GSM BT
Application Note

GSM/GPRS Module Series

Rev. GSM_BT_Application_Note_V1.2

Date: 2015-06-25

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.

Office 501, Building 13, No.99, Tianzhou Road, Shanghai, China, 200233

Tel: +86 21 5108 6236

Mail: info@quectel.com

Or our local office, for more information, please visit:

http://www.quectel.com/support/salesupport.aspx

For technical support, to report documentation errors, please visit:

http://www.quectel.com/support/techsupport.aspx

Or Email: Support@quectel.com

About the Document

History

Revision	Date	Author	Description
1.0	2015-01-13	Victor CHEN	Initial
1.1	2015-03-13	Victor CHEN	1. Added independence description of Bluetooth function in Chapter 1.5 2. Added notes for operation in Chapter 2.1 3. Added pairing mode in Chapter 2.5 4. Added AT+QBTCONND in Chapter 3.2.14
1.2	2015-06-25	Hyman DING/ Ramos ZHANG	1. Added description of HFG in Chapter 3.2.12-3.2.18 2. Added AT+QBTHFGAUDIO in Chapter 3.2.27 3. Added Transparent Access Mode in SPP profile 4. Added example for Transparent Access Mode 5. Added HFP AT command and example

Contents

Table Index

1 Introduction

This document mainly introduces how to use the BT (Bluetooth) function of Quectel standard module.

This document is applicable to Quectel M66 module.

1.1. General Introduction

BT is a proprietary open wireless technology standard for exchanging data over short distances (using short-wavelength radio transmissions in ISM band from 2400-2480MHz) from fixed and mobile devices, creating personal area networks with high levels of security. BT was standardized as IEEE802.15.1. And Quectel modules support BT3.0.

1.2. BT Profiles

To use BT wireless technology, a device must be able to interpret certain BT profiles. BT profiles are definitions of possible applications and specify general behavior that BT enables devices to communicate with other BT devices. There is a wide range of BT profiles describing many different types of applications or use cases for devices. Through the guidance provided by the BT specification, you can create applications to work with other BT devices. Besides of all profiles, there are four basic ones: GAP, SDAP, SPP and GOEP profiles. Quectel modules support SPP and HFP.

1.3. BT Address

Each BT transceiver is allocated with a unique 48-bit BT device address. This address is derived from the IEEE802 standard. This 48-bit address is divided into three fields:

● LAP field: lower address consisting of 24 bits
● UAP field: upper address consisting of 8 bits
● NAP field: non-significant address consisting of 16 bits

The LAP and UAP indicates the significant address of the BT device address. Quectel module with BT function will specify an address before it leaves the factory.

1.4. AT Interface for BT Function

As module solution, we provide a set of AT commands to support basic BT operations, including scanning, pairing, connecting and so on. Our modules support SPP, HFP profile. The SPP profile could be used to communicate between module and others BT device via serial port. And through the HFP profile, the module can act as a HF (Hands-Free Unit) device or AG (Audio Gateway) device.

1.5. Independence of BT

The Bluetooth function could work without SIM card or in flight mode (**AT+CFUN=0/4**).

2 BT Operation

To operate BT function, you should connect the module to a BT device, and turn on BT power by AT commands. The first time using BT device on Quectel module, you need to scan and pair the device with module to ensure the BT device is connected successfully.

The following figure shows the detailed operation of BT (SPP profile) by AT commands as an example.
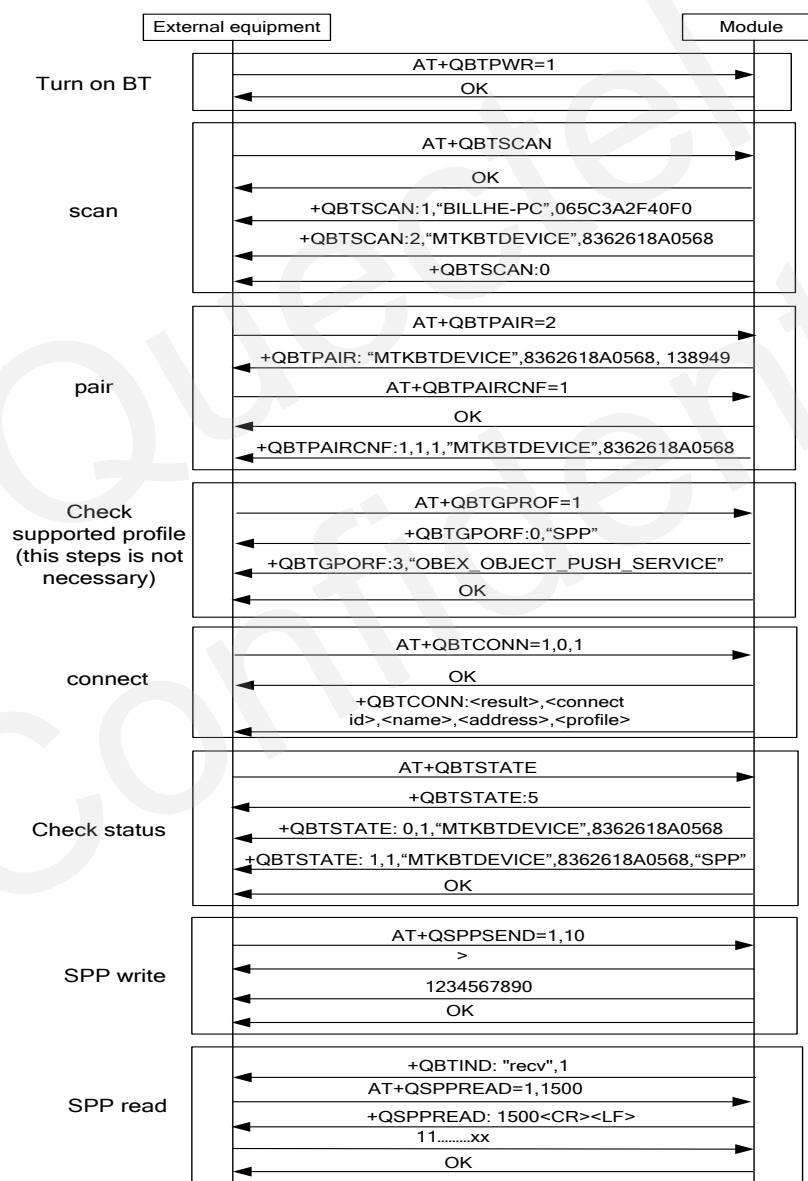
Figure 1: BT (SPP profile) Operation Flow Chat

2.1. Notes for Operation

2.1.1. Number of Searched Devices

The max number of searched devices is 20 in a single scanning process.

2.1.2. Number of Paired Devices

The max number of paired devices stored in your device is 10. If you want to continue to pair new device after the space was full, you must remove the paired devices to free the storage space (by command "**AT+QBTUNPAIR=<paired device id>**").

2.1.3. Special Function

If you want to use other BT device address and the password (PINCODE Pair) to connect SPP directly without scanning (pairing process will work internally), you can use the command "**AT+QBTCONND=<addr>[,<mode>[,<password>]]**". Refer to Chapter 2.7.3 to get more details.

2.2. Power On/Off

To use the BT function, you must power on BT device first.

AT+QBTPWR=1	//Power on BT device
OK	
AT+QBTPWR?	//Check power status
+QBTPWR: 1	//BT device is powered on
OK	

Turn off BT device if not used.

AT+QBTPWR=0	//Power off the BT device.
OK	

NOTE

While BT is turned on, the module is visible by other BT devices. To make the module covered, you need to set it as invisible.

2.3. Attribution Configurations

2.3.1. Change BT Name

You can change the BT name by executing **AT+QBTNAME**.

```
AT+QBTNAME="MYBTDEVICE"
OK
AT+QBTNAME?
+QBTNAME: "MYBTDEVICE"

OK
```

2.3.2. Set BT Visibility

```
AT+QBTVISB=0                              //Invisible
OK
AT+QBTVISB=1                              //Visible forever
OK
AT+QBTVISB=2,60                           //Visible for 60 seconds
OK

+QBTVISB: 0                               //Visible time out, invisible now
```

2.3.3. Read Address

```
AT+QBTADDR?
+QBTADDR: A662616202C3

OK
```

2.4. Scan BT Devices

The module scans and displays device ID, name and address of all available BT devices around it.

If the device you want to pair with is not in the list, ensure that the device is turned on and visible. You could use **AT+QBTSCANC** to cancel the scan process and **AT+QBTSCAN** to restart.

2.4.1. Scan until Timeout

```
AT+QBTSCAN                                    //Start to scan
OK

+QBTSCAN: 1,"BILLHE-PC",065C3A2F40F0

+QBTSCAN: 2,"GAOSHENGHAI",8E1F49F4402C

+QBTSCAN: 3,"BTDEVICE",8362618A0568

+QBTSCAN: 4,"QUECTEL-RAMOS",0AB6F27D001A

+QBTSCAN: 0                                   //Complete
```

2.4.2. Cancel Scan

```
AT+QBTSCAN                                    //Start to scan
OK

+QBTSCAN: 1,"M66",8362618A0568

+QBTSCAN: 2,"QUECTEL-RAMOS",0AB6F27D001A

AT+QBTSCANC                                   //Cancel scan
+QBTSCAN: 0

OK
```

2.5. Pair/Unpair BT Device

The visible devices will be shown in the list with the device ID, name and address. **AT+QBTSTATE** can be used to query the pairing status. You need to select the device scan ID, and use **AT+QBTPAIR=<scan id>** and **AT+QBTPAIRCNF=1** to pair and confirm it, then pairing result will be shown.

After successfully paired, the module will show the paired device ID. You can also use the **AT+QBTSTATE** to query the pairing status. Meanwhile, pairing information is saved as non-volatile; you only need to pair the device once.

Currently, Quectel supports two pairing modes:

- **PINCODE**

PIN Code pairing requires both sides of Bluetooth devices input matching password (If both of them have input devices), or one inputs matching password while the other automatic confirms it.

- **SSP--Numeric Compare Mode (6-bit random password comparison)**

In SSP (Simple Safe Pair) mode, protocol will generate a digital code, and both sides need to confirm the code. This mode is used between devices which have display function and yes/no enabled function, such as mobile phone or module.

In a pairing scene, pairing between A and B depends on the matching mode supported by both sides. If A and B both support SSP, then the two sides pairing mode is SSP. If A and B only support PINCODE, or one is SSP, the other is PINCODE, then the pairing mode will be PINCODE.

2.5.1. Originated Pair SSP (Numeric Compare Mode)

AT+QBTPAIR=2	//Use the device ID
OK	
+QBTPAIR: "M66",8362618A0568,138949	//Indicate pairing status
AT+QBTPAIRCNF=1	//Accept pairing
OK	
+QBTPAIRCNF: 1,1,1,"M66",8362618A0568	//Pairing result

2.5.2. Originated Pair PINCODE

AT+QBTPAIR=2	
OK	
+QBTPAIR: "M66",8362618A0568	//Indicate pairing status
AT+QBTPAIRCNF=1,"1234"	//Accept pairing with password, double quote is required
OK	
+QBTPAIRCNF: 1,1,1,"M66",8362618A0568	//Pairing result

2.5.3. Terminated Pair SSP (Numeric Compare Mode)

+QBTIND: "pair","btd_left",656261222B10,138949	//Indicate pairing status
AT+QBTPAIRCNF=1	//Accept pairing
OK	

+QBTPAIRCNF: 1,1,1,"btd_left",656261222B10   //Pairing result

2.5.4. Terminated Pairing PINCODE

+QBTIND: "pair","btd_left",656261222B10   //Indicate pairing status
AT+QBTPAIRCNF=1,"1234"   //Accept pairing with password, double quote is
                                                  required.

OK

+QBTPAIRCNF: 1,1,1,"btd_left",656261222B10   //Pairing result

2.5.5. Unpair BT Device

You could use the **AT+QBTUNPAIR=<paired device id>** to unpair the paired BT device according to device ID. You can also use the **AT+QBTSTATE** to query the pairing status.

AT+QBTUNPAIR=1   //Unpair the paired BT device according to device ID
OK

2.6. Get the Supported Profile

After being paired, you could use the **AT+QBTGPROF=<paired device id>** to check the profiles provided by device.

AT+QBTGPROF=1   //1: paired device ID
+QBTGPROF: 0,"SPP"

+QBTGPROF: 3,"OBEX_OBJECT_PUSH_SERVICE"

+QBTGPROF: 5,"HF_PROFILE"

OK

2.7. Connect/Disconnect

You can use the **AT+QBTCONN=<paired device id>,<profile id>[,<mode>]** to connect the BT device with the paired ID, and **AT+QBTDISCONN=<connected id>** to disconnect the BT device with the connected ID. You can also use the **AT+QBTSTATE** to query the connection state.

2.7.1. SPP Connect/Disconnect

The following examples mainly introduce how to connection/disconnect SPP.

2.7.1.1. Initiate Connection (Buffer Access Mode)

AT+QBTCONN=1,0,1	//Initiate SPP connection
OK	
+QBTCONN: 1,1,"btd_left",656261222B10,"SPP"	//After connecting successfully, MCU can send and receive data via **AT+QSPPSEND** or **AT+QSPPREAD**

2.7.1.2. Initiate Connection (AT Command Mode)

AT+QBTCONN=1,0,0	//Initiate SPP connection
OK	
+QBTCONN: 1,1,"btd_left",656261222B10,"SPP"	//After connecting successfully, the data sent by the other side of the device will be treated as AT Command (Remote AT)

2.7.1.3. Initiate Connection (Transparent Access Mode)

AT+QBTCONN=1,0,2	//Initiate SPP connection
OK	
+QBTCONN: 1,1,"btd_left",656261222B10,"SPP"	
CONNECT	
//Enter data transparent access mode now. Read and write data directly	

2.7.1.4. Accept Connection (Buffer Access Mode)

+QBTIND: "conn","btd_left",656261222B10,"SPP" //Indicate connection state	
AT+QBTACPT=1,1	//Accept connection
OK	
+QBTACPT: 1, 2,"btd_left",656261222B10,"SPP" //After connecting successfully, MCU can send and	

receive data via **AT+QSPPSEND** or **AT+QSPPREAD**

2.7.1.5. Accept Connection (AT Command Mode)

+QBTIND: "conn","btd_left",656261222B10,"SPP" //Indicate connection state
AT+QBTACPT=1,0   //Accept connection
OK

+QBTACPT: 1, 2,"btd_left",656261222B10,"SPP" //After connecting successfully, the data sent by the other side of the device will be treated as AT Command (Remote AT)

2.7.1.6. Accept Connection (Transparent Access Mode)

+QBTIND: "conn","btd_left",656261222B10,"SPP" //Indicate connection state
AT+QBTACPT=1,2   //Accept connection
OK

+QBTACPT: 1, 2,"btd_left",656261222B10,"SPP"

CONNECT

//Enter data transparent access mode now. Read and write data directly

2.7.1.7. Initiate Disconnection (Buffer Access Mode and AT Command Mode)

AT+QBTDISCONN=2   //2: Connected ID
OK

+QBTDISCONN: 2,1,"btd_left",656261222B10,"SPP"   //Disconnecting result

2.7.1.8. Initiate Disconnection (Transparent Access Mode)

AT+QBTDISCONN=2   //2: Connected ID
OK

CLOSED   //Exit from the transparent access mode first

+QBTDISCONN: 2,1,"btd_left",656261222B10,"SPP"   //Disconnecting result

2.7.1.9. Disconnect Indication (Buffer Access Mode and AT Command Mode)

+QBTIND: "disc",1,1,"btd_left",656261222B10,"SPP"   //URC, Indicate disconnection state

2.7.1.10. Disconnect Indication (Transparent Access Mode)

CLOSED   //Exit from the transparent access mode

+QBTIND: "disc",1,1,"btd_left",656261222B10,"SPP"   //URC, indicate disconnection state

2.7.2. HFP Connect /Disconnect

2.7.2.1. Initiate Connection with HF Device

AT+QBTCONN=4,6   //4: Paired ID; 6: profile ID
OK

+QBTCONN: 1,0,"Motorola H270",00241C6743E9,"HFG_PROFILE"

2.7.2.2. Initiate Connection with AG Device

AT+QBTCONN=5,5   //5: Paired ID; 5: profile ID
OK

+QBTCONN: 1,0,"HUAWEI P6-C00",80717A48820B,"HF_PROFILE"

2.7.2.3. Accept Connection

When the module works as AG or HF, the other BT device can connect to the module directly without the need of any accept command.

+QBTCONN: 1,0,"HUAWEI P6-C00",80717A48820B,"HF_PROFILE"   //AG device connected to the module

+QBTCONN: 1,0,"Motorola H270",00241C6743E9,"HFG_PROFILE"   //HF device connected to the module

2.7.2.4. Initiate Disconnection with HF Device

AT+QBTDISCONN=0   //0: Connected ID
OK

+QBTDISCONN: 0,4,"Motorola H270", 00241C6743E9,"HFG_PROFILE" //Disconnecting result

2.7.2.5. Disconnect Indication with HF Device

+QBTIND: "disc",0,4,"Motorola H270",00241C6743E9,"HFG_PROFILE"

//URC, indicate disconnection state. 0: connected ID; 4: paired ID.

2.7.2.6. Disconnect Indication with AG device

+QBTIND: "disc",0,5,"HUAWEI P6-C00",80717A48820B,"HF_PROFILE"

//URC, indicate disconnection state. 0: connected ID; 5: paired ID.

2.7.3. Use BT Device Address to Connect SPP Directly

If you want to use the other BT device address and the password (PINCODE Pair) to connect SPP directly, you can use the command **AT+QBTCONND=<addr>[,<mode>[,<password>]]**.

//After BT is powered on, you can send the AT Command directly.

AT+QBTCONND=00025BB3A978,1,"0000"   //SPP connect buffer access mode.

+QBTPAIR: 00025BB3A978,"MPT-II",0   //First time to connect

+QBTPAIRCNF: 1,2,1,"MPT-II",00025BB3A978

+QBTCONN: 1,1,"MPT-II",00025BB3A978,"SPP"   //Connected successfully

OK
AT+QSPPSEND=1,94   //Send data to print
>
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqr stuvwxyz{|}~
OK
AT+QBTDISCONN   //In this condition, you can use this AT command
                                               without connected ID

OK

+QBTDISCONN: 1,1,"MPT-II",B3A9785B0002,"SPP"

//SPP connected as transparent access mode directly

AT+QBTCONND=F0728C0D50AE,2
OK

+QBTCONN: 1,1,"GT-e",F0728C0D50AE,"SPP"

CONNECT	//Enter transparent access mode
xxxxxxxxxxxxxxxxxxxxxxxx	//Receive data from remote BT device
+++	//Send "**+++**" to exit from transparent access mode
OK	//Exit from the transparent access mode, but do not disconnect
ATO	//**ATO** command enter the transparent mode
CONNECT	//Enter the transparent mode again

NOTES

1. There is no need to scan and pair.
2. This connection can only be used for SPP connection.

2.8. Query the Device Status Information

You could use the **AT+QBTSTATE** to get the working, pairing and connecting information. Pairing information is non-volatile. You can also use this command to query whether the device has been paired before. If so, you can directly use the paired information to connect to the device. Meanwhile, you can use this command to check the connection status.

AT+QBTSTATE
+QBTSTATE: 5

+QBTSTATE: 0,1,"M66",8362618A0568

+QBTSTATE: 1,1,"M66",8362618A0568,"SPP"

OK

2.9. SPP AT Command Mode

In AT command mode, the SPP data sent to module will be treated as AT Command (Remote AT)

```
+QBTIND: "conn","GT-eeeee",F0728C0D50AE,"SPP"
AT+QBTACPT=1,0                      //Connected as AT command mode
OK


+QBTCONN: 1,2,"GT-eeeee",F0728C0D50AE,"SPP"

//Remote device:

ATI
ATI
Quectel_Ltd
Quectel_M66
Revision: M66FAR01A05


OK
```

2.10. SPP Buffer Access Mode

In buffer mode, the SPP data will be buffered. URC will be reported to notify the data arriving. Command is also provided to send and read the SPP data.

2.10.1. Read Data

In buffer access mode, after receiving data, the module will report URC as **+QBTIND: "recv",<connect ed id>** to notify the host. Then host can retrieve data by **AT+QSPPREAD=<connected id>,<readlength>**. Notes: if the buffer is not empty, and the module receives data again, it will not report a new URC until all the received data has been retrieved by **AT+QSPPREAD=<connected id>,<readlength>** from buffer.

```
+QBTIND: "recv",1                   //<connected id>=1 receives data notification
AT+QSPPREAD=1,1500                   //Read data from <connected id>=1, max length is 1500 bytes
+QSPPREAD: 1500                      //Read 1500 bytes
11………xx


OK
AT+QSPPREAD=1,1500                   //Continue to read
+QSPPREAD: 500                       //Only 500
11………xx
```

OK
AT+QSPPREAD=1,1500   //Continue to read
+QSPPREAD: 0   //The buffer is empty, all data has been read out

OK

2.10.2. Send Data

AT+QSPPSEND=1,10   //Send 10 bytes to **<connected id>=1**
>1234567890
OK
AT+QSPPSEND=1,0   //Query the sent and unacknowledged data
+QSPPSEND: 10,10,0

OK

……
AT+QSPPSEND=1,800   //Send 800 bytes to **<connected id>=1**
>1234567890xxxxx…
+CME ERROR: 8020   //The buffer is full and the 800 bytes not send. You can send
                                       the data when the buffer has enough space

//When the buffer has enough space, you can send the data again

AT+QSPPSEND=1,200
>1234567890xxxxx…
OK

2.11. SPP Transparent Access Mode

In transparent access mode, the corresponding UART port enters into the exclusive mode. The data received from UART port will be sent to BT directly, and the received data from BT will be outputted to UART port directly as well. "**+++**" is used to exit from transparent access mode. When "**+++**" returns OK, the module will be switched to buffer access mode. **ATO** can be used to return back to transparent access mode. Note: In transparent mode, host cannot execute AT command. If the BT connection is closed, the module will report CLOSED and exit from the transparent access mode.

You can use "**+++**" or DTR (**AT&D1** should be set) to exit from the transparent access mode. To prevent the "**+++**" from being misinterpreted as data, you should comply with the following sequence:

1)   Do not input any character within T1 time (1 second) before inputting "**+++**".
2)   Input "**+++**", time gap between each "**+**" cannot be longer than 20ms, and no other characters can

    be inputted during this time.
3) Do not input any character within T1 time (1 second) after "**+++**" has been inputted.
4) Exit from transparent access mode, return OK.

ATO will change the access mode of connection which exits from transparent access mode lately. If it enters into transparent access mode successfully, CONNECT will be returned. If there is no connection enters into transparent access mode before, **ATO** will return NO CARRIET.

AT+QBTCONN=1,0,2	//Connect to the paired device 1, with transparent access mode
OK	
+QBTCONN: 1,1,"GT-e",F0728C0D50AE,"SPP"	
CONNECT	//Enter into transparent access mode
Aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa	//Received data from remote BT
+++	//Send "**+++**" to exit from transparent access mode
OK	//Exit from the transparent access mode, but do not disconnect
ATO	//**ATO** command enter the transparent mode
CONNECT	//Enter the transparent mode again
Bbbbbbbbbbbbbbbbbbb	//Received data
CLOSED	//Disconnect passively
+QBTIND: "disc",1,1,"GT-e",F0728C0D50AE,"SPP"	
/********************* accept connect ************************************/	
+QBTIND: "conn","GT-e",F0728C0D50AE,"SPP"	
AT+QBTACPT=1,2	//Accept connection, with transparent access mode
OK	
+QBTACPT: 1,2,"GT-e",F0728C0D50AE,"SPP"	//URC connect successfully
CONNECT	//Transparent access mode
+++	//Input "**+++**"
OK	//Exit from transparent access mode
ATO	//**ATO** command
CONNECT	//Enter transparent access mode again
+++	//Input "**+++**"
OK	//Exit from transparent access mode
AT+QBTDISCONN=2	//Disconnection initiatively
OK	

CLOSED

+QBTDISCONN: 2,1,"GT-e",F0728C0D50AE,"SPP"

2.12. HFP Connection

2.12.1. Module Acts as AG Device

Quectel module supports HFP profile, the following connection sequence shows how the module acts as an AG device.

//Connect to the Bluetooth headset first.

AT+QBTSCAN
OK

+QBTSCAN: 1,"Motorola H270",00241C6743E9   //Found the Bluetooth headset
AT+QBTSCANC   //Stop scanning
+QBTSCAN: 0

OK
AT+QBTPAIR=1   //Pairing
OK

+QBTPAIRCNF: 1,1,0,"Motorola H270",00241C6743E9   //Paired successfully
AT+QBTCONN=1,6   //Connect HFG profile
OK

+QBTCONN: 1,1,"A16S",00241C6743E9,"HFG_PROFILE"   //Connected successfully

//If the Bluetooth headset has been paired with the module. Turn on the Bluetooth headset, it will connect to the module automatically.

+QBTCONN: 1,1,"A16S",00241C6743E9,"HFG_PROFILE"   //Connect to the module automatically

//When the Bluetooth headset is connected to the module, the speech voice will output by the Bluetooth headset. And you can set the speech voice output path by **AT+QBTHFGAUDIO**.

AT+QBTHFGAUDIO=1   //Speech voice output from module side
OK

AT+QBTHFGAUDIO=0   //Speech voice output from HF device side
OK

2.12.2. Module Acts as HF Device

The following connection sequence shows how the module acts as a HF device.

```
AT+QBTSCAN
OK

+QBTSCAN: 1,"GT-e",F0728C0D50AE          //Found the phone device
AT+QBTSCANC                              //Stop scanning
+QBTSCAN: 0

OK
AT+QBTPAIR=1                             //Pairing
OK

+QBTPAIR: "GT-e",F0728C0D50AE,348531
AT+QBTPAIRCNF=1
OK

+QBTPAIRCNF: 1,1,1,"GT-e",F0728C0D50AE
AT+QBTCONN=1,5                           //Connect HF profile, module acts as a hand free device
OK

+QBTCONN: 1,1,"GT-e",F0728C0D50AE,"HF_PROFILE"    //Connected successfully

//After connecting to AG device, you can use the following command to control the AG device (phone)

AT+QBTATD=10086                          //Call the 10086
OK

+QBTIND: "ring",1,1,10086

+QBTIND: "ring",4,1,10086

AT+QBTATH                                //Hung up voice call
OK

+QBTIND: "ring",0,1,10086

//When a call is coming, the following URC will be output

+QBTIND: "ring",2,1,055165869386         //A call is coming
AT+QBTATA                                //Answer the call
OK
```

3 BT Commands Description

3.1. AT Command Syntax

Test Command	AT+<x>=?	This command returns the list of parameters and value ranges set by the corresponding Write Command or internal processes.
Read Command	AT+<x>?	This command returns the currently set value of the parameter or parameters.
Write Command	AT+<x>=<…>	This command sets the user-definable parameter values.
Execution Command	AT+<x>	This command reads non-variable parameters affected by internal processes in the GSM engine.

3.2. Description of AT Command

3.2.1. AT+QBTPWR   Power On/Off BT

AT+QBTPWR Power On/Off BT	
Test Command **AT+QBTPWR=?**	Response **+QBTPWR: (list of supported <on_off>s)** **OK**
Read Command **AT+QBTPWR?**	Response **+QBTPWR: <power status>** **OK** **+CME ERROR: 100**
Write Command **AT+QBTPWR=<on_off>**	Response **OK** **+CME ERROR: 100**

Parameter

<on_off>	0	Off
	1	On

<power status>	0	Power off
	1	Power on

Example

AT+QBTPWR=1	//Power on BT
OK	

3.2.2. AT+QBTNAME   Read/Write BT Device Local Name

AT+QBTNAME Read/Write BT Device Local Name	
Test Command AT+QBTNAME=?	Response OK ERROR
Read Command AT+QBTNAME?	Response +QBTNAME: <device name> OK +CME ERROR: <errorcode>
Write Command AT+QBTNAME=<device name>	Response OK +CME ERROR: <errorcode>

Parameter

<device name>   String type, device name

> **NOTE**
>
> Max length of the device name is 54 bytes.

Example

AT+QBTNAME?	
+QBTNAME: "MYBTDEVICE"	
OK	

3.2.3. AT+QBTADDR Read BT Device Local Address

AT+QBTADDR Read BT Device Local Address	
Test Command AT+QBTADDR=?	Response **OK** **ERROR**
Read Command AT+QBTADDR?	Response **+QBTADDR: <addr>** **OK** **+CME ERROR: <errorcode>**

Parameter

<addr>	Device address (e.g.: A662616202C3, means: LAP: 0xA66261, UAP: 0x62, NAP: 0x02c3)

Example

AT+QBTADDR?
+QBTADDR: A662616202C3

OK

3.2.4. AT+QBTVISB Set BT Visibility

AT+QBTVISB Set BT Visibility	
Test Command AT+QBTVISB=?	Response **+QBTVISB:** (list of supported **<n>**s),(list of supported **<time>**s) **OK**
Read Command AT+QBTVISB?	Response **+QBTVISB: <n>** **OK** **+CME ERROR: <errorcode>**
Write Command AT+QBTVISB=<n>[,<time>]	Response **OK** **+CME ERROR: <errorcode>**

	If <n>=2, response:
	OK
	+QBTVISB: 0

Parameter

<n>	0	Invisible
	1	Visible forever
	2	Visibility temporary on, can be discovered by other BT device for some time
<time>	Integer type, visible time, unit: second	
	1-255	Visible timer

Example

AT+QBTVISB=0	//Invisible
OK	
AT+QBTVISB=1	
OK	
AT+QBTVISB=2,60	
OK	
+QBTVISB: 0	//Become invisible after time out

3.2.5. AT+QBTSCAN   Scan BT Devices

AT+QBTSCAN Scan BT Devices	
Test Command **AT+QBTSCAN=?**	Response **+QBTSCAN: (list of supported <timeout>s),(list of supported <deviceNumber>s),(list of supported <Cod>s)** **OK**
Read Command **AT+QBTSCAN?**	Response **+QBTSCAN: <timeout>,<deviceNumber>,<Cod>** **OK** **+CME ERROR: <errorcode>**
Write Command **AT+QBTSCAN=<timeout>[,<deviceNumber>[,<Cod>]]**	Response **OK** **+QBTSCAN: <searched device id>,<name>,<addr>** **…** **+QBTSCAN: 0[,<errorcode>]**

	+CME ERROR: <errorcode>
Execute Command **AT+QBTSCAN**	Response **OK** **+QBTSCAN: <searched device id>,<name>,<addr>** **…** **+QBTSCAN: 0[,<errorcode>]** **+CME ERROR: <errorcode>**

Parameter

<timeout>	1-255	The max number of seconds for searching. Default value: 60 (s)
<searched device id>	0	Search completed
	1-20	Assign an ID for searched device
<deviceNumber>	1-20	The max number of device allowed to search
<Cod>	0-255	The Class of Device/Service (CoD). Refer to BT specification for details
<name>		String type, device name
<addr>		Device address
<errorcode>		Refer to Chapter 5

Example

AT+QBTSCAN=60,10,0
OK

+QBTSCAN: 1,"BILLHE-PC",065C3A2F40F0

+QBTSCAN: 2,"bt_dev1",016261E65467

+QBTSCAN: 3,"2233",326261926A88
AT+QBTSCANC
+QBTSCAN: 0

OK
AT+QBTSCAN?
+QBTSCAN: 60,10,0

OK

3.2.6. AT+QBTSCANC   Stop Scanning

AT+QBTSCANC Stop Scanning	
Execute Command **AT+QBTSCANC**	Response **OK** **+CME ERROR: <errorcode>**

3.2.7. AT+QBTPAIR   Pair BT Device

AT+QBTPAIR Pair BT Device	
Test Command **AT+QBTPAIR=?**	Response **+QBTPAIR:** (list of supported **<searched device id>**s) **OK**
Write Command **AT+QBTPAIR=<searched device id>**	Response **OK** **+QBTPAIR: <name>, <addr>[,<numericcompare>]** **+CME ERROR: <errorcode>**

Parameter

<searched device id>	Searched device ID, range: 1-20
<name>	String type, device name
<addr>	Device address
<numericcompare>	The 6 bytes random password in Numeric Compare Mode

3.2.8. +QBTIND: "pair"   Pairing Indication

+QBTIND: "pair" Pairing Indication	
+QBTIND: "pair",<name>,<addr> **[,<numericcompare>]**	

Parameter

<name>	String type, device name
<addr>	Device address
<numericcompare>	The 6 bytes random password in Numeric Compare Mode

When receiving a pairing indication:

- Use **AT+QBTPAIRCNF=1** command without passkey if there is a **<numericcompare>** parameter
- Use **AT+QBTPAIRCNF=1,<passkey>** command with passkey if there is no **<numericcompare>** parameter.

3.2.9. AT+QBTPAIRCNF   Pair Confirmation

AT+QBTPAIRCNF Pair Confirmation	
Test Command **AT+QBTPAIRCNF=?**	Response **+QBTPAIRCNF:** (list of supported **<op>**s), (list of supported **<passkey>**s) **OK**
Write Command **AT+QBTPAIRCNF=<op>[,<passkey>]**	Response **OK** **+QBTPAIRCNF:** **<result>,<paired device id>,<is first>,<name> ,<addr>** **+CME ERROR: <errorcode>**

Parameter

<op>	0	Reject pairing
	1	Accept pairing
<passkey>	String type, passkey, 4 bytes	
<result>	0	Failed
	1	Success
<paired device id>	1-10	The allocated paired device ID
<is first>	Whether it is the first time to pair	
	0	No
	1	Yes
<name>	String type, device name	
<addr>	Device address	
<passkey>	String type, passkey, 4 bytes	

3.2.10. AT+QBTUNPAIR Delete Paired BT Device

AT+QBTUNPAIR Delete Paired BT Device	
Test Command **AT+QBTUNPAIR=?**	Response **+QBTUNPAIR:** (list of supported**<paired device id>**s) **OK**
Write Command **AT+QBTUNPAIR=<paired device id>**	Response **OK** **+CME ERROR: <errorcode>**

Parameter

<paired device id>	1-10	Paired device ID

3.2.11. AT+QBTSTATE Query the Device Status

AT+QBTSTATE Query the Device Status	
Test Command **AT+QBTSTATE=?**	Response **+QBTSTATE:** (list of supported **<state>**s) **OK**
Execute Command **AT+QBTSTATE**	Response **+QBTSTATE: <state>** **+QBTSTATE: 0, <paired device id>,<name>,<addr>** … **+QBTSTATE: 1, <connected id>,<name>,<addr>,<profile name>** … **OK** **+CME ERROR:<errorcode>**

Parameter

<state>	0	Initial on
	1	Deactivating
	2	Activating
	5	Idle
	6	Scanning

	7	Scan result indication
	8	Scan cancelling
	9	Initiate pairing
	12	Connecting
	13	Un-pairing
	14	Deleting paired device
	15	Deleting all
	16	Disconnecting
	19	Numeric confirm
	20	Numeric confirm response
	25	Connect accept confirm indication
	26	Service refreshing
	29	Device name setting
	30	Authentication setting
	31	Release all connection
	36	Activating profiles
<paired device id>	1-10	Paired device ID
<connected id>	0-2	Connected device ID
<name>	String type, device name	
<addr>	Device address	
<profile name>	String type, profile name	
	"SPP"	
	"OBEX_PBA_PROFILE_CLIENT"	
	"OBEX_PBA_PROFILE"	
	"OBEX_OBJECT_PUSH_SERVICE"	
	"OBEX_OBJECT_PUSH_CLIENT"	
	"HF_PROFILE"	
	"HFG_PROFILE"	

3.2.12. AT+QBTGPROF   Get the Supported Profile

AT+QBTGPROF Get the Supported Profile	
Test Command **AT+QBTGPROF=?**	Response **OK** **ERROR**
Write Command **AT+QBTGPROF=<paired device id>**	Response **+QBTGPROF: <profile id>,<profile name>** **…** **OK** **+CME ERROR: <errorcode>**

Parameter

<paired device id>	1-10	Paired device ID
<profile id>	0	"SPP",0x1101
	1	"OBEX_PBA_PROFILE_CLIENT",0x112E
	2	"OBEX_PBA_PROFILE",0x112F
	3	"OBEX_OBJECT_PUSH_SERVICE",0x1105
	4	"OBEX_OBJECT_PUSH_CLIENT",0xfffd
	5	"HF_PROFILE",0x111E
	6	"HFG_PROFILE",0x111F
<profile name>	String type, profile name	
	"SPP"	
	"OBEX_PBA_PROFILE_CLIENT"	
	"OBEX_PBA_PROFILE"	
	"OBEX_OBJECT_PUSH_SERVICE"	
	"OBEX_OBJECT_PUSH_CLIENT"	
	"HF_PROFILE"	
	"HFG_PROFILE"	

NOTE

It returns the profiles supported by both the local device and the other side device.

3.2.13. AT+QBTCONN   Connect BT Device

AT+QBTCONN Connect BT Device	
Test Command **AT+QBTCONN=?**	Response **+QBTCONN:** (list of supported **<paired device id>**s)**,** (list of supported **<profile id>**s)**,** (list of supported **<mode>**s) **OK**
Write Command **AT+QBTCONN=<paired device id>,<profile id>[,<mode>]**	Response **OK** **+QBTCONN:** **<result>,<connected id>,<name>,<addr>,<profile name>** **+CME ERROR: <errorcode>**

Parameter

<result>	0	Failed
	1	Successful

<paired device id>	1-10	Paired device ID
<profile id>	0	"SPP",0x1101
	1	"OBEX_PBA_PROFILE_CLIENT",0x112E
	2	"OBEX_PBA_PROFILE",0x112F
	3	"OBEX_OBJECT_PUSH_SERVICE",0x1105
	4	"OBEX_OBJECT_PUSH_CLIENT",0xfffd
	5	"HF_PROFILE",0x111E
	6	"HFG_PROFILE",0x111F
<mode>		Used for SPP connection
	0	AT command mode
	<u>1</u>	Buffer Access Mode
	2	Transparent Access Mode
<connected id>	0-2	Connected device ID
<name>		String type, device name
<addr>		Device address
<profile name>		String type, profile name
		"SPP"
		"OBEX_PBA_PROFILE_CLIENT"
		"OBEX_PBA_PROFILE"
		"OBEX_OBJECT_PUSH_SERVICE"
		"OBEX_OBJECT_PUSH_CLIENT"
		"HF_PROFILE"
		"HFG_PROFILE"

NOTE

Quectel does not support to change the mode when SPP is successfully connected. You have to disconnect SPP first.

3.2.14. AT+QBTCONND　Use BT Device Address to Connect SPP Directly

AT+QBTCONND　Use BT Device Address to Connect SPP Directly	
Test Command **AT+QBTCONND=?**	Response **+QBTCONND:** (list of supported **<addr>**s),(list of supported **<mode>**s)**,** (list of supported **<password>**s), **OK**
Write Command **AT+QBTCONND=<addr>[,<mode>[,<password>]]**	Response **OK** **+QBTCONN:** **<result>,<connected id>,<name>,<addr>,<profile name>** **+CME ERROR: <errorcode>**

Parameter

<result>	0	Failed
	1	Successful
<mode>	Used for SPP connection	
	0	AT command mode
	1	Buffer Access Mode
	2	Transparent Access Mode
<connected id>	0-2	Connected device ID
<name>	String type, device name	
<addr>	Device address, does not need double quotes.	
	Sequence: Nap,UAP,LAP	
<password>	String type, the password in PINCODE pair, double quotes are required	
<profile name>	String type, profile name	
	"SPP"	
	"OBEX_PBA_PROFILE_CLIENT"	
	"OBEX_PBA_PROFILE"	
	"OBEX_OBJECT_PUSH_SERVICE"	
	"OBEX_OBJECT_PUSH_CLIENT"	
	"HF_PROFILE"	
	"HFG_PROFILE"	

NOTE

Direct SPP connection will only use the Bluetooth address with the sequence NAP-UAP-LAP. "Originated PINCODE Pair (with password)" or "Originated Pair SSP (Numeric Compare Mode)" is internally used in this connection, so you do not need to scan or concern the pairing process.

Example

```
AT+QBTADDR?
+QBTADDR: 02C362A66261

OK
AT+QBTCONND=00025BB3A978,1,"0000"
+QBTPAIR:00025BB3A978, "MPT-II",0              //First time to connect

+QBTPAIRCNF:1,2,1,"MPT-II",00025BB3A978

+QBTCONN:1,1, "MPT-II",00025BB3A978,"SPP"      //Connected successfully

OK
AT+QSPPSEND=1,94                               //Send data to print
>
```

!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqr
stuvwxyz{|}~

OK
AT+QBTDISCONN   //In this condition, you can use this AT Command without connected ID.
OK

+QBTDISCONN: 1,1,"MPT-II",B3A9785B0002,"SPP"

3.2.15. +QBTIND: "conn"   Connecting Indication

+QBTIND: "conn" Connecting Indication	
+QBTIND: "conn",<name>, <addr>,<profile name>	

Parameter

<name>	String type, device name
<addr>	Device address
<profile name>	String type, profile name
	"SPP"
	"OBEX_PBA_PROFILE_CLIENT"
	"OBEX_PBA_PROFILE"
	"OBEX_OBJECT_PUSH_SERVICE"
	"OBEX_OBJECT_PUSH_CLIENT"
	"HF_PROFILE"
	"HFG_PROFILE"

3.2.16. AT+QBTACPT   Accept Connection

AT+QBTACPT Accept Connection	
Test Command **AT+QBTACPT=?**	Response **+QBTACPT:** (list of supported**<op>**s) **OK**
Write Command **AT+QBTACPT=<op>[,<mode>]**	Response **OK** **+QBTACPT:** **<result>,<connected id>,<name>,<addr>,<profile name>** **+CME ERROR: <errorcode>**

Parameter

<op>	0	Reject connection
	1	Accept connection
<result>	0	Fail
	1	Success
<mode>		Used for SPP connection
	0	AT command mode
	<u>1</u>	Buffer Access Mode
	2	Transparent Access Mode
<connected id>	0-2	Connected device ID
<name>		String type, device name
<addr>		Device address
<profile name>		String type, profile name
		"SPP"
		"OBEX_PBA_PROFILE_CLIENT"
		"OBEX_PBA_PROFILE"
		"OBEX_OBJECT_PUSH_SERVICE"
		"OBEX_OBJECT_PUSH_CLIENT"
		"HF_PROFILE"
		"HFG_PROFILE"

3.2.17. AT+QBTDISCONN   Disconnect

AT+QBTDISCONN Disconnect	
Test Command **AT+QBTDISCONN=?**	Response **+QBTDISCONN:** (list of supported**<connected id>**s) **OK**
Write Command **AT+QBTDISCONN=<connected id>**	Response **OK** **+QBTDISCONN: <connected id>,<paired device id>,** **<name>,<addr>,<profile name>** **+CME ERROR: <errorcode>**

Parameter

<connected id>		Connected device ID.
	0	Connection 0
	1	Connection 1
	2	Connection 2
<paired device id>	The paired device ID	

<name>	String type, device name
<addr>	Device address
<profile name>	String type, profile name
	"SPP"
	"OBEX_PBA_PROFILE_CLIENT"
	"OBEX_PBA_PROFILE"
	"OBEX_OBJECT_PUSH_SERVICE"
	"OBEX_OBJECT_PUSH_CLIENT"
	"HF_PROFILE"
	"HFG_PROFILE"

3.2.18. +QBTIND: "disc" Disconnect Indication

+QBTIND: "disc" Disconnect Indication	
+QBTIND: "disc",<connected id>,<paired device id>,<name>, <addr>,<profile name>	

Parameter

<connected id>	The device connected id.
<paired device id>	The paired device ID
<name>	String type, device name
<addr>	Device address
<profile name>	String type, profile name
	"SPP"
	"OBEX_PBA_PROFILE_CLIENT"
	"OBEX_PBA_PROFILE"
	"OBEX_OBJECT_PUSH_SERVICE"
	"OBEX_OBJECT_PUSH_CLIENT"
	"HF_PROFILE"
	"HFG_PROFILE"

3.2.19. AT+QSPPSEND Send Data

AT+QSPPSEND Send Data	
Test Command **AT+QSPPSEND=?**	Response: **+QSPPSEND:** (list of supported **<connected id>**s)[,(list of supported **<sendlength>**s)**]**

	OK
Write Command Send data with changeable length **AT+QSPPSEND=<connected id>** Response ">", then type data to send, tap CTRL+Z to send, tap ESC to cancel the operation	Response: **>** **<input data>** **CTRL+Z** If connection has been established and sending is successful, response: **OK** If connection has been established but sending buffer does not have enough space, response: **+CME ERROR: 8020** If connection has not been established, abnormally closed, or parameter is incorrect, response: **+CME ERROR: <errorcode>**
Write Command Send data with fixed length **AT+QSPPSEND=<connected id>,<sendlength>** Response ">", type data until the data length is equal to **<length>**	Response: **>** **<input data with specified length>** If connection has been established and sending is successful, response: **OK** If connection has been established but sending buffer does not have enough space , response: **+CME ERROR: 8020** If connection has not been established, abnormally closed, or parameter is incorrect, response: **+CME ERROR: <errorcode>**
Write Command When <length> is 0, query the sent data **AT+QSPPSEND=<connected id>,0**	Response If the specified <channel> connection has existed, return: **+QSPPSEND: <totalsendlength>,<sentbytes>,<leftbytes>** **OK** Else, return: **+CME ERROR: <errorcode>**

Parameter

\<connected id\>	Connected ID
\<sendlength\>	Integer type, the length of data to be sent, which cannot exceed 1460
\<totalsendlength\>	Integer type, the total length of sent data
\<sentbytes\>	Integer type, the length of data already sent
\<leftbytes\>	Integer type, the length of the left data

3.2.20. AT+QSPPREAD   Read Data

AT+QSPPREAD Read Data	
Test Command **AT+QSPPREAD=?**	Response: **+QSPPREAD:** (list of supported**\<connected id\>**s),(list of supported **\<readlength\>**) **OK**
Write Command **AT+QSPPREAD=\<connected id\>,\<readlength\>**	Response If the specified channel has received data, response: **+QSPPREAD: \<readactuallength\>\<CR\>\<LF\>\<data\>** **OK** If no data, return: **+QSPPREAD: 0** **OK** If connection does not exist, return: **+CME ERROR: \<errorcode\>**
Write Command When **\<readlength\>** is 0, query retrieved data **AT+QSPPREAD=\<conencted id\>,0**	Response: If the specified connection has existed, return: **+QSPPREAD:** **\<totalreceivelength\>,\<havereadlength\>,\<unreadlength\>** **OK** Else, response: **+CME ERROR: \<errorcode\>**

Parameter

\<connected id\>	Connected ID
\<readlength\>	The maximum length of data to be retrieved, the range is 0-1500
\<totalreceivelength\>	The total length of received data

| <havereadlength> | The read length of received data |
| <unreadlength> | The unread length of received data |

3.2.21. +QBTIND: "recv"   Incoming Data Indication

In buffer access mode, after receiving data, the module will report URC as **+QBTIND: "recv",<connected id>** to notify the host. Then host can retrieve data by **AT+QSPPREAD=<connected id>,<readlength>**. Notes: if the buffer is not empty, and the module receives data again, it will not report a new URC until all the received data has been retrieved by **AT+QSPPREAD=<connected id>,<readlength>** from buffer.

+QBTIND: "recv" Incoming Data Indication	
+QBTIND: "recv",<connected id>	The URC of incoming data in buffer access mode. Can receive data by **AT+QSPPREAD**.

Parameter

| <connected id> | Connected ID |

3.2.22. +QBTIND: "ring"   Call Indication

+QBTIND: " ring " Call Indication	
+QBTIND: ring",<call_state >,<call_id> ,<number>	

Parameter

< call_state >	The call state
	0 Idle state
	1 Outgoing state
	2 Incoming state
	4 Active state
	8 Hold state
<call_id>	The call ID
<number>	The call number

3.2.23. AT+QBTATA   Answer Incoming Call

AT+QBTATA Answer Incoming Call	
Execute Command	Response:
AT+QBTATA	**OK**
	+CME ERROR: <errorcode>

3.2.24. AT+QBTATH   Hung up Voice Call

AT+QBTATH Hung up Voice Call	
Execute Command	Response:
AT+QBTATH	**OK**
	+CME ERROR: <errorcode>

3.2.25. AT+QBTATD   Dial Voice Call

AT+QBTATD Dial Voice Call	
Write Command	Response:
AT+QBTATD=<number>	**OK**
	+CME ERROR: <errorcode>

Parameter

<number>	Phone number
	Dialing digits: *,#,0-9

3.2.26. AT+QBTATDL   Redial Last Phone Number

AT+QBTATDL Redial Last Phone Number	
Execute Command	Response:
AT+QBTATDL	**OK**
	+CME ERROR: <errorcode>

3.2.27. AT+QBTHFGAUDIO   Select Audio Output Path While Connecting with HFG Device

AT+QBTHFGAUDIO Select Audio Output Path While Connecting with HFG Device	
Test Command **AT+QBTHFGAUDIO=?**	Response: **+QBTHFGAUDIO:** (list of supported**<path>**s) **OK**
Read Command **AT+QBTHFGAUDIO?**	Response: **+QBTHFGAUDIO: <path>** **OK**
Write Command **AT+QBTHFGAUDIO=<path>**	Response **OK** **+CME ERROR: <errorcode>**

Parameter

<path>	Audio output path while the module is connecting with HF device
	0 Audio output from HF device side
	1 Audio output from module side

4 Appendix A Reference

Table 1: Related Documents

SN	Document Name	Remark
[1]	GSM 07.07	Digital cellular telecommunications (Phase 2+); AT command set for GSM Mobile Equipment (ME)
[2]	GSM 07.10	Support GSM 07.10 multiplexing protocol
[3]	Specification of the Bluetooth System - Specification Volume 1	Bluetooth Specification from Http://www.bluetooth.com

Table 2: Terms and Abbreviations

Abbreviation	Description
SPP	Serial Port Profile
URC	Unsolicited Result Code
BT	Bluetooth
SSP	Simple Safe Pair
HFP	Hands-Free Profile
HF	Hands-Free Unit
AG	Audio Gateway

5 Appendix B Summary of <errorcode>

Table 3: Different Coding Schemes of +CME ERROR: <errorcode>

Code of <errorcode>	Meaning
8002	Device ID invalid
8003	Busy in scanning
8004	Param invalid
8005	Busy in pairing
8006	Already paired
8007	Scan error
8008	Power request failed
8009	Set request failed
8010	Read addr request failed
8011	Set visible request failed
8012	Scan request failed
8013	Scan cancel request failed
8014	Pair request failed
8015	Pair confirm request failed
8016	Unpair request failed
8017	Get profile request failed
8018	Connect request failed
8019	Connect accept request failed
8101	Disconnect request failed

8102	Check state request failed
8107	HFATD require failed
8108	HFATA require failed
8109	HFATH require failed
8110	BTCLCC require failed
8111	HFRVC require failed
8020	BT buffer full
8021	BT not connect
8022	BT memory wrong
8023	The data exceeds the specified length
8024	BT disconnected
8201	BT already connected
4	Operation not supported