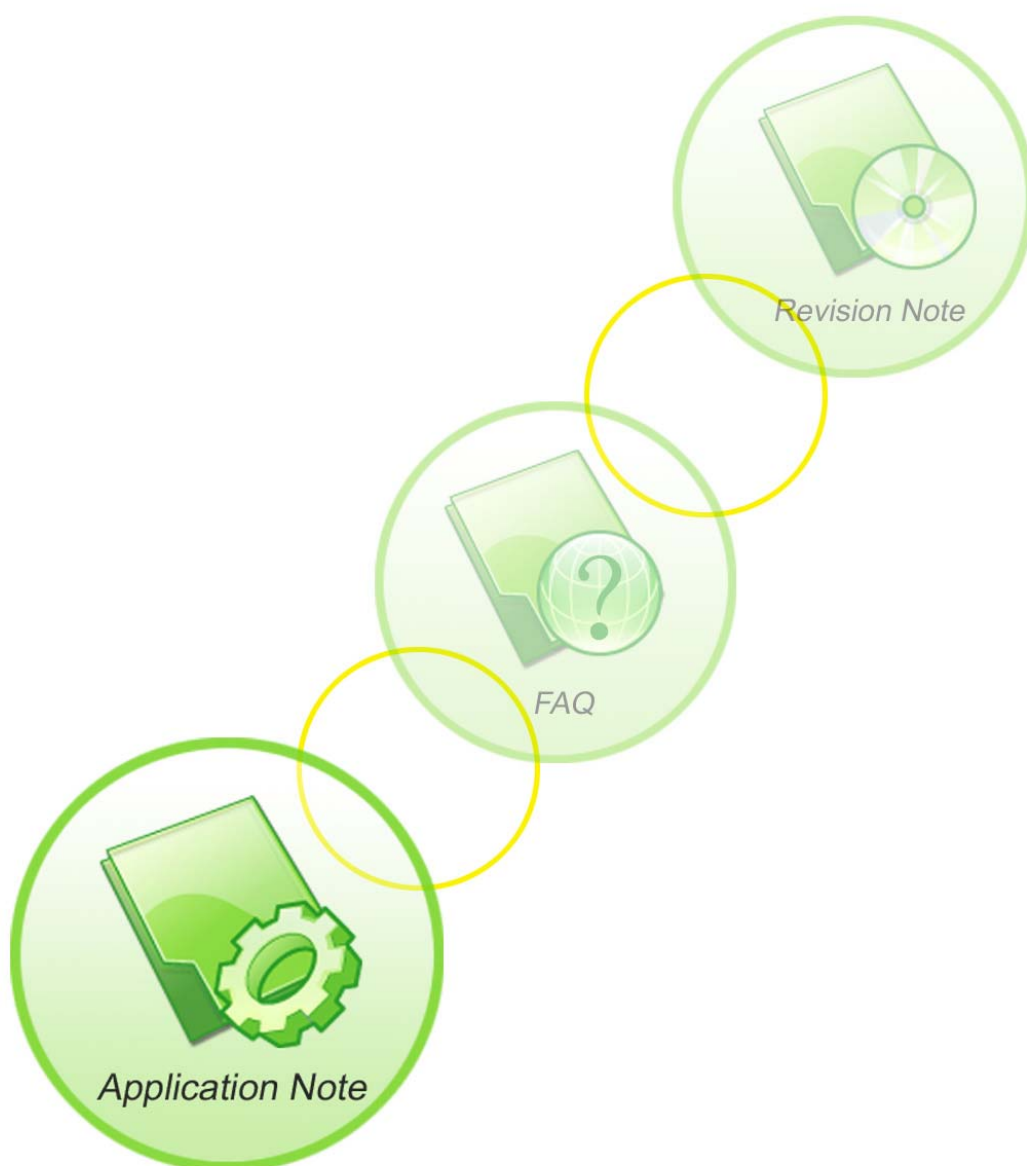




A company of SIM Tech

SIM800 Series _Bluetooth_ Application Note_V1.00



| | |
|----------------------------|--|
| Document Title | SIM800 Series_Bluetooth_Application Note |
| Version | 1.00 |
| Date | 2013-11-07 |
| Status | Release |
| Document Control ID | SIM800 Series_Bluetooth_Application Note_V1.00.doc |

General Notes

Simcom offers this information as a service to its customers, to support application and engineering efforts that use the products designed by Simcom. The information provided is based upon requirements specifically provided to Simcom by the customers. Simcom has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession. Furthermore, system validation of this product designed by SIMCOM within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

Copyright

This document contains proprietary technical information which is the property of SIMCOM Limited., copying of this document and giving it to others and the using or communication of the contents thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights reserved in the event of grant of a patent or the registration of a utility model or design. All specification supplied herein are subject to change without notice at any time.

Copyright © SIMCom Wireless Solutions Ltd. 2013

Content

| | |
|---|-----------|
| Version Histor..... | 5 |
| 1 Bluetooth Function..... | 6 |
| 1.1. Bluetooth Introduction | 6 |
| 1.2. Bluetooth Profile | 6 |
| 1.3. Bluetooth Device Address..... | 6 |
| 1.4. AT Interface for Bluetooth Function | 6 |
| 2. AT Command | 7 |
| 2.1. AT+BTHOST Inquiry and set host device name | 7 |
| 2.2. AT+BTSTATUS Inquiry current BT device status..... | 8 |
| 2.3. AT+BTPOWER Power on/off BT radio | 9 |
| 2.4. AT+BTPAIR Pair BT device..... | 9 |
| 2.5. AT+BTUNPAIR Unpair BT device | 10 |
| 2.6. AT+BTSCAN Scan surrounding BT device | 11 |
| 2.7. AT+BTCONNECT Connect paired BT device..... | 11 |
| 2.8. AT+BTDISCONN Disconnect BT connection | 12 |
| 2.9. AT+BTGETPROF Get profile provided by paired device..... | 12 |
| 2.10. AT+BTACPT Accept connecting request..... | 13 |
| 2.11. AT+BTOPPACPT Accept OPP service..... | 14 |
| 2.12. AT+BTOPPPUSH Push OPP object to paired device..... | 14 |
| 2.13. AT+BTSPGET Get data from BT serial port as client based on SPP service..... | 15 |
| 2.14. AT+BTSPSEND Send data to BT serial port as client based on SPP service..... | 16 |
| 2.15. AT+BTATA Answer incoming call..... | 17 |
| 2.16. AT+BTATDL Redial last number | 17 |
| 2.17. AT+BTATH Hung up voice call..... | 17 |
| 2.18. AT+BTVGS Configure voice volume..... | 17 |
| 2.19. AT+BTVGM Configure MIC gain level..... | 18 |
| 2.20. AT+BTATD Dial voice call..... | 18 |
| 2.21. AT+BTRSSI Get RSSI of connected BT device..... | 19 |
| 3. CME Error Code | 20 |
| 4 Examples..... | 21 |
| 4.1 Accept request from other BT device..... | 21 |
| 4.2 Send pairing request to other BT device | 21 |
| 4.3 Get the profile provided by paired device | 22 |
| 4.4 Connect service..... | 23 |
| 4.5 Accept file from paired device | 23 |
| 4.6 Send file to other paired BT device..... | 23 |
| 4.7 AT Channel mode, Module as client | 23 |
| 4.8 AT Channel mode, module as server..... | 24 |
| 4.9 Exchanged mode setup..... | 24 |

| | | |
|----------------------|---------------------------------|-----------|
| 4.10 | Data Exchanged | 26 |
| 4.11 | Module as SPP server..... | 26 |
| Appendix..... | | 27 |
| A. | Reference | 27 |
| B. | Profile..... | 27 |
| C. | Glossary and Abbreviation | 28 |

Version History

| Date | Version | Description | Owner |
|------------|---------|-------------|------------|
| 2013-11-07 | 1.00 | Original | Ping Zhang |
| | | | |

Scope

This document describes how to use the AT command about Bluetooth and some application note. The document can apply to “1308B01SIM800H32_BT” version.

1 Bluetooth Function

1.1. Bluetooth Introduction

Bluetooth is a wireless technology standard for exchanging data over short distances (using short-wavelength radio transmissions in the ISM band from 2400–2480 MHz) from fixed and mobile devices, creating personal area networks (PANs) with high levels of security. Bluetooth was standardized as IEEE 802.15.1

1.2. Bluetooth Profile

To use Bluetooth wireless technology, a device has to be able to interpret certain Bluetooth profiles, which are definitions of possible applications and specify general behaviors that Bluetooth enabled devices use to communicate with other Bluetooth devices. These profiles include settings to parametrize and to control the communication from start. Adherence to profiles saves the time for transmitting the parameters anew before the bi-directional link becomes effective. There are a wide range of Bluetooth profiles that describe many different types of applications or use cases for devices.

Besides of all profiles, there have four basic ones, they are GAP/SDAP/SPP/GOEP Profile.

1.3. Bluetooth Device Address

The Bluetooth device address stores the network address of a Bluetooth-enabled device. It is used to identify a particular device during operations such as connecting to, pairing with, or activating the device.

A Bluetooth-enabled device address is a unique, 48 bits address containing the following three fields:

- LAP field: lower part of the address containing 24 bits.
- UAP field: upper part of the address containing 8 bits.
- NAP field: non-significant part of the address containing 16 bits.

The LAP and the UAP represent the significant address part (SAP) of the Bluetooth device address.

1.4. AT Interface for Bluetooth Function

As module solution, we provide series of AT interface to operate Bluetooth function, including pairing, bonding, pushing or receiving file.

Also including interface for SPP service, which could communicate between Bluetooth device and others via serial port.

2. AT Command

| Command | Description |
|---------------|---|
| AT+BTHOST | Inquiry and set host device name |
| AT+BTSTATUS | Inquiry current BT device status |
| AT+BTPOWER | Power on or power off BT radio |
| AT+BTPAIR | Pair BT device |
| AT+BTSCAN | Scan surrounding BT device |
| AT+BTUNPAIR | Unpair BT device |
| AT+BTCONNECT | Connect paired BT device |
| AT+BTDISCONN | Disconnect BT device |
| AT+BTGETPROF | Get profile provided by paired device |
| AT+BTACPT | Accept connecting request |
| AT+BTOPPACPT | Accept OPP service |
| AT+BTOPPPUSH | Push OPP object to paired device |
| AT+BTSPPESEND | Send data to BT serial port as client based on SPP service |
| AT+BTSPPEGET | Get data from BT serial port as client based on SPP service |
| AT+BTATA | Answer incoming call |
| AT+BTATDL | Redial last number |
| AT+BTATH | Hung up voice call |
| AT+BTVGS | Configure voice volume |
| AT+BTVGM | Configure MIC volume |
| AT+BTATD | Dial up a voice call |
| AT+BTRSSI | Get RSSI of connected device |

2.1. AT+BTHOST Inquiry and set host device name

| AT+BTHOST | |
|------------------------------------|---|
| Test command AT+BTHOST=? | <p>Response</p> <p>+BTHOST: (1,max length of the device <name>)</p> <p>OK</p> <p>Parameters See Write Command</p> |
| Read command AT+BTHOST? | <p>Response</p> <p>+BTHOST: <name>, <address></p> <p>OK</p> |

| | |
|--|--|
| | Parameters See Write Command |
| Write command AT+BTHOST=<name> | Response OK |
| | Parameter <name> device name <address> device address |
| Note | Max length of <name> is 18 bytes, and display in UTF-8 code. |

2.2. AT+BTSTATUS Inquiry current BT device status

| | |
|--------------------------------------|--|
| AT+BTSTATUS | |
| Test Command AT+BTSTATUS=? | Response OK |
| | Parameters See Read Command |
| Read Command AT+BTSTATUS? | <p>Response</p> <p>If unpaired before: +BTSTATUS: <status></p> <p>If paired before but unconnected: +BTSTATUS: <status> P: <paired id>, <name> <address></p> <p>If paired and connected: +BTSTATUS: <status> P: <paired id>, <name> <address> C: <connected id>, <name>, <address>, <profile name></p> <p>OK</p> <p>Parameter</p> <p><status></p> <ul style="list-style-type: none"> 0 Initial 1 Disactivating 2 Activating 5 Idle 6 Scanning 7 Inquiry_Res_Ind 8 stopping scanning 9 Bonding 12 Connecting 13 Unpairing 14 Deleting paired device 15 Deleting all paired device 16 Disconnecting 19 Pairing confirm 20 Waiting for remote confirm |

| | |
|------|--|
| | 25 Accepting connection 26 SDC Refreshing 29 Setting host name 30 Releasing all connection 31 Releasing connection 36 Activating service <paired id> paired device ID <connected id> connected device ID <name> device name <address> device address <profile name> profile |
| Note | Max length of <name> is 18 bytes, 18 bytes in UTF-8 code |

2.3. AT+BTPOWER Power on/off BT radio

| AT+BTPOWER | |
|--|---|
| Test Command AT+BTPOWER=? | Response +BTPOWER: (list of supported <n>s) OK |
| | Parameters See Write Command |
| Write Command AT+BTPOWER=<n> | Response OK |
| | parameter <n> 0 power off BT radio 1 power on BT radio |
| Note | After power off BT radio, should wait 25s at least to re-power on BT radio. |

2.4. AT+BTPAIR Pair BT device

| AT+BTPAIR | |
|---|--|
| Test Command AT+BTPAIR=? | Response +BTPAIR: 0,(list of supported <device ID>s) +BTPAIR: 1,(list of supported <confirm>s) +BTPAIR: 2,(length of supported <passkey>s) OK |
| | Parameters See Write Command |
| Write Command 1) active AT+BTPAIR=0, | Response OK |

| | | | | | | | | | | | | | | | | | | | | | |
|---|--|--------------------------|--------------|------------------------|----------------------|------------------------|---------------------------|-------------------|--|---------------------|----------------|------------------------|-------------------|-------------------------|------------------|---------------------|-------------|------------------------|----------------|-------------------------|------------------|
| <p><device ID></p> <p>2) passive with digital key request AT+BTPAIR=1, <confirm></p> <p>3) passive with passkey request AT+BTPAIR=2, <passkey></p> | <p>If digital key exchanged +BTPAIRING: <name>,<address>,<passcode></p> <p>If passkey exchanged: +BTPAIRING: <name>,<address></p> <p>If passive mode with success: +BTPAIR: <id>,<name>,<address></p> <p>If passive mode with failure: +BTPAIR: 0</p> <p>Parameter</p> <table border="1"> <tr><td><device ID></td><td>BT device ID</td></tr> <tr><td><confirm></td><td>1 accept 0 reject</td></tr> <tr><td><passkey></td><td>passkey, length is (4-16)</td></tr> <tr><td><id></td><td>0 paired failed ≥1 paired device ID</td></tr> <tr><td><name></td><td>BT device name</td></tr> <tr><td><address></td><td>BT device address</td></tr> <tr><td><passcode></td><td>Digital password</td></tr> </table> <p>URC</p> <p>If there is incoming request: +BTPAIRING: <name>,<address>,<passcode></p> <p>or +BTPAIRING: <name>,<address></p> <p>Parameter</p> <table border="1"> <tr><td><name></td><td>device name</td></tr> <tr><td><address></td><td>device address</td></tr> <tr><td><passcode></td><td>digital password</td></tr> </table> | <device ID> | BT device ID | <confirm> | 1 accept 0 reject | <passkey> | passkey, length is (4-16) | <id> | 0 paired failed ≥1 paired device ID | <name> | BT device name | <address> | BT device address | <passcode> | Digital password | <name> | device name | <address> | device address | <passcode> | digital password |
| <device ID> | BT device ID | | | | | | | | | | | | | | | | | | | | |
| <confirm> | 1 accept 0 reject | | | | | | | | | | | | | | | | | | | | |
| <passkey> | passkey, length is (4-16) | | | | | | | | | | | | | | | | | | | | |
| <id> | 0 paired failed ≥1 paired device ID | | | | | | | | | | | | | | | | | | | | |
| <name> | BT device name | | | | | | | | | | | | | | | | | | | | |
| <address> | BT device address | | | | | | | | | | | | | | | | | | | | |
| <passcode> | Digital password | | | | | | | | | | | | | | | | | | | | |
| <name> | device name | | | | | | | | | | | | | | | | | | | | |
| <address> | device address | | | | | | | | | | | | | | | | | | | | |
| <passcode> | digital password | | | | | | | | | | | | | | | | | | | | |
| <p>Note</p> | <p>1. Max length of <name> is 18 bytes, 18 bytes in UTF-8 code</p> <p>2. Pairing timeout is around 15s each side</p> | | | | | | | | | | | | | | | | | | | | |

2.5. AT+BTUNPAIR Unpair BT device

| AT+BTUNPAIR | |
|---|--|
| <p>Test Command AT+BTUNPAIR=?</p> | <p>Response +BTUNPAIR: (list of supported <device ID>s)</p> <p>OK</p> <p>Parameter See Write Command</p> |
| <p>Write Command AT+BTUNPAIR=<device ID></p> | <p>Response OK</p> <p>Parameter</p> |

| | |
|--|--|
| | <p><device ID> Paired Device ID.</p> <p>0 delete all the paired device</p> <p>1 delete the the paired device corresponding to ID</p> |
|--|--|

2.6. AT+BTSCAN Scan surrounding BT device

| AT+BTSCAN | |
|--|--|
| Test Command AT+BTSCAN=? | <p>Response</p> <p>+BTSCAN: (list of supported <switch>s), (list of supported <Timer>s)</p> <p>OK</p> <p>Parameter</p> <p>See Write Command</p> |
| Write Command AT+BTSCAN=<switch>[,<Timer>] | <p>Response</p> <p>OK</p> <p>If BT device scanned: +BTSCAN: <status>,<device ID>,<name>,<address></p> <p>If terminate: +BTSCAN: <status></p> <p>Parameter</p> <p><switch> 1 start 0 stop</p> <p><status> 0 BT device found 1 scanning finished 2 scanning stop 3 scanning failed</p> <p><Timer> scanning time 10-60s</p> <p><device ID> BT device ID scanned</p> <p><name> BT device name</p> <p><address> BT device address</p> |
| Note | <p>1. Max length of <name> is 18 bytes, 18 bytes in UTF-8 code</p> <p>2. If <timer> omitted, the default value is 30s</p> |

2.7. AT+BTCONNECT Connect paired BT device

| AT+BTCONNECT | |
|---------------------------------------|--|
| Test Command AT+BTCONNECT=? | <p>Response</p> <p>+BTCONNECT: (list of supported <device ID>s), (list of supported <profile ID>s)</p> <p>OK</p> <p>Parameter</p> <p>See Write Command</p> |

| | |
|--|---|
| Write Command AT+BTCONNE CT=<device ID>,<profile ID> | <p>Response</p> <p>OK</p> <p>If OK: +BTCONNECT: <id>,<name>,<address>,<profile name></p> <p>If failed: +BTCONNECT: 0</p> |
| | <p>Parameter</p> <p><device ID> ID of paired BT device</p> <p>< profile ID> BT profile ID</p> <p><id> ID of connected BT device</p> <p><name> BT device name</p> <p><address> BT device adress</p> <p><profile name> BT device service name</p> |
| Note | <p>1. Max length of <name> is 18 bytes, 18 bytes in UTF-8 code</p> <p>2. Connection timeout is around 20s</p> <p>3. if incoming request, there will be URC +BTCONNECTING: <address>,<profile name></p> |

2.8. AT+BTDISCONN Disconnect BT connection

| | |
|---|--|
| AT+BTDISCONN | |
| Test Command AT+BTDISCON N=? | <p>Response</p> <p>+BTDISCONN: (list of supported <device ID>s)</p> <p>OK</p> |
| | <p>Parameter</p> <p>See Write Command</p> |
| Write Command AT+BTDISCON N=<device ID> | <p>Response</p> <p>OK</p> <p>+BTDISCONN: <name>,<address>,<profile name></p> |
| | <p>Parameter</p> <p><device ID> connected device ID</p> <p><name> device name</p> <p><address> devie address</p> <p><profile name> profile service</p> |
| Note | <p>1. Max length of <name> is 18 bytes, 18 bytes in UTF-8 code</p> <p>2. If disconnected by remote, there still be URC: +BTDISCONN</p> |

2.9. AT+BTGETPROF Get profile provided by paired device

| | |
|------------------------------------|--|
| AT+BTGETPROF | |
| Test Command AT+BTGETPRO | <p>Response</p> <p>+BTGETPROF: (list of supported <device ID>s)</p> |

| | |
|---|--|
| F=? | OK |
| | Parameter See Write Command |
| Write Command AT+BTGETPRO F=<device ID> | Response OK +BTGETPROF: <profile ID>,<name> |
| | Parameter <device ID> Paired Device ID <profile ID> profile ID <name> profile name |

2.10. AT+BTACPT Accept connecting request

| | |
|---|--|
| AT+BTACPT | |
| Test Command AT+BTACPT=? | Response +BTACPT: (list of supported <confirm>s) OK |
| | |
| Write Command AT+BTACPT=<confirm> | Response OK If connected successfully, then will report: +BTCONNECT: <id>,<name>,<address>,<profile name> If connecting failed: + BTDISCONN: <name>,<address>,<profile name> |
| | Parameter <confirm> 1 accept 0 reject <id> >0 connected device ID <name> device name <address> device address <profile name> profile name |
| | URC If incoming connecting request: +BTCONNECTING: <address>,<profile name> |
| | Parameter <address> device address <profile name> profile name |

| | |
|------|--|
| Note | Max length of <name> is 18 bytes, 18 bytes in UTF-8 code |
|------|--|

2.11. AT+BTOPPACPT Accept OPP service

| AT+BTOPPACPT | |
|--|---|
| Test Command AT+BTOPPACPT=? | Response +BTOPPACPT: (list of supported <confirm>s),(list of supported<drv>) OK |
| Write Command AT+BTOPPACPT=<confirm>[,<drv>] | Response OK +BTOPPPUSH: <status> Parameter <div> <confirm> 1 Accept 0 Reject <drv> 0 internal flash memory 1 external memory card <status> 0 failed 1 successful </div> URC: If there has an incoming opp file, there will be a URC report. +BTOPPPUSHING: <name>, <file name> Parameter <name> device name <file name> file name |
| Note | 1. Max length of <name> is 18 bytes, 18 bytes in UTF-8 code 2. File is stored in path: C:\User\BtReceived\ for internal memory card, D:\BtReceived\ for external memory card. At the first time to use SD card, customer must execute “AT+SD2PCM=0” and “AT&W”, then reboot the module. |

2.12. AT+BTOPPPUSH Push OPP object to paired device

| AT+BTOPPPUSH | |
|---------------------------------------|---|
| Test Command AT+BTOPPPUSH=? | Response +BTOPPPUSH: (list of supported <device ID>s), (length of supported <string>s) OK |
| | Parameter |

| | |
|--|---|
| | See Write Command |
| Write Command AT+BTOPPPUS HT=<device ID >,<string> | Response OK +BTOPPPUSH: <para> Parameter <device ID> Paired Device ID <string> file name include complete path, lenght (4-259) <para> 0 Send failed 1 Send successfully 2 Server issue |
| Note | |

2.13. AT+BTSPGET Get data from BT serial port as client based on SPP service

| | |
|---|--|
| AT+BTSPGET | |
| Test Command AT+BTSPGET =? | Response +BTSPGET: (list of supported <command>s) OK Parameter See Write Command |
| Read Command AT+BTSPGET ? | Response +BTSPGET: <command> OK Parameter See Write Command |
| Write Command AT+BTSPGET =<command>[, <reqLength>][,<s howWithHex>] | Response OK or Error If command value is 2,return format: +BTSPGET: <port ID>,<cnfLen1> OK If command value is 3,return format: +BTSPDATA: <port ID>,<repLength>,<data string> OK Automatic mode, the header will be +BTSPDATA, manual mode, the header will be +BTSPGET. |

| | |
|------|---|
| | <p>Parameter</p> <p><command> 0 Automatic mode. Data will be output in decimal system 1 manual mode. There will be an indication when first package arrived 2 Inquiry data length under manual mode 3 Get data under manual mode.</p> <p><reqLength> 1-1024 , the length of data requested, only valid under manual mode</p> <p><showWithHex> 1, displayed with HEX, only valid under manual mode</p> <p><port ID> Serial port ID</p> <p><cnfLen1> 1-1024, character length</p> <p><data string> string printed</p> |
| Note | <p>URC</p> <p>When module receive SPP data,there will be UCR report:</p> <ol style="list-style-type: none"> Automatic mode +BTSPPDATA: <port ID>,<cnfLen2>,<data string> Manual mode +BTSPPMAN: <port ID> <p>Parameter</p> <p><cnfLen2> 1-1024, length of printed character</p> |

2.14. AT+BTSPSEND Send data to BT serial port as client based on SPP service

| AT+BTSPSEND | |
|---|---|
| Write Command AT+BTSPSEN D=<length> | <p>Response</p> <p>></p> <p>If successful, SEND OK</p> <p>Or if failed, SEND FAIL</p> |
| | <p>Parameter</p> <p><length> 1-1024, the length of data will be sent.</p> <p>When the length of inputing data is up to <length> specified, the package will be sent out automatically. ESC key is used to quit in the middle of process.</p> |
| Execute Command AT+BTSPSEN D | <p>Response</p> <p>></p> <p>If successful, SEND OK</p> <p>Or failed, SEND FAIL</p> |

| | |
|--|---|
| | Under this mode, <Ctrl+z> will submit the package, ESC will quit the process. |
|--|---|

2.15. AT+BTATA Answer incoming call

| AT+BTATA | |
|------------------------------------|--|
| Execute Command AT+BTATA | Response OK |
| Note | When module connected with smartphone as an earphone, if here comes incoming call, the call would be answered through this command |

2.16. AT+BTATDL Redial last number

| AT+BTATDL | |
|-------------------------------------|---|
| Execute Command AT+BTATDL | Response OK |
| Note | When module connected with smartphone as an earphone, would redial last number through this command |

2.17. AT+BTATH Hung up voice call

| AT+BTATH | |
|------------------------------------|---|
| Execute Command AT+BTATH | Response OK |
| Note | When module connected with smartphone as an earphone, the incoming call would be hung up through this command |

2.18. AT+BTVGS Configure voice volume

| AT+BTVGS | |
|---|--|
| Test Command AT+BTVGS=? | Response +BTVGS: (<gain> range) |
| | OK |
| | Module is Earphone mode |
| Read Command AT+BTVGS? | Response +BTVGS: <gain> |
| | OK |
| | |
| Write Command AT+BTVGS=<gain> | Response OK |
| | Parameter <gain> volume |
| | This command is used to configure call volume when the module is |

| | |
|------|---|
| | connected with smartphone as an earphone |
| Note | For some smartphone,after connected with BT earphone,the current call volume may not be transmitted to earphone,thus the return value of the read command may be 0.But after setting once,the value would be correct. |

2.19. AT+BTVMG Configure MIC gain level

| AT+BTVMG | |
|---|--|
| Test Command AT+BTVMG=? | Response +BTVMG: (<gain> range) OK |
| Read Command AT+BTVMG? | Response +BTVMG: <gain> OK |
| Write Command AT+BTVMG=<gain> | Response OK Parameter <gain> MIC gain level This command is used set MIC volume when the module is connected with smartphone as an earphone |
| Note | For some smartphone,after connected with BT earphone,the current MIC volume may not be transmitted to earphone,thus the return value of the read command may be 0.But after setting once,the value would be correct. |

2.20. AT+BTATD Dial voice call

| AT+BTATD | |
|---|--|
| Test Command AT+BTATD=? | Response +BTATD: (<number> length range) OK |
| Write Command AT+BTATD=<number> | Response OK Parameter <number> phone number Module as earphone connected to smartphone, this command could make an outgoing call |
| Note | |

2.21. AT+BTRSSI Get RSSI of connected BT device

| AT+BTRSSI | |
|--|---|
| Test Command AT+BTRSSI=? | Response +BTRSSI: (<device ID>) |
| | OK |
| Write Command AT+BTRSSI=<device ID> | Response +BTRSSI: <rss> |
| | OK |
| | Parameter <device ID> Connected Device ID <rss> -122...0 RSSI value of BT device |
| Note | RSSI value is negative, the smaller value represents the worse signal |

3. CME Error Code

+CME ERROR: <err>

| Code | Description |
|------|----------------------------|
| 1000 | Return fail |
| 1002 | Not power on |
| 1003 | State not idle |
| 1004 | Malloc error |
| 1010 | Scan fail |
| 1011 | scan return error |
| 1020 | Out of scanning count |
| 1021 | Out of profile id count |
| 1025 | Out of pairing count |
| 1026 | Bond error |
| 1027 | Device has Bonded |
| 1030 | Debond error |
| 1031 | Get device info error |
| 1032 | Service refresh error |
| 1033 | Profile connect error |
| 1040 | OPP handle error |
| 1041 | OPP send error |
| 1045 | OPP send error by server |
| 1046 | Get index by profile error |
| 1047 | Connect not support |
| 1048 | Disconnect not support |
| 1049 | Active or address error |
| 1050 | Only connect one device |
| 1055 | Spp is not connect |
| 1099 | BTAUD attach error |

4 Examples

There are some examples to explain how to use these commands.

In the “Grammar” columns of following tables, input of AT commands are in black, module return values are in blue.

4.1 Accept request from other BT device

| Command | Description |
|---|--|
| AT+BTPOWER=1 OK | Power on BT radio |
| +BTPAIRING: "PC-NS130100361",34:c7:31:aa:37:5b,763191 | Incoming digital key request from other BT device |
| AT+BTPAIR=1,1 OK +BTPAIR: 1,"PC-NS130100361",34:c7:31:aa:37:5b | Accept pairing request, and paired successfully |
| +BTPAIRING: "Jabra BT160",00:16:8f:0d:65:82 | Incoming passkey request from other BT device |
| AT+BTPAIR=2,0000 OK +BTPAIR: 2,"LBH505",50:5b:0b:0a:10:32 | Accept pairing request, and paired successfully.Default passkey of other BT device is 0000.If not, please change this value according to other device's passkey. |

4.2 Send pairing request to other BT device

| Command | Description |
|--|---------------------------------|
| AT+BTPOWER=1 OK | Power on BT radio |
| AT+BTSCAN=1,20 OK +BTSCAN: 0,1,"PC-NS130100361",34:c7:31:aa:37:5b +BTSCAN: 0,2,"ADMIN-9A6E040AC",68:5d:43:ec:fe:72 +BTSCAN: 0,3,"LIB-PC",c8:f7:33:43:48:e6 +BTSCAN: | Inquiring surrounding BT device |

| | |
|--|--|
| 0,4,"MK-FUJIANJUN",88:53:2e:e8:9d:0f +BTSCAN: 0,5,"MTKBTDEVICE",45:8c:96:3e:66:01 +BTSCAN: 0,6,"MK-ZHANZHIMIN",00:1a:7d:da:71:10 +BTSCAN: 0,7,"Jabra BT160",00:16:8f:0d:65:82 +BTSCAN: 1 | |
| AT+BTPAIR=0,6 OK | Try to pair the sixth BT device in the view list |
| +BTPAIRING: "MK-ZHANZHIMIN",00:1a:7d:da:71:10,76319 1 AT+BTPAIR=1,1 OK +BTPAIR: 1,"MK-ZHANZHIMIN",00:1a:7d:da:71:10 | Answer to the pairing request in digital key mode |
| AT+BTPAIR=0,7 OK | Try to pair the seventh BT device in the view list |
| +BTPAIRING: "Jabra BT160",00:16:8f:0d:65:82 AT+BTPAIR =2,0000 OK +BTPAIR: 2,"Jabra BT160",00:16:8f:0d:65:82 | Answer to the pairing request in passkey mode |

4.3 Get the profile provided by paired device

| Command | Description |
|---|--|
| | Configure based on example 4.2 |
| AT+BTGETPROF=1 +BTGETPROF: 1,"A2DP(Source)" +BTGETPROF: 2,"HFP(AG)" +BTGETPROF: 8,"AVRCP(Target)" +BTGETPROF: 3,"A2DP" +BTGETPROF: 4,"SPP" +BTGETPROF: 6,"HFP" +BTGETPROF: 5,"HSP" OK | Get the profile of first paired device in list |

4.4 Connect service

| Command | Description |
|--|--|
| | Get Profile based on example 4.3 |
| AT+BTCONNECT=1,2 OK +BTCONNECT: 1,"MK-ZHANZHIMIN",00:1a:7d:da:71:10,"HFP(AG)" | Connect with the second profile service of first paired device,"HFP(AG)" |

4.5 Accept file from paired device

| Command | Description |
|--|---|
| | Pairing device based on example 4.2 |
| +BTOPPPUSHING: "MK-ZHANZHIMIN","link.txt" | Incoming opp pushing service from paired device |
| AT+BTOPPACPT=1 OK +BTOPPPUSH: 1 | Accept file(stored in internal memory card by default,input "AT+BTOPPACPT=1,1" if want it stored in external memory |

4.6 Send file to other paired BT device

| Command | Description |
|---|---------------------------------------|
| | Pairing device based on example 4.2 |
| AT+BTOPPPUSH=1,c:\User\BtReceived\link.txt OK +BTOPPPUSH: 1 | Sending file and waiting for response |
| | |

4.7 AT Channel mode, Module as client

SPP service has two kinds of connection. One is client mode via AT command channel (shorted as AT channel mode following), another is peer to peer mode to exchange data (shorted as exchanged mode following).

This section is for AT channel mode as client.

| Command | Description |
|---------|---|
| | Based on example 4.3, get profile. Supposed local device ID is 34:c7:31:aa:37:5b, another remote device ID is 12:34:56:78:90:12, name is BTOTHER, unconnected. |

| | |
|----------------------------------|--|
| AT+BTSPPGET=0 | Set report-auto mode |
| OK | |
| AT+BTCONNECT=1,4 | Connect server |
| OK | |
| +BTCONNECT: 1," | Report automatically once ok |
| BTOTHER",12:34:56:78:90:12,"SPP" | |
| AT+BTSPPSSEND | Send data here when get propomt ">", and |
| >AT+CREG? | then the input characters and the response |
| SEND OK | will output here |
| | |
| +BTSPPDATA: 19,1,A | Input characters |
| +BTSPPDATA: 19,3,T+C | |
| +BTSPPDATA: 19,25,REG? | |
| +CREG: 0,0 | Response |
| OK | |

4.8 AT Channel mode, module as server

| Command | Description |
|------------------|---|
| | Supposed module acts as serve, connection is available, but first data package is not "SIMCOMSPPFORAPP" |
| AT+BTSPPSSEND=10 | At this moment, data can not be sent |
| ERROR | |

4.9 Exchanged mode setup

| Command | Description |
|------------------|---|
| | Based on example 4.3, get profile. Supposed local device ID is 34:c7:31:aa:37:5b, another remote device ID is 12:34:56:78:90:12, name is BTOTHER, unconnected. |
| AT+BTSPPGET=0 | Set auto-report mode\ |
| OK | |
| AT+BTCONNECT=1,4 | Connect remote pear |

| | |
|--|--|
| <pre> OK +BTCONNECT: 1," BTOTHER",12:34:56:78:90:12,"SPP" AT+BTSPGET? +BTSPGET: 0 OK AT+BTSPSEND > SIMCOMSPFORAPP +BTSPDATA: 19,10,1234567890 AT OK AT+BTSPGET=1 ERROR </pre> | <p>Report automatically once ready</p> <p>Get SPP report mode</p> <p>After connected, receiving mode can not be configured</p> |
| <pre> AT+BTSPGET=1 OK AT+BTCONNECT=1,4 OK +BTCONNECT: 1," BTOTHER",12:34:56:78:90:12,"SPP" AT+BTSPSEND > SIMCOMSPFORAPP +BTSPMAN: 19 AT OK AT+BTSPGET=2 +BTSPGET: 19,10 OK AT+BTSPGET=3,3 +BTSPGET:19,3,123 OK AT+BTSPGET=3,10,1 +BTSPGET: 19,7,34353637383930 </pre> | <p>Supposed manual mode ok</p> <p>connect</p> <p>Inquiry data length</p> <p>Read 3 bytes</p> <p>Read 10 bytes and display in hex</p> |

OK

+BTSPPMAN: 19

AT+BTSPPGET=?

+BTSPPGET: (0-3)

OK

4.10 Data Exchanged

| Command | Description |
|--|-------------|
| AT+BTSPPSSEND=10 >1234567890 SEND OK | |
| AT+BTSPPSSEND >abcdefg SEND OK | |

4.11 Module as SPP server

This SPP connection is launched by remote BT device,module will play a role as BT serial server after accept this connection request.Then remote device can access the moduel's AT serial port with AT interchange though the established serial channel.

| Command | Description |
|--|--|
| | Based on example 4.2 |
| +BTCONNECTING: 00:1a:7d:da:71:10,"SPP" | SPP Request from remote device |
| AT+BTACPT=1 OK +BTCONNECT: 1, "MK-ZHANZHIMIN",00:1a:7d:da:71:10,"SPP" | Accept request,module as serial server,remote device can access module's AT port through the BT connection |

Appendix

A. Reference

| ID | Document | Remark |
|-----|---------------------------------|--------|
| [1] | SIM800 Series AT Command Manual | |
| | | |

B. Profile

| Profile | Introduction |
|---------|--|
| SPP | Abbreviation of Serial Port Profile,to implement BT serial port function.Module an transmit data to connected BT device throuth AT+BTSPSEND after successfully applying this profile.The module will receive data report +BTSPDATA in automatic mode,and +BTSPPMAN in manual mode. |
| OPP | Abbreviation of OPP Object Push Profile,to implement pushing BT object.This unction is used between the two paired BT devices,AT+BTOPPPUSH to push file, AT+OPACPT to receive the pushed file. |
| HFP/HSP | Abbreviation of Handsfree Profile/Headset Profile, i.e. BT earphone function. HFP is the enhanced version of HSP,so even if the other BT device just supports HSP,SIM800H still can connect the BT device with HFP.Module's call voice would be displayed from BT earphone after this profile being connected.When the module play a role as smart phone,BT earphone could control the call operation(e.g.hang up,answer,redial) . |
| A2DP | Abbreviation of Advanced Audio Distribution Profile,which is advanced rotocol for audio frequency distribution.Earphone will activate AVRCP connection after the profile being connected.It is mainly used to for BT earphone to transmit Hi-Q audio frequency.If be suffixed with source,it means this device is audio frequency source,i.e. paly a role as smartphone. |
| AVRCP | Abbreviation of Audio Video Remote Control Profile,is AV remote control protocol.This profile depends on A2DP and only could be connected after the A2DP connection is established.It is mainly used for BT earphone to control the edia function of smartphone.If be suffixed with target,it means this device is controlling target,i.e.paly a role as smart phone. |
| HFP(AG) | This profile si HFP,i.e. paly a role as BT earphone.After the module connected with smartphone,the call voice of smartphone could be displayed by the module's audil channel.Also the call operation of smartphone can be controlled by those commands such as AT+BTATD, AT+BTATH, AT+BTATA. |
| HFG | This profile is HFP,but plays a role as smartphone at this moment.After the |

module connected with smartphone, there will display such information indicates profile being connected successfully. If the module plays a role of earphone, then the information displayed after connection will be HFP(AG).

C. Glossary and Abbreviation

| Glossary | Discription |
|----------|-------------------------------------|
| EVB | Evaluation Board |
| BT | Blue tooth |
| PROFILE | Bluetooth function protocol |
| SPP | Serial Port Profile |
| OPP | OPP Object Push Profile |
| A2DP | Advanced Audio Distribution Profile |
| AVRCP | Audio Video Remote Control Profile |
| HSP | BT handset protocol |
| HFP | HandFree application protocol |
| URC | Unsolicited Result Code |
| TE | Terminal Equipment |
| TA | Terminal Adapter |
| DTE | Data Terminal Equipment |
| DCE | Data Communication Equipment |
| ME | Mobile Equipment |
| MS | Mobile station |

Contact us:

Shanghai SIMCom Wireless Solutions Co.,Ltd.

Address: Building A, SIM Technology Building, No. 633, Jinzhong Road, Shanghai,
P. R. China 200335

Tel: +86 21 3252 3300

Fax: +86 21 3252 3020

URL: www.sim.com/wm

SIMCOM CONFIDENTIAL FILE