FSC-BT616 Programming User Guide

Release 5.0.2

Table of contents

1 Introduction **1**

1.1 Description 1

1.2 Module Default Settings 1

2 Hardware Description **2**

2.1 Pin Diagram 2

2.2 Pin Description 3

2.3 Hardware Design Notes 3

3 Function Description **5**

3.1 GPIO state 5

3.2 Mode 6

3.3 GATT Throughput Service 6

3.4 Performance parameters 7

3.5 Data transfer rate 7

3.6 Low Power description 7

4 AT command description **8**

4.1 Command description 8

4.2 Command 8

4.3 Command Format 8

4.4 Event 9

4.5 Event Format 9

5 Commands Table **11**

5.1 AT - Serial communication test 11

5.2 AT+VER - Get Firmware Version 11

5.3 AT+ADDR - Get MAC Address 12

5.4 AT+NAME - Get/Set Local Name 13

5.5 AT+BAUD - Get/Set Uart Baudrate 14

Shenzhen Feasycom Technology Co.,Ltd.

5.6 AT+TXPOWER - Get/Set TX Power . . . 15
5.7 AT+LPM - On/off low power config . . . 16
5.8 AT+ADVIN - Get/Set advertising interval . . . 16
5.9 AT+DISC - Disconnect specified connection . . . 17
5.10 AT+TPMODE - Get/Set throughput mode . . . 18
5.11 AT+ADVDATA - Get/Set advertising manufacturer specific data . . . 19
5.12 AT+IBEACON - Get/Set Switch for BLE broadcast 0xFF . . . 20
5.13 AT+LESEND - Send Data to Remote Device . . . 20
5.14 AT+REBOOT - Soft Reboot . . . 21
5.15 AT+RESTORE - Restore Factory Settings . . . 21
5.16 AT+PIN - Get/Set Pin Code . . . 22
5.17 AT+TYPE - Get/Set How the binding is initiated . . . 22
5.18 AT+UUIDCFG - UUID used for write through transmission . . . 23
5.19 AT+PIOCFG - Get/Set pin function switch . . . 24
5.20 AT+CONFIG - Initiates connection parameter updates . . . 25
5.21 AT+STATE - Query connection status . . . 25

6 Event Table **27**
6.1 +GATTDATA - GATT Received Incoming Data . . . 27

7 Application scenarios **28**
7.1 Get/Set the default parameters of the module . . . 28
7.2 Process of sending data . . . 29
7.3 The module acts as the central to connect to the remote device . . . 31

8 FAQ **33**
8.1 How does an IOS mobile phone scan the Bluetooth MAC address? . . . 33

9 Appendix **35**
9.1 Download PDF Document . . . 35

Shenzhen Feasycom Technology Co., Ltd.

Chapter 1

Introduction

[中文]

1.1 Description

This design guide is suitable for engineers to develop FSC-BT616 series Bluetooth modules

1.2 Module Default Settings

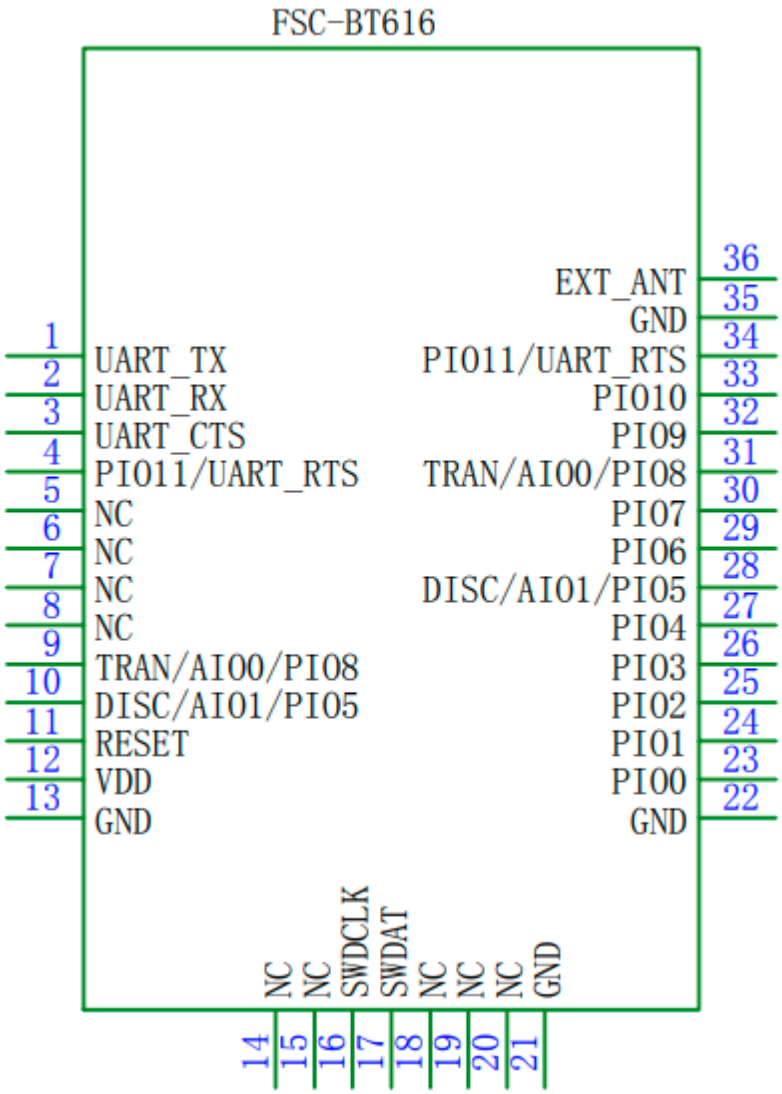
Name	Feasycom
Service UUID	FFF0
Write UUID	FFF2
Notify UUID	FFF1
UART Baudrate	115200/8/N/1

Shenzhen Feasycom Technology Co.,Ltd.

Chapter 2

Hardware Description

2.1 Pin Diagram



nology Co.,Ltd.

2.2 Pin Description

Pin	Pin Name	Type	Pin Descriptions
1	UART_TX	O	UART TX
2	UART_RX	I	UART RX
3	UART_RTS	I/O	UART RTS
4	UART_CTS	I/O	UART CTS
11	RESET	I	Restore Factory Settings
12	VDD	Power	Power supply for I/O ports, DC 3.3V
13	GND	GND	GND
16	SWCLK	I/O	SWD CLK line(Default)
17	SWDIO	I/O	SWD DATA line(Default)
32	LED	O	Output square wave in pairing mode, output high level when bluetooth is connected
33	STATUS	O	bluetooth is not connected to output low level, connected to output high level
36	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/GND/UART_RX/UART_TX to use
- If the MCU needs to obtain the connection status of the Bluetooth module, it needs to be connected to the STATUS pin
- After drawing the schematic diagram, please send it to Feasycom for review, so as to pre-

Shenzhen Feasycom Technology Co.,Ltd.

vent the Bluetooth distance from not reaching the best effect

- The module supports UART wake-up
- VDD/GND/RESET/SWCLK/SWDIO is burner hole, you can reserve solder joints for testing

Shenzhen Feasycom Technology Co.,Ltd.

Chapter 3

Function Description

3.1 GPIO state

LED PIN is PIN 32

State	Description
0.83Hz square wave	disconnect state
high level	connected state

Connection status PIN is PIN 33

State	Description
low level	disconnect state
high level	connected state

Shenzhen Feasycom Technology Co.,Ltd.

3.2 Mode

Throughput mode	when disconnected, the data received by the serial port is analyzed according to the AT command; when connected, all the data received by the serial port is sent to the remote Bluetooth as it is.
Command mode	when disconnected, the data received by the serial port is analyzed according to the AT command; when connected, The data received by the serial port is still parsed according to the AT command. When it is necessary to send data to the remote end, send AT+LESEND command.

3.3 GATT Throughput Service

type	UUID	Permissions	Description
Service	0xFFF0		GATT Through-put Service
Write	0xFFF2	Write， Write Without Response	APP send data to module
Notify	0xFFF1	Notify	Module send data to remote

Shenzhen Feasycom Technology Co.,Ltd.

3.4 Performance parameters

type	Min.	Type.	Max.	Description
power on		50ms		loading finished
wake up		100ms		wake up from sleep

3.5 Data transfer rate

Baudrate	packet size	interval	conn inter-val	transmit mode	rate
921600	244	5ms	15ms	Notify(2M PHY)	48800 Byte/s

3.6 Low Power description

The module supports one low-power mode, serial port wake-up mode.

Mode	command	hardware link-ing	sleep	Description
serial mode	AT+LPM=1		The serial port automatically sleeps if no data communication occurs within 10 seconds After the serial port receives the first frame of data, the serial port exits the sleep	The first frame of wake up data will lost Simple logic, save I/O

Shenzhen Feasycom Technology Co.,Ltd.

Chapter 4

AT command description

4.1 Command description

Throughout this specification

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

4.2 Command

The command is the control command sent by the host to the module actively. After receiving the | command, the module needs to reply **<CR><LF>OK<CR><LF>** as a response.

4.3 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

Shenzhen Feasycom Technology Co.,Ltd.

- 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>**
- Modules should always return the result of instruction execution (**OK** on success, **ERROR** on failure)

Example:

Read module's name

<< AT+VER

>> +VER=1.0.0,FSC-BT616

Write invalid baudrate

<< AT+BAUD=0

>> ERROR

4.4 Event

Event is the data that the module actively sends to the host. Generally used to indicate a change of

state or data received.

4.5 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 ','

Shenzhen Feasycom Technology Co.,Ltd.

Example:

Receive data from APP in command mode

>> +DATA=0,10,1234567890

Shenzhen Feasycom Technology Co.,Ltd.

Chapter 5

Commands Table

5.1 AT - Serial communication test

Command	**AT**
Response	**OK**
Description	When powering on or changing the baud rate, test the UART communication between the host and the module

Example:

<< AT

>> OK

5.2 AT+VER - Get Firmware Version

Command	**AT+VER**
Response	**+VER=Param**
`Param`	Firmware version
Description	Get Firmware Version

Shenzhen Feasycom Technology Co.,Ltd.

Example:

<< AT+VER

>> +VER=5.0.2,BT616

>> OK

5.3 AT+ADDR - Get MAC Address

Command	**AT+ADDR**
Response	**+ADDR=Param**
`Param`	Module’s MAC address (12 Bytes ASCII)
Description	Get MAC Address

Example:

<< AT+ADDR

>> +ADDR=DC0D30010203

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

5.4 AT+NAME - Get/Set Local Name

Command	**AT+NAME{=Param1{,Param2}}**
Param1	local name(1~29 Bytes ASCII)
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**
Param	local name
Description	Write local name if parameter exist, otherwise read current local name

Example:

Read current local name

<< AT+NAME

>> +NAME=Feasycom

>> OK

Change module’s local name to “ABC”

<< AT+NAME=ABC

>> OK

Change module’s local name to “ABC” and enable suffix

<< AT+NAME=ABC,1

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

5.5 AT+BAUD - Get/Set Uart Baudrate

Command	**AT+BAUD{=Param}**
`Param`	baudrate (1200/2400/4800/9600/19200/38400/57600/115200/ 230400/460800/921600, default:115200)
Response	**+BAUD=Param**
`Param`	baudrate
Description	Module will change baudrate to target value immediately

Example:

Query baudrate

<< AT+BAUD

>> +BAUD=115200

>> OK

Change baudrate

<< AT+BAUD=9600

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

5.6 AT+TXPOWER - Get/Set TX Power

Command	**AT+TXPOWER{=Param}**
`Param`	tx power (0~C, default:7) 0: -21dbm 1: -18dbm 2: -15dbm 3: -12dbm 4: -9dbm 5: -6dbm 6: -3dbm 7: 0dbm 8: 1dbm 9: 2dbm A: 3dbm B: 4dbm C: 5dbm
Response	**+TXPOWER=Param**
`Param`	tx power
Description	Get/Set TX Power

Example:

Get tx power

<< AT+TXPOWER

>> +TXPOWER=7

>> OK

Set tx power to 5dbm

<< AT+TXPOWER=C

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

5.7 AT+LPM - On/off low power config

Command	**AT+LPM{=Param}**
`Param`	low power mode (0/1, default: 0) 0: disable 1: enable low power and wake up by UART
Response	**+LPM=Param**
`Param`	low power mode
Description	Get/Set low power config

Example:

Get low power config

<< AT+LPM

>> +LPM=0

>> OK

Set low power config

<< AT+LPM=1

>> OK

5.8 AT+ADVIN - Get/Set advertising interval

Command	**AT+ADVIN{=Param}**
`Param`	advertising interval(20~10000 ms, default: 152ms)
Response	**+ADVIN=Param**
`Param`	advertising interval
Description	Get/Set advertising interval

Shenzhen Feasycom Technology Co.,Ltd.

Example:

Get advertising interval

<< AT+ADVIN

>> +ADVIN=152

>> OK

Set advertising interval to 100ms

<< AT+ADVIN=100

>> OK

5.9 AT+DISC - Disconnect specified connection

Command	**AT+DISC**
Response	**OK**
Description	Disconnect Bluetooth connection

Example:

Disconnect device

<< AT+DISC

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

5.10 AT+TPMODE - Get/Set throughput mode

Command	**AT+TPMODE{=Param}**
`Param`	Throughput mode (0~1, default 1) 0: Command Mode 1: Throughput Mode
Response	**+TPMODE=Param**
`Param`	Throughput mode
Description	When bluetooth connected and in throughput mode,the AT command will be de-active,every byte received via physical UART will be sent to remote.

Example:

Throughput Mode

<< AT+TPMODE=1

>> OK

Command Mode

<< AT+TPMODE=0

>> OK

ShenZhen Feasycom Technology Co.,Ltd.

5.11 AT+ADVDATA - Get/Set advertising manufacturer specific data

Command	**AT+ADVDATA{=Param}**
`Param`	manufacturer specific data, tag 0xff
Response	**+ADVDATA=Param**
`Param`	manufacturer specific data, tag 0xff
Description	Get/Set advertising manufacturer specific data

Example:

Get advertising manufacturer specific data, x is hexadecimal data

<< AT+ADVDATA

>> +ADVDATA=\x03\x01\x02

>> OK

Set advertising manufacturer specific data, x is hexadecimal data

<< AT+ADVDATA=\x03\x01\x02

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

5.12 AT+IBEACON - Get/Set Switch for BLE broadcast 0xFF

Command	**AT+IBEACON{=Param}**
`Param`	mode（0~1，default: 1） 0: Turn off BLE broadcast 0xFF 1: Example Enable BLE broadcast 0xFF
Response	**+IBEACON=Param**
Description	After this function is enabled,BLE switches between two types of broadcast data

Example:

Example Enable BLE broadcast 0xFF

<< AT+IBEACON=1

>> OK

Turn off BLE broadcast 0xFF

<< AT+IBEACON=0

>> OK

5.13 AT+LESEND - Send Data to Remote Device

Command	**AT+LESEND{=Param1{,Param2}}**
`Param1`	data len
`Param2`	data
Response	**OK**
Description	send data to remote device in Command Mode

Shenzhen Feasycom Technology Co.,Ltd.

Example:

Send data to connected devices

<< AT+LESEND=4,2022

>> OK

5.14 AT+REBOOT - Soft Reboot

Command	**AT+REBOOT**
Response	**OK**
Description	Module release all Bluetooth connections then reboot

Example:

<< AT+REBOOT

>> OK

5.15 AT+RESTORE - Restore Factory Settings

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

Example:

<< AT+RESTORE

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

5.16 AT+PIN - Get/Set Pin Code

Command	**AT+PIN{=Param}**
`Param`	Pin Code (000000 ~ 999999, default 000000)
Response	**+PIN=Param**
`Param`	Pin Code
Description	Get/Set Pin Code

Example:

Get Pin Code

<< AT+PIN

>> +PIN=000000

>> OK

Set Pin Code 123456

<< AT+PIN=123456

>> OK

5.17 AT+TYPE - Get/Set How the binding is initiated

Command	**AT+TYPE{=Param}**
`Param`	mode（0~1，default: 0） 0: Wait for the primary end to initiate 1: Take the initiative
Response	**+TYPE=Param**
Description	Change the way the binding is initiated

Example:

Shenzhen Feasycom Technology Co.,Ltd.

Set active initiation

<< AT+TYPE=1

>> OK

Set waiting for the primary end to initiate

<< AT+TYPE=0

>> OK

5.18 AT+UUIDCFG - UUID used for write through transmission

Command	**AT+UUIDCFG=Param1{,Param2, Param3{,Param4, Param5{,Param6, Param7}}}**
`Param1`	Set the number of UUids

`Param2`
`Param4`
`Param6`
	The last parameter sets the type of the UUID(1/2/3)
	1: Change the UUID of the SERVER
	2: Change the UUID of notify
	3: Change the write UUID

`Param3`
`Param5`
`Param7`
	The UUID value can be 16 or 128 case-insensitive and can contain a hyphen (-)

Response	**OK**
Description	Change the way the binding is initiated

Example:

Set the SERVER UUID to 0xFFE0

<< AT+UUIDCFG=1,1,FFE0

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

Set the notify UUID 为 11223344-5566-7788-9900-aabbccddeeff,write UUID 为 aabbccdd-eeff-0011-2233-445566778899

<<

AT+UUIDCFG=2,2,1122334455667788990aabbccddeeff,3,aabbccddeeff00112233445566778899

>> OK

5.19 AT+PIOCFG - Get/Set pin function switch

Command	**AT+PIOCFG{=Param1{,Param2}}**
`Param1`	operating mode(0~1， default: 0).PIN9 Low level transparent mode, high level command mode. 0: Turn off work mode pin control 1: Enable work mode pin control
`Param2`	Disconnect the connection (0~1, default: 1). The falling edge of PIN10 is disconnected. 0: Turn off the disconnect pin function 1: Enable the disconnect pin function
Response	**+PIOCFG=Param1,Param2**
Description	Modify the pin function switch

Example:

Enable pin control working mode

<< AT+PIOCFG=1

>> OK

Disable the function of two pins

<< AT+PIOCFG=0,0

Shenzhen Feasycom Technology Co.,Ltd.

>> OK

5.20 AT+CONFIG - Initiates connection parameter updates

Command	**AT+CONFIG=Param1,Param2,Param3,Param4**
`Param1`	Minimum connection interval
`Param2`	Maximum connection interval
`Param3`	Connection delay
`Param4`	connection timeout
Response	**OK**
Description	Initiates connection parameter updates

Example:

Initiates connection parameter updates

<< AT+CONFIG=15,30,0,3000

>> OK

5.21 AT+STATE - Query connection status

Command	**AT+STATE**
Response	**+STATE=Param**
`Param`	Connected or not(0/1) 0: ununited 1: connected
Description	Query connection status

Example:

Shenzhen Feasycom Technology Co.,Ltd.

Query connection status

<< AT+STATE

>> +STATE

>> OK

Shenzhen Feasycom Technology Co.,Ltd.

Chapter 6

Event Table

6.1 +GATTDATA - GATT Received Incoming Data

Indication	**+GATTDATA=Param1,Param2,Param3**
`Param1`	Index
`Param2`	Payload length
`Param2`	Payload
Description	In the throughput mode: report data without +GATTDATA header In the command mode: report data with +GATTDATA header +GATTDATA=1,5,12345

Example:

received data 1234567890

>> +GATTDATA=10,1234567890

Shenzhen Feasycom Technology Co.,Ltd.

Chapter 7

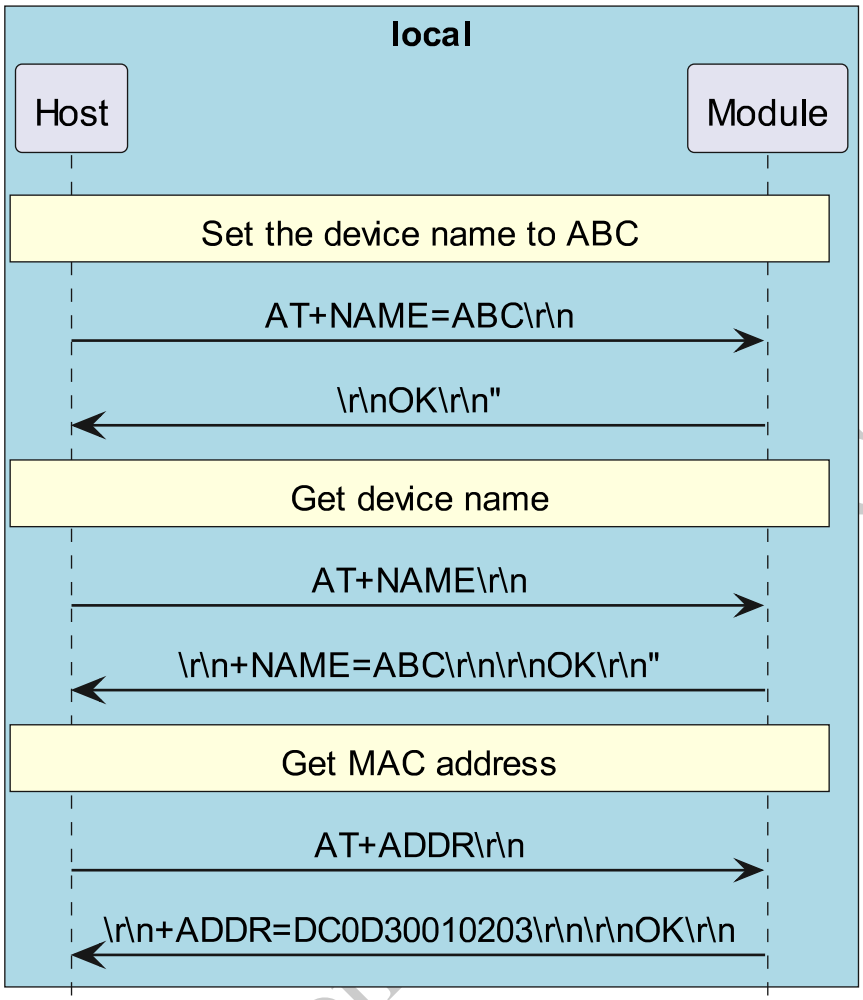
Application scenarios

7.1 Get/Set the default parameters of the module

When the module is disconnect state, it will parse the uart data according to the AT command. host can get and set the default parameters of the module, as shown in the figure below:

1. Set the device name to ABC
2. Get device name
3. Get MAC address

Shenzhen Feasycom Technology Co.,Ltd.



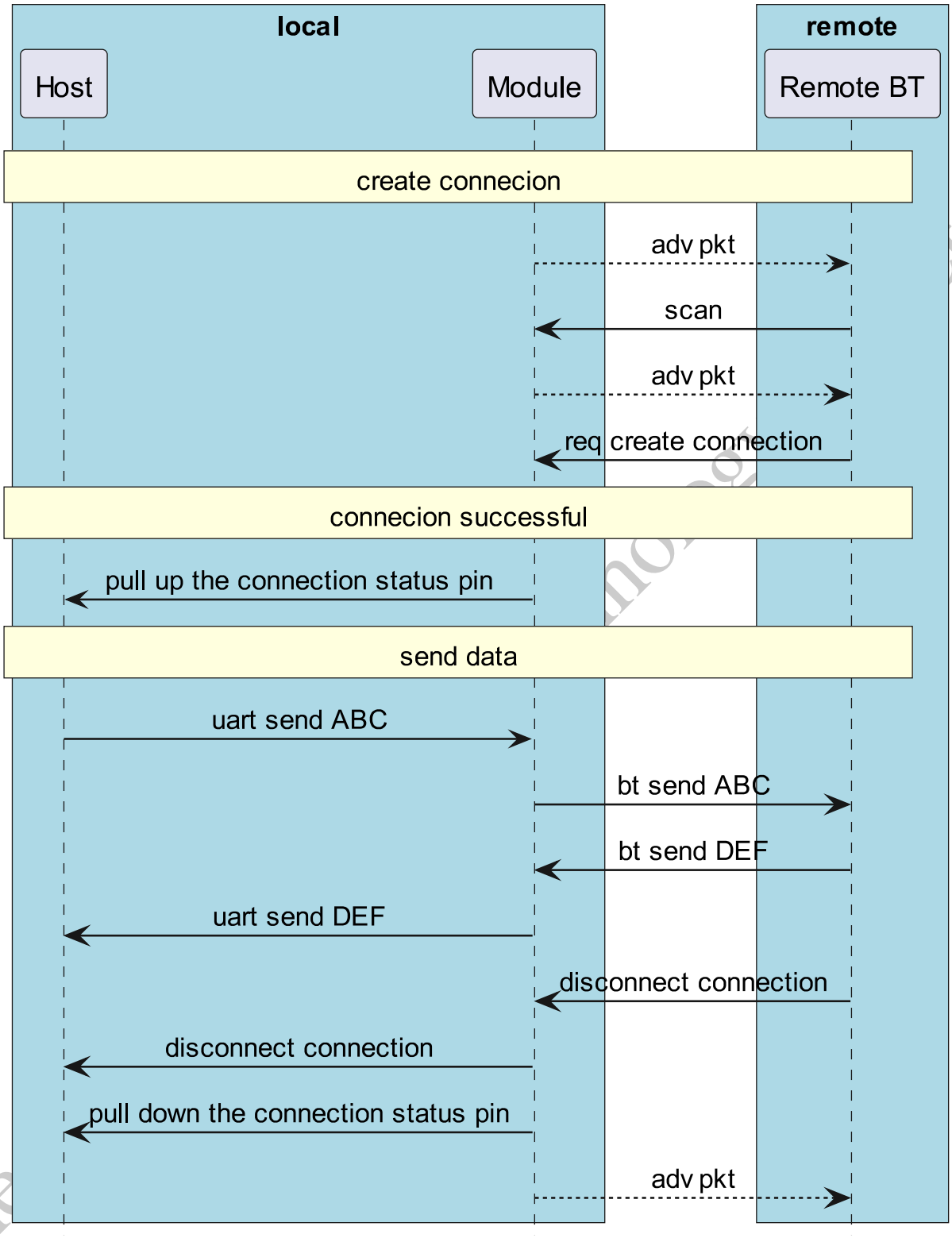
7.2 Process of sending data

When the module is powered on, it will continue to send advertising data, and the remote device (mobile phone) can scan the advertising packet. And initiate a connection request to the module.

After the connection is successful, the module will pull up the connection status pin to notify the host that the Bluetooth connection is successful. The host can send data to the remote device through

the Bluetooth module, and the remote device can also send data to the host.

Shenzhen Feaycon Co.,Ltd.

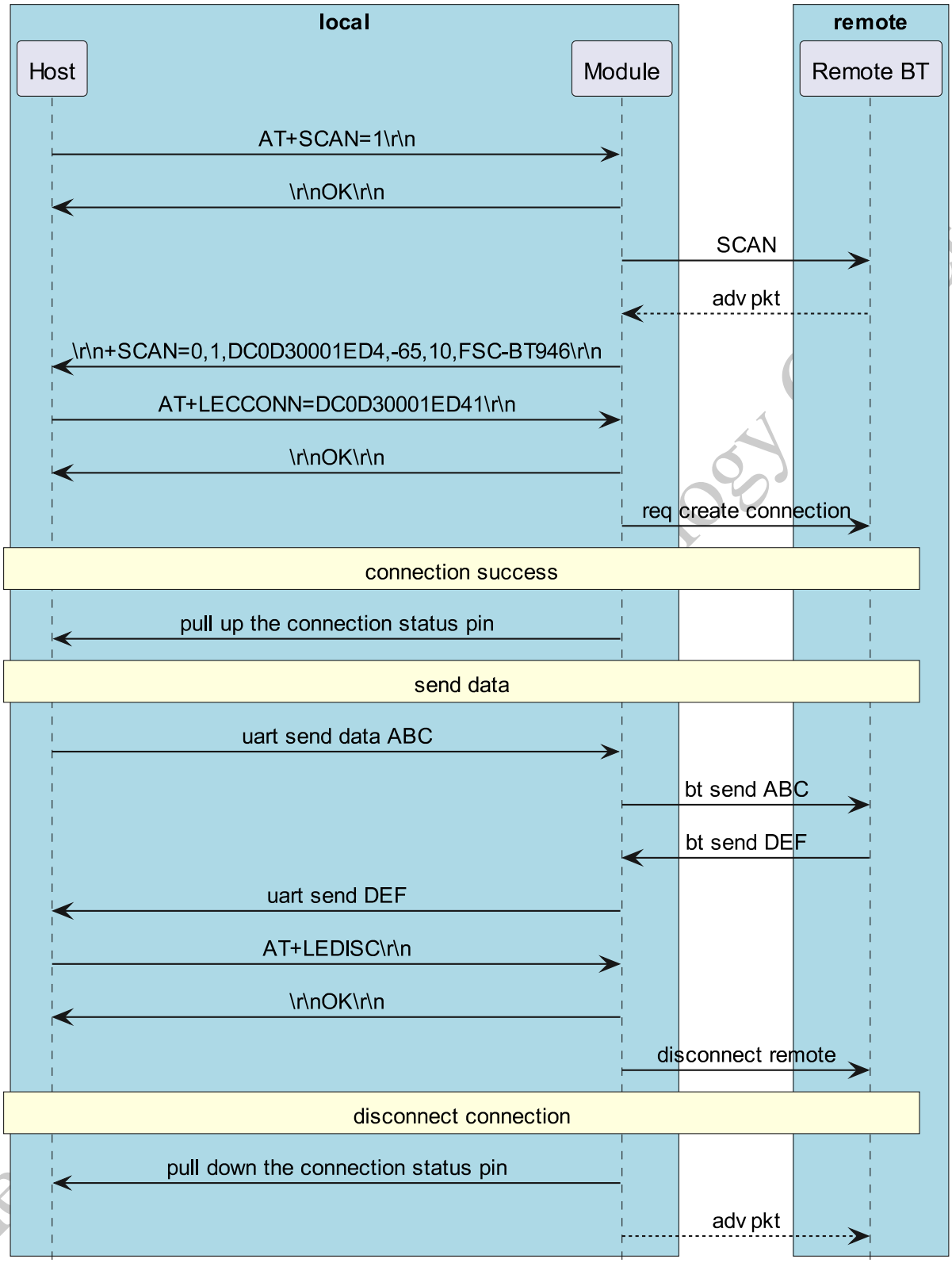

local
remote
Host
Module
Remote BT
create connecion
adv pkt
scan
adv pkt
req create connection
connecion successful
pull up the connection status pin
send data
uart send ABC
bt send ABC
bt send DEF
uart send DEF
disconnect connection
disconnect connection
pull down the connection status pin
adv pkt

7.3 The module acts as the central to connect to the remote device

The module can be used as a master device to connect to the slave device, and the host can send commands to control the module to scan and connect and disconnect. The figure below shows the

process of connecting other devices:

Shenzhen Feasycom Technology Co.,Ltd.


local
remote
Host
Module
Remote BT
AT+SCAN=1\r\n
\r\nOK\r\n
SCAN
adv pkt
\r\n+SCAN=0,1,DC0D30001ED4,-65,10,FSC-BT946\r\n
AT+LECCONN=DC0D30001ED41\r\n
\r\nOK\r\n
req create connection
connection success
pull up the connection status pin
send data
uart send data ABC
bt send ABC
bt send DEF
uart send DEF
AT+LEDISC\r\n
\r\nOK\r\n
disconnect remote
disconnect connection
pull down the connection status pin
adv pkt

Chapter 8

FAQ

8.1 How does an IOS mobile phone scan the Bluetooth MAC address?

For security reasons, the IOS system converts the Bluetooth MAC address into a UUID at the bottom layer and sends it to the upper layer application. Therefore, the APP cannot get the MAC address of the device.

FSC-BT616 will put the MAC address in the advertising packet by default, and the APP can get the MAC address from the advertising packet through the following methods.

```
- (void)centralManager:(CBCentralManager *)central
 ↪didDiscoverPeripheral:(CBPeripheral *)peripheral
 ↪advertisementData:(NSDictionary *)advertisementData
 ↪RSSI:(NSNumber *)RSSI
{
    if(![self describeDictonary:advertisementData])
    {
        NSLog(@"is not fsc module");
        return;
    }
}

- (Boolean)describeDictonary: (NSDictionary *) dict
```

(continues on next page)

Shenzhen Feasycom Technology Co.,Ltd.

(continued from previous page)

```
{
    NSArray *keys;
    id key;
    keys = [dict allKeys];
    for(int i = 0; i < [keys count]; i++)
    {
        key = [keys objectAtIndex:i];
        if([key isEqualToString:@
↪"kCBAdvDataManufacturerData"])
        {
            NSData *tempValue = [dict objectForKey:key];
            const Byte *tempByte = [tempValue bytes];
            if([tempValue length] == 6)
            {
                // The parameter after tempByte is the␣
↪Bluetooth address
                return true
            }
        }else if([key isEqualToString:@"kCBAdvDataLocalName
↪"])
        {
            //there is name
            //NSString *szName = [dict objectForKey: key];
        }
    }
    return false;
}
```

Shenzhen ,Ltd.

Chapter 9

Appendix

9.1 Download PDF Document

Download PDF Document

Shenzhen Feasycom Technology Co.,Ltd.