process and use notify information in the callback function.

Note: MTU refers to the Maximum Transmission Unit (MTU), the maximum packet size (in Bytes) that a communication protocol can pass.

- **Module Parameter Configuration [Service UUID:0xFF90]**

Characteristic UUID	Operation	Bytes	Default Value	Remarks
FF91	Read / Write	16	Tv430u-x xxxxxxx (ASCII string with terminator)	Device name, XXXXXXXX for the last four Bytes of MAC address.
FF92	Read / Write	1	0	Set Bluetooth connection interval: **0: 20 ms** 1: 30 ms 2: 50 ms 3: 100ms 4: 200 ms 5: 300 ms 6: 400 ms 7: 500 ms 8: 1000 ms 9: 1500 ms 10: 2000 ms
FF93	Read / Write	1	5	Set baud rate of serial ports: 0: 4800 bps 1: 9600 bps 2: 19200 bps 3: 38400 bps 4: 57600 bps **5: 115200 bps**
FF94	Write	1	None	Channel to control remote reset and recovery: - Remote reset control by writing **0x55**. - Remote light recovery control and reset by writing **0x35** (recover user data only). - Remote deep recovery control by writing **0x36** (factory reset) and reset
FF95	Read / Write	1	3	Set broadcast cycle:

				0: 20 ms 1: 50 ms 2: 100 ms **3: 200 ms** 4: 500 ms 5: 1000 ms 6: 1500 ms 7: 2000 ms 8: 2500 ms 9: 3000 ms 10: 4000 ms 11: 5000 ms
FF96	Read / Write	2	0x5624	Set product identification code
FF97	Read / Write	1	7	Set transmission power: 0: -21 dBm 1: -18 dBm 2: -15 dBm 3: -12 dBm 4: -9 dBm 5: -6 dBm 6: -3 dBm **7: 0 dBm** 8: 1 dBm 9: 2 dBm
FF98	Read / Write	16	Default broadcast	Set customized broadcast data Customizing broadcast data: 0 < n <= 16 See details in "Broadcast Data Configuration".

Remark: The channel is for module parameter configuration.

FF91 is the channel for setting device name. Device name can be acquired and set by read and write to this channel accordingly. The length of device name set must meet the condition: 0 < L < 17. And **the name is suggested to end with the terminator ('\0')**. The default name is "Tvvvvv- XXXXXXX\0" (16 Byte), wherein vvvv is the current firmware version number and XXXXXXXX is the last four Bytes of MAC address.

FF92 is the channel to set the connection interval. Connection interval between mobile devices and the module can be set by write to this channel. Thus, the device power consumption and the data throughput can be controlled in a flexible way. In order to raise the connection speed, the setting of connection interval will not be saved. After the command is executed, the following confirmation will be got from TX:

"TTM:TIMEOUT\r\n\0" means timeout and the change failed.

"TTM:OK\r\n\0" means the change is successful and the new connection interval is applied.

Test shows that it takes around 30 s to wait when the connection interval is changed from 500 ms to another interval by iPhone 4S (iOS 5.1.1). But it will be effectively very quickly if the connection interval is changed from a high frequency one (such as: 100 ms) which is affected by BLE protocols.

FF93 is the channel to set baud rate. Baud rate can be set by read and write to the channel. The new baud rate will be effective in two seconds after set and can be saved after power-off. 5 (115200 bps) is by default.

FF94 is the channel to control remote reset and recovery. Various controlling functions can be realized by writing different values to the channel.

1. Write **0x55** to this channel will software-reset the module.
2. Write **0x35** to the channel will light-recovery the module. All user settings will be recovered to the factory defaults, including I/O output status, PWM initialization mode and user password. Afterwards, the module will be reset.
3. Write **0x36** to the channel will deep-recovery the module. All system settings will be recovered to the factory defaults and the module will be reset afterwards.

FF95 is the channel to set broadcast cycle. Broadcast cycle can be set by read and write to this channel. The setting can be saved after power-off. 3 (200 ms) is by default.

FF96 is the channel to set product ID by read and write to the channel. APP can filter and connect to the specific product through this code. The setting can be saved after power-off. 0x5246 is by default.

FF97 is the channel to set transmit power by write to this channel. **The setting cannot be saved after power-off. 7 (0 dBm) is by default.**

FF98 is the channel to set broadcast contents. Broadcast data can be customized by write to this channel. The setting can be saved after power-off. When the data is all 0 (16 Byte), it is regarded that default broadcast data is used instead of customized data. (See details in "Broadcast Data Configuration").

- **Device Information [Service UUID:0x180A]**

Characteristic UUID	Operation	Bytes	Default Value	Remarks
2A23	Read	8	xxxxxx0000xxxxxx (Hex)	System ID, wherein xxxxxxxxxxxx is the MAC address of module, with low Byte in front.
2A26	Read	6	V4.30u (ASCII)	Firmware version of the module.

Remark: This channel is for module information read.

Acquire module information by read this channel 2A23. For example: xxxxxx0000xxxxxx, wherein xx part is the MAC address of the module in six Bytes (low Byte in the front).

Acquire module version number by read this channel 2A26. For example: Vx.xx, wherein xx is firmware version number.

- **Anti-hijacking Password [Service UUID:0xFFC0]**

The module supports anti-hijacking password. Unauthorized mobile devices (or mobile phones) is prevent from being connected to the module effectively by this service. The initial password is 000000 (ASCII). In this case, APP does not need pairing with the module during connecting, so it is regarded as no use of password and any mobile device with specified APP can connect to the module.

The new password (not all zero) is set and saved by APP. If a new password (not all zero) is set, anti-hijacking is enabled. A password once configured requests will be submitted within **20 s** after APP connects to the module. Otherwise the connection is broken up. Any write operation except for password submission cannot be executed before APP submits the correct password.

If the password needs to be recovered, **the module must be reset first by pull-low RESTORE (IO0) pin for 5 s and the operation must be done within 30 s after connection set-up.** For safety, password read is not supported, and all passwords are kept by APP.

A password channel is provided to realize the submission, modification and cancellation of the password by protocol. Meanwhile, event notify service of password is also provided to inform APP of the results of password operations, including 4 events: right password, error password, successful password update and cancel password.

Characteristic UUID	Operation	Bytes	Default Value	Remarks
FFC1 (handle:0x0045)	Write (Saved after power-off)	12	"123456123456"(ASCII)	Submit current password of 123456, and the new password must be same as the previous one.
			"123456888888"(ASCII)	Change the previous password of 123456 into the new one of 888888, and the previous password must be correct.
			"888888000000"(ASCII)	Cancel password (by changing the password into the default value 000000, and the previous password must be correct.
FFC2 (handle:0x0048)	Notify	1	0(PWD_RIGHT_EVENT)	Right password.
			1(PWD_ERROR_EVENT)	Error password
			2(PWD_UPDATED_EVENT)	Successful password update.
			3(PWD_CANCEL_EVENT)	Cancel password.

Remark:

1. Password is all in 12-Byte ASCII, wherein the red part is the current password and the blue part is the new password.

2. Current password is "000000" by default before modified by APP.
3. The execution result of related password operations can be generated in this channel by enabling notification of FFC2(01 00 is needed to be written into 0x0048+1 = 0x0049 by BTool).
4. When APP submits "123456123456", it means the new password is the same with the current one. And APP will be notified in channel FFC2 of “notify:0(PWD_ RIGHT_EVENT)”. It shows the password submission is correct.
5. When APP submits the password (red part) is different from the current one, such as: "123455xxxxxx", regardless of the value of “xxxxxx” part, APP will be notified in channel FFC2 of “notify: 1(PWD_ ERROR _EVENT)”. It shows the password submission is wrong.
6. When APP submits "123456888888", it means the new password is "888888" and the current password is "123456". APP will be notified in channel FFC2 of “notify: 2(PWD_ UPDATED _EVENT”. It shows the password update is successful.
7. When APP submits "888888000000", it means the new password will be changed to an all-zero value. APP will be notified in channel FFC2 of “notify: 3(PWD_ CANCEL _EVENT)”. It shows the password is cancelled.

8 Broadcast Data Configuration

- Default Broadcast Data

When the module EN pin is set low, the module will broadcast at an interval of 200 ms. In the domain of the broadcast data GAP_ADTYPE_MANUFACTURER_SPECIFIC (iOS officially defined programming macro), the following contents are included (default of 3 Bytes):

{

0x00,0x00, Customized device type code, 00 00 is by default, and can be set by AT command.

0x00, Percentage of module power supply (2.0 V = 0%).

}

- Customized Broadcast Data

Customizing the broadcast content can be realized by AT command, and the maximum length is16 Bytes (in blue). In the broadcast data GAP_ADTYPE_MANUFACTURER_SPECIFIC domain will contain the following content, and the length is 2 + n Bytes:

{

0x00,0x00, Customized device type code, 00 00 is by default, can be set by AT command;

Data [n], Customized broadcast data, n < = 16;

}

Note:

Customized broadcast data can be modified by AT command and saved after power-off. After re-power on, last-time customized broadcast data will be shown. If customized broadcast data is all 0 (16 Byte), the customized broadcast will not be used but the system default broadcast contents. To avoid the extra power consumption caused by too long broadcast data, customized broadcast data can be set to be any value in 1 Byte.

9 System Reset and Recovery

There are three methods of module reset, among which the third one can recovery system parameters:

1. Reset module by AT command (See details in "AT Command").
2. Remote reset module by the service channel interface of APP [See details in "BLE Protocol (APP Interface)").
3. Reset module by RESET pin of the module (See details in "Module Parameter Configuration"). **30 seconds after power-on**, set the pin low and hold **for 5 s**, the module will recover the parameters before user modified (light recovery, reset the module immediately after release press). 30 seconds after power-on, set the pin low and hold **for 20 s**, the module will be factory reset (deep recovery) immediately. This pin is with an internal pull-up, and the module will not enter recovery mode by default.

- **System parameters reset in light recovery including:**

A. Anti-hijack password recovers to "000000". No password will be used by default.

- **System parameters reset in deep recovery including:**

A. Anti-hijack password recovers to "000000". No password will be used by default.
B. Serial port baud rate recovers to 115200 bps.
C. Device name recovers to "Tv430u-XXXXXXXX" and X is the last four Bytes of MAC address.
D. Broadcast cycle recovers to 3 (200 ms).
E. Connection interval recovers to 20 ms.
F. Transmit power recovers to 0 dBm.
G. Product ID recovers to 0x5246.
H. Customized broadcast length recovers to 0.
I. All customized broadcast data recovers to 0. Default broadcast data is used but customized broadcast data).

10 Master Reference Code (Transparent Transmission)

Logical relationship: The module uses two IO ports of BCTS, BRTS to send and receive notification and control. These two IOs are in high level and will be triggered when put low in normal. If the module has data to send, BCTS will be set low to inform the microcontroller to receive. If the microcontroller has data to send, BRTS will set low to notification module to receive. The schematic code is as follows:

```
void main(void)
{
  EN = 0;                                          // EN, start to broadcast
  while(!BLEMoudleAck("TTM:OK"));                  // Wait the mobile device to scan and connect
                                                   // Wait for successful connection, can also join in the time-limited waiting
                                                   // Can also judge the level of the connection prompt signal line
  BRTS = 0;                                        // BRTS is set low, and notify CC2640R2F module to ready for receiving
  halMcuWaitMs(50);                                // Delay for 50 ms
  UARTWrite(  HAL_UART_PORT_0,  "TTM:CIT-100ms",   // Modify the connection interval, and get confirmation from the serial port
14);
  halMcuWaitMs(50);                                // Delay for 50 ms, and ensure the data is transmitted
  BRTS = 1;                                        // BRTS is set high, and transmission is finished
  while(!BLEMoudleAck("TTM:OK"));                  // Wait for successful configuration, can also join in the time-limited waiting

  while(1){                                        // Cyclic test of transmission and receiving
   while(1){
    if(BCTS == 0){                                 // Check, if BCTS is set low, the module will be ready for receiving
    while(BCTS==0);                                // Wait for transmission to be finished, can also do time-limited waiting
    if(UARTRead(uartBuffer) == SUCCESS)            // Read data from serial port
    {... ...}                                      // Use data
  }
  BRTS = 0;                                        // RTS is set low, notify CC2640R2F module to ready
```

```
                                                    for receiving
    halMcuWaitMs(50);                               // Delay for 50 ms
    send_TX("1234567890");                          // Transmit any data (Beyond 200 Byte)
    halMcuWaitMs(50);                               // Delay for 50 ms, and ensure the data is transmitted
    BRTS = 1;                                       // BRTS is set high, and transmission is finished
    halMcuWaitMs(20);                               // Delay to send the next packet, the delay depends on
                                                    the packet size.

    }
  }
}}
```

11 iOS APP Programming Reference

The module is always to broadcast as slave, waiting for mobile phone to scan and connect as master. The scanning and connection are usually completed by APP. Due to the particularity of BLE protocol, there is no need to scan and connect Bluetooth LE devices in the system settings of the Smart phone. Smart devices are responsible for BLE connection, communication, disconnection, etc. And usually it is implemented by the APP.

Regarding BLE programming in iOS, the key point is the read, write and enable notify switch to **Characteristic (or called channel)** to. To read and write in the channel can realize the direct control on the direct-drive mode functions of the module and no extra MCU is needed. Typical functions that are involved are as follows:

```
/*!
* @method writeValue:forCharacteristic:withResponse:
* @param data The value to write.
* @param characteristic The characteristic on which to perform the write operation.
* @param type The type of write to be executed.
* @discussion Write the value of a characteristic.
* The passed data is copied and can be disposed of after the call finishes.
* The relevant delegate callback will then be invoked with the status of the request.
* @see peripheral:didWriteValueForCharacteristic:error:
*/
- (void)writeValue:(NSData *)data forCharacteristic:(CBCharacteristic *)characteristic type:(CBCharacteristicWriteType)type;
```

Note: to write to a characteristic.

```
NSData *d = [[NSData alloc] initWithBytes:&data length:mdata.length];
[p writeValue:d
forCharacteristic:c
type:CBCharacteristicWriteWithoutResponse];
```

```
/*!
* @method readValueForCharacteristic:
* @param characteristic The characteristic for which the value needs to be read.
* @discussion Fetch the value of a characteristic.
* The relevant delegate callback will then be invoked with the status of the request.
* @see peripheral:didUpdateValueForCharacteristic:error:
```

```
*/
- (void)readValueForCharacteristic:(CBCharacteristic *)characteristic;
```

Note: to read a characteristic

[p readValueForCharacteristic:c];

```
/*!
* @method setNotifyValue:forCharacteristic:
* @param notifyValue The value to set the client configuration descriptor to.
* @param characteristic The characteristic containing the client configuration.
* @discussion Ask to start/stop receiving notifications for a characteristic.
* The relevant delegate callback will then be invoked with the status of the request.
* @see peripheral:didUpdateNotificationStateForCharacteristic:error:
*/
- (void)setNotifyValue:(BOOL)notifyValue forCharacteristic:(CBCharacteristic *)characteristic;
```

Note: to open a characteristic notify enable switch.

[self setNotifyValue:YES forCharacteristic:c];//open notify enable switch.

[self setNotifyValue:NO forCharacteristic:c]; //close notify enable switch.

```
/*
* @method didUpdateValueForCharacteristic
* @param peripheral Pheripheral that got updated
* @param characteristic Characteristic that got updated
* @error error Error message if something went wrong
* @discussion didUpdateValueForCharacteristic is called when CoreBluetooth has updated a
* characteristic for a peripheral. All reads and notifications come here to be processed.
*
*/
-(void)peripheral:(CBPeripheral*)peripheral didUpdateValueForCharacteristic:(CBCharacteristic *)characteristic error:(NSError *)error
```

Note: after each reading operation, this callback function will be performed. The application layer saves the data that is read in this function.

About the details of scanning, connecting, and other communication operations, please refer to the test APP source code (BLE Transmit Module v1.29) for transparent transmission in iOS, in which it realizes, for FFE9 and FFE4, the operations of data transmit from BLE to serial port and from serial port to BLE characteristics (notify and write). Other

controls on direct-drive functions are similar, all by reading or writing to certain characteristic. The only difference is the characteristic UUID and the Bytes of reading and writing operations.

12 Application and Implementation

12.1 Basic Operation of Hardware Design

1. It is recommended to offer the module with a DC stabilized power supply, a tiny power supply ripple coefficient and the reliable ground. Please pay attention to the correct connection between the positive and negative poles of the power supply. Otherwise, the reverse connection may cause permanent damage to the module;
2. Please ensure the supply voltage is between the recommended values. The module will be permanently damaged if the voltage exceeds the maximum value. Please ensure the stable power supply and no frequently fluctuated voltage.
3. When designing the power supply circuit for the module, it is recommended to reserve more than 30% of the margin, which is beneficial to the long-term stable operation of the whole machine. The module should be far away from the power electromagnetic, transformer, high-frequency wiring and other parts with large electromagnetic interference.
4. The bottom of module should avoid high-frequency digital routing, high-frequency analog routing and power routing. If it has to route the wire on the bottom of module, for example, it is assumed that the module is soldered to the Top Layer, the copper must be spread on the connection part of the top layer and the module, and be close to the digital part of module and routed in the Bottom Layer (all copper is well grounded).
5. Assuming that the module is soldered or placed in the Top Layer, it is also wrong to randomly route the Bottom Layer or other layers, which will affect the spurs and receiving sensitivity of the module to some degrees;
6. Assuming that there are devices with large electromagnetic interference around the module, which will greatly affect the module performance. It is recommended to stay away from the module according to the strength of the interference. If circumstances permit, appropriate isolation and shielding can be done.
7. Assuming that there are routings of large electromagnetic interference around the module (high-frequency digital, high-frequency analog, power routings), which will also greatly affect the module performance. It is recommended to stay away from the module according to the strength of the interference. If circumstances permit, appropriate isolation and shielding can be done.
8. It is recommended to stay away from the devices whose TTL protocol is the same 2.4 GHz physical layer, for example: USB 3.0.
9. The antenna installation structure has a great influence on the module performance. It is necessary to ensure the antenna is exposed and preferably vertically upward. When the module is installed inside of the case, a high-quality antenna extension wire can be used to extend the antenna to the outside of the case.
10. The antenna must not be installed inside the metal case, which will cause the transmission distance to be greatly weakened.
11. The recommendation of antenna layout.

    The inverted-F antenna position on PCB is free space electromagnetic radiation. The location and layout of antenna is a key factor to increase the data rate and transmission range.

    Therefore, the layout of the module antenna location and routing is recommended as follows:

    （1） Place the antenna on the edge (corner) of the PCB.

    （2） Make sure that there is no signal line or copper foil in each layer below the antenna.

（3） It is the best to hollow out the red part of the antenna position in the following figure ensure that S11 of the module is minimally affected.

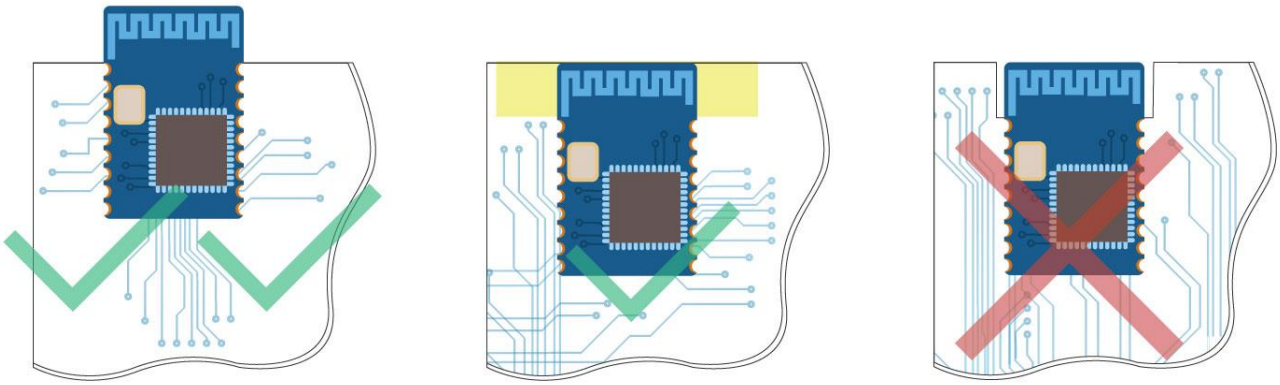


Figure 9. Recommendation of Antenna Layout

12. Antenna

RF-BM-S02I module is integrated the IPEX version 1 antenna seat, the specification of antenna seat is as follow:

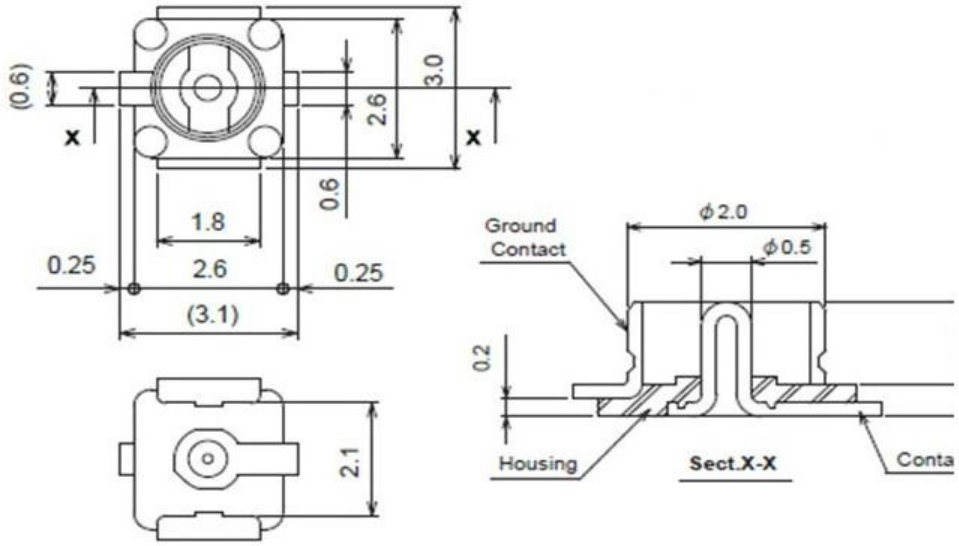


Figure 10. Specification of Antenna Seat

The specification of IPEX wire end is as follow:

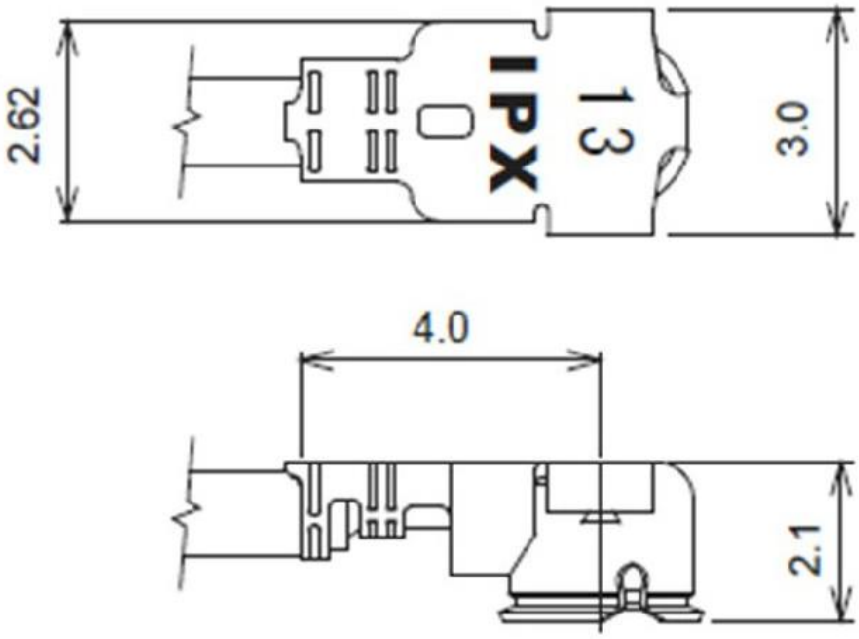


Figure 11. Specification of IPEX Wire

12.2 Trouble Shooting

12.2.1 Unsatisfactory Transmission Distance

1. When there is a linear communication obstacle, the communication distance will be correspondingly weakened. Temperature, humidity, and co-channel interference will lead to an increase in communication packet loss rate. The performances of ground absorption and reflection of radio waves will be poor, when the module is tested close to the ground.
2. Seawater has a strong ability to absorb radio waves, so the test results by seaside are poor.
3. The signal attenuation will be very obvious, if there is a metal near the antenna or the module is placed inside of the metal shell.
4. The incorrect power register set or the high data rate in an open air may shorten the communication distance. The higher the data rate, the closer the distance.
5. The low voltage of the power supply is lower than the recommended value at ambient temperature, and the lower the voltage, the smaller the power is.
6. The unmatchable antennas and module or the poor quality of antenna will affect the communication distance.

12.2.2 Vulnerable Module

1. Please ensure the supply voltage is between the recommended values. The module will be permanently damaged if the voltage exceeds the maximum value. Please ensure the stable power supply and no frequently fluctuated voltage.
2. Please ensure the anti-static installation and the electrostatic sensitivity of high-frequency devices.

3. Due to some humidity sensitive components, please ensure the suitable humidity during installation and application. If there is no special demand, it is not recommended to use at too high or too low temperature.

12.2.3 High Bit Error Rate

1. There are co-channel signal interferences nearby. It is recommended to be away from the interference sources or modify the frequency and channel to avoid interferences.
2. The unsatisfactory power supply may also cause garbled. It is necessary to ensure the power supply reliability.
3. If the extension wire or feeder wire is of poor quality or too long, the bit error rate will be high.

12.3 Electrostatics Discharge Warnings

The module will be damaged for the discharge of static. RF-star suggest that all modules should follow the 3 precautions below:

1. According to the anti-static measures, bare hands are not allowed to touch modules.
2. Modules must be placed in anti- static areas.
3. Take the anti-static circuitry (when inputting HV or VHF) into consideration in product design.
   Static may result in the degradation in performance of module, even causing the failure.

12.4 Soldering and Reflow Condition

1. Heating method: Conventional Convection or IR/convection.
2. Temperature measurement: Thermocouple d = 0.1 mm to 0.2 mm CA (K) or CC (T) at soldering portion or equivalent methods.
3. Solder paste composition: Sn/3.0 Ag/0.5 Cu
4. Allowable reflow soldering times: 2 times based on the following reflow soldering profile.
5. Temperature profile: Reflow soldering shall be done according to the following temperature profile.
6. Peak temperature: 245 ℃.

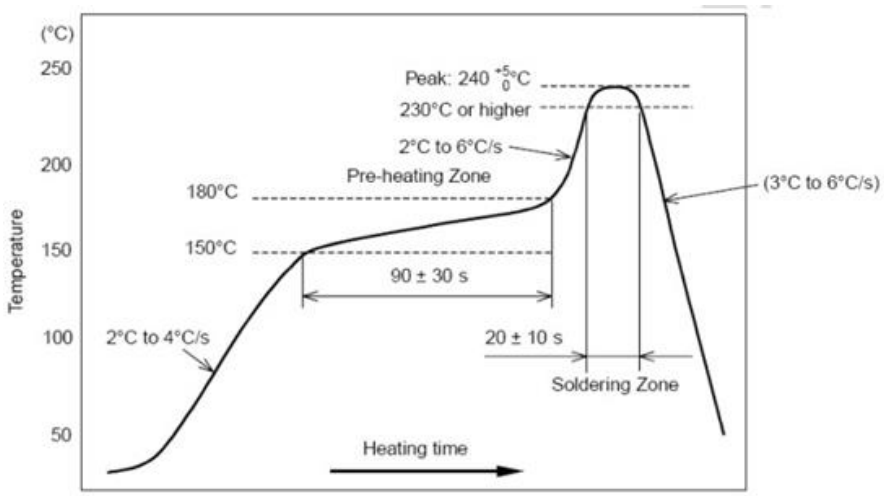


Figure 12. Recommended Reflow for Lead Free Solder

13 Revision History

Version No.	Date	Description
V4.30u	2019.02.21	The initial version released.
V4.30u	2019.03.19	Add directional connection function. Modify abnormal disconnection of anti-hijacking password. Modify high-speed transparent transmission packet stacking. Modify the 15-Byte limit of name modification. Improve character attribute description.
V4.30u	2019.04.01	Redefine master-slave role functionality. Modify disable connection of master after abnormal disconnection of slave. Modify disable EN and the long light of broadcast indication LED during disconnection.
V4.30u	2019.04.22	Update power consumption parameters. Add hardware specifications.
V4.30u	2019.07.04	Update rate and packet instructions. Update hardware specifications.
V4.30u	2019.11.18	Add data delay configuration.

Note: The protocol is updated from time to time. Before using this document, please make sure it is the latest version.

14 Contact Us

SHENZHEN RF-STAR TECHNOLOGY CO., LTD.

Shenzhen HQ:

Add.: Room 601, Block C, Skyworth Building, High-tech Park, Nanshan District, Shenzhen, Guangdong, China

Tel.: 86-755-8632 9687

Chengdu Branch:

Add.: No. B4-12, Building No.1, No. 1480 Tianfu Road North Section (Incubation Park), High-Tech Zone, Chengdu, China (Sichuan) Free Trade Zone, 610000

Tel.: 86-28-6577 5970

Email: sunny@szrfstar.com, sales@szrfstar.com

Web.: www.szrfstar.com