



FEASYCOM®

FSC-BT80X

Source Programming User Guide

Version 2.0

Copyright © 2013-2019 Feasycom Technology Co., Ltd. All Rights Reserved.

Revision History

Version	Date	Notes	Author
1.0	2015/10/10	First Release	Tony
1.1	2017/05/08	Add commands	Navy
2.0	2019/11/20	Add commands	Navy

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.....	5
1.1 Terms.....	5
1.2 Hardware Interface.....	5
1.3 Supported Bluetooth Profile	5
1.4 Command Format.....	5
1.5 Indication Format.....	6
1.6 Module Default Settings	6
2.Command Table	7
2.1 General Commands	7
2.1.1 AT Command Test.....	7
2.1.2 Bluetooth Profile Selection <i><need reboot></i>	7
2.1.3 Read Firmware Version	8
2.1.4 Read BR/EDR MAC Address.....	8
2.1.5 Read BLE MAC Address	9
2.1.6 Read/Write BR/EDR Local Name	9
2.1.7 Read/Write BLE Local Name.....	10
2.1.8 Read/Write UART Baudrate.....	10
2.1.9 Read/Write Pin Code.....	10
2.1.10 Read/Write Class Of Device <i><reboot required></i>	11
2.1.11 Read/Clear Paired Record	11
2.1.12 Turn On/Off Throughput Mode	11
2.1.13 Read Module States.....	12
2.1.14 Turn On/Off Power On Auto Reconnect <i><need reboot></i>	12
2.1.15 Turn On/Off Pairing/Advertising Mode.....	13
2.1.16 Turn On/Off BT Radio.....	13
2.1.17 Scan Nearby Devices	13
2.1.18 Turn On/Off Auto Link	14
2.1.19 Speaker Volume Setting.....	14
2.1.20 Microphone Gain Setting.....	14
2.1.21 I2S/PCM Format Configuration <i><need reboot></i>	15
2.1.22 Release All Connections.....	15
2.1.23 Enable Print Log.....	15
2.1.24 Soft Reboot.....	16
2.1.25 Restore Factory Settings.....	16
2.2 HFP Commands	16
2.2.1 Read HFP State.....	16
2.2.2 Establish HFP Connection	16
2.2.3 Release HFP Connection.....	17
2.2.4 Establish/Release Voice Audio.....	17
2.3 A2DP Commands.....	17
2.3.1 Read A2DP State.....	17
2.3.2 Establish A2DP Connection	17

2.3.3 Release A2DP Connection	18
2.3.4 Read A2DP Encoder	18
2.3.5 Establish/Release A2DP Audio Connection.....	18
2.4 Bluetooth Serial Commands (BR/EDR SPP).....	19
2.4.1 Read SPP State	19
2.4.2 Establish SPP Connection	19
2.4.3 Release SPP Connection	19
2.4.4 Send Data Via SPP.....	19
2.5 Bluetooth Serial Commands (LE GATT)	20
2.5.1 Read GATT State.....	20
2.5.2 Release GATT Connection.....	20
2.5.3 Establish GATT Client Connection.....	20
2.5.4 Send Data Via GATT.....	20
3. Indication Table	21
3.1 General Indications	21
3.1.1 Device State	21
3.1.2 Scan Result.....	21
3.1.3 Paired Success.....	22
3.2 HFP Indications	22
3.2.1 HFP State	22
3.2.2 HFP Device Information	22
3.2.3 HFP Voice Audio State	23
3.3 A2DP Indications.....	23
3.3.1 A2DP State	23
3.3.2 A2DP Device Information.....	23
3.4 Bluetooth Serial Indications.....	23
3.4.1 SPP State	23
3.4.2 GATT State.....	24
3.4.3 SPP Device Information.....	24
3.4.4 SPP Received Data	24
3.4.5 GATT Received Data	24
3.5 GPIO Indications	25
3.5.1 LED Pin	25
3.5.2 State Pin	25

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)
- HFP Source (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
```

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
```

## 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

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

#### 2.1.2 Bluetooth Profile Selection <need reboot>

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+PROFILE{=Param}<br>Param: A base-10 representation of a bit field, default:85, for each bit:<br>BIT[0]   SPP (Serial Port Profile)<br>BIT[1]   GATT Server (Generic Attribute Profile)<br>BIT[2]   GATT Client (Generic Attribute Profile)<br>BIT[3]   HFP Sink (Hands-Free Profile)<br>BIT[4]   HFP Source (Hands-Free Profile)<br>BIT[5]   A2DP Sink (Advanced Audio Distribution Profile)<br>BIT[6]   A2DP Source (Advanced Audio Distribution Profile)<br>BIT[7]   AVRCP Controller (Audio/Video remote controller Profile)<br>BIT[8]   AVRCP Target (Audio/Video remote controller Profile)<br>BIT[9]   HID Keyboard (Human Interface Profile)<br>BIT[10]  PBAP Server (Phonebook Access Profile) |
| <b>Response:</b> +PROFILE=Param                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Description:</b> GATT Server and Client, HFP Sink and Source, A2DP Sink and Source, AVRCP Controller and Target cannot be enabled both because of mutual exclusion.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

BT80X only supports HFP Source,A2DP Source, SPP, GATT Server and GATT Client  
Module will soft reboot if profile selection changed

**Example:** Read current profile selection

<< AT+PROFILE

>> +PROFILE=85

**Example:** Enable GATT Server ,SPP ,A2DP Source

<< AT+PROFILE=67

>> OK

### 2.1.3 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

### 2.1.4 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



## 2.1.5 Read BLE MAC Address

|                                                                                   |
|-----------------------------------------------------------------------------------|
| <b>Format:</b> AT+LEADDR                                                          |
| <b>Response:</b> +LEADDR=Param<br>Param: Module's LE MAC address (12 Bytes ASCII) |

## 2.1.6 Read/Write BR/EDR Local Name

|                                                                                                                                                                                                                                                                                                  |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+NAME {=Param1{, Param2}}<br>Param1: BR/EDR local name (1~31 Bytes ASCII, default: FSC-BT80X)<br>Param2: MAC address suffix (0/1, default:0)<br>(0) Disable suffix<br>(1) Enable suffix "-XXXX" (lower 4 bytes of MAC address) after local name                                 |
| <b>Response:</b> +NAME=Param                                                                                                                                                                                                                                                                     |
| <b>Description:</b> Write local name if parameter existence, otherwise read current local name                                                                                                                                                                                                   |
| <b>Example:</b> Read current BR/EDR local name<br><< AT+NAME<br>>> +NAME=Feasycom<br>>> OK<br><b>Example:</b> Change module's BR/EDR local name to "ABC"<br><< AT+NAME=ABC<br>>> OK<br><b>Example:</b> Change module's BR/EDR local name to "ABC" and enable suffix<br><< AT+NAME=ABC,1<br>>> OK |

## 2.1.7 Read/Write BLE Local Name

**Format:** AT+LENAM {=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:** +LENAM=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 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

### 2.1.10 Read/Write Class Of Device <need reboot>

**Format:** AT+COD{=Param}

Param: Class of device (6 bytes ASCII, default: 5A020C)

**Response:** +COD=Param

### 2.1.11 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
```

### 2.1.12 Turn On/Off Throughput Mode

**Format:** AT+TPMODE{=Param}

Param: Throughput mode (0/1, default:0)

(0) Turn Off

(1) Turn On

|                                                                                                                                                                                       |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Response:</b> +TPMODE=Param                                                                                                                                                        |
| <b>Description:</b> 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 |
| <b>Example:</b> Read current throughput mode<br><< AT+TPMODE<br>>> +TPMODE=1<br>>> OK<br><b>Example:</b> Turn off throughput mode<br><< AT+TPMODE=0<br>>> OK                          |

### 2.1.13 Read Module States

|                                                                                                                                                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+STAT                                                                                                                                                                      |
| <b>Response:</b> +STAT=Param1, Param2, Param3, Param4, Param5, Param6<br>Param1: DEVSTAT<br>Param2: SPPSTAT<br>Param3: GATTSTAT<br>Param4: HFPSTAT<br>Param5: A2DPSTAT<br>Param6: AVRCPSTAT |
| <b>Description:</b> Refer to chapter 3 for state description, state may have different meanings according to profile selection                                                              |

### 2.1.14 Turn On/Off Power On Auto Reconnect <need reboot>

|                                                                                                                              |
|------------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+AUTOCONN{=Param}<br>Param: Option (0~127, default:2)<br>(0) Turn Off<br>(1-15) Turn on and reconnect times |
| <b>Response:</b> +AUTOCONN=Param                                                                                             |

**Description:** Module will attempt to connect last device after power on if set

### 2.1.15 Turn On/Off Pairing/Advertising Mode

**Format:** AT+PAIR=Param

Param: Pair mode (0/1)

(0) Turn Off

(1) Turn On

**Response:** OK

**Description:** Module will enter pair mode itself if no connection established, and leave pair mode otherwise

### 2.1.16 Turn On/Off BT Radio

**Format:** AT+BTEN{=Param}

Param: BT Radio (0/1, default:1)

(0) Turn Off

(1) Turn On

**Response:** +BTEN=Param

**Description:** Module will disable all Bluetooth function if BT Radio off

### 2.1.17 Scan Nearby Devices

**Format:** AT+SCAN =Param

Param:(0~2)

(0) Stop scan

(1) Scan nearby BR/EDR devices

(2) Scan nearby BLE devices

**Description:** Refer to Chapter 3 for format description of scan result

### 2.1.18 Turn On/Off Auto Link

|                                                                                                                                                                                                                             |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+AUTOLINK{=Param}<br>Param: Option (0/1, default:0)<br>(0) Turn Off<br>(1) Turn on                                                                                                                         |
| <b>Response:</b> + AUTOLINK =Param                                                                                                                                                                                          |
| <b>Description:</b> if set, Module will scan nearby devices for 5.12 seconds, find the one which has best signal strength, then connect to it automatically.<br>Module is not connected, it will always scan nearby devices |

### 2.1.19 Speaker Volume Setting

|                                                                                                                                                          |
|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+SPKVOL{=Param}<br>Param: audio speaker volume (0~15, default:12)                                                                       |
| <b>Response:</b> +SPKVOL =Param                                                                                                                          |
| <b>Example:</b> Read current speaker volume<br><< AT+SPKVOL<br>>> +SPKVOL=14<br><b>Example:</b> set audio speaker volume to 9<br><< AT+SPKVOL=9<br>>> OK |

### 2.1.20 Microphone Gain Setting

|                                                                                                                         |
|-------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+MICGAIN{=Param}<br>Param: Microphone input gain (0~15, default:12)                                    |
| <b>Response:</b> + MICGAIN=Param                                                                                        |
| <b>Description:</b> Command only effective for module which has internal codec, volume 0 will mute the microphone input |

### 2.1.21 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

**Note:** HFP only supports I2S master/slave, 16-bit resolution, 8kHz  
A2DP does not support 48kHz

**Example:** Read current I2S/PCM configuration

```
<< AT+I2SCFG
```

```
>> +I2SCFG=0
```

**Example:** Set I2S/PCM configuration to: I2S master, 16-bit resolution, 44.1kHz.

I2S LRCLK: 44100Hz

I2S BCLK: 1.411MHz (44100Hz \* 16bit \* 2Stereo)

```
<< AT+I2SCFG=5
```

```
>> OK
```

### 2.1.22 Release All Connections

**Format:** AT+DSCA

**Description:** Module release all Bluetooth connections with remote device

### 2.1.23 Enable Print Log

**Format:** AT+PRINT{=Param}

Param: (0~1)  
(0) Disable  
(1) Enable

**Description:** Enable module print the log(including states of profiles) to uart.

## 2.1.24 Soft Reboot

**Format:** AT+REBOOT

**Description:** Module release all Bluetooth connections with remote device then reboot

## 2.1.25 Restore Factory Settings

**Format:** AT+RESTORE

**Description:** Module restore all factory settings then reboot

# 2.2 HFP Commands

## 2.2.1 Read HFP State

**Format:** AT+HFPSTAT

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

## 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 Establish/Release Voice Audio

**Format:** AT+HFPAUDIO{=Param}

Param: Operation (0/1)

(0) Release voice audio connection with remote hfp sink device

(1) Establish voice audio connection with remote hfp sink device

## 2.3 A2DP Commands

### 2.3.1 Read A2DP State

**Format:** AT+A2DPSTAT

**Response:** +A2DPSTAT=Param

Param: Refer to Chapter 3 for state description

### 2.3.2 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

### 2.3.3 Release A2DP Connection

**Format:** AT+A2DPDISC

**Description:** Release current A2DP connection with remote device

### 2.3.4 Read A2DP Encoder

**Format:** AT+A2DPENC

**Response:** +A2DPENC=Param

Param:(0~4)

(0) INVALID

(1) SBC

(2) FASTSTREAM

(3) APTX

(4) APTX-LL

### 2.3.5 Establish/Release A2DP Audio Connection

**Format:** AT+A2DPAUDIO{=Param}

Param: Operation (0/1)

(0) Release A2DP audio connection with remote a2dp sink device

(1) Establish A2DP audio connection with remote a2dp sink device

## 2.4 Bluetooth Serial Commands (BR/EDR SPP)

### 2.4.1 Read SPP State

|                                                                                    |
|------------------------------------------------------------------------------------|
| <b>Format:</b> AT+SPPSTAT                                                          |
| <b>Response:</b> +SPPSTAT=Param<br>Param: Refer to Chapter 3 for state description |

### 2.4.2 Establish SPP Connection

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

### 2.4.3 Release SPP Connection

|                                                                       |
|-----------------------------------------------------------------------|
| <b>Format:</b> AT+SPPDISC                                             |
| <b>Description:</b> Release current SPP connection with remote device |

### 2.4.4 Send Data Via SPP

|                                                                                                                 |
|-----------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+SPPSEND=Param1, Param2<br>Param1: Payload length (1~236)<br>Param2: Payload (1~236Bytes UTF8) |
| <b>Description:</b> If throughput mode is on, this command is de-active                                         |
| <b>Example:</b> Send data "1234567890" to remote device via SPP<br><< AT+SPPSEND=10,1234567890<br>>> OK         |

## 2.5 Bluetooth Serial Commands (LE GATT)

### 2.5.1 Read GATT State

|                                                                                     |
|-------------------------------------------------------------------------------------|
| <b>Format:</b> AT+GATTSTAT                                                          |
| <b>Response:</b> +GATTSTAT=Param<br>Param: Refer to Chapter 3 for state description |

### 2.5.2 Release GATT Connection

|                                                                        |
|------------------------------------------------------------------------|
| <b>Format:</b> AT+GATTDISC                                             |
| <b>Description:</b> Release current GATT connection with remote device |

### 2.5.3 Establish GATT Client Connection

|                                                                                                                                                                                                                                                                                                                                                                        |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+GATTCONN=Param1{,Param2,Param3,Param4}<br>Param1: MAC address of target device & MAC address type (13 Bytes ASCII)<br>Param2: Service-UUID, Support 16 Bit and 128 Bit (4 Bytes/32 Bytes ASCII)<br>Param3: Write-UUID, Support 16 Bit and 128Bit (4 Bytes/32 Bytes ASCII)<br>Param4: Notify-UUID, Support 16 Bit and 128Bit (4 Bytes/32 Bytes ASCII) |
| <b>Example:</b> Specified remote device service connections<br><< AT+LECCONN=123456ABCDEF0,FFF0,FFF2,FFF1<br>>> OK                                                                                                                                                                                                                                                     |

### 2.5.4 Send Data Via GATT

|                                                                                                                   |
|-------------------------------------------------------------------------------------------------------------------|
| <b>Format:</b> AT+GATTSEND=Param1, Param2<br>Param1: Payload length (1~100)<br>Param2: Payload (1~100 Bytes UTF8) |
| <b>Description:</b> 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 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

## 3.3 A2DP Indications

### 3.3.1 A2DP State

**Format:** +A2DPSTAT=Param

Param:(0~5)

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

### 3.3.2 A2DP Device Information

**Format:** +A2DPDEV=Param

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

## 3.4 Bluetooth Serial Indications

### 3.4.1 SPP State

**Format:** +SPPSTAT=Param

Param:(0~3)

- (0) Unsupported
- (1) Standby
- (2) Connecting
- (3) Connected

### 3.4.2 GATT State

**Format:** +GATTSTAT=Param

Param:(0~3)

(0) Unsupported

(1) Standby

(2) Connecting

(3) Connected

### 3.4.3 SPP Device Information

**Format:** +SPPDEV=Param

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

### 3.4.4 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

### 3.4.5 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



## 3.5 GPIO Indications

### 3.5.1 LED Pin

#### **LED0(Output)**

|              |                     |
|--------------|---------------------|
| Low Level    | Initializing        |
| Blink in 1Hz | Ready to connecting |
| High Level   | Connected           |

### 3.5.2 State Pin

#### **LED1(Output)**

|            |                       |
|------------|-----------------------|
| Low Level  | SPP/GATT Disconnected |
| High Level | SPP/GATT Connected    |