FSC-BT80X

Sink Programming User Guide

Version 3.4

Revision History

Version	Data	Notes	Author
1.0	2015/10/10	First Release	Tony
2.0	2016/05/20	Add commands	Navy
3.0	2017/11/13	Add A2DP Source profiles	Navy
3.1	2018/01/25	Add commands	Navy
3.2	2018/08/31	Update Commands	Navy
3.3	2018/12/21	Add commands	Navy
3.4	2019/04/26	Update A2DP Source profiles Commands	Carfei

Contact Us:

Shenzhen Feasycom Technology Co., Ltd
Web: www.feasycom.com
Email:support@feasycom.com
Tel: +86-755-27924639,+86-755-23062695
Address: Room 2004-2005,20th Floor, Huichao Technology Building, Jinhai Road, Xixiang, Baoan District, Shenzhen,518100, China.

Contents

1. Introduction

This specification presents design guidelines for software engineers that use FSC-BT80X series modules for Bluetooth requirements.

1.1 Terms

Throughout this specification:

- { }　：Content between {...} is optional
- <<　：Content behind << represents a *COMMAND* sent from Host to Module
- >>　：Content behind >> represents a *RESPONSE* sent from Module to Host

1.2 Hardware Interface

- GPIO
- PWM
- UART
- I2C Master/Slave
- I2S Master/Slave
- Analog Input/Output

1.3 Supported Bluetooth Profile

- SPP (Serial Port Profile)
- GATT Server (Generic Attribute Profile)
- GATT Client (Generic Attribute Profile)
- HFP Sink (Hands-Free 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 Keyboard (Human Interface Profile)
- PBAP Server (Phonebook Access Profile)

1.4 Command Format

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

- All commands start with “AT”, end with <CR><LF>

- <CR> stands for "carriage return", corresponding hex is 0x0D
- <LF> stands for "line feed", corresponding hex is 0x0A
- If command has parameter, parameter keep behind "="
- If command has multiple parameters, parameter must be separated by ","
- If command has response, response start with <CR><LF>, end with <CR><LF>
- Module will always report command's execution result using "OK" for success or "ERROR" for failure

e.g.

1. Read module's BR/EDR local name
   << *AT+NAME*
   >> *+NAME=Feasycom*
   >> *OK*

2. Pick up an incoming call when no call incoming actually
   << *AT+HFPANSW*
   >> *ERROR*

1.5 Indication Format

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

- All indications start with <CR><LF>, end with <CR><LF>
- If indication has parameter, parameter keep behind "="
- If indication has multiple parameters, parameter must be separated by ","
- Hex value <FF> will be used instead of "," in some special indications

e.g.

1. Received "1234567890" from mobile phone via SPP profile
   >> *+SPPDATA=10,1234567890*
2. Call number "10086" use a mobile phone when HFP connected
   >> *+HFPSTAT=4*
   +HFPCID=10086
   +HFPCIE=China Mobile
   +HFPAUDIO=1
   +HFPSTAT=6

1.6 Module Default Settings

Local Name (BR/EDR)	FSC-BT80X
Local Name (LE)	FSC-BT80X-LE
Pin Code	0000
Secure Simple Pairing Mode	On
Physical UART Baudrate	115200bps/8/N/1

2.Command Table

2.1 General Commands

2.1.1 AT Command Test

Format: AT
Response: OK
Description: Test the communication between HOST and Module after power on, UART baudrate changed and etc.
Example: AT command test << *AT* >> *OK*

2.1.2 Read Firmware Version

Format: AT+VER
Response: +VER=Param Param: Firmware version (24 Bytes ASCII)
Example: Read module's firmware version << *AT+VER* >> *+VER=FSC-BT80X,V1.0.0,20160120* >> *OK*

2.1.3 Read BR/EDR MAC Address

Format: AT+ADDR
Response: +ADDR=Param Param: Module's BR/EDR MAC address (12 Bytes ASCII)

Example: Read Module's BR/EDR MAC address << *AT+ADDR* >> *+ADDR=DC0D30123456* >> *OK*

2.1.4 Read BLE MAC Address

Format: AT+LEADDR
Response: +LEADDR=Param Param: Module's LE MAC address (12 Bytes ASCII)

2.1.5 Read/Write BR/EDR Local Name

Format: AT+NAME {=Param1{, Param2}} Param1: BR/EDR local name (1~31 Bytes ASCII, default: FSC-BT80X) Param2: MAC address suffix (0/1, default:0) (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 existence, otherwise read current local name
Example: Read current BR/EDR local name << *AT+NAME* >> *+NAME=Feasycom* >> *OK* **Example**: Change module's BR/EDR local name to "ABC" << *AT+NAME=ABC* >> *OK* **Example**: Change module's BR/EDR local name to "ABC" and enable suffix << *AT+NAME=ABC,1* >> *OK*

2.1.6 Read/Write BLE Local Name

Format: AT+LENAME {=Param1{, Param2}} Param1: BLE local name (1~25 Bytes ASCII, default: FSC-BT80X-LE) Param2: MAC address suffix (0/1, default:0) (0) Disable suffix (1) Enable suffix "-XXXX" (lower 4 bytes of MAC address) after local name
Response: +LENAME=Param

2.1.7 Read/Write BLE Random Addr Configuation

Format: AT+LECFG {=Param} Param1: BLE Random addr enable(0/1, default:1) (0) Disable (1) Enable
Response: +LECFG=Param

2.1.8 Read/Write UART Baudrate

Format: AT+BAUD{=Param} Param: Baudrate (9600/19200/38400/57600/115200/230400/460800/921600, default:115200)
Response: +BAUD=Param
Description: Module's baudrate will be changed immediately after received this command

2.1.9 Read/Write UART Configuration*<need reboot>*

Format: AT+UARTCFG{=Param} Param: Uart Configuration(0/1, default:0) (0) Disable CTS/RTS (1) Enable CTS/RTS

Response: +UARTCFG= Param

2.1.10 Read/Write Pin Code

Format: AT+PIN{=Param} Param: Pin code (4~15 Bytes ASCII, default:0000)
Response: +PIN=Param
Example: Read module's pin code **<<** *AT+PIN* **>>** *+PIN=0000* **>>** *OK* **Example**: Change module's pin code to "12345678" **<<** *AT+PIN=12345678* **>>** *OK*

2.1.11 Turn On/Off Secure Simple Pairing *<need reboot>*

Format: AT+SSP{=Param} Param: Simple pairing (0/1, default:1) (0) Turn off (1) Turn on
Response: +SSP=Param
Description: Pin code input will be bypassed if simple pairing is on in pairing procedure

2.1.12 Read/Write Class Of Device *<need reboot>*

Format: AT+COD{=Param} Param: Class of device (6 bytes ASCII, default:240404 Handsfree device)
Response: +COD=Param

Description: Filter nearby BR/EDR devices using COD when in scan state in A2DP master mode. Configure type of device in A2DP slave mode.

2.1.13 Read/Clear Paired Record

Format: 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} 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, iPhone* *+PLIST=2, A0BC30075421, Samsung S8* *+PLIST=E* >> *OK* **Example**: Clear module's paired record << *AT+PLIST=0* >> *OK*

2.1.14 Turn On/Off Throughput Mode *<need reboot>*

Format: 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
Example: Read current throughput mode << *AT+TPMODE* >> *+TPMODE=1* >> *OK* **Example:** Turn off throughput mode << *AT+TPMODE=0* >> *OK*

2.1.15 Read Module States

Format: AT+STAT
Response: +STAT=Param1, Param2, Param3, Param4, Param5, Param6, Param7, Param8 Param1: DEVSTAT Param2: SPPSTAT Param3: GATTSTAT Param4: HFPSTAT Param5: A2DPSTAT Param6: AVRCPSTAT Param7: HIDSTAT Param8: PBSTAT
Description: Refer to chapter 3 for state description, state may have different meanings according to profile selection

2.1.16 Turn On/Off Power On Auto Reconnect *<need reboot>*

Format: AT+AUTOCONN{=Param} Param: Option (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

2.1.17 Scan Nearby Devices

Format: AT+SCAN {=Param} Param1:(0~1) (0) Stop scan (1) Scan nearby BR/EDR devices
Description: Refer to Chapter 3 for format description of scan result when in A2DP Slave mode. Scan and connect nearby BR/EDR devices when in A2DP master mode.

2.1.18 Auto Scan Enable

Format: AT+INQCFG {=Param} Param: (0/1, default:1) (0) Disable (1) Enable
Response: +INQCFG=Param
Description: Auto Scan and connect nearby BR/EDR devices when in A2DP master mode

2.1.19 Speaker Volume Setting

Format: AT+SPKVOL{=Param} Param: ('+'/'-')
Response: +SPKVOL =Param
Example: Read current speaker volume << *AT+SPKVOL* >> *+SPKVOL=14* **Example**: Increase audio speaker volume << *AT+SPKVOL=+* >> *OK*

2.1.20 I2S/PCM Format Configuration *<need reboot>*

Format: +I2SCFG{=Param}
Param: A base-10 representation of a bit field, default:0, for each bit:

BIT[0]	0: Disable I2S/PCM for audio input/output
	1: Enable I2S/PCM for audio input/output
BIT[1]	0: I2S/PCM master role
	1: I2S/PCM slave role
BIT[2]	0: 48000Hz sample rate
	1: 44100Hz sample rate
BIT[3-4]	00: I2S Philips standard format
BIT[5-6]	00: 16-bit resolution
	01: 24-bit resolution
	10: 32-bit resolution

Example: Read current I2S/PCM configuration
<< AT+I2SCFG
>> +I2SCFG=0
Example: Set I2S/PCM configuration to: I2S master, 32-bit resolution,48kHz.
I2S LRCLK: 48000Hz
I2S BCLK: 3.072MHz (48000Hz * 32bit * 2Stereo)
<< AT+I2SCFG=65
>> OK

2.1.21 SPDIF Format Configuration *<need reboot>*

Format: +SPDIFCFG{=Param}
Param: A base-10 representation of a bit field, default:0, for each bit:

BIT[0]	0: Disable SPDIF for audio output
	1: Enable SPDIF for audio output

Example: Read current SPDIF configuration
<< AT+SPDIFCFG
>> +SPDIFCFG=0

2.1.22 Release All Connections

Format: AT+DSCA
Description: Module release all Bluetooth connections with remote device

2.1.23 Soft Reboot

Format: AT+REBOOT
Description: Module release all Bluetooth connections with remote device then reboot

2.1.24 Restore Factory Settings

Format: AT+RESTORE
Description: Module restore all factory settings then reboot

2.1.25 Close AT Air Command Mode

Format: AT+CLOSEAT
Description: Close AT Air Command Mode

2.1.26 Enable Module

Format: AT+BTEN {=Param} Param: (0~1) (0) Enter Pairing Mode (1) Quit Pairing Mode
Description: Disconnect all device connected and put module enter unconnectable and undiscoverable state .Always effective even reboot.

2.1.27 Enter Pairing Mode

Format: AT+PAIR{=Param} Param: (0~1) (0) Enter Pairing Mode (1) Quit Pairing Mode
Description: put module enter or quit connectable and discoverable state.

2.2 HFP Commands

2.2.1 Read HFP State

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

2.2.2 Establish HFP Connection

Format: AT+HFPCONN{=Param} Param: MAC address of target device (12 Bytes ASCII)
Description: Module will reconnect to last HFP device if parameter not exist

Example: Connect to last HFP device << *AT+HFPCONN* >> *OK* **Example2**: Connect to specific HFP device with MAC address << *AT+HFPCONN=1C5CF226D773* >> *OK*

2.2.3 Release HFP Connection

Format: AT+HFPDISC
Description: Release current HFP connection with remote device

2.2.4 Dial/Redial Phone Number

Format: AT+HFPDIAL{=Param} Param: Phone number (1~25 Bytes ASCII)
Description: Dial specific number if parameter existence, otherwise redial
Example: Redial << *AT+HFPDIAL* >> *OK* **Example:** Dial number “075527924639” << *AT+HFPDIAL=075527924639* >> *OK*

2.2.5 Send DTMF code

Format: AT+HFPDTMF=Param Param: DTMF code (0~9/#/*)
Example: Send DTMF code “#” while talking << *AT+HFPDTMF=#* >> *OK*

2.2.6 Pick Up Incoming Call

Format: AT+HFPANSW
Description: Pick up an incoming call

2.2.7 Reject/Hung up Call

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

2.2.8 Transfer Voice Audio

Format: AT+HFPADTS{=Param} Param: Transfer direction (0/1) (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 existence

2.2.9 Mute Mic

Format: AT+MUTEMIC{=Param} Param: mute mic(0/1) (0) unmute (1) mute
Description: mute mic when call active

2.3 A2DP/AVRCP Commands

2.3.1 Read A2DP State

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

2.3.2 Read/Write A2DP Role

Format: AT+A2DPROLE{=Param} Param: (0/1 default:0) (0) Slave (1) Master
Description: Receive audio sink from remote device in slave role. Lanuch audio source to remote device in master role. Master mode default settings: 1. Enable auto scan(+INQCFG=1) Auto clear paired device list and reboot after role changed.

2.3.3 Establish A2DP Connection

Format: AT+A2DPCONN{=Param} Param: MAC address of target device (12 Bytes ASCII)
Description: Module will reconnect to last A2DP device if no parameter exist

Example: Connect to last A2DP device

<< *AT+A2DPCONN*

>> *OK*

Example2: Connect to specific A2DP device with MAC address

<< *AT+A2DPCONN=1C5CF226D773*

>> *OK*

2.3.4 Release A2DP Connection

Format: AT+A2DPDISC

Description: Release current A2DP connection with remote device

2.3.5 Read A2DP Decoder

Format: AT+A2DPDEC

Response: +A2DPDEC=Param

Param:(0~7)

(0) INVALID
(1) SBC(decoder/encoder configurable)
(2) MP3(decoder configurable)
(3) AAC(decoder configurable)
(4) FASTSTREAM(decoder/encoder configurable)
(5) APTX(decoder/encoder configurable)
(6) APTX-LL(decoder/encoder configurable)
(7) APTX-HD(decoder/encoder configurable)

Description: Default support (1)SBC (5)APTX (7)APTX-HD

2.3.6 Read/Write AVRCP Configuration

Format: AT+AVRCPCFG{=Param}

Param: A base-10 representation of a bit field, default:9, for each bit:

BIT[0] Auto get track ID3 information (title, artist, album) on track changed.default:1

BIT[1-3] Auto get track state (play progress) if value > 0. default:5(second)

Example: Read AVRCP configuration << *AT+ AVRCPCFG* >> *+ AVRCPCFG =9* *OK* **Example**: Get track play progress every 1 second << *AT+ AVRCPCFG =3* >> *OK*
Description: Refer to Chapter 3 for indication format of track information and track state

2.3.7 Track Play/Pause

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

2.3.8 Track Play

Format: AT+PLAY
Description: Send play command to remote media player

2.3.9 Track Pause

Format: AT+PAUSE
Description: Send pause command to remote media player

2.3.10 Track Stop

Format: AT+STOP
Description: Send stop command to remote media player

2.3.11 Track Forward

Format: AT+FORWARD
Description: Send forward command to remote media player

2.3.12 Track Backward

Format: AT+BACKWARD
Description: Send backward command to remote media player

2.4 Phonebook Access Commands

2.4.1 Download Phonebook

Format: AT+PBDOWN=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)
Response: +PBDATA=Param1<FF>Param2<FF>Param3 {<FF>Param4} Param: Refer to Chapter 3 for format description of received phonebook data

2.5 Bluetooth Serial Commands (BR/EDR SPP)

2.5.1 Read SPP State

Format: AT+SPPSTAT

Response: +SPPSTAT=Param Param: Refer to Chapter 3 for state description

2.5.2 Establish SPP Connection

Format: AT+SPPCONN=Param Param: MAC address of target device (12 Bytes ASCII)
Description: If target device is mobile phone, mobile phone must have initialized a RFCOMM service before this

2.5.3 Release SPP Connection

Format: AT+SPPDISC
Description: Release current SPP connection with remote device

2.5.4 Send Data Via SPP

Format: AT+SPPSEND=Param1, Param2 Param1: Payload length (1~236) Param2: Payload (1~236Bytes 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*

2.6 Bluetooth Serial Commands (LE GATT)

2.6.1 Read GATT State

Format: AT+GATTSTAT
Response: +GATTSTAT=Param Param: Refer to Chapter 3 for state description

2.6.2 Release GATT Connection

Format: AT+GATTDISC
Description: Release current GATT connection with remote device

2.6.3 Send Data Via GATT

Format: AT+GATTSEND=Param1, Param2 Param1: Payload length (1~100) Param2: Payload (1~100 Bytes UTF8)
Description: If throughput mode is on, this command is de-active
Example: Send data "1234567890" to remote device via GATT << *AT+GATTSEND=10,1234567890* >> *OK*

3. Indication Table

3.1 General Indications

3.1.1 Device State

Format: +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 Non Discoverable; 1: BR/EDR Discoverable BIT[2] 0: BLE Non Advertising; 1: BLE Advertising BIT[3] 0: BR/EDR Non Scanning; 1: BR/EDR Scanning BIT[4] 0: BLE Non Scanning; 1: BLE Scanning
Example: Module is power on, discoverable and advertising **>>** *+DEVSTAT=7*

3.1.2 Scan Result

Format: +SCAN =Param1, Param2, Param3, Param4, Param5, Param6 Param1: Index (1~8) 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 (-255 ~ 0) Param5: Size of Param6 if exist Param6: Device Name for BR/EDR devices or advertising data for LE devices
Description: Param5/Param6 may not exist if remote device out of distance
Example: Scan BR/EDR nearby devices in 6.4s << *AT+SCAN=1,5* >> *OK* *+SCAN=1,2, DC0D30000003, -32,8, Feasycom* *+SCAN=2,2, DC0D30000044, -64,8, Feasycom_1234*

+SCAN=3,2, DC0D30000097, -47,8, TESTHID

3.1.3 Paired Success

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

3.2 HFP Indications

3.2.1 HFP State

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

3.2.2 HFP Device Information

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

Example: HFP connect success with device
>> *+HFPDEV=1C5CF226D774,iPhone*

3.2.3 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

3.2.4 Incoming/Outgoing Call Name

Format: +HFPCIE=Param

Param:(UTF8), Call name

Description: Not every mobile phone support this indication

3.2.5 HFP Voice Audio State

Format: +HFPAUDIO=Param

Param:(0/1)

(0) HFP voice audio disconnected, audio input/output routed to remote device

(1) HFP voice audio connected, audio input/output routed to module

3.2.6 HFP Device Network Signal Strength

Format: +HFPSIG=Param

Param:(0~5) Network signal strength of remote device

3.2.7 HFP Device Network Operator Selection

Format: +HFPNET=Param

Param:(UTF8) Network operator selection of remote device

3.2.8 HFP Device Roaming State

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

3.2.9 HFP Device Battery Level

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

3.3 A2DP/AVRCP Indications

3.3.1 A2DP State

Format: +A2DPSTAT=Param
Param:(0~4)

(0) Unsupported
(1) Standby
(2) Connecting
(3) Connected
(4) Streaming

3.3.2 A2DP Device Information

Format: +A2DPDEV=Param
Param: (12 Bytes ASCII), Remote device's MAC address of current A2DP connection

3.3.3 AVRCP State

Format: +AVRCPSTAT=Param
Param:(0~3)

(0) Unsupported
(1) Standby
(2) Connecting

(3) Connected

3.3.4 Media Player State

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

3.3.5 Media Player Play Progress

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

Example: Read media player play progress every 3s
<< *AT+AVRCPCFG=7*
>> *+TRACKSTAT=1,142000,248000*
+TRACKSTAT=1,145000,248000
+TRACKSTAT=1,148000,248000

3.3.6 Media Track Information

Format: +TRACKINFO=Param1 <FF> Param2 <FF> Param3
Param1: title
Param2: artist
Param3: ablum

Example: Phone playing song “Creep-Radio Head”
>> *+TRACKINFO=Creep <FF> Radiohead <FF> Pablo Honey*

3.4 Phonebook Access Indications

3.4.1 PB Entries Of Remote Devices

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

3.4.2 Received Phonebook Data

Format1: +PBDATA=Param1 <FF> Param2 xFF \xParam3 {xFF Param4}
Param1: Type
(0) Phonebook (SIM Storage)
(1) Phonebook (Phone Storage)
(2) Received call log
(3) Dialed call log
(4) Missed call log
Param2: (UTF8), Name
Param3: (ASCII), Number
Param4: (15 Bytes ASCII), Call time
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

3.5 Bluetooth Serial Indications

3.5.1 SPP State

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

3.5.2 GATT State

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

3.5.3 SPP Device Information

Format: +SPPDEV=Param
Param: (12 Bytes ASCII), Remote device's MAC address of current SPP connection

3.5.4 GATT Device Information

Format: +GATTDEV=Param
Param: (12 Bytes ASCII), Remote device's MAC address of current GATT connection

3.5.5 SPP Received Data

Format: +SPPDATA=Param1, Param2 Param1: Payload length Param2: Payload
Description: If throughput mode is on, only Param2 will be present
Example: Received data "1234567890" from remote device via SPP << *+SPPDATA=10,1234567890*

3.5.6 GATT Received Data

Format: +GATTDATA=Param1, Param2 Param1: Payload length Param2: Payload
Description: If throughput mode is on, only Param2 will be present
Example: Received data "1234567890" from remote device via GATT << *+GATTDATA=10,1234567890*

3.6 GPIO Indications

3.6.1 LED Pin

LED0(Output)	
Low Level	Initializing
Blink in 1Hz	Ready to connecting
High Level	Connected

3.6.2 State Pin

LED1(Output)	
Low Level	SPP/GATT Disconnected

High Level	SPP/GATT Connected