FSC-BT90X

BT4.2 Programming User Guide

Version 3.2

Revision History

Version	Data	Notes	Author
1.0	2015/09/10	First Release	Tony
2.0	2016/03/21	Add LE GATT Commands	Jerry
3.0	2016/11/12	Add HID Commands	Navy
3.1	2016/12/13	Add GPIO Indications	Navy
3.2	2019/03/22	Build for BT90X Add GATT Client relevant content Add LENAME\LEADDR\HIDMMU commands Change usage of HIDSEND\PLIST commands Change usage of SPPSTAT indication Correct usage of SCAN indication Delete HSM\PIOCFG\FLOWCTL commands	Jerry

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-BT90X for Bluetooth requirements. It applies to FSC-BT901, FSC-BT906 devices.

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
- SPI Master
- I2C Master/Slave
- Analog Input/Output

1.3 Supported Bluetooth Profile

- SPP (Serial Port Profile)
- GATT Server (Generic Attribute Profile)
- GATT Client (Generic Attribute Profile)
- HID Keyboard (Human Interface 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. Write a baudrate which is not supported
   << *AT+BAUD=0*
   >> *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 ","

e.g.

1. Received "1234567890" from mobile phone via SPP profile
   >> *+SPPDATA=10,1234567890*

1.6 Module Default Settings

Local Name (BR/EDR)	Feasycom
Local Name (LE)	FeasycomLE
Pin Code	0000
Secure Simple Pairing Mode	OFF
Physical UART Baudrate	115200bps/8/N/1

2. Command Table

2.1 General Commands

2.1.1 UART Communication Test

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

2.1.2 Read Firmware Version

Format: AT+VER
Response: +VER=Param Param: Firmware version (16 Bytes ASCII)
Example: Read module's firmware version << *AT+VER* >> *+VER=BT901,2.2.9,20181001* >> *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 Local Name

Format: AT+NAME {=Param1{, Param2}}
Param1: BR/EDR local name (1~31 Bytes ASCII, default: Feasycom)
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: FeasycomLE)
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 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 "1234"
<< *AT+PIN=1234*
>> *OK*

2.1.8 Turn On/Off Secure Simple Pairing

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

2.1.9 Read/Write UART Baudrate

Format: AT+BAUD{=Param} Param: Baudrate (2400/4800/9600/19200/38400/57600/115200/230400/256000/460800/512000/921600, default:115200)
Response: +BAUD=Param
Description: Module's baudrate will be changed immediately after received this command

2.1.10 Read/Write Class Of Device

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

2.1.11 Read/Write Work Mode

Format: AT+MODE{=Param} Param: Work Mode (1~4, default:4) (1) SPP Mode (2) HID Mode (3) BLE Mode (4) SPP+BLE Mode
Response: +MODE=Param
Description: After the command is executed, the module switches to the new Work Mode
Example: Read current Work Mode << *AT+MODE* >> *+MODE=4* >> *OK* **Example:** Change module's Work Mode to HID Mode << *AT+MODE=2* >> *OK*

2.1.12 Read/Clear Paired Record

Format: AT+PLIST{=Param} Param: Control method(0) (0) Clear all paired record
Response1: +PLIST= { **Response2**: +PLIST=Param1, Param2 Param1: (1~8) Paired device's index Param2: (MAC) Paired device's MAC address **Response3**: +PLIST=}
Example: Read module's paired record << *AT+PLIST* >> *+PLIST= {* *+PLIST=1,1C5CF226D773* *+PLIST=2, A0BC30075421* *+PLIST=}* >> *OK* **Example**: Clear module's paired record << *AT+PLIST=0* >> *OK*

2.1.13 Turn On/Off Throughput Mode

Format: AT+TPMODE{=Param} Param: Throughput mode (0/1, default:0) (0) Turn Off (1) Turn On
Response: +TPMODE=Param
Description: When SPP/HID/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*

2.1.14 Turn On/Off Low Power Mode

Format: AT+LPM{=Param} Param: Low Power Mode (0/1, default:0) (0) Turn Off (1) Turn On
Response: +LPM=Param
Description: This instruction is only applicable to BT816S module
Example: Read current Low Power Mode *<< AT+LPM* *>> +LPM=0* *>> OK* **Example:** Turn on Low Power Mode *<< AT+LPM=1* *>> OK*

2.1.15 Release All Connections

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

2.1.16 Soft Reboot

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

2.1.17 Restore Factory Settings

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

2.1.18 Scan Nearby Devices

Format: AT+SCAN =Param1{, Param2{, Param3}} Param1:(0~3) (0) Stop scan (1) Scan nearby BR/EDR devices (2) Scan nearby BLE devices (3) Scan nearby BR/EDR/BLE devices Param2:(1~48) Scan period. unit:1.28s, default:12.8s Param3:(1~25 Bytes ASCII) Name filter. Filter scan results with name if set
Description: Refer to Chapter 3 for format description of scan result

2.2 Bluetooth Serial Commands (BR/EDR SPP)

2.2.1 Read SPP State

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

2.2.2 Turn On/Off SPP Power On Auto Reconnect

Format: AT+SPPAC{=Param} Param: Option (0/1, default:0) (0) Turn Off (1) Turn On

Response: +SPPAC=Param
Description: Module will attempt to connect last device after power on if set the param as 1

2.2.3 Establish SPP Connection

Format: AT+SPPCONN{=Param} Param: MAC address of target device (12 Bytes ASCII)
Description: If the parameter does not exist, the module will attempt to connect to the last device

2.2.4 Release SPP Connection

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

2.2.5 Send Data Via SPP

Format: AT+SPPSEND=Param1, Param2 Param1: Payload length (1~256) Param2: Payload (1~256Bytes 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*

2.3 Bluetooth Serial Commands (LE GATT Server)

2.3.1 Read GATT Server State

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

2.3.2 Release GATT Connection

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

2.3.3 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*

2.4 Bluetooth Serial Commands (LE GATT Client)

2.3.1 Read GATT Client State

Format: AT+LECSTAT

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

2.3.2 Establish GATT Connection

Format: AT+ LECCONN=Param1, Param2, Param3, Param4, Param5 Param1: Remote device's LE MAC address (12 Bytes ASCII) Param2: MAC address type(0~1) Param3: Service UUID (16 or 128 bits Hex) Param4: Write UUID (16 or 128 bits Hex) Param5: Notify UUID (16 or 128 bits Hex)
Description: Establish GATT connection with remote device by specific UUIDs
Example: Connect to remote device via GATT by 16 bits UUID << *AT+LECCONN=DD0D30101234,0,FFF0,FFF2,FFF1* >> *OK* **Example**: Connect to remote device via GATT by 128 bits UUID << *AT+LECCONN=000D30101234,1,49535343FE7D4AE58FA99FAFD205E455,49535343 884143F4A8D4ECBE34729BB3,495353431E4D4BD9BA6123C647249616* >> *OK*

2.3.3 Release GATT Connection

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

2.3.4 Send Data Via GATT

Format: AT+LECSEND=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+LECSEND=10,1234567890* >> *OK*

2.5 Bluetooth Serial Commands (BR/EDR HID)

2.5.1 Read HID State

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

2.5.2 Turn On/Off HID Power On Auto Reconnect

Format: AT+HIDAC{=Param} Param: Option (0/1, default:1) (0) Turn Off (1) Turn On
Response: +HIDAC=Param
Description: Module will attempt to connect last device after power on if set the param as 1

2.5.3 Establish HID Connection

Format: AT+HIDCONN{=Param} Param: MAC address of target device (12 Bytes ASCII)
Description: If the parameter does not exist, the module will attempt to connect to the last device

2.5.4 Release HID Connection

Format: AT+HIDDISC
Description: Release current HID connection with remote device

2.5.5 Read/Write HID Send Delay

Format: AT+HIDDLY{=Param} Param: HID Send Delay (2~4 Bytes ASCII,Default:10)
Response: +HIDDLY=Param
Description: Different phones may require different delay settings to achieve the best HID transmission speed and stability
Example: Read current HID Send Delay << *AT+HIDDLY* >> *+HIDDLY=10* >> *OK*

2.5.6 IOS Device On-screen Keyboard Toggle

Format: AT+HIDOSK
Description: This instruction applies only to the IOS Device

2.5.7 Send Data Via HID

Format: AT+HIDSEND=Param1, Param2 Param1: Payload length (even, 2,4,6,...,256) Param2: Payload (2,4,6,...,256Bytes HID key)
Description: If throughput mode is on, this command is de-active

Example: Send data "12" to remote device via HID, '\x00\x1E\x00\x1F' below is four bytes of Hex, and it's NOT printable string, except this, all the other characters are ASCII. << *AT+HIDSEND=4,\x00\x1E\x00\x1F* >> *OK*

2.5.8 Read HID Transmit Buffer Realtime Count

Format: AT+HIDMMU
Response: +HIDMMU =Param1, Param2 Param1: HID transmit buffer total size (1~8192) Param2: HID transmit buffer rest size (1~8192)
Description: When total size equals to rest size, this means HID transmit buffer is empty.
Example: Read HID transmit buffer realtime count << *AT+HIDMMU* >> *+HIDMMU=4096,4090* >> *OK*

3. Indication Table

3.1 General Indications

3.1.1 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 nearby BR/EDR devices

<< *AT+SCAN=1*

>> *OK*

+SCAN=1,2, DC0D30000003, -32,8, Feasycom

+SCAN=2,2, DC0D30000044, -64,8, Feasycom

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

3.2 Bluetooth Serial Indications

3.2.1 SPP State

Format: +SPPSTAT=Param

Param:(0~4)

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

3.2.2 GATT State

Format: +GATTSTAT=Param

Param:(0~3)

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

3.2.3 HID State

Format: +HIDSTAT=Param

Param:(0~3)

(0) Unsupported
(1) Standby

(2) Connecting (3) Connected

3.2.4 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*

3.2.5 GATT Server 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*

3.2.6 GATT Client Received Data

Format: +LECDATA=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 << *+LECDATA=10,1234567890*

3.3 GPIO Indications

3.3.1 LED Pin

PIN32 (Output)	
Low Level	Initializing
Blink in 1Hz	Ready to connecting
High Level	Connected

3.3.2 State Pin

PIN33 (Output)	
Low Level	Disconnected
High Level	Connected