BT1026 programming user guide

Release 2.0

Table of contents

Chapter 1

Introduction

[中文]

1.1 Description

This design guide is suitable for engineers to develop FSC-BT1026 series Bluetooth modules, also suitable for BT806、BT802、BT1006 series modules

1.2 Module Default Settings

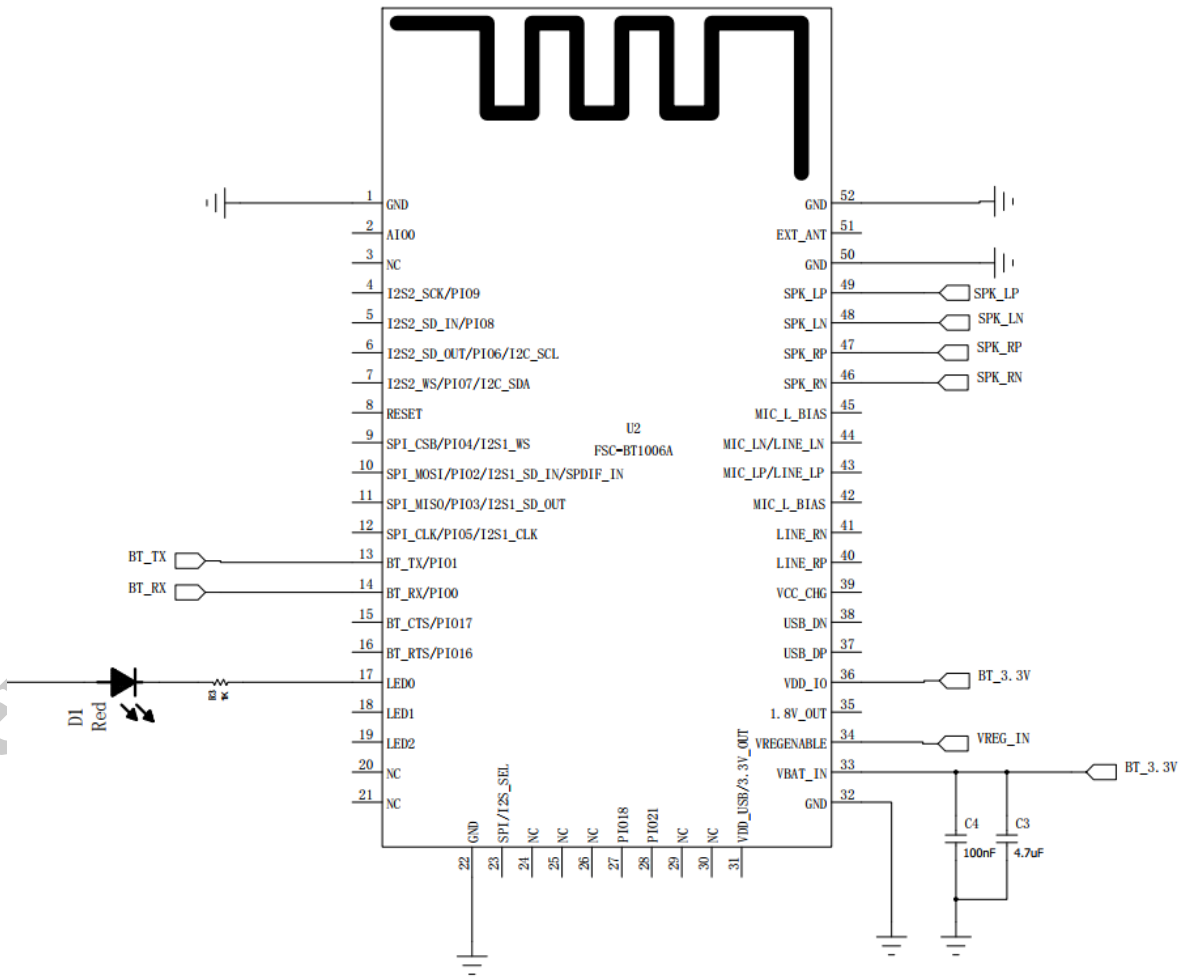
Name	FSC-BT1026-XXXX
LE-Name	FSC-BT1026-LE-XXXX
Pin Code	0000
Secure Simple Pairing Mode	ON
UART Baudrate	115200/8/N/1

Chapter 2

Hardware Description

2.1 Pin Diagram

FSC-BT1006 as an example:



2.2 Pin Description

Pin	Pin Name	Type	Pin Descriptions
4	I2S_CLK	I/O	I2S BCLK
5	I2S_IN	I	I2S DATA IN
6	I2S_OUT	O	I2S DATA OUT
7	I2S_WS	I/O	I2S SYNC
8	RESET	I	External reset input: active Low
13	UART_TX	O	UART TX
14	UART_RX	I	UART RX
15	UART_CTS	I/O	UART CTS
16	UART_RTS	I/O	UART RTS(default: PA mute pin)
17	LED0	I/O	Output square wave in pairing mode, output high level when bluetooth is connected
18	LED1	I/O	SPP/GATT is not connected to output low level, connected to output high level
32	GND	GND	GND
33	VDD	VDD	Power supply for I/O ports, DC 3.3V
34	VREG_IN	I	Typically connected to an ON/OFF push button
36	VDD_IO	I	pio supply
43	MIC_LP	Audio	MIC0/Line_IN differential L input, positive
44	MIC_LN	Audio	MIC0/Line_IN differential L input, negative
45	MIC_BIAS	Audio	MIC Power Supplies
46	SPK_RN	Audio	Headphone/speaker differential R output, negative
47	SPK_RP	Audio	Headphone/speaker differential R output, positive
48	SPK_LN	Audio	Headphone/speaker differential L output, negative
49	SPK_LP	Audio	Headphone/speaker differential L output, positive
51	EXT_ANT	ANT	Change the 0 ohm resistance near the antenna, you can connect an external Bluetooth antenna

2.3 Hardware Design Notes

- The simple test of the module only needs to connect VDD/VDD_IO/VREG_IN/GND/UART_RX/UART_TX to use
- VREG_IN defaults to ON/OFF push button. Power on: short press for 1 second. Power off: long press for 3 seconds.
- If no ON/OFF push button is required, the power-on delay is 100ms before pulling up VREG
- After drawing the schematic diagram, please send it to Feasycom for review,so as to avoid the Bluetooth distance not reaching the best effect

Chapter 3

Function Description

3.1 Compare related products

Module	BT1026A	BT1026B	BT1026C	BT1026D	BT1026E	BT1035	BT806A	BT806B	BT1006A
Chip	QCC3021	QCC3031	QCC3024	QCC3034	QCC5125	QCC3056	CSR8670	CSR8675	QCC3007
TX&RX	RX	RX	RX	RX	RX	TX	TX&RX	TX&RX	RX
APTX	x	√	x	√	√	√	√	√	x
APTX-HD	x	√	x	√	√	√	√	√	x
APTX-LL	x	√	x	√	√	√	√	√	x
APTX-AD	x	x	x	x	√	√	x	x	x
LDAC	x	x	x	x	√	x	x	x	x
CVC	HandsFree	HandsFree	Headset	Headset	HandsFree	x	HandsFree	HandsFree	HandsFree
Upgrade	OTA/UART	OTA/UART	OTA/UART	OTA/UART	OTA/UART	OTA/UART	UART	UART	UART
Version	5.1	5.1	5.1	5.1	5.1	5.2	5.0	5.0	5.0

Note:

- BT1006 series modules have been discontinued. Corresponding replacement modules:BT1026
- BT1026B, BT1026D APTX-LL need to purchase license, the default firmware does not support. If you want to test APTX-LL, please contact Feasycom.
- BT1026E aptx series decoding and cvc need to purchase license

3.2 Profiles & Features

- SPP (Serial Port Profile)
- GATTS (Generic Attribute Profile LE-Peripheral role)

- GATTC (Generic Attribute Profile LE-Central role)
- HFP-HF (Hands-Free Profile)
- HFP-AG (Hands-Free-AG Profile)
- A2DP-Sink (Advanced Audio Distribution Profile)
- A2DP-Source (Advanced Audio Distribution Profile)
- AVRCP-Controller (Audio/Video remote controller Profile)
- AVRCP-Target (Audio/Video remote controller Profile)
- HID-DEVICE (Human Interface Profile)
- PBAP (Phonebook Access Profile)

3.3 GATT Default service and characteristic

Type	UUID	Characteristic	Description
Service	0xFFF0		throughput services
Write	0xFFF2	Write，Write Without Response	app send to module
Notify	0xFFF1	Notify	module send to app

Chapter 4

Command Description

4.1 Terms

- Throughout this specification:
- Content between { } is optional
- Content behind << represents a COMMAND from Host
- Content behind >> represents a RESPONSE/EVENT to Host

4.2 Command Format

AT+Command{=Param1{,Param2{,Param3…}}}<CR><LF>

- All commands start with “AT”, end with <CR><LF>
- <CR> means “carriage return”, corresponds to hex value 0x0D
- <LF> means “line feed”, corresponds to hex value 0x0A
- If Command has Parameter, Parameter follows behind ‘=’
- If Command has multiple Parameters, Parameter must be separated by ‘,’
- If Command has Response, Response starts with <CR><LF>, ends with <CR><LF>
- Module will always report command’s execution result by using “OK” for success or “ERROR” for failure

Example:Read module’s BR/EDR local name

<< AT+NAME

>> +NAME=FSC-BT1026-XXXX

>> OK

Example:Pick up an incoming call when no call incoming actually

<< AT+HFPANSW

>> ERROR

4.3 Event Format

<CR><LF>+Indication{=Param1{,Param2{,Param3…}}}<CR><LF>

- All Events start with <CR><LF>, end with <CR><LF>
- If Event has Parameter, Parameter follow behind ‘=’
- If Event has multiple Parameters, Parameter must be separated by ‘,’

Example: Received “1234567890” from mobile phone via SPP profile

>> +SPPDATA=10,1234567890

Example: Dial number “10086” using a mobile phone when HFP connected

>> +HFPSTAT=4

>> +HFPAUDIO=1

>> +HFPCID=10086

>> +HFPSTAT=6

Chapter 5

Commands Table

5.1 General Commands

5.1.1 AT+HELP - Firmware Function/Command Summary

Command	**AT+HELP**
Response	<FIRMWARE FUNCTION: appropriate working scenario > <OTA PATH: latest suitable firmware path on server for upgrade On-The-Air> <ENABLED PROFILES: LINKS: ON/OFF> … … <COMMAND SUMMARY: DESCRIPTION: PROFILE CATEGORY>
Description	Using help command to get the basic summary information

5.1.2 AT+VER - Get Firmware Version

Command	**AT+VER**
Response	**+VER=Param1,Param2,Param3**
`Param1`	Module type
`Param2`	Firmware version
`Param3`	Firmware Compile time

Example:

<< AT+VER

>> +VER=BT1036,V2.6.1,20220922

>> OK

5.1.3 AT+BAUD - Get/Set Uart Baudrate

Command	**AT+BAUD{=Param}**
`Param`	9600/19200/38400/57600/115200(default)/230400/460800/921600
Response	**+BAUD=Param**
Description	Module will change baudrate to target value immediately or after reboot depending on firmware

Example:Read module’s baudrate

<< AT+BAUD

>> +BAUD=115200

>> OK

<< AT+BAUD=921600

>> OK

Module switch baudrate to 921600 immediately, host change baudrate as well, query module’s name at new baudrate

<< AT+NAME

>> +NAME=FSC-BT1036-1F26

>> OK

5.1.4 AT+I2SCFG - Get/Set I2S Settings

Command	**AT+I2SCFG{=Param}**
Param	A base-10 representation of a bit field, default:0, for each bit
BIT[0]	0:disable; 1:enable
BIT[1]	0:master; 1:slave
BIT[2]	0:FS=48000Hz;1:FS=44100Hz
BIT[3]	0:left justified; 1:right justified
BIT[4]	0:data 1 bit delay; 1:data no delay
BIT[5-6]	00: bit depth=16bits 01: bit depth=24bits 10: bit depth=32bits
BIT[7]	0:WS at high level，SD data is right channel 1:WS at high level，SD data is left channel
Response	**+I2SCFG=Param**
Note	BIT[4] is valid only at the condition of BIT[3]=0. BIT[7] Partial firmware support

Example:usual configuration and description

1	I2S Master; Sample rate=48000Hz; Resolution=16bits; Bit clock= 48000*16*2ch=1.536Mhz
3	I2S Slave; Sample rate=48000Hz; Resolution=16bits; Bit clock= 48000*16*2ch=1.536Mhz
65	I2S Master; Sample rate=48000Hz; Resolution=32bits; Bit clock= 48000*32*2ch=3.072Mhz
67	I2S Slave; Sample rate=48000Hz; Resolution=32bits; Bit clock= 48000*32*2ch=3.072Mhz

5.1.5 AT+SPDIFCFG - SPDIF Format Configuration

Command	**AT+SPDIFCFG{=Param}**
`Param`	0-Disable SPDIF for audio output 1-Enable SPDIF for audio output
Description	SPDIF output only support: BT802、BT806、BT1026A、BT1026C、BT1026E If the audio output is spdif, the I2S configuration is invalid.

5.1.6 AT+MICGAIN - Get/Set Analog Input Gain

Command	**AT+MICGAIN{=Param}**
`Param`	Gain（‘+’/ ‘- ‘)
Description	Adjust input gain

5.1.7 AT+SPKVOL - Get/Set Analog Output Volume

Command	**AT+SPKVOL{=Param}**
`Param`	Volume（‘+’/ ‘-’, default:15)
Description	Adjust Output volume, Adjust A2DP volume when play music,adjust HFP volume when talking

5.1.8 AT+REBOOT - Soft Reboot

Command	**AT+REBOOT**
Response	**OK**
Description	Module release all Bluetooth connections with remote device then re-boot

5.1.9 AT+RESTORE - Restore Factory Settings

Command	**AT+RESTORE**
Response	**OK**
Description	Module restore all factory settings then reboot

5.1.10 AT+BTEN - Bluetooth On/Off

Command	**AT+BTEN{=Param}**
`Param`	0-Power off 1-Power on

5.1.11 AT+PROFILE - Bluetooth Profile Selection

Command	**AT+PROFILE{=Param}**
`Param`	A base-10 representation of a bit field, for each bit:
`BIT[0]`	SPP (Serial Port Profile)
`BIT[1]`	GATT Server (Generic Attribute Profile)
`BIT[2]`	GATT Client (Generic Attribute Profile)
`BIT[3]`	HFP-HF (Hands-Free Profile Handsfree)
`BIT[4]`	HFP-AG (Hands-Free Profile Audio Gateway)
`BIT[5]`	A2DP Sink (Advanced Audio Distribution Profile)
`BIT[6]`	A2DP Source (Advanced Audio Distribution Profile)
`BIT[7]`	AVRCP Controller (Audio/Video remote controller Profile)
`BIT[8]`	AVRCP Target (Audio/Video remote controller Profile)
`BIT[9]`	HID Keyboard (Human Interface Profile)
`BIT[10]`	PBAP Server (Phonebook Access Profile)
Response	**+PROFILE=Param**
Description	GATT Server and Client, HFP Sink and Source, A2DP Sink and Source, AVRCP Controller and Target cannot be enabled simultaneously because of mutual exclusion.

Example:Read current profile selection

<< AT+PROFILE

>> +PROFILE=1195

Example:Enable SPP, GATT Server, HFP Source, A2DP Source profile, disable the others

<< AT+PROFILE=83

>> OK

Example:Enable SPP, A2DP Sink, disable the others

<< AT+PROFILE=33

>> OK

5.1.12 AT+AUTOCONN - Turn On/Off Power On Auto Reconnect

Command	**AT+AUTOCONN{=Param}**
Param	(0~15, default:3) (0) Turn Off (1-15) Turn on and reconnect times
Response	**+AUTOCONN=Param**
Description	Module will attempt to connect last device after power on if set.

5.1.13 AT+STAT - Get All Profile State

Command	**AT+STAT**
Response	**+STAT=Param1, Param2, Param3…**
Param1	DEVSTAT
Param2	SPPSTAT
Param3	GATTSTAT
Param4	HFPSTAT
Param5	A2DPSTAT
Param6	AVRCPSTAT
Param7	HIDSTAT
Param8	PBSTAT

5.1.14 AT+DEVSTAT - Read Device State

Command	**AT+DEVSTAT**
Response	**+DEVSTAT=Param**
`Param`	A base-10 representation of a bit field, for each bit:
`BIT[0]`	0: Power Off; 1: Power On
`BIT[1]`	0: BR/EDR Not Discoverable; 1: BR/EDR Discoverable
`BIT[2]`	0: BLE Not Advertising; 1: BLE Advertising
`BIT[3]`	0: BR/EDR Not Scanning; 1: BR/EDR Scanning
`BIT[4]`	0: BLE Not Scanning; 1: BLE Scanning

Example:usual combination and description

0	Device power off
1	Device power on
3	Device power on, BR/EDR Discoverable
5	Device power on, BR/EDR Not Discoverable, BLE Advertising
7	Device power on, BR/EDR Discoverable, BLE Advertising
13	Device power on, BR/EDR Not Discoverable, BLE Advertising, Scanning nearby BR/EDR devices

5.1.15 AT+ADDR - Get BR/EDR MAC Address

Command	**AT+ADDR**
Response	**+ADDR=Param**
`Param`	Module’s BR/EDR MAC address (12 Bytes ASCII)

Example:

<< AT+ADDR

>> +ADDR=DC0D30010203

>> OK

5.1.16 AT+LEADDR - Get BLE MAC Address

Command	**AT+LEADDR**
Response	**+LEADDR=Param**
`Param`	Module’s LE MAC address (12 Bytes ASCII)

5.1.17 AT+NAME - Get/Set BR/EDR Local Name

Command	**AT+NAME{=Param1{,Param2}}**
`Param1`	BR/EDR local name(1~31 Bytes ASCII)
`Param2`	MAC address suffix(0/1,default:1) 0: Disable suffix 1: Enable suffix “-XXXX”(lower 4 bytes of MAC address) after local name
Response	**+NAME=Param**
Description	Write local name if parameter exist, otherwise read current local name

Example:Read current BR/EDR local name

<< AT+NAME

>> +NAME=FSC-BT1036-XXXX

>> OK

Example:Change module’s BR/EDR local name to “ABC”,and disable suffix

<< AT+NAME=ABC,0

>> OK

Example:Change module’s BR/EDR local name to “ABC”and enable suffix

<< AT+NAME=ABC,1

>> OK

5.1.18 AT+LENAME - Get/Set BLE Local Name

Command	**AT+LENAME{=Param1{,Param2}}**
Param1	BLE local name(1~25 Bytes ASCII)
Param2	MAC address suffix(0/1,default:1) 0-Disable suffix 1-Enable suffix “-XXXX”(lower 4 bytes of MAC address) after local name
Response	**+LENAME=Param**

5.1.19 AT+SSP - Get/Set BR/EDR Pairing Mode

Command	**AT+SSP{=Param}**
Param	Pairing mode (0~3, default:2) (0) Legacy pairing, use pin code for pairing (1) Secure simple pairing, auto pairing (2) Secure simple pairing, display yes/no in pairing
Response	**+SSP=Param**
Note	need reboot

5.1.20 AT+PIN - Get/Set BR/EDR Pin Code

Command	**AT+PIN{=Param}**
Param	Pin code(4~15 Bytes ASCII, default:0000)
Response	**+PIN=Param**
Description	Pin code only work in legacy pairing mode, see AT+SSP

Example:Read module’s pin code

<< AT+PIN

>> +PIN=0000

>> OK

Example: Change module’s pin code to “1234”

<< AT+PIN=1234

>> OK

5.1.21 AT+COD: Get/Set Device Class

Command	**AT+COD=Param**
`Param`	Class of device(6 bytes ASCII, default:240408 Handsfree device)
Response	**+COD=Param**

Related configuration reference: COD.

5.1.22 AT+PAIR: Get/Set BR/EDR/BLE Visibility

Command	AT+PAIR=Param
`Param`	Mode(0-1) 0: Leave BR/EDR/BLE discoverable mode (stop advertising/broadcasting) 1: Enter BR/EDR/BLE discoverable mode (start advertising/broadcasting)
Description	Module will always be discoverable if no device connected (BR/EDR or BLE), and be undiscoverable if connected with remote device, unless received this command

5.1.23 AT+SCAN - Scan Nearby Devices

Command	AT+SCAN=Param
`Param`	scan type(0~1) 0: Stop scan 1: Scan nearby BR/EDR devices
Description	Format description reference: *+SCAN - Scan Result*

5.1.24 AT+PLIST - Get/Delete Paired List

Command	**AT+PLIST{=Param}**
Param	(0/1~8/12 Bytes MAC address) (0) Clear all paired record (1~8) Clear specific paired record with index (MAC) Clear specific paired record with MAC address
Response1	**+PLIST=Param1, Param2, Param3{,Param4}**
Param1	(1~8) Paired device’s index
Param2	(MAC) Paired device’s MAC address
Param3	(UTF8) Paired device’s name
Response2	**+PLIST=E** End of the paired record

Example:Read module’s paired record

<< AT+PLIST

>> +PLIST=1,1C5CF226D773, iPhone12

+PLIST=2, A0BC30075421, Samsung S8

+PLIST=E

>> OK

Example:Clear module’s paired record

<< AT+PLIST=0

>> OK

5.1.25 AT+DSCA - Release All Connections

Command	**AT+DSCA**
Description	Module release all Bluetooth connections with remote device

5.1.26 AT+TPMODE - Turn On/Off Throughput Mode

Command	**AT+TPMODE{=Param}**
Param	Throughput mode(0~1，default:0）0: Turn Off 1: Turn On
Response	**+TPMODE=Param**
Description	When SPP/GATT profile connected and throughput mode is on, the AT command will be de-active, every byte received via physical UART will be sent to air, vice visa

5.1.27 AT+AUXCFG - Audio Input Mode Configuration

Command	**AT+AUXCFG=Param1**
Param1	mode(0-3,default:) 0: BT Mode 1: LineIn Mode 2: Spdif Mode 3: I2S Mode
Description	This command is used to configure the input mode of the module, BT1006A, BT1026C, BT1026D do not support spdif After BT1026A, BT1026B,are set to spdif, PA_MUTE pin will be modified to 20 pin

Example:Set LineIn mode input

<< AT+AUXCFG=1

>> OK

Some old firmware use AT+LINECFG=1 command to set

5.1.28 AT+PRINT - Turn on/off reporting data

Command	**AT+PRINT{=Param}**
Param	0-off 1-on 。default: Open report data

5.1.29 AT+TONEPLAY - play tone

Command	**AT+TONEPLAY{=Param}**
Param	tone(0-94)

5.1.30 AT+MUTEPIO - Set PA_MUTE pin

Command	**AT+MUTEPIO{=Param}**
Param	PIO(0-63, 1026C default:PIO23)

5.1.31 AT+MICMUTE - mute MIC

Command	**AT+MICMUTE=Param**
Param	0-unmute 1-mute
Description	mute mic when call active

5.1.32 AT+SPKMUTE - mute Speaker

Command	**AT+MICMUTE=Param**
Param	0-unmute 1-mute

5.2 HFP Commands

5.2.1 AT+HFPSTAT - Read HFP State

Command	**AT+HFPSTAT**
Response	**+HFPSTAT=Param**
Description	Format description reference: *+HFPSTAT - HFP State*

5.2.2 AT+HFPRES - Get/Set HFP Sample Rate

Command	**AT+HFPRES{=Param}**
`Param`	Sample rate (Hz) for HFP voice call, 0:16k 1-48K
Response	**+HFPRES=Param**

5.2.3 AT+HFPCONN - Establish HFP Connection

Command	**AT+HFPCONN{=Param}**
`Param`	MAC address of target device (12 Bytes ASCII)
Description	Reconnect to last HFP device if parameter not exist

Example:Connect to last HFP device

<< AT+HFPCONN

>> OK

Example:Connect to specific HFP device with MAC address

<< AT+HFPCONN=1C5CF226D773

>> OK

5.2.4 AT+HFPDISC - Release HFP Connection

Command	**AT+HFPDISC**
Description	Release current HFP connection with remote device

5.2.5 AT+HFPDIAL - Dial/Redial Phone Number

Command	**AT+HFPDIAL{=Param}**
`Param`	Phone number (1~25 Bytes ASCII)
Description	Dial specific number if parameter exist, otherwise redial

Example:Redial

<< AT+HFPDIAL

>> OK

Example:Dial number “075527924639”

<< AT+HFPDIAL=075527924639

>> OK

5.2.6 AT+HFPDTMF - Send DTMF code

Command	**AT+HFPDTMF{=Param}**
`Param`	**DTMF (0~9/#/*)**

Example:Send DTMF code “#” while talking

<< AT+HFPDTMF=#

>> OK

5.2.7 AT+HFPANSW - Pick Up Incoming Call

Command	AT+HFPANSW
Description	Pick up an incoming call

5.2.8 AT+HFPCHUP - Reject/Hung up Call

Command	AT+HFPCHUP
Description	Reject incoming call or hung up outgoing/active call

5.2.9 AT+HFPADTS - Transfer Voice Audio Between Local and Remote Device

Command	AT+HFPADTS=Param
`Param`	0: Transfer voice audio from module to remote device 1: Transfer voice audio from remote device to module
Description	Transfer voice audio between module and remote device by default if no parameter set

5.2.10 AT+HFPVR - Start/Stop Voice Recognition of Remote Device

Command	AT+HFPVR=Param
`Param`	0-Stop 1-Start
Description	Start/Stop Voice Recognition of Remote Device (such as Siri for iOS devices)

5.2.11 AT+HFPSCO - HFP SCO Configuration

Command	AT+HFPSCO=Param
Param	0-default 1-always to HF 2-always to AG

5.2.12 AT+HFPBATT - Send device battery level

Command	AT+HFPBATT=Param
Param	level(0-9)

5.3 A2DP/AVRCP Commands

5.3.1 AT+A2DPSTAT - Read A2DP State

Command	AT+A2DPSTAT
Response	+A2DPSTAT=Param
Description	Format description reference: *+A2DPSTAT - A2DP State*

5.3.2 AT+A2DPCONN - Establish A2DP Connection

Command	AT+A2DPCONN{=Param}
Param	MAC address of target device (12 Bytes ASCII)
Description	Reconnect to last A2DP device if parameter not exist

5.3.3 AT+A2DPDISC - Release A2DP Connection

Command	AT+A2DPDISC
Description	Release current A2DP connection with remote device

5.3.4 AT+A2DPCFG - Read/Write A2DP Configuration

Command	AT+A2DPCFG=Param
Param	A base-10 representation of a bit field, for each bit
BIT[0]	0: Disable AAC Codec 1: Enable AAC Codec
BIT[1]	0: Disable APTX Codec 1: Enable APTX Codec
BIT[2]	0: Disable APTX-LL Codec 1: Enable APTX-LL Codec
BIT[3]	0: Disable APTX-HD Codec 1: Enable APTX-HD Codec
BIT[4]	0: Disable APTX-AD Codec 1: Enable APTX-AD Codec
BIT[5]	0: Disable LDAC Codec 1: Enable LDAC Codec

Example:Read current A2DP configuration

<< AT+A2DPCFG

>> +A2DPCFG=0

>> OK

Example::Set A2DP configuration to: Only enable AAC Codec

<< AT+A2DPCFG=1

>> OK

5.3.5 AT+A2DPDEC - Read A2DP Decoder

Command	AT+A2DPDEC
Response	+A2DPDEC=Param
Param	1:SBC 3:AAC 5:APTX 7:APTX-HD 8:APTX-LL 9:APTX-AD 10:LDAC

5.3.6 AT+A2DPENC - Read A2DP Encoder

Command	AT+A2DPENC
Response	+A2DPENC=Param
`Param`	1:SBC 3:AAC 5:APTX 7:APTX-HD 8:APTX-LL 9:APTX-AD 10:LDAC

5.3.7 AT+A2DPAUDIO - Establish/Release A2DP Audio Connection

Command	AT+A2DPAUDIO{=Param}
`Param`	0-Release A2DP audio connection with remote a2dp sink device 1-Establish A2DP audio connection with remote a2dp sink device
Note	A2DP SOURCE only supported

5.3.8 AT+AVRCPSTAT - Read AVRCP State

Command	AT+AVRCPSTAT
Response	+AVRCPSTAT=Param
Description	Format description reference: *+AVRCPSTAT - AVRCP State*

5.3.9 AT+AVRCPCFG - Get/Set AVRCP Configuration

Command	AT+AVRCPCFG{=Param}
`Param`	A base-10 representation of a bit field, default:3, for each bit:
`BIT[0]`	Auto get track ID3 information (title, artist, album) on track changed.default:1
`BIT[1-3]`	Auto get track play progress if value > 0. default:5 second

Example: Read AVRCP configuration

<< AT+AVRCPCFG

>> +AVRCPCFG=1

OK

Example: Get track play progress every 1 second

<< AT+AVRCPCFG=3

>> OK

5.3.10 AT+PLAYPAUSE - Track Play/Pause

Command	**AT+PLAYPAUSE**
Description	Send play or pause command to remote media player according to current play status

5.3.11 AT+PLAY - Track Play

Command	**AT+PLAY**
Description	Send play command to remote media player

5.3.12 AT+PAUSE - Track Pause

Command	**AT+PAUSE**
Description	Send pause command to remote media player

5.3.13 AT+STOP - Track Stop

Command	**AT+STOP**
Description	Send stop command to remote media player

5.3.14 AT+FORWARD - Track Forward

Command	AT+FORWARD
Description	Send forward command to remote media player

5.3.15 AT+BACKWARD - Track Backward

Command	AT+BACKWARD
Description	Send backward command to remote media player

5.3.16 AT+FFWD - Track FastForward

Command	AT+FFWD=Param
`Param`	0-Fast Forward Release, 1-Fast Forward Press
Description	Send fast forward command to remote media player

5.3.17 AT+RWD - Track Rewind

Command	AT+RWD=Param
`Param`	0-Rewind Release, 1-Rewind Press
Description	Send rewind command to remote media player

5.4 PBAP Commands

5.4.1 AT+PBSTAT - Read PBAP state

Command	AT+PBSTAT
Response	+PBATAT=Param
Description	Format description reference: *+PBSTAT - PBAP State*

5.4.2 AT+PBCONN - Establish PBAP Connection

Command	**AT+PBCONN{=Param}**
`Param`	MAC address of target device (12 Bytes ASCII)
Description	Module will use current HFP device’ MAC address if parameter not exist For some firmware release, module will establish PBAP connection automatically on received command AT+PBDOWN

5.4.3 AT+PBDISC - Release PBAP Connection

Command	**AT+PBDISC**
Description	Release current PBAP connection with remote device

5.4.4 AT+PBDOWN - Download Phonebook

Command	Param1{,Param2}
Param1	Phonebook type(0-5) (0) Phonebook (SIM Storage) (1) Phonebook (Phone Storage) (2) Received call log (3) Dialed call log (4) Missed call log (5) All call log
Param2	Max items (1~65535, default:3000 for phonebook; 50 for call log)
Description	For some phones (e.g. iPhone), the contact download permission must be turned on in phone’s Bluetooth settting refer to application note for more description: *Phonebook downloading*

5.5 SPP Commands

5.5.1 AT+SPPSTAT - Read SPP State

Command	AT+SPPSTAT
Response	+SPPATAT=Param
Description	Format description reference: *+SPPSTAT - SPP State*

5.5.2 AT+SPPCONN - Establish SPP Connection

Command	AT+SPPCONN{=Param}
Param	MAC address of target device (12 Bytes ASCII)

5.5.3 AT+SPPDISC - Release SPP Connection

Command	**AT+SPPDISC**
Description	Release current SPP connection with remote device

5.5.4 AT+SPPSEND - Send Data Via SPP

Command	**AT+SPPSEND=Param1,Param2**
`Param1`	Payload length (1~492)
`Param2`	Payload (1~492 Bytes UTF8)
Description	If throughput mode is on, this command is de-active

Example: Send data “1234567890” to remote device via SPP

<< AT+SPPSEND=10,1234567890

>> OK

5.6 GATT Commands

5.6.1 AT+GATTSTAT - Read GATT State

Command	**AT+GATTSTAT**
Response	+GATTATAT=Param
Description	Format description reference: *+GATTSTAT - GATT State*

5.6.2 AT+GATTDISC - Release GATT Connection

Command	**AT+GATTDISC**
Description	Release current GATT connection with remote device

5.6.3 AT+GATTSEND - Send Data Via GATT

Command	**AT+GATTSEND=Param1,Param2**
`Param1`	Payload length (1~492)
`Param2`	Payload (1~492 Bytes UTF8)
Description	If throughput mode is on, this command is de-active

Example: Send data “1234567890” to remote device via GATT

<< AT+SPPSEND=10,1234567890

>> OK

5.7 HID Commands

5.7.1 AT+HIDSTAT - Read HID State

Command	**AT+HIDSTAT**
Response	+HIDATAT=Param
Description	Format description reference: *+HIDSTAT - HID State*

5.7.2 AT+HIDCONN - Establish HID Connection

Command	**AT+HIDCONN{=Param}**
`Param`	MAC address of target device (12 Bytes ASCII)

5.7.3 AT+HIDDISC - Release HID Connection

Command	**AT+HIDDISC**
Description	Release current HID connection with remote device

5.7.4 AT+HIDMODE - Get/Set HID Input Mode

Command	AT+HIDMODE{=Param}
Param	HID keyboard input mode (0~1), default 1 (0) Hex key code (1) Ascii key code (English)
Note	Module can support various keyboard language with specify firmware, such as:TURKEY SPAIN PORTUGAL FRANCE GERMANY ITALY CZECH JAPAN

5.7.5 AT+HIDREL - Read/write HID key value auto release

Command	AT+HIDREL{=Param}
Param	HID Key value auto release(0~1), default 1 (0) manual release (1) auto release
Note	If HIDREL=0, the user needs to send \x00 \x00 to manually send the bounce key

5.7.6 AT+HIDDLY - Get/Set HID Report Period

Command	AT+HIDDLY{=Param}
Param	HID report period in millisecond, default 10 ms

5.7.7 AT+HIDSEND - Send HID Keyboard Report

Command	**AT+HIDSEND=Param1,Param2**
`Param1`	Report length
`Param2`	Report payload
Note	For special key code: 0x0D -> ENTER 0x08 -> BACKSPACE 0x09 -> TAB 0x20 -> SPACE

Example: Send key code ‘A’ to remote device (on AT+HIDMODE=1)

>> AT+HIDSEND=1,A

<< OK

Example: Send key code ‘A’ to remote device (on AT+HIDMODE=0)

<< AT+HIDSEND=4, x00 x04

>> OK

Note: As payload is hex value, hence actual command is:

41 54 2B 48 49 44 53 45 4E 44 3D 34 2C 00 04 0d 0a

Where:

00 : modifier

04 : key code

Module will auto send debounce key code by itself

5.7.8 AT+HIDCMD - Send HID User Report

Command	AT+HIDCMD=Param
Param	2 bytes hid user report e.g., for iPhone: Play/Pause: 00 CD Stop: 00 B7 Forward: 00 B5 Backward: 00 B6 Fast Forward: 00 B3 Rewind:00 B4 Record:00 B2 VolumpUp:00 E9 VolumpDn:00 EA Mute:00 E2 On screen keyboard Toggle:01 AE

Example: Send Volume Up to iPhone

<< AT+HIDCMD= x00 xE9

>> OK

Note: As the payload is hex value, hence actual command is:

41 54 2B 48 49 44 43 4D 44 3D 00 E9 0D 0A

Chapter 6

Events Table

6.1 General Events

6.1.1 +SCAN - Scan Result

Format1	**+SCAN =Param1,Param2,Param3, Param4,Param5,Param6**
Param1	Index
Param2	Device address type (0~2) (0) LE public address (1) LE random address (2) BR/EDR address
Param3	MAC address (12 Bytes ASCII)
Param4	RSSI (-127 ~ -1)
Param5	Size of Param6 if exist
Param6	Device Name for BR/EDR devices or advertising data for LE devices

Example: Scan BR/EDR nearby devices

<< AT+SCAN=1

>> OK

>> +SCAN=1,3,B019C66209FA,-32,9,wt-iphone

>> +SCAN=2,0,DC0D30000053,-74,5,BW226

>> +SCAN=3,0,00158354F994,-43,9,LAPTOP-3L

6.1.2 +PAIRED - Pair Result

Format	+PAIRED=Param
Param	MAC address (12 Bytes ASCII) of current pairing device

6.2 HFP Events

6.2.1 +HFPSTAT - HFP State

Format	+HFPSTAT=Param
Param1	(0~6) (0) Unsupported (1) Standby (2) Connecting (3) Connected (4) Outgoing call (5) Incoming call (6) Active call

6.2.2 +HFPDEV - HFP Remote Device Information

Format	+HFPDEV=Param1{,Param2}
Param1	(12 Bytes ASCII), Remote MAC address of current HFP connection
Param2	(UTF8), Remote device name of current HFP connection

Example: HFP connect success with device

>> +HFPDEV=1C5CF226D774, iPhone

6.2.3 +HFPCID - Incoming/Outgoing Call Number

Format	+HFPCID=Param
Param	(1~25 Bytes ASCII), Call number

Example: Dial number 10086

<< AT+HFPDIAL=10086

>> +HFPSTAT=4

>> +HFPCID=10086

>> +HFPCIE=China Mobile

>> +HFPAUDIO=1

Example: Incoming call with number 13265463800

>> +HFPSTAT=5

>> +HFPCID=13265463800

>> +HFPCIE=Jerry

>> +HFPAUDIO=1

6.2.4 +HFPCIE - Incoming/Outgoing Call Name

Format	+HFPCIE=Param
Param	(UTF8), Call name
Note	This event is supported by IOS mobile phones, but not supported by most Android phones

6.2.5 +HFPAUDIO - HFP Voice Audio State

Format	+HFPAUDIO=Param
Param	(0) HFP voice audio disconnected, audio input/output routed to remote device (1) HFP voice audio connected, audio input/output routed to module

6.2.6 +HFPSIG - HFP Remote Device Network Signal Strength

Format	+HFPSIG=Param
Param	(0~5)Network signal strength of remote device

6.2.7 +HFPROAM - HFP Remote Device Roaming State

Format	+HFPROAM=Param
Param	(0/1) Roaming state of remote device

6.2.8 +HFPBATT - HFP Remote Device Battery Level

Format	+HFPBATT=Param
Param	(0~5) Battery level of remote device

6.2.9 +HFPNET - HFP Remote Device Network Operator Selection

Format	+HFPNET=Param
Param	（UTF8）Network operator selection of remote device

6.2.10 +HFPMANU - HFP Remote Device Manufacture

Format	+HFPMANU=Param
Param	（UTF8）Manufacture name of remote device

6.2.11 +HFPNUM - HFP Remote Device Phone Subscriber Number

Format	+HFPNUM=Param
Param	(ASCII)Phone subscriber number of remote device

6.2.12 +HFPIBR - HFP Remote Device In-band-ring Support

Format	+HFPIBR=Param
Param	(0/1) In-band-ring support
Description	Report whether the current connected phone support in-band-ring

6.3 A2DP/AVRCP Events

6.3.1 +A2DPSTAT - A2DP State

Format	+A2DPSTAT=Param
Param	(0) Unsupported (1) Standby (2) Connecting (3) Connected (4) Streaming

6.3.2 +A2DPDEV - A2DP Remote Device Information

Format	+A2DPDEV=Param1{,Param2}
Param1	(12 Bytes ASCII), Remote device’s MAC address of current A2DP connection
Param2	(UTF8), Remote device’s name of current A2DP connection

6.3.3 +AVRCPSTAT - AVRCP State

Format	+AVRCPSTAT=Param
Param	(0) Unsupported (1) Standby (2) Connecting (3) Connected

6.3.4 +PLAYSTAT - Media Player State

Format	+PLAYSTAT=Param
Param	(0) Stopped (1) Playing (2) Paused (3) Fast Forwarding (4) Fast Rewinding

6.3.5 +TRACKSTAT - Media Player Play Progress

Format	+TRACKSTAT=Param1,Param2,Param3
Param1	(0~4), Media Player State, see +PLAYSTAT
Param2	(Decimal ASCII),Elapsed time of current track in millisecond
Param3	(Decimal ASCII),Total time of current track in millisecond

Example: Read media player play progress every 1s

>> +TRACKSTAT=1,54000,322000

>> +TRACKSTAT=1,55000,322000

>> +TRACKSTAT=1,56000,322000

6.3.6 +TRACKINFO - Media Track Information

Format	+TRACKINFO=Param1,Param2,Param3
Param1	title
Param2	artist
Param3	album

Example: Phone playing song “Creep-Radio Head”

>> +TRACKINFO=Creep , Radiohead , Pablo Honey

6.4 Phonebook Access Events

6.4.1 +PBSTAT - PBAP State

Format	+PBSTAT=Param
`Param`	(0) Unsupported (1) Standby (2) Connecting (3) Connected (4) Downloading

6.4.2 +PBCNT - Phonebook Entries of Remote Device

Format	+PBCNT=Param
`Param`	Phonebook entries of remote device

6.4.3 +PBDATA - Phonebook Data

Format1	**+PBDATA=Param1<FF>Param2<FF>Param3{<FF>Param4}**
Param1	Type (0) Phonebook (SIM Storage) (1) Phonebook (Phone Storage) (2) Received call log (3) Dialed call log (4) Missed call log
Param2	Name
Param3	Number
Param4	(15 Bytes ASCII), Call time if current download type is call log Format: Year(4Bytes) Month(2Bytes) Day(2Bytes) T(1Byte) Hour(2Bytes) Minute(2Bytes) Second(2Bytes). e.g. 20161012T152826 represents 2016/10/12/15/28/26
Format2	+PBDATA=E: Download complete
Description	Call time may not exist for some mobile phones

Example: Download all phonebook

<< AT+PBDOWN=1

>> +PBCNT=234

+PBDATA=1<FF>Jack<FF>18219146201

+PBDATA=1<FF>kenan<FF>8613771972680

……

+PBDATA=E

Example: Download 10 dialed call log

<< AT+PBDOWN=3,10

>> +PBDATA=3<FF>China Mobile<FF>10086<FF>20171013T103516

+PBDATA=3<FF>Jerry<FF>18688967507<FF>20171012T152826

……

+PBDATA=E

6.5 SPP Events

6.5.1 +SPPSTAT - SPP State

Format	+SPPSTAT=Param
Param	(0) Unsupported (1) Standby (2) Connecting (3) Connected

6.5.2 +SPPDATA - SPP Received Incoming Data

Format	+SPPDATA=Param1,Param2
Param1	Payload length
Param2	Payload

Example: Received data “1234567890” from remote device via SPP

<< +SPPDATA=10,1234567890

6.6 GATT Events

6.6.1 +GATTSTAT - GATT State

Format	+GATTSTAT=Param
Param	(0) Unsupported (1) Standby (2) Connecting (3) Connected

6.6.2 +GATTDATA - GATT Received Incoming Data

Format	+SPPDATA=Param1,Param2
Param1	Payload length
Param2	Payload

Example: Received data “1234567890” from remote device via GATT

<< +GATTDATA=10,1234567890

6.7 HID Events

6.7.1 +HIDSTAT - HID State

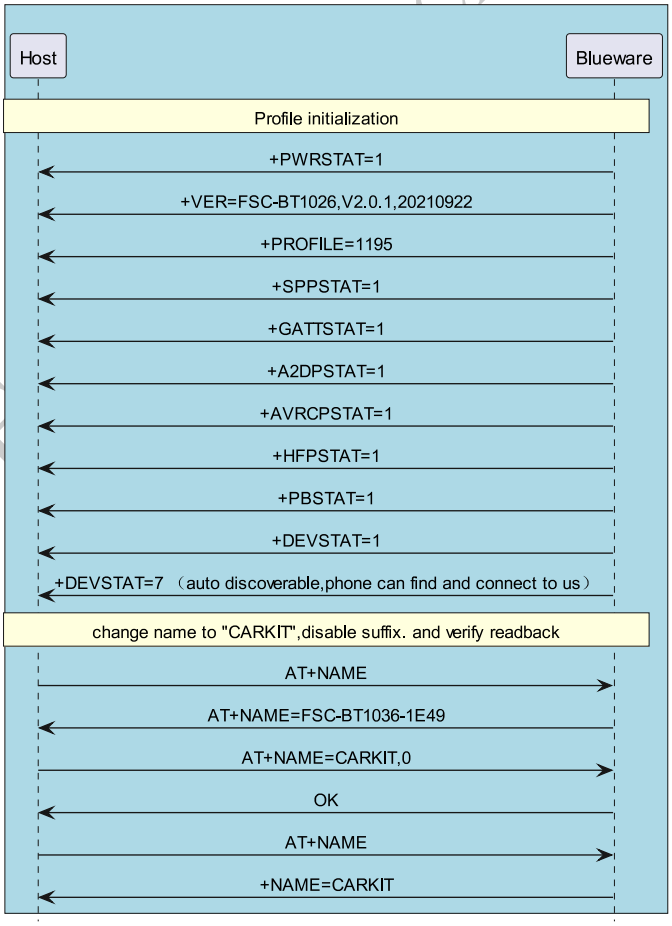
Format	+HIDSTAT=Param
Param	(0) Unsupported (1) Standby (2) Connecting (3) Connected

Chapter 7

Application scenarios

7.1 profiles initializing and change parameter

The following figure shows Profile initialization and name modification

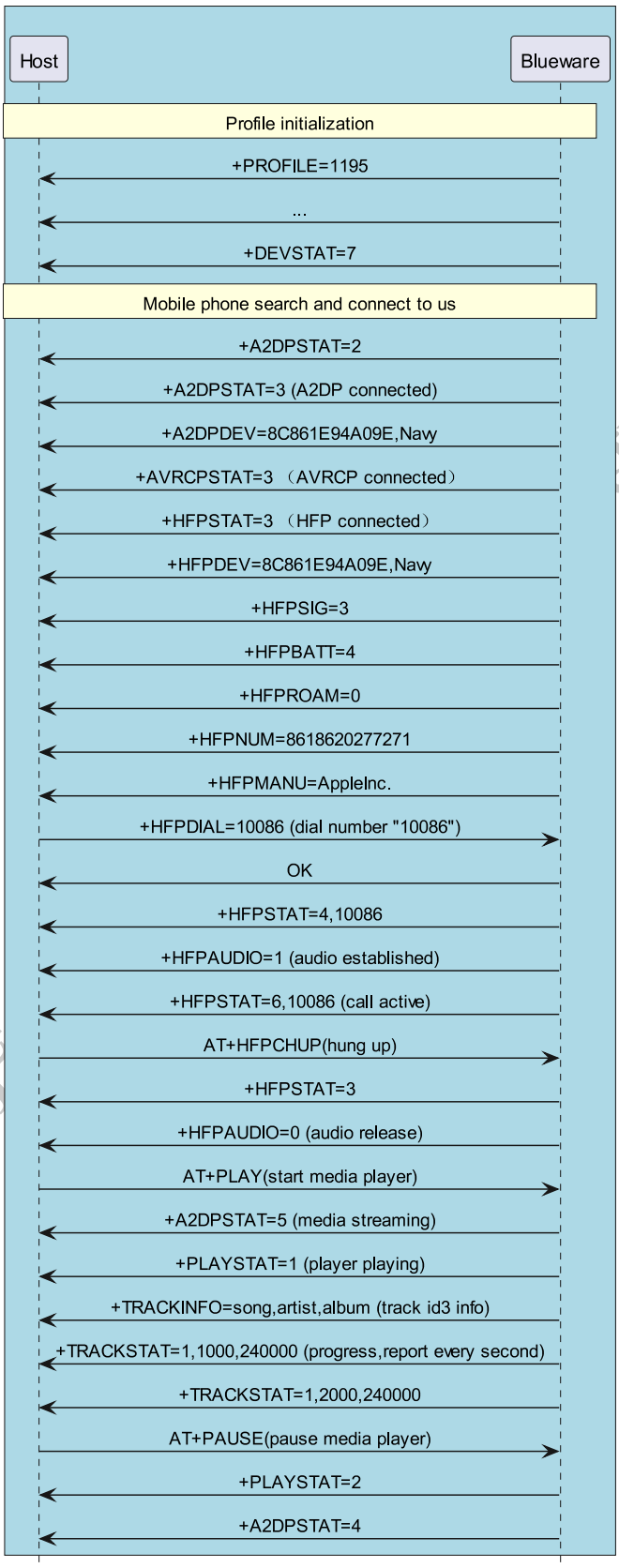


MCU change device name CARKIT reference code:

```
void change_name(void)
{
    uart_send("AT+NAME\r\n");
    if(uart_read("+NAME",name_buf))
    {
        if(memcmp(name_buf,"CARKIT",6))
        {
            uart_send("AT+NAME=CARKIT,0\r\n");      //defalut disable MAC address suffix
            uart_send("AT+NAME\r\n"); // read bt name
            if(uart_read("+NAME",name_buf))
            {
                if(memcmp(name_buf,"CARKIT",6))
                {
                    //change name fail
                }
                else
                {
                    //change name success
                }
            }
        }
    }
}
```

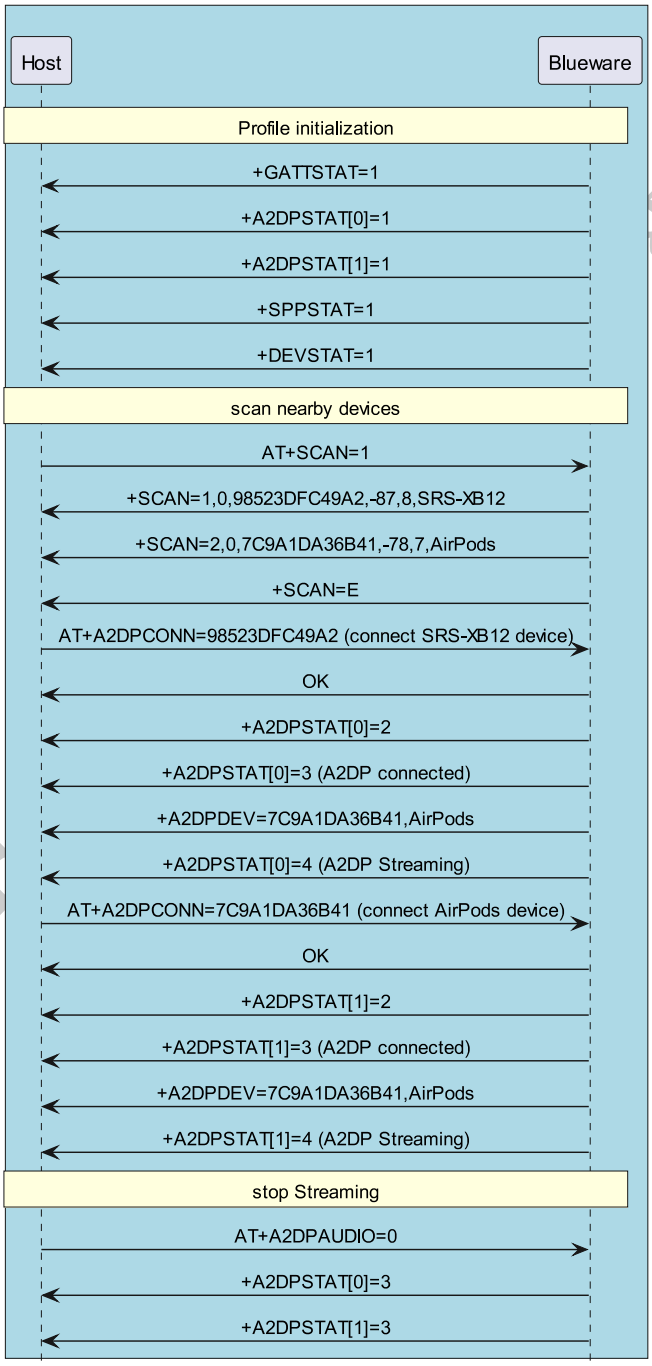
Note: modify any parameters, it is recommended to query first and then modify the final verification

7.2 Sink mode connection

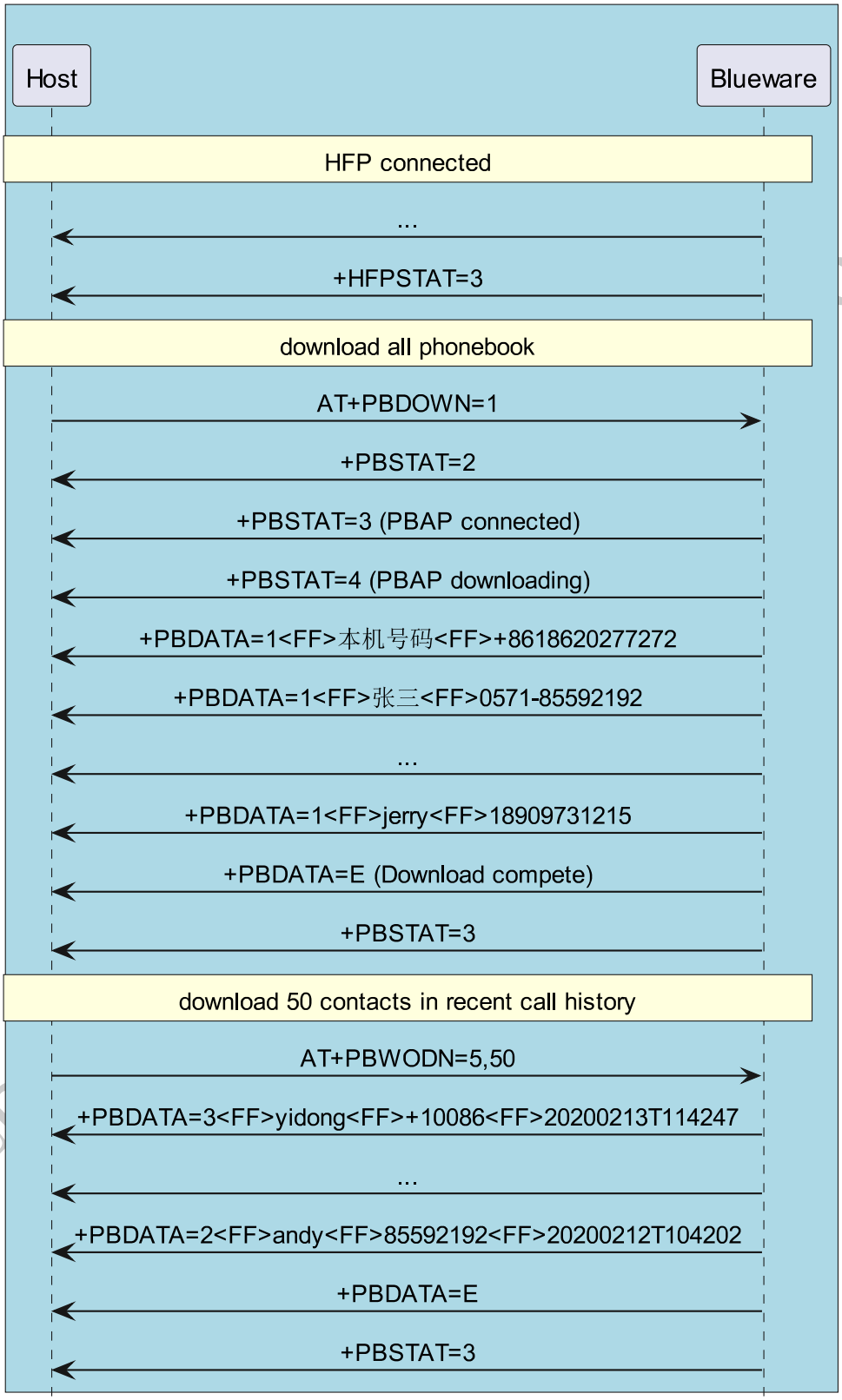


7.3 Source mode connection

Note: The transmission(source) mode connection needs to configure the module to A2DP Source, HFP Source, Only BT80X series and BT1035 support, 80X need to send AT+PROFILE configuration to source mode



7.4 Phonebook downloading



Chapter 8

Appendix

8.1 Download PDF Document

Download PDF Document