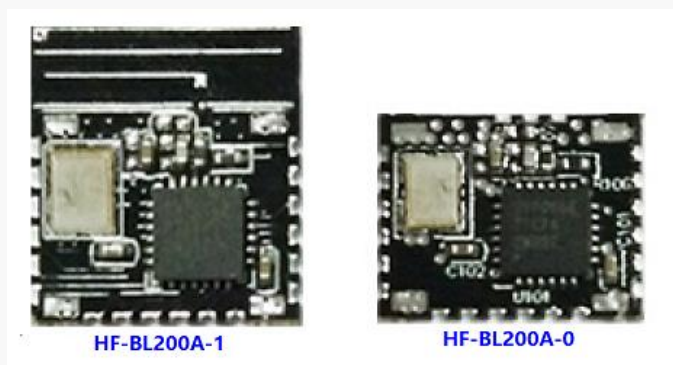


HF-BL200A

Bluetooth Low Energy (BLE 4.1) Module

User Manual

V 1.2



Overview of Characteristic

- ◇ Support IEEE 802.15.1 BT4.1 Wireless Standard
- ◇ Support wireless upgrade (OTA)
- ◇ UART Transparent Transmit Module
- ◇ Supper low sleep mode power 3.5uA, excellence power save scheme suitable for battery-powered applications
- ◇ Support internal / external antenna options
- ◇ Single 2.7V~3.6V power supply
- ◇ Smallest size :
 - internal antenna: 12.7mm x 12mm x 2mm
 - external antenna: 9.5mm x 12mm x 2mm
- ◇ SIG QDID: 91979
- ◇ ROHS compliance

Table of Contents

LIST OF FIGURES	5
LIST OF TABLES	5
1. PRODUCT OVERVIEW	7
1.1. General Description	7
1.2. Device Feature.....	7
1.3. Key Application	7
1.4. Device Parameter	8
1.5. Order information.....	8
2. HARDWARE INTRODUCTION	10
2.1. Pin Definition	10
2.2. Electrical Specification.....	12
2.3. Mechanical size	13
2.4. Antenna Layout	14
3. UART TRANSPARENT TRANSMIT MODE	16
3.1. Transparent Transmit Mode.....	16
3.2. Transparent Transmit Mode Pin Description	18
3.3. Factory Default Parameter	18
3.3.1. Device Name	18
3.3.2. Broadcast Parameter	19
3.3.3. Transmit Power	19
3.3.4. Connect Parameter	19
3.3.5. Baud rate	19
3.3.6. Send Latency Parameter.....	19
4. UART AT COMMAND DEFINITION	20
4.1. Module Operation Mode Configure	20
4.2. AT: Command Overview	20
4.2.1. Command Format.....	21
4.2.2. Command List.....	22
4.3. AT Command.....	24
4.3.1. Help command.....	24
4.3.2. Set Module NAME	24
4.3.3. Query module Name.....	25
4.3.4. Set Connect Parameter	25
4.3.5. Query Connect Parameter.....	26
4.3.6. Set Baud rate	27
4.3.7. Query Baud rate.....	27
4.3.8. Get MAC Address	28
4.3.9. Query Software Version.....	28
4.3.10. Set Transmit Power.....	28

4.3.11.	Query Transmit Power	29
4.3.12.	Set Broadcast Parameter.....	30
4.3.13.	Query Broadcast Parameter	30
4.3.14.	Set User-define Broadcast Data	31
4.3.15.	Query User-define Broadcast Data	31
4.3.16.	Set Auto-broadcast Switch.....	32
4.3.17.	Query Auto-broadcast Switch	32
4.3.18.	Start Broadcast.....	32
4.3.19.	Stop Broadcast.....	33
4.3.20.	Query Current BLE Subsystem Status	34
4.3.21.	Disconnect.....	34
4.3.22.	Set Device Verify Code	35
4.3.23.	Query Device Verify Code.....	35
4.3.24.	Set UART Output Data Latency	36
4.3.25.	Save Parameter	37
4.3.26.	Module Restore	37
4.3.27.	Set Module Operate Mode	37
4.3.28.	Module BLE subsystem Status Notification	38
4.3.29.	Module CPU Status Notification.....	38
4.3.30.	Restore Factory Setting via At Command.....	39
4.3.31.	Enable Module Deep Sleep	39
4.3.32.	Restore Factory Setting via Hardware Method	40
4.3.33.	Change Service UUID.....	40
4.3.34.	Change APP Receive Channel UUID	41
4.3.35.	Change APP Send Channel UUID.....	41
5.	APP COMMAND.....	42
5.1.	Channel and Data Description.....	42
5.2.	Data Format	42
5.3.	Command Content	43
5.3.1.	Command Type: 0x0E	43
5.3.2.	Command Type: 0x0F.....	43
6.	BASIC COMMUNICATION MECHANISIM	45
6.1.	Application Service Data Channel (user-define application service UUID: 0x2B00) .	45
6.1.1.	Module->APP, UART Data Channel 【feature UUID:0x2B10】	45
6.1.2.	APP->Module, UART Data Channel 【feature UUID:0x2B11】	45
6.1.3.	APP->Module, OTA Mode Switch 【Feature UUID:0x2B12】	46
6.1.4.	APP->Module, APP Command Channel 【Feature UUID:0x2B13】	46
6.2.	Battery Service Data Channel.....	46
6.2.1.	APP->Module, Battery Data Channel 【Feature UUID:0x2A19】	47
6.3.	OTA Service Data Channel.....	47
6.3.1.	APP->Module, OTA Data Channel	47
7.	PACKAGE INFORMATION.....	48
7.1.	Reflow Soldering Profile	48

7.2. Handling Instruction48

7.3. Shipping Information49

APPENDIX A: UART TRANSPARENT TRANSMIT TYPICAL APPLICATION50

APPENDIX D: CONTACT INFORMATION52

LIST OF FIGURES

Figure 1.	HF-BL200A order information	9
Figure 2.	HF-BL200A Pin Definition	10
Figure 3.	HF-BL200A-0 Mechanical size.....	13
Figure 4.	HF-BL200A-1 Mechanical size.....	14
Figure 5.	HF-BL200A Module Reference Placement	15
Figure 6.	Basic HF-BL200A wireless network structure	16
Figure 7.	HF-BL200A Default UART Parameter	20
Figure 8.	"AT:HELP" list All command.....	21
Figure 9.	Module UART Output Data Latency Set Map	37
Figure 10.	Module UART Receive Enable Signal Map.....	40
Figure 11.	Reflow Soldering Profile	48
Figure 12.	Package Information.....	49
Figure 13.	UART Transparent Transmit Typical Application	50

LIST OF TABLES

Table 1	HF-BL200A Module Technique Specification	8
Table 2	HF-BL200A Pin Function Definition.....	10
Table 3	Limited Specification	12
Table 4	Electrical Specification	12
Table 5	RF specification	12
Table 8	Module connect interval 20ms communication mode example	17
Table 9	Error Code list.....	22
Table 10	AT Command List.....	22
Table 11	Command Data from APP to Module	42
Table 12	Feedback Data from Module to APP	42
Table 13	Command Type 0x0F command list.....	43
Table 14	Command Type 0x0F Feedback List	44
Table 15	Description of User-define Service of All Channel	45
Table 16	0x2B10 feature UART Module->APP Channel Description	45
Table 17	0x2B11 Feature UART APP->Module Channel Description	45
Table 18	0x2B12 Feature OTA Mode Switch Channel Description	46
Table 19	0x2B13 Feature APP Command Channel Description	46
Table 20	Description of Battery Service of All Channel	46
Table 21	0x2A19 Feature Battery Power Channel Feature Description	47
Table 22	Description of OTA Service of All Channel	47
Table 23	Feature OTA Data Channel Feature Description	47
Table 24	Reflow Soldering Specification.....	48

History

V1.0 02-02-2018. First Version

V1.1 02-02-2018. add requirement for PIN3 connect to GND

V1.2 08-13-2019. add requirement for PIN16 capacity

1. PRODUCT OVERVIEW

1.1. General Description

HF-BL200A Bluetooth Low Energy module is a high performance IOT module designed by High Flying. It provide a solution for connecting things to Bluetooth wireless network and data transmitting via UART interface. With the feature of low power, small size, high anti-interference performance, the module integrates PCB antenna and use open stamp type interface which enable customer have more flexibility on software and product structure, and solve the RF hardware design and debug issue.

1.2. Device Feature

- 2.7~3.6V Single power supply Smallest size
- internal antenna: 12.7mm x 12mm x 2mm
- external antenna: 9.5mm x 12mm x 2mm
- With 32 bit high performance RISC 48MHz ARM M0 MCU, MCU build-in 12KB SRAM and 512KB Flash
- Support low power & multiple level power management mode
- Full peripheral interface
 - GPIO interface
 - Serial Communication (UART application)
 - ADC interface
- BQB

1.3. Key Application

- Smart LED lighting
- Smart toy
- Electronic Scale
- Smart Cup
- Smart Home Appliance
- OBD
- IOT, Smart Home Automation
- Sports, fitness, customer electronics products
- Smart instrument, data acquisition, sensor
- PC, Tablet peripheral interface

1.4. Device Parameter

Table 1 HF-BL200A Module Technique Specification

Class	Item	Condition	Parameter	
Wireless parameter	Certificate		BQB	
	Wireless standard		802.15.1	
	Frequency range		2.402GHz-2.480GHz	
	Data rate		1Mbps@2.4GHz	
	Tx power		Max=+10 dBm,	
	Rx sensitivity		-93 dBm	
Hardware parameter	Data interface		UART SPI,I2C PWM	
	Operating voltage		2.7~3.6V	
	Power consumption	Dormant		2uA
		Sleep		3~11uA
		DIG Active		5mA
		RF Active		20mA
	Average power consumption	20ms broadcast		3.2mA
		0.5ms broadcast		130uA
		4s broadcast		1.3uA
		30ms connection		19uA
		100ms connection		290uA
		4s connection		10uA
	Work temperature		-40°C- 85°C	
	Storage temperature		-45°C- 125°C	
Size		HF-BL200A-1: 12.7mm x 12mm x 2mm HF-BL200A-0: 9.5mm x 12mm x 2mm		

Power details refer to Table 6

1.5. Order information

As customer demand, HF-BL200A can provide variant physical version, detail part number as below:

HF-BL200A-X

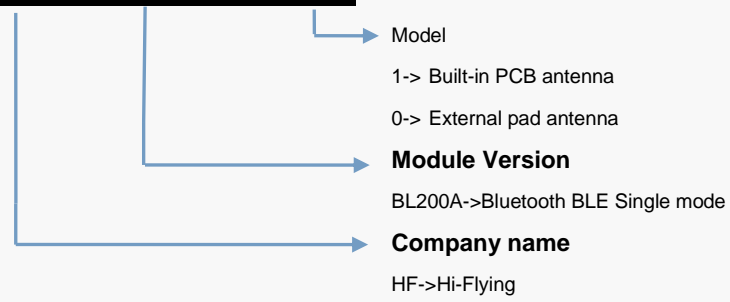


Figure 1. HF-BL200A order information

2. HARDWARE INTRODUCTION

2.1. Pin Definition

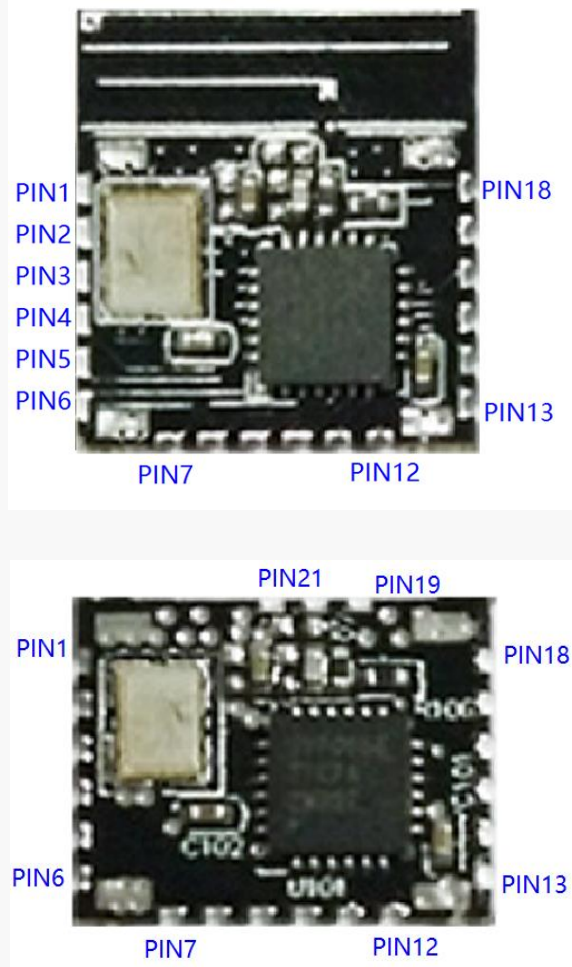


Figure 2. HF-BL200A Pin Definition

Table 2 HF-BL200A Pin Function Definition

Pin	Net name	Features	Description
1,2,13 19,21	GND	Power	Ground
3	VEQ	Power	e-fuse parameter programming pin. Connect this pin direct to GND
4	IO23	D/A I/O	GPIO/AIO 1
5	IO24	D/A I/O	Add 4.7K resistor external GPIO/PWM0/AIO2
6	IO01	D/A I/O	GPIO/SPI-M CLK/PWM1/AIO3
7	IO02	D/A I/O	GPIO/SPI-M MISO/PWM2/AIO4
8	MANUFACTURE	DIG I/O	Add 4.7K resistor external

			Pull high for more than 5 seconds to restore the factory settings, IO03/SPI-M CSN/PWM3
9	WAKE	DIG I/O	Pull low to wake up Bluetooth IO04 / SPI-M MOSI / PWM4
10	BCTS	DIG I/O	The default output low, high data output is sent, used to wake up the MCU, the data is sent after the output low IO05 / I2C-M
11	IO06	DIG I/O	GPIO/I2C-M
12	VDD	Power	Power input, 2.7V ~ 3.6V
14	UART_TX	DIG I/O	3.3V TTL IO09
15	UART_RX	DIG I/O	3.3V TTL IO10
16	RESET	DIG I	Add external 0.1uF capacity to GND Hardware reset, active low
17	UART_DEB_TX	DIG I/O	Debug UART_TX IO15
18	UART_DEB_RX	DIG I/O	Debug UART_TX IO16
20	RF	RF	RF antenna pin

note:

1: **Connect PIN3(VEQ) to GND.**

2: In normal startup, **IO03 and IO24 must be connected with 4.7K pull-down resistor**. When it needs to enter programming mode, reset IO03 and use UART_DEB_TX and UART_DEB_RX to program the program.

2.2. Electrical Specification

Table 3 Limited Specification

Parameter	Description	Min Value	Typ. Value	Max value	Unit
VPIN	Pin voltage limit	-0.1	-	VDD	V
T_VRISE	Power up time	5	-	100	mS
ESD_HBM				4000	V
ESD_MM				100	V
ESD_CDM				500	V
Storage temperature range		-50		150	°C

Table 4 Electrical Specification

Parameter	Condition	Min Value	Typ. Value	Max Value	Unit
Work voltage		2.7	3.3	3.6	V
Work temperature		-40		85	°C
I_DORM	Dormant status		2		uA
I_SLEEP	Sleep status		3		uA
I_TX0DBM	TX Mode 0dBm		23		mA
I_TX10DBM	TX Mode, 10dBm		45		mA
I_TXN20DBM	TX Mode -20dBm		15		mA
I_RX	RX default		20		mA
	System clock =24Mhz		7.1		mA
	System clock =12Mhz		4		mA
	System clock =6Mhz		2.5		mA
	System clock =3Mhz		1.7		mA
Sleep mode	Operating 3Mhz		1.1		mA
Deep sleep mode	ECO Stop, WCO work		1.3		uA
Stop mode	ECO Stop, WCO work		60		nA

Table 5 RF specification

Parameter	Condition	Min value	Typ. Value	Max value	Unit
Receiving sensitivity			-93		dBm
Operating frequency range		2400		2483	MHz
Frequency deviation		225	250	275	KHz
Output Power			19		dBm
Frequency tolerance			±50		Ppm

Signal strength accuracy			±5		dBm
Signal accuracy			1		dBm

2.3. Mechanical size

HF-BL200A physical size (unit: mm) as below:

HF-BL200A-0 Pads: Size 9.5mm X 12mm, pad pitch 1.27mm

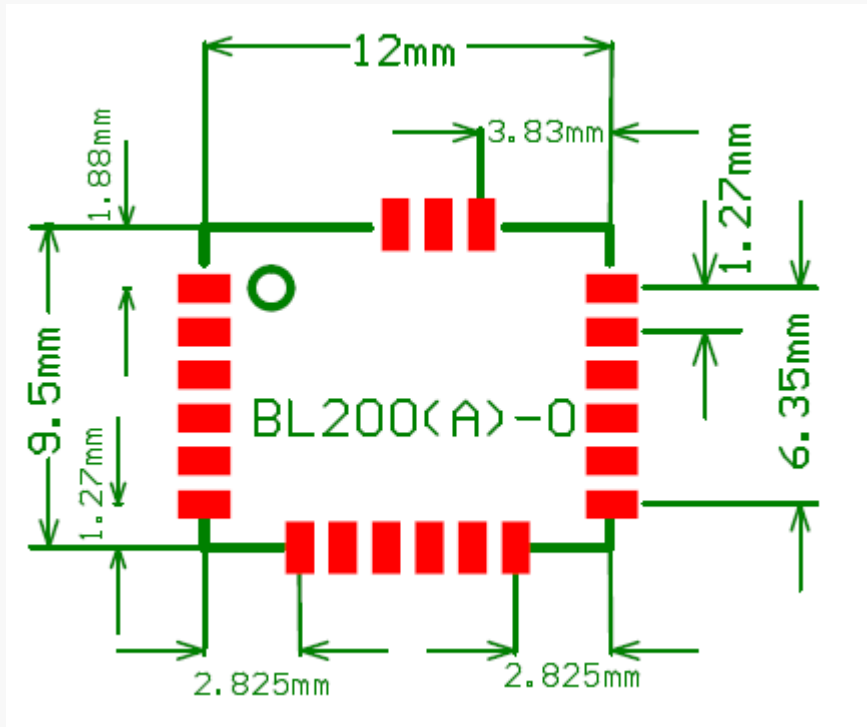


Figure 3. HF-BL200A-0 Mechanical size

HF-BL200A-1 Pads: Size 12.7mm X 12mm, pad pitch 1.27mm

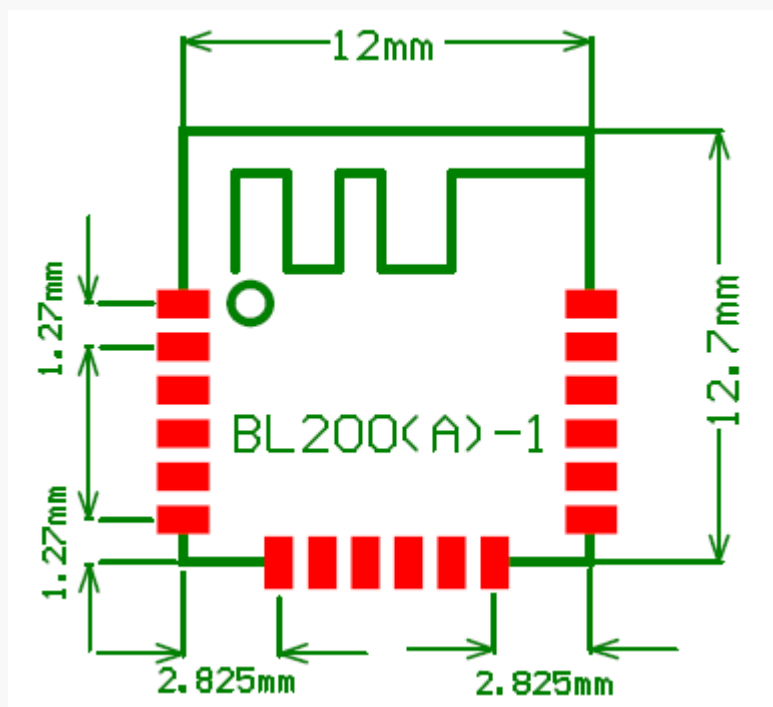


Figure 4. HF-BL200A-1 Mechanical size

2.4. Antenna Layout

HF-BL200A supports internal antenna.. Customers need to obey following antenna design rules and module location suggestions:

- ✓ For user PCB, place module on the edge area of the PCB as possible, or suspend the antenna area.
- ✓ Module antenna (5mmx12mm) correspondent area can't put components or paste GND, the surrounding components or GND should be as far as possible from antenna place.
- ✓ Antenna must be away from high components at least 10mm.
- ✓ Antenna can't be shielded by any metal enclosure.

High Flying suggests to locate HF-BL200A as Figure 5 to reduce the influence to antenna and wireless signal as much as possible, or contact High Flying technique people for support

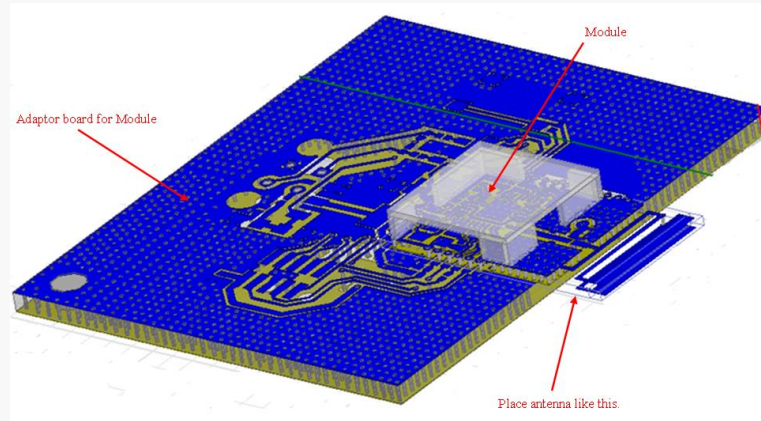


Figure 5. HF-BL200A Module Reference Placement

3. UART TRANSPARENT TRANSMIT MODE

3.1. Transparent Transmit Mode

Transparent Transmit means the two way communication between device and mobile terminal via the connecting with serial interface and device MCU. Device can set HF-BL200A baud rate and connect interval via AT command through serial interface (refer to “Serial AT command Description” for details). Target different baud rate and BLE connect interval or different packet sending interval, module would have different data handling capacity. The default baud rate of HF-BL200A is 115200bps, it is recommended to set baud rate at 115200bps when transmitting big data or in highly real-time application. Baud rate support save after power off.

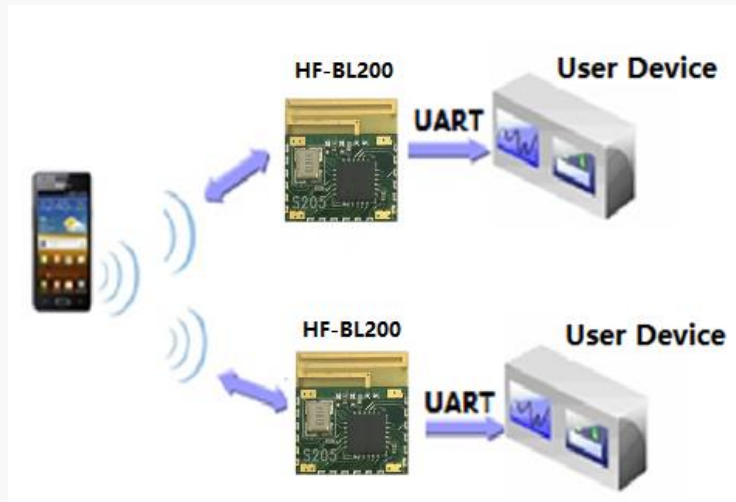


Figure 6. Basic HF-BL200A wireless network structure

Module BLE connect interval is 20ms, and it is able to send almost 200 byte data one time via serial interface. Theoretically, module is able to send data up to 4KB/S. Below is the detail description about transparent transmit.

Module can receive almost 200 byte data from serial interface one time. Module will automatically send sub-packet according to data size, each packet limit is 20 bytes. The data packet sending from mobile device to module must be separated to sub-packet(1-20 byte each). After receiving the sub-packets, module will send to MCU in proper order.

Serial hardware protocol: 115200 bps, 8,non-parity, 1 stop bit

After receiving data from MCU serial port, module will judge if it is AT command. If it is, module will execute the command. If not, module will transmit the data to APP under BLE connection and notify mode enabled status.

After module receives APP data, it will output the data to MCU via serial interface.

If module system status changed, module will send status notification character to MCU, such as disconnect, connect, broadcast.

Bluetooth protocol require that the minimum connect interval is 7.5ms. Default connect interval is 20ms (comply with IOS regulation), if it is needed to save power and apply low speed transmit mode. User can adjust connect interval through AT command (the maximum connect interval is 2000ms, comply with IOS regulation)

The calculation of transmit speed : the data packet of each connection interval from APP to module is 1 packet, max 4 packet can be transmitted from module to APP in each interval (the packet is related to module CPU capacity) . There is maximum 20 bytes in each packet, if connect interval is T (unit :ms), then max transmit speed V (unit is byte/s) is :

$$V_{module} = 20 \times 4 \times 1000 / T \quad (V \text{ only related } T)$$

$$V_{app} = 20 \times 1000 / T \quad (V \text{ only related } T)$$

V_{app}: if module connect interval is 20ms, then each interval can transmit max 20 bytes, so theoretically the max speed is 20x50 =1k byte/s. Test shows the transmit from APP to module is stable and reliable.

V_{module}: if module connect interval is 20ms, then each interval can transmit maximum 80 bytes, so theoretically the maximum speed is 80x 50=4k byte/s. Test shows that if transmit speed is under 2kbyte/s, there is few chances of leaking packet. For safety consideration, it is recommended to do verification and re-transmit at up level whatever it is low speed or high speed application. In Android application, it is recommended to lower connect interval to increase speed.

Below is the example of 20ms connect interval communication mode, user can self-configure. The lower the Speed, the lower the packet lost ratio.

Table 8 Module connect interval 20ms communication mode example

通讯参考模式	BLE连接间隔 T (ms)	理论最高转发能力 V (byte/s) V = 80*1000/T	串口包长度 L (byte)	串口发包间隔 TS (ms) 当 L<80 时, TS >= T 当 80<L<160 时, TS >= T*2 当 160<L<200 时, TS >= T*3	实际转发速率 V0 (byte/s) V0 = L*1000/TS	备注
1	20	4K	80	TS >= T 即可, 若取 TS=20ms	80*1000/20 = 4K	TS 偏小, 不推荐
2	20	4K	200	TS >= T*3 即可, 若取 TS=70ms	200*1000/70 = 2.8K	
3	20	4K	200	TS >= T*3 即可, 若取 TS=80ms	200*1000/80 = 2.5K	
4	20	4K	80	TS >= T 即可, 若取 TS=35ms	80*1000/30 = 2.6K	
5	20	4K	70	TS >= T 即可, 若取 TS=30ms	70*1000/30 = 2.3K	
6	20	4K	60	TS >= T 即可, 若取 TS=30ms	60*1000/30 = 2K	
7	20	4K	40	TS >= T 即可, 若取 TS=30ms	40*1000/30 = 1.3K	
8	20	4K	20	TS >= T 即可, 若取 TS=30ms	20*1000/30 = 666byte	

Remark: user can set particular communication mode according to actual application, packet length can be set between 80byte < L < 200byte , the relationship as below as per BLE protocol:

If $L < 80$, $TS \geq T$;

If $80 < L < 160$, $TS \geq T * 2$;

If $160 < L < 200$, $TS \geq T * 3$;

All transmit mode is safe as long as it meets above conditions, $TS=T$ is recommended, $TS=T*2$, $TS=T*3$ is ok but the packet lost ratio is high, must add verification and re-transmit mechanism. In another words, if apply 80byte $< L < 200$ byte ,serial data can transmit to module in one time, but need to reserve time for sending data from module to Bluetooth, or there will occur rear-end. E.g. Set connect interval $T=20$ ms, if data length is $L=200$, TS must over $T*3 = 60$ ms , $TS=70$ ms is a reasonable choice.

Serial data length can be any value under 200 byte. In order to achieve the best efficiency and avoid full load operation , it is recommended to use 20,40,60 byte serial data packet, set packet interval over 20ms.

Remark: in IOS, the function is used to call Characteristic is:
BCharacteristicWriteWithRespons

e parameter, "write with response" mode would lower transmit efficiency to ensure the accuracy of each packet. Using " CBCharacteristicWriteWithoutResponse" ; "write without response" mode would promote transmit efficiency , but the accuracy of packet need to be verified by APP up level.

3.2. Transparent Transmit Mode Pin Description

All pins of Transparent Transmit : UART_TX、UART_RX、WAKEUP、BCTS、XRES、MANUFACTURE

Min. pins of Transparent Transmit: UART_TX、UART_RX、WAKEUP

Details refer to 《Appendix A》

WEAKEUP: inside high impedance, pin must be connected. Pull up to stop UART receive function. UART is unable to receive data (UART able to send data). Pull down to enable UART to receive data.

MANUFACTURE: Pull down inside, and it can be no connection. Pull up outside and power on or reset, keep high level MANUFACTURE at least 5 second. Module restore factory setting

XRES: Pull up inside, can be no connection

BCTS: In normal time, output low level; BCTS output high level before sending data from module, after sending , BCTS output low level.

3.3. Factory Default Parameter

3.3.1. Device Name

Device NAME: "HF-BL200A"

3.3.2. Broadcast Parameter

Broadcast Min. interval: "0800", means 500ms

Broadcast Max. interval: "0800", means 500ms

Broadcast type: "0", means public

Broadcast channel : "7", means apply channel index 37 38 39

3.3.3. Transmit Power

Broadcast transmit power : "8", means 0dbm

Connect transmit power: "8" means 0dbm

3.3.4. Connect Parameter

Min interval : "0016", means 20ms

Max interval: "0032", means 40ms

Slave Latency: "0", means 0

Connect timeout: "0200", means 2000ms

3.3.5. Baud rate

Baud rate: "115200"

3.3.6. Send Latency Parameter

Serial sending Latency : "01", means 1ms

4. UART AT COMMAND DEFINITION

4.1. Module Operation Mode Configure

After HF-BL200A power on, module enter transparent transmit mode, default UART configuration parameter is as below:

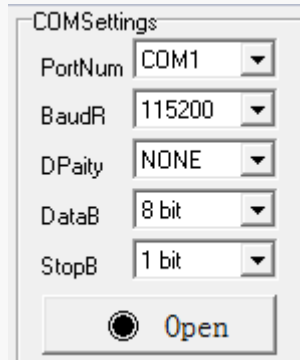


Figure 7. HF-BL200A Default UART Parameter

User can configure the module via AT command through UART or use Evaluation Kit to test. Evaluation kit refer to Appendix B.

<Description>: AT command debug tool recommend use “ friend serial debug assistant” and SecureCRT tool ,which can be download from company website.

4.2. AT: Command Overview

AT: command can be inputted via serial debug tool or compile as below photo shows, AT:HELP is a help command, list all commands and description.

Remark: the end of each command need add return or line feed character.

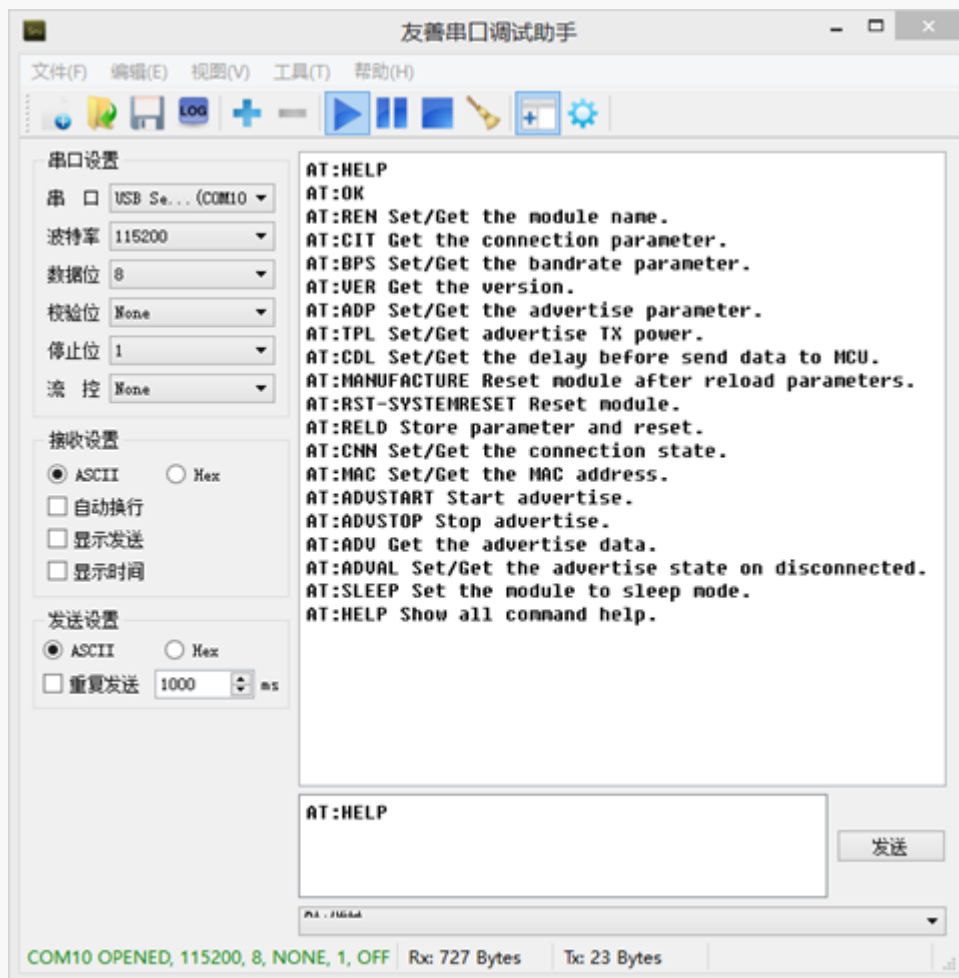


Figure 8. "AT:HELP" list All command

4.2.1. Command Format

AT: command based on ASCII code, format as below:

➤ Format description

- <>: mean the part must include
- [: mean optional part

➤ Command message

AT: <CMD>[op][para-1,para-2,para-3,para-4...]<CR><LF>

- AT: : command message prefix
- CMD: command character string
- [op] : command operate character, can be parameter setting or query
 - ◆ "-": means parameter setting
 - ◆ "?": means query
- [para-n] : parameter setting code, if query, it is not required.
- <CR>: end , ASCII code 0x0d;
- <LF>: return, ASCII code 0x0a

<description>:

When input command, "AT: <CMD>" character is capital word, parameter parts is the same.

➤ Feedback message

AT: <RSP>[op] [para-1,para-2,para-3,para-4...]<CR><LF><CR><LF>

- AT: feedback message prefix;
- RSP: feedback character string, include:
 - ◆ "OK": success
 - ◆ "ERP": failed
 - ◆ "DENY": operation denied
 - ◆ <CMD>: original command
- [op]: -
- [para-n]: the feedback parameter when query or error code when error happened
- <CR>: ASCII code 0x0d
- <LF>: ASCII code 0x0a

➤ Error code

Table 9 Error Code list

Error code	Description
ERP	Invalid command
WRONG	Invalid parameter
DENY	Operation denied

4.2.2. Command List

Table 10 AT Command List

Command Type	Data transmit direction	Save or not	Function description	Command detail
"AT:HELP"	MCU->Module	No	Helo command	4.3.1
" AT:REN-" + Name	MCU->Module	Yes	Set module NAME	4.3.2
" AT:REN? "	MCU->Module	Query	Query module NAME	4.3.3
"AT+CIT-" +IntervalMin+IntervalMax +Slave Latency+connSupervision Timeout	MCU->Module	Yes	Set connect parameter	4.3.4
"AT+CIT? "	MCU->Module	Query	Query connect parameter	4.3.5
"AT:BPS-" + baudrate	MCU->Module	Yes	Set baud rate	4.3.6
"AT:BPS? "	MCU->Module	Query	Query baud rate	4.3.7

"AT:MAC? "	(APP)MCU->Module	Query	Query MAC	4.3.9
"AT:VER? "	(APP)MCU->Module	Query	Query software version	4.3.10
"AT:TPL-" +Data0+Data1	MCU->Module	Yes	Set transmit power	4.3.11
"AT:TPL? "	MCU->Module	Query	Query transmit power	4.3.12
"AT:ADP-" +parameters	MCU->Module	Yes	Set broadcast parameter	4.3.13
"AT:ADP? "	MCU->Module	Query	Query broadcast parameter	4.3.14
"AT:ADV-" + Data	MCU->Module	Yes	Set user-define broadcast data	4.3.15
"AT:ADV? "	MCU->Module	Query	Query user-define broadcast data	4.3.16
"AT:ADVAL"+data	MCU->Module	Yes	Set auto-broadcast switch	4.3.17
"AT:ADVAL? "	MCU->Module	Query	Query auto-broadcast switch	4.3.18
"AT:ADVSTART"	MCU->Module	No	Start broadcast	4.3.19
"AT:ADVSTOP"	MCU->Module	No	Stop broadcast	4.3.20
"AT:CNN? "	MCU->Module	Query	Query current BLE subsystem status	4.3.21
"AT:CNN-D"	MCU->Module	No	Disconnect	4.3.22
"AT:PID-" + Data	(APP)MCU->Module	Yes	Ser device verify code	4.3.23
"AT:PID? "	(APP)MCU->Module	Query	Query device verify code	4.3.24
"AT:CDL-" +X	MCU->Module	Yes	Set UART output data latency	4.3.25
"AT:SAVE"	(APP) MCU->Module	No	Save parameter	4.3.26
"AT:RST"	MCU->Module	No	Restore module	4.3.27
"AT:SLEEP-" +command	MCU->Module	No	Set module enter sleep mode	4.3.28
"AT:CNN-" +status	Module->MCU	Notification	Module BLE subsystem status Notification	4.3.33
"AT:CPU-" +status	Module->MCU	Notification	Module CPU status Notification	4.3.34

"AT:RELD"	MCU->Module	no	Restore factory setting via AT command	4.3.35
PIN32->WAKEUP high level outside	MCU->module	Notification	Enable module CPU subsystem deep sleep (BLE subsystem isolated operation)	4.3.36
PIN29->MANUFACTURE keep high level at least 5s when power on	MCU->Module	-	Restore factory parameter via hardware	4.3.37

* remark: UART serial interface command must end with <CR><LF>, feedback must end with<CR><LF>

*remark: (APP) means the command is also applicable to APP command, APP command no need <CR><LF>

4.3. AT Command

4.3.1. Help command

Function: display all command description

Effective time: effective immediately

Command:

"AT:HELP"+<CR><LF>

feedback:

"AT:OK" +<CR><LF>+data mean command success, feedback description data

Data: list all command description data

"AT:ERP\r\n" means invalid command, query failed

4.3.2. Set Module NAME

Function: set module name, max length is 11 character

Effective time: effective after re-broadcast

Command:

"AT:REN-" + Name+<CR><LF>

Name : display the name, max 11 character

Feedback:

"AT:OK\r\n" means success

"AT:WRONG\r\n" means invalid parameter, set failed

"AT:ERP\r\n" means invalid command, set failed

4.3.3. Query module Name

Function: query module current Name

Command

"AT:REN?" +<CR><LF>

'?': means query

Feedback

"AT:REN-" + devicename +<CR><LF> means success

Devicename: device name can be displayed, max 11 character, default factory name is "HF-BL200A"

"AT:ERP\r\n0" means invalid command, failed

4.3.4. Set Connect Parameter

Function: set BLE connect parameter

Effective time: effect immediately at connection status or effective after connection

Command:

"AT:CIT-" + IntervalMin + IntervalMax + Slave Latency + connSupervision Timeout +<CR><LF>

IntervalMin: 4 character means, HighByte [...] LowByte; '0'~'9' means 0~9, "0200"-> decimal 200*1.25ms

IntervalMax: 4 byte means, HighByte [...] LowByte; '0'~'9' means 0~9, "1000"-> decimal 1000*1.25ms

Slave Latency: 1 character, '0'~'9' means 0~9,

connSupervision Timeout: 4 character means, HighCharacter [...] LowCharacter, '0'~'9' means 0~9, "2000"-> decimal 2000*10ms

when connect to IOS APP, Apple has their own regulation, for parameter must comply :

1. $\text{IntervalMax}^* (\text{Slave Latency}+1) \llcorner = 2 \text{ seconds}$
2. $\text{IntervalMin} \geq 20\text{ms}$
3. $\text{IntervalMin} + 20\text{ms} \leq \text{IntervalMax}$
4. $\text{Slave Latency} \leq 4$
5. $\text{connSupervision Timeout} \leq 6 \text{ seconds}$
6. $\text{IntervalMax}^*(\text{Slave Latency}+1)*3 < \text{connSupervisionTimeout}$

Feedback:

"AT:WRONG\r\n" means invalid parameter, modify failed

"AT:OK\r\n" means success, run with new connect interval

"AT:ERP\r\n" means invalid command, modify failed

4.3.5. Query Connect Parameter

Function: Query module connect parameter

Effective time: immediately

Command:

"AT:CIT?" + <CR><LF>

Feedback:

"AT+CIT-" + IntervalMin + IntervalMax + Slave Latency + connSupervision Timeout + <CR><LF>

IntervalMin: 4 character means , HighByte [...] LowByte; '0'~'9' means 0~9, "0200"-> decimal 200*1.25ms

IntervalMax: 4 character means, HighByte [...] LowByte; '0'~'9' means 0~9, "1000"-> decimal 1000*1.25ms

Slave Latency: 1 character, '0'~'9' means 0~9,

connSupervision Timeout: 4 character means , HighByte[...]LowByte, '0'~'9' means 0~9, "2000"-> decimal 2000*10ms

when connect to IOS APP , Apple has their own regulation, four parameter must comply:

7. $\text{IntervalMax}^* (\text{Slave Latency}+1) \llcorner = 2 \text{ seconds}$
8. $\text{IntervalMin} \geq 20\text{ms}$
9. $\text{IntervalMin} + 20\text{ms} \leq \text{IntervalMax}$

10. Slave Latency \leq 4
11. connSupervision Timeout \leq 6 seconds
12. IntervalMax*(Slave Latency+1)*3<connSupervisionTimeout

"AT:ERP\r\n" means invalid command, query failed

4.3.6. Set Baud rate

Function: modify UART Baud rate

Effective time: effective after receiving the success feedback data

Command:

"AT:BPS-"+baud rate+<CR><LF>

Baudrate:6 character, '0'~'9' means 0~9,"115200"-> decimal 115200

Feedback:

"AT:OK\r\n" means success

"AT:WRONG\r\n" means invalid parameter, modify failed

"AT:ERP\r\n" means invalid command, modify failed.

Remark: only apply regular baud rate 4800,9600,19200,38400,57600,115200, other parameter would be consider invalid.

4.3.7. Query Baud rate

Function: query current baud rate

Effective time: immediately

Command:

"AT:BPS? "+<CR><LF>

Feedback:

"AT:BPS-"+baud rate+<CR><LF> means success

Baudrate:6 character , '0'~'9' means 0~9,"115200"-> decimal 115200,"038400"-> decimal 38400

"AT:ERP\r\n" means invalid command, query failed

4.3.8. Get MAC Address

Function: Get Bluetooth MAC

Command:

```
"AT:MAC?"<CR><LF>
```

APP Command do not need <CR><LF>

Feedback:

```
" AT:MAC-" + MAC + <CR><LF>
```

MAC: 12 character can represent 6 byte data, each 2 character represent the four bit in high order and the four bit in low order of one byte.

e.g. : "123456789012" -> 0x12, 0x34, 0x56, 0x78, 0x90, 0x12, character can only be '0'~'9', 'A'~'F'

APP Command do not need <CR><LF>

4.3.9. Query Software Version

function: Query device current software version

Effective time: immediately

Command:

```
"AT:VER?"<CR><LF>
```

APP Command do not need <CR><LF>

Feedback:

```
"AT:VER-" + version + <CR><LF>
```

 means query success

APP Command do not need <CR><LF>

Version: character string, e.g. "100-CU-V*.*)"

```
"AT:ERP\r\n"
```

 means invalid command, query failed

APP Command do not need "\r\n"

4.3.10. Set Transmit Power

Function: modify device transmit power, broadcast power and connect power

Effective time: immediately

Command

"AT:TPL-"+Data0+Data1+<CR><LF>

Data0: broadcast power, one character, '1'~'4' means 1~4,

Data1: connect power, one character, '1'~'4' means 1~4,

1-> -47dbm

2-> -7dbm

3-> 2dbm

4-> 8dbm

Feedback:

"AT:OK\r\n" means command successfully received

"AT:WRONG\r\n" means invalid parameter, modify failed, if not '1'~'4', then it is invalid parameter

"AT:ERP\r\n" means invalid command, modify failed

4.3.11. Query Transmit Power

Function: query device current transmit power , broadcast power, connect power

Effective time: immediately

Command:

"AT:TPL? "+<CR><LF>

'?': means query command

Feedback:

"AT:TPL-"+Data0+Data1+<CR><LF>

Data0: broadcast power ,one character, '1'~'8' means 1~8,

Data1: connect power, one character, '1'~'8' means 1~8,

1-> -18dbm

2-> -12dbm

3-> -6dbm

4-> -3dbm

5-> -2dbm

6-> -1dbm

7-> +3dbm

8-> 0dbm

"AT:ERP\r\n" means invalid command, query failed

4.3.12. Set Broadcast Parameter

Function: set broadcast parameter, min. interval/max interval/broadcast type/broadcast channel

Effective time: effect in next broadcast

Command:

"AT:ADP"+parameters+<CR><LF>

Parameters: includes 13 character, min interval ->4 character, max interval ->4 character, broadcast type->1 character, broadcast channel ->1 character

Min interval: HighByte|...|LowByte, '0'~'9' means 0~9("100"-> decimal 100), unit : 0.625ms

Max interval: HighByte|...|LowByte, '0'~'9' means 0~9("1200"-> decimal 1200), unit: 0.625ms

Broadcast type: '0'~'1' means 0~1, 0->public, 1->random, only public type available temporarily

Broadcast channel : '1'~'7' means 1~7, the three bit in low order correspond each channel index. bit0->channel37, bit1->channel->38, bit2->channel39, the setting can be combined

Feedback:

"AT:OK\r\n" means command successful received

"AT:WRONG\r\n" means invalid parameter, change failed

"AT:ERP\r\n" means invalid command, change failed

4.3.13. Query Broadcast Parameter

Function: query broadcast parameter, min interval/ max interval/ broadcast type/broadcast channel

Effective time: immediately

Command:

"AT:ADP?" + <CR><LF>

Feedback:

"AT:ADP"+parameters+<CR><LF>

Parameters: include 13 character, min interval ->4 character, max interval->4 character, broadcast type->1 character, broadcast channel ->1 character

Min interval: HighByte|...|LowByte, '0'~'9' means 0~9("100"-> decimal 100) , unit: 0.625ms

Max interval: HighByte|...|LowByte, '0'~'9' means 0~9("1200"-> decimal 1200) , unit: 0.625ms

Broadcast: '0'~'1' means 0~1, 0->public,1->random

Broadcast channel: '1'~'7' means 1~7, the three bit in low order correspond each channel index , bit0->channel37, bit1->channel->38,bit2->channel39, the setting can be combined.

"AT:ERP\r\n" means invalid command, query failed

4.3.14. Set User-define Broadcast Data

Function: set the user-define data in broadcast, the max data length is 11 bytes

Effective time: effective in next broadcast

Command:

"AT:ADV-" + Data + <CR><LF>

Data: max length is 11 bytes, data format must follow Bluetooth regulation

Feedback:

"AT:OK\r\n" means command successful received

"AT:WRONG\r\n" means invalid parameter, change failed

"AT:ERP\r\n" means invalid command, change failed

4.3.15. Query User-define Broadcast Data

Function: query the data in user-define broadcast, max data length is 11 bytes

Effective time: immediately

Command:

"AT:ADV? " + <CR><LF>

Feedback:

"AT:ADV-" + data + <CR><LF> means command successful received and feedback data

Data: max length is 11 bytes, data format must follow Bluetooth regulation

"AT:ERP\r\n" means invalid command, change failed

4.3.16. Set Auto-broadcast Switch

Function: auto start broadcast when module power on and disconnection

Effective time: effective when disconnect

Command:

```
"AT:ADVAL-"+data+<CR><LF>
```

Data: '0'->disable, '1'-> means enable

Feedback:

```
"AT:OK\r
```

```
\n" means command successfully received
```

```
"AT:WRONG\r\n" means invalid parameter, command failed
```

```
"AT:ERP\r\n" means invalid command, command failed
```

4.3.17. Query Auto-broadcast Switch

Function: query the functions status of auto-broadcast

Effective time: immediately

Command:

```
"AT:ADVAL? "+<CR><LF>
```

Feedback:

```
"AT:ADVAL-"+status+<CR><LF> feedback current auto-broadcast switch function parameter
```

Status: each character indicate a status

'0'-> disable

'1'-> enable

```
"AT:WRONG\r\n" means invalid parameter, query failed
```

```
"AT:ERP\r\n" means invalid command, query failed
```

4.3.18. Start Broadcast

Function: device start broadcast

Effective time: do not execute if under broadcast status and connect status, execute immediately under other status

Command:

"AT:ADVSTART"+ <CR><LF>

Feedback:

"AT:OK\r\n" means command successfully received and start broadcast

"AT:ADVSTART-"+status+<CR><LF> means command successfully received, but status incorrect

Status: each character indicate a status

'0'-> CYBLE_CNN_INITIALIZING

'1'-> CYBLE_CNN_ADVERTISING

'2'-> CYBLE_CNN_CONNECTED

'3'-> CYBLE_CNN_DISCONNECTED

'4'-> CYBLE_CNN_STOPPED

'5'-> CYBLE_CNN_CONNECTING

"AT:ERP\r\n" means invalid command, command failed

4.3.19. Stop Broadcast

Function: stop broadcast

Effective time: execute immediately at broadcast status; do not execute if at other status

Command:

"AT:ADVSTOP"+ <CR><LF>

Feedback:

"AT:OK\r\n" means command successfully received and stop broadcast

"AT:ADVSTOP-"+status+<CR><LF> means command successfully received, but status incorrect

Status: each character indicate a status

'0'-> CYBLE_CNN_INITIALIZING

'1'-> CYBLE_CNN_ADVERTISING

'2'-> CYBLE_CNN_CONNECTED

'3'-> CYBLE_CNN_DISCONNECTED

'4'-> CYBLE_CNN_STOPPED

'5'-> CYBLE_CNN_CONNECTING

"AT:ERP\r\n" means invalid command, command failed

4.3.20. Query Current BLE Subsystem Status

Function: query current device status

Effective time: immediately

Command:

"AT:CNN? "+<CR><LF>

Feedback:

"AT:CNN-"+status+<CR><LF> means query success

Status: each character indicate a status

'0'-> CYBLE_CNN_INITIALIZING

'1'-> CYBLE_CNN_ADVERTISING

'2'-> CYBLE_CNN_CONNECTED

'3'-> CYBLE_CNN_DISCONNECTED

'4'-> CYBLE_CNN_STOPPED

'5'-> CYBLE_CNN_CONNECTING

"AT:ERP\r\n" means invalid command, query failed

4.3.21. Disconnect

Function: device disconnect

Effective time: effective immediately if device at connect status; if at other status, do not execute command

Command:

"AT:CNN-D"+<CR><LF>

Feedback:

"AT:OK\r\n" means command received and executed

"AT:CNN-"+status+<CR><LF> means command received, but not executed because status incorrect

Status: each character indicate a status

'0'-> CYBLE_CNN_INITIALIZING

'1'-> CYBLE_CNN_ADVERTISING

'2'-> CYBLE_CNN_CONNECTED

'3'-> CYBLE_CNN_DISCONNECTED

'4'-> CYBLE_CNN_STOPPED

'5'-> CYBLE_CNN_CONNECTING

"AT:ERP\r\n" means invalid command, change failed

4.3.22. Set Device Verify Code

Function: change device verify code, all "0" means no verify code

Effective time: effective in next Bluetooth connection

Command:

"AT:PID-"+ Data+<CR><LF>

APP command no need <CR><LF>

Data:4 character means 1 hexadecimal data, '0'~'9','A'~'F' means 0~9, A~F ("12AD"-> hexadecimal 0x12AD)

Feedback:

"AT:OK\r\n" means command successfully received.

"AT:WRONG\r\n" means invalid parameter, change failed

"AT:ERP\r\n" means invalid command, change failed

APP command no need "\r\n"

4.3.23. Query Device Verify Code

Function: query device verify code parameter, all '0' means no verify code

Effective time: immediately

Command:

"AT:PID? "+<CR><LF>

'?': means query command

APP command no need<CR><LF>

Feedback:

"AT:PID-" + Data + <CR><LF>

APP Command no need<CR><LF>

Data: 4 Character means 1 hexadecimal data, '0'~'9', 'A'~'F' means 0~9, A~F ("12AD"-> hexadecimal 0x12AD)

"AT:ERP\r\n" means invalid command, query failed

APP Command no need "\r\n"

4.3.24. Set UART Output Data Latency

Function: receive data from APP, then output low level in BCTS inform outside MCU, send data after the set latency time; during sending, BCTS keep low level until sending finished, set BCTS high level; AT command feedback data from UART is not effected by this

Effective time: immediately

"AT:CDL-" + X + <CR><LF>

X: 2 character means 1 byte, "10"-> decimal 10, unit ms, max 10ms

Feedback:

"AT:OK\r\n" means command successfully received

"AT:WRONG\r\n" means invalid parameter, change failed

"AT:ERP\r\n" means invalid command, change failed

In order to enable user CPU to have enough time to wake up from sleep and ready to receive, module provide a Latency (X) setting. Set BCTS low level before sending data from UART, and the data latency between BCTS to module TX is set by the parameter. Module can assure min latency over "X", the actual latency is $T = (X + Y)$ ms, and $500\mu s < Y < 1ms$. The parameter can be saved after power off.

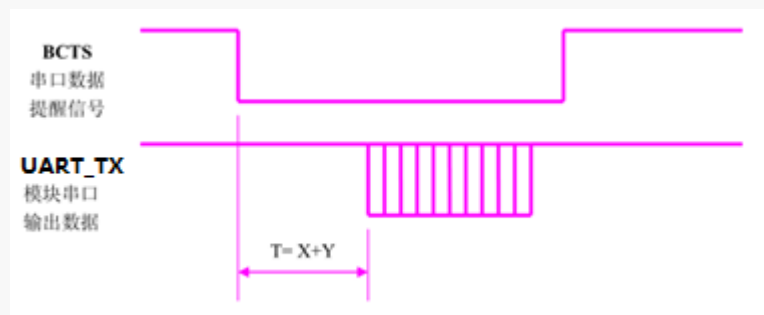


Figure 9. Module UART Output Data Latency Set Map

4.3.25. Save Parameter

Function: save the changed parameter(the save will stop BLE function and other interrupt, after save, system will restore)

Effective time: immediately, save will stop BLE function and other interrupt, after save, system will restore

Command:

"AT:SAVE"+<CR><LF>

APP command no need <CR><LF>

Feedback:

"AT:OK\r\n" means command correct

"AT:ERP\r\n" means invalid command, command failed

APP Command no need "\r\n"

4.3.26. Module Restore

Function: module restore, system software restore

Effective time: stop BLE function ,execute immediately

Command:

"AT:RST"+<CR><LF>

Feedback:

"AT:OK\r\n" means command successfully received

"AT:ERP\r\n" means invalid command, change failed

4.3.27. Set Module Operate Mode

function: stop BLE subsystem, force system enter Deepsleep、hibernate、stop mode,

Effective time: stop BLE function, immediately

Command:

"AT:SLEEP-"+command+<CR><LF>

Command: 1-> allow CPU system enter deep sleep, process via UART

2->force CPU enter Hibernate, BLE subsystem must stop, wake up through wake-up pin only

3->force CPU stop, BLE subsystem must stop , restore via wake-up pin and xres only

Feedback:

"AT:OK\r\n\r\n0" means command success

"AT:WRONG\r\n\r\n" means invalid parameter, command failed

"AT:ERP\r\n\r\n" means invalid command, command failed

4.3.28. Module BLE subsystem Status Notification

Function: module provide MCU the current BLE subsystem status variation

Notify time: notify when status changed

Command:

"AT:CNN-"+status+<CR><LF>

Status: each character indicate one status

'0'-> CYBLE_CNN_INITIALIZING

'1'-> CYBLE_CNN_ADVERTISING

'2'-> CYBLE_CNN_CONNECTED

'3'-> CYBLE_CNN_DISCONNECTED

'4'-> CYBLE_CNN_STOPPED

'5'-> CYBLE_CNN_CONNECTING

4.3.29. Module CPU Status Notification

Function: module provide current CPU status to MCU

Notify time: notify when status changed

Command:

"AT:CPU-"+status+<CR><LF>

Status: each character indicate one status

'0'-> CPU power on

'1'-> CPU deep sleep

'2'-> CPU operate or sleep

'3'-> CPU Hibernate

'4'-> CPU stop

4.3.30. Restore Factory Setting via At Command

function: all changeable parameter restore to factory setting, include baud rate, device name, transmit power, user-define broadcast data, UART data latency data, verify code ,broadcast parameter, connect parameter

Effective time: immediately, system compulsory restore

Command:

"AT:RELD"+<CR><LF>

Feedback:

"AT:OK\r\n" means command success, system restore

"AT:ERP\r\n" means invalid command, restore failed

4.3.31. Enable Module Deep Sleep

Function: enable module CPU subsystem deep sleep, (BLE subsystem isolate operation), non deep sleep mode module can receive data via UART, under deep sleep module UART function is disabled

Effective time: immediately

Command:

PIN32->WAKEUP: high level ->CPU subsystem enter deep sleep , module UART don't receive data

PIN32->WAKEUP: low level-> CPU is not allowed to enter deep sleep, module UART able to 不允 receive data

Need to pay attention to actual level and data sending sequence, sending data must wait 50 us after set high level to low level, after sending finished, and must waiting another 50 us to set low level to high level

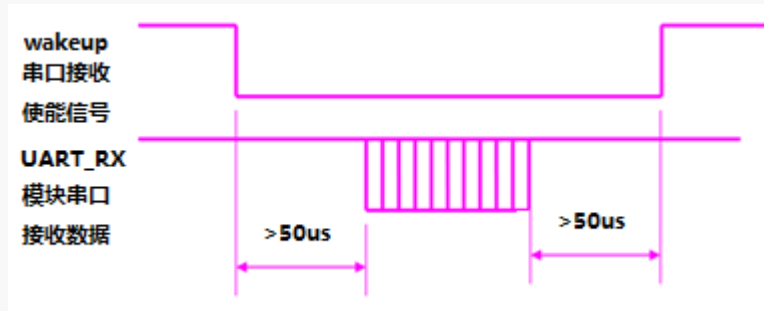


Figure 10. Module UART Receive Enable Signal Map

Feedback:

null

4.3.32. Restore Factory Setting via Hardware Method

Function: all changeable parameter restore to factory setting, include baud rate, device name, transmit power, user-define broadcast data, UART data latency data, verify code ,broadcast parameter, connect parameter

Effective time: immediately

Command:

PIN29->MANUFACTURE: pin pull up to power on or restore system, keep pin high level over 5 seconds after system started

Feedback:

"AT:RELOAD\r\n" means accept command to restore factory setting

4.3.33. Change Service UUID

Function: Set service UUID

Effective time: Need reboot

Command:

"AT:UIDS"+<CR><LF>

Feedback:

"AT:OK\r\n" means command successfully received

"AT:ERP\r\n" means invalid command, change failed

For APP to change this, does not need “\r\n”

4.3.34. Change APP Receive Channel UUID

Function: Set Receive Channel UUID

Effective time: Need reboot

Command:

"AT:UIDR"+<CR><LF>

Feedback:

"AT:OK\r\n" means command successfully received

"AT:ERP\r\n" means invalid command, change failed

For APP to change this, does not need “\r\n”

4.3.35. Change APP Send Channel UUID

Function: Set Send Channel UUID

Effective time: Need reboot

Command:

"AT:UIDW"+<CR><LF>

Feedback:

"AT:OK\r\n" means command successfully received

"AT:ERP\r\n" means invalid command, change failed

For APP to change this, does not need “\r\n”

5. APP COMMAND

5.1. Channel and Data Description

- APP command don't send in groups
- Maxium data of one group command is 20 bytes
- Data Channel: 0x2B13 send WriteWithResponse and receive notify

5.2. Data Format

- Bye length (1 byte) + command type (1 byte) + command content (max 18 bytes)
- APP-->MODULE

Table 11 Command Data from APP to Module

Byte length	Data[0]	1 byte	Include all byte of command type and command content
Command type	Data[1]	1 byte	0x0E: almost 18 bytes in one group AT command 0x0F: send verify code
Command content	Data[2]~data[19]	18 bytes	Details refer to command list (command type)

- APP<--MODULE

Table 12 Feedback Data from Module to APP

Byte length	Data[0]	1 byte	Including all bytes from command type and command content
Feedback type	Data[1]	1 byte	0x0E: maxium 18 bytes in one group AT command 0x0F: feedback verify result
Command content	Data[2]~data[19]	18 bytes	Details refer to feedback list (feedback type)

5.3. Command Content

5.3.1. Command Type: 0x0E

Command:

Function: data write in the form of 0x2B13 write with response , verify code command refer to AT command 4.3.23 &4.3.24, query version command refer to AT command 4.3.10

Feedback:

Channel: feedback data from 0x2B13 notify

Data content:

verify code command refer to AT command 4.3.23 &4.3.24, query version command refer to AT command 4.3.10

5.3.2. Command Type: 0x0F

Command:

Function: send verify code from APP to module, if no verification after connection, module will not execute any other command from APP, if module self verification is null or "0000", then verify code is no needed.

Verify code have time limit, if no verification within 10 seconds, module will disconnect with APP

Command Content:

Table 13 Command Type 0x0F command list

Byte length	Data[0]	1 byte	1~19	Include all bytes from command type and command content
Command type	Data[1]	1 byte	0x0F	Verify code command
Command	Data[2]~data[19]	18 bytes	Data[2]~data[19]	Command content

Feedback:

Channel: feedback data from 0x2B13 notify

Data Content:

Table 14 Command Type 0x0F Feedback List

Byte length	Data[0]	1 byte	2	Include all bytes from command type and command content
command type	Data[1]	1 byte	0x0F	Verified and feedback
Command content	Data[2]	1 byte	Data[2]	Verification feedback 0x00: success 0x01: failed 0x02: no verify code

6. BASIC COMMUNICATION MECHANISIM

6.1. Application Service Data Channel (user-define application service UUID: 0x2B00)

Table 15 Description of User-define Service of All Channel

UUID	Channel attribute	function
0x2B10	Notify/WriteWithResponse	APP send enable command to module via this notify channel
		Under notify enable status, module send data to APP via notify channel
0x2B11	Read/WriteWithoutResponse	APP send data to module
0x2B12	WriteWithResponse	APP send OTA mode switch command
0x2B13	Notify/WriteWithResponse	APP command

6.1.1. Module->APP, UART Data Channel 【feature UUID:0x2B10】

Table 16 0x2B10 feature UART Module->APP Channel Description

UUID	Executable operation	Bytes	Default value	Remark
0x2B10	Notify/WriteWithResponse	20	Null	Module receive data from UART RX , and notify APP via notify channel

Remark: data input from UART and output to Bluetooth. If turn on notify enable switch, outside MCU will send data to module RX via UART, and create a notify event in the channel, APP can directly process in the callback function.

6.1.2. APP->Module, UART Data Channel 【feature UUID:0x2B11】

Table 17 0x2B11 Feature UART APP->Module Channel Description

Feature value UUID	Executable operation	Byte	Default value	remark

0x2B11	Read/WriteWithout Response	20	null	APP write data to module via "write" channel, module output data via UART
--------	----------------------------	----	------	---

Description : input from Bluetooth and output via UART. APP write data via "Write" channel, the data will output via UART TX.

6.1.3. APP->Module, OTA Mode Switch 【Feature UUID:0x2B12】

Table 18 0x2B12 Feature OTA Mode Switch Channel Description

Feature value UUID	Executable operation	Byte	Default value	remark
0x2B12	WriteWithoutResponse	20	Null	Mode switch: switch from application mode to OTA status, command data is : "bootloader"

6.1.4. APP->Module, APP Command Channel 【Feature UUID:0x2B13】

Table 19 0x2B13 Feature APP Command Channel Description

Feature value UUID	Executable operation	Byte	Default value	remark
0x2B13	Notify/WriteWithResponse	20	null	APP command, details refer to AT command :4.3.9、4.3.10、4.3.23、4.3.24 Note: APP do not need to add the "AT:" prefix for AT comamnd

6.2. Battery Service Data Channel

Battery service UUID: 0x180F

Table 20 Description of Battery Service of All Channel

UUID	Channel attribute	function
0x2A19	read	Read electricity

6.2.1. APP->Module, Battery Data Channel 【Feature UUID:0x2A19】

Table 21 0x2A19 Feature Battery Power Channel Feature Description

Feature value UUID	Executable operation	byte	Default value	remark
0x2A19	Read	1	null	Unit is 20mV

6.3. OTA Service Data Channel

OTA service UUID: 00060000-F8CE-11E4-ABF4-0002A5D5C51B

Table 22 Description of OTA Service of All Channel

UUID	Channel attribute	function
00060001-F8CE-11E4-ABF4-0002A5D5C51B	WriteWithResponse	OTA data transmit

6.3.1. APP->Module, OTA Data Channel

Table 23 Feature OTA Data Channel Feature Description

Feature value UUID	Executable operation	byte	Default value	remark
00060001-F8CE-11E4-ABF4-0002A5D5C51B	WriteWithResponse	20	null	OTA data under "bootloader" mode, OTA mode receive program data and cover FLASH relative area

7. PACKAGE INFORMATION

7.1. Reflow Soldering Profile

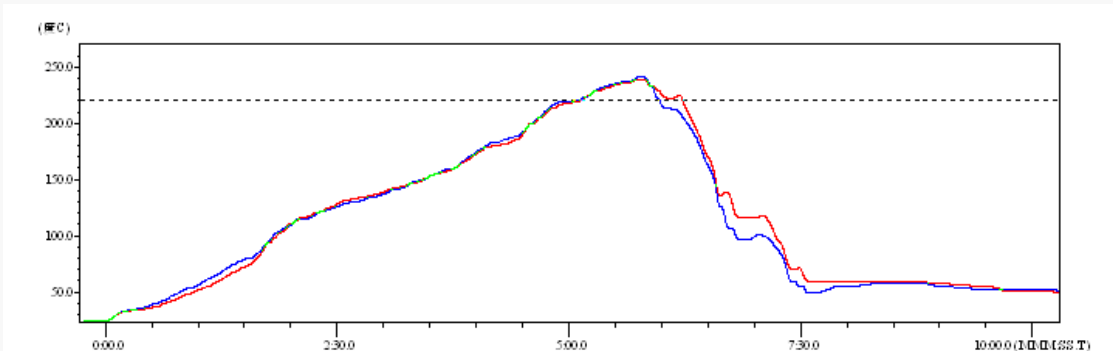


Figure 11. Reflow Soldering Profile

Table 24 Reflow Soldering Specification

Number	Item	Temp (°C)	Time(s)
1	Reflow time	220 °C above	35~55 s
2	Peak temp	Max 260°C	

- Remark:**
1. Recommend to supply N2 for reflow oven
 2. O2 content lower than 300ppm;

7.2. Handling Instruction

1. Shelf life in sealed bag: 12 months at <30°C, <60% humidity.
2. After bag opened, device need to be re-baked if window time over 168 hours
3. Recommended to over bake with N2 supplied
4. Recommended to reflow oven with N2 supplied
5. Bake required with 24 hours at 125+5°C,
6. Recommended to store at ≦10% RH with vacuum packing
7. If SMT process needs twice reflow
 - (1) TOP side (2) BOT side

Case 1: BLE module mounted on PCB TOP, need to bake TOP side after BOT side processed 168 hours (window time)

Case 2: BLE module mounted on PCB BOT side ,follow normal bake rule.

Remark: window time means it is been 168 hours since last bake to next flow.

7.3. Shipping Information

Tape& reel

Size :TBD mm



Carton

size: TBD mm (inside)



Figure 12. Package Information

APPENDIX A: UART TRANSPARENT TRANSMIT TYPICAL APPLICATION

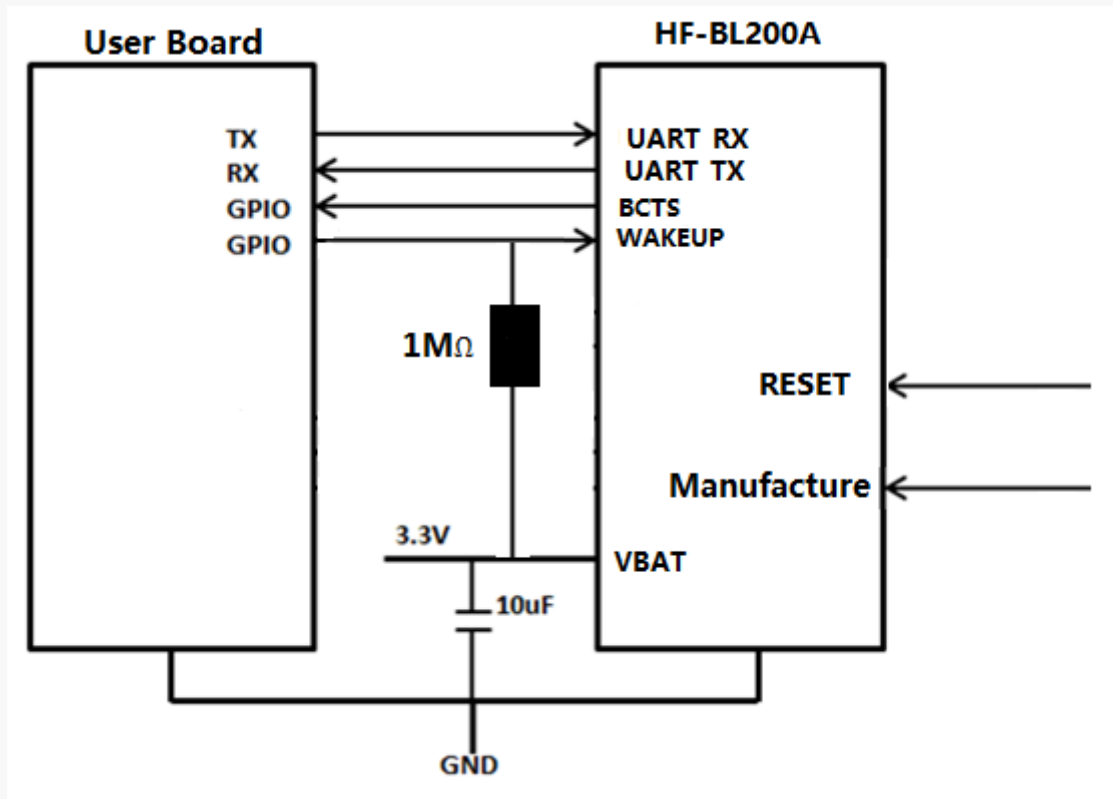


Figure 13. UART Transparent Transmit Typical Application

XRES-----module reset signal, input, low level valid

There is a pull-up resistor to VBAT, no need external pull up resistor. When module power on or failure, MCU will reset the module, pull down at least 10 ms then pull-up.

BCTS----- Module UART output notify signal, output (wakeup external MCU, remind MCU there is data output)

Module will pull down BCTS, and delay Xms before output data; X set by external MCU via At command, in default there is no latency. After sending data, BCTS pull-up, if there is new data sending BCTS will pull-up after sending all the new data. .

WAKEUP----- UART wakeup enable, Stop Mode Wakeup,(recommend to add external Pull-up resistor)

Pin Mode: WAKEUP pin is high resistor input. Can't be suspended, can be controller by pull-up or pull down or external MCU

UART Wakeup Enable: when WAKEUP is pull-up, all system enter into DEEPSLEEP ; when WAKEUP is pull-down, system enter into SLEEP mode. When sending data from external MCU to module, need to pull-down WAKEUP pin, after sending finished, pull-up WAKEUP to lower the power.

There must be a latency from WAKEUP pull-down to MCU sending data, the time is no less 拉低 than 2ms。

Stop Mode Wakeup: WAKEUP pull-down, AT command send out Stop command, when system enter into STOP mode, can be wakeup through pull-up WAKEUP.

UART_TX/RX-----UART data send/receive signal。

Manufacture (RELD) -----restore factory setting via hardware method: keep high level at least 5 second when power on or restore

APPENDIX D: CONTACT INFORMATION

Address: Room 1002 ,#1Building,No.3000 Longdong Avenue, Pudong District, Shanghai, China

Post Code: 201203

Web: www.hi-flying.com

Online consult: 400-189-3108

Email: sales@hi-flying.com

For more information ,please visit <http://www.hi-flying.com/>

© Copyright High-Flying, May, 2011

The information disclosed herein is proprietary to High-Flying and is not to be used by or disclosed to unauthorized persons without the written consent of High-Flying. The recipient of this document shall respect the security status of the information.

The master of this document is stored on an electronic database and is "write-protected" and may be altered only by authorized persons at High-Flying. Viewing of the master document electronically on electronic database ensures access to the current issue. Any other copies must be regarded as uncontrolled copies.