BlueSoleil Firmware Programming Manual

August 6, 2013

Version 2.2

VERSION HISTORY

REVISION	AMENDMENT	DATE	AUTHOR
1.0	Initial version	2012-06-25	Wang Yuqiang Zhu Yong
1.1	Added certification, contact information, copyright	2012-08-08	Niu Chong Huang Ruixue
1.2	1. Update commands' response. See the chapter 12. 2. Update the chapter 2. 3. Update format.	2012-12-04	Li Li
1.3	1. Update commands, See the chapter 4.7,4.11, 6.2, 6.4, 6.11, 6.12, 6.13, 12.2, 12.3, 12.4, 16.4. 2. Update format.	2012-12-13	Li Li
1.4	1. Add 4.13 EDFU command; 2. Add 7.16 HFSRVC command; 3. Add 7.17 HFCHLD command; 4. Add 5.2 ROLE indication; 5. Add 11.3 A2DPAUDIO indication 6. Update 7.10, 4.8, 2.1.4, 4.10; 7. Update 8.4, 8.5, 8.6, 8,7, 8.8, 9.2,18.6 8. Add 20,21 Opps	2013-03-07	Wang Yuqiang
1.5	1. Add 9.4 PBCHARSET indication.	2013-03-13	WangYuqiang
1.6	1. Chapter 12, update commands, delete AVRCP TG relevant commands; 2. Chapter 13,delete AVRCP TG relevant indications; 3. Add chapter 22, AVRCP TG commands; 4. Add chapter 23, AVRCP TG indications.	2013-03-14	WangYuqiang
1.7	1. Add chapter 24, HFP AG commands; 2. Add chapter 25, HFP AG indications.	2013-03-17	WangYuqiang

1.8	1.Add AT+B UART command 2.Update PBAP & MAP commands/indications in chapter 12,13,14,15	2013-4-19	WangYuqiang
1.9	1. Default speaker/microphone values set to 15. 2. Add HID interfaces. 3. Update MAP chapter structure. 4. Correct chapter reference mistakes.	2013-6-18	Yan Laijun Li Li
2.0	Add 4.15 AT+B SCOD command. Add 4.15 AT+B GCOD command.	2013-7-10	Wang yuqiang
2.1	Add 6.14 AT+B HFSCFG command Add 6.15 AT+B HFGCFG command	2013-7-18	Wang yuqiang
2.2	1. Add 4.11 AT+B SPRO command; 2. Update 20.1.1 HID Connect Status; 3. Add HID stream methods in 20.4 and 20.5; 4. Add 20.4 AT+B PINRES command and 21.2 AT-B PININD event; 5. Add PIO Assignments; 6. Compatible with 40e-multiprofile AT commands;	2013-8-6	Wang yuqiang Yan Laijun

Contents

1 Introduction

IVT BlueSoleil module i50e and i40e are *Bluetooth* Classic 2.1+EDR modules. They are embedded with BlueSoleil firmware which supports HFP (Hands-free Profile), HSP (Headset Profile), A2DP (Advanced Audio Distribution Profile), AVRCP (Audio & Video Remote Control Profile), PBAP (Phonebook Access Profile), MAP (Message Access Profile), SPP (Serial Port Profile),OPP (Object Push Profile) and HID (Human Interface Device) Profile. BlueSoleil i40e does not support A2DP, AVRCP, and HFP AG role.

There are two roles in BlueSoleil Firmware: the module acts as controller role, and MCU acts as host role. *Bluetooth* functions are embedded in the controller unit and the application is running in the host unit. To achieve high-level hardware integration, the host unit communicates with the controller unit via UART by using well defined AT commands.

This document addresses BlueSoleil firmware's default setting, and AT commands.

2 Important Notes

This chapter discusses BlueSoleil firmware defaulting setting and parameter maximum length.

2.1 Default Settings

This chapter introduces BlueSoleil firmware's defaulting setting.

2.1.1 UART Default Setting

The UART default setting is 115200, 8, N, 1

2.1.2 Default Device Name

i50e default device name is I50E_STD. i40e default device name is I40E_MHF.

2.1.3 Default PIN Code

The default PIN Code is 0000.

2.1.4 Default Device Type

The default device type is 0x001F00 (Unknown *Bluetooth* device).

2.1.5 Default Speaker Volume

The default speaker volume is 15. It can be changed with the command in the chapter 6.11.

2.1.6 Default Microphone Volume

The default microphone volume is 15. It can be changed with the command in the chapter 6.12.

2.2 Boot-up Timing

Host shall open the UART port immediately after power on the *Bluetooth* module. The host can receive the "AT-B INIT 0\r" string when the *Bluetooth* system initializing finished.

2.3 Parameter Maximum Length

This chapter introduces parameters' maximum length.

2.3.1 *Bluetooth* Software Version Information

An ASCII code string like "I50E.STD.0.20121101.1". The maximum length is 22 bytes.

2.3.2 *Bluetooth* Device Address

An ASCII code string like "1234567890AB", the length is 12 bytes.

2.3.3 *Bluetooth* Device Name

The maximum length of the *Bluetooth* device name is a 31 bytes UTF8 code string with a

mix of ‘A’-‘Z’,’a’-‘z’,’0’-‘9’. The length of the local device name must be between 1 byte and 31 byte.

If the remote device name is non-English letters, the host unit shall call a UTF8 to ASCII converter to display characters correctly. Please refer to the PC host APP code for more information.

2.3.4 PIN Code

The maximum length of PIN code is 16 bytes. Only ‘0’-‘9’ is admitted.

2.3.5 Dial Number Length

The number is dialed out. There is no limit to number length on HF device side, but we had better limit it to less than 40 bytes.

2.3.6 Caller ID Display Length

There is no limit to number length on HF device side, and it depends on mobile phone side, but we had better limit it to less than 40 bytes.

3 AT Command Format

This chapter introduces the AT commands’ format. Some responses will not be returned immediately. Where applicable, an approximate delay time will be included to notify the response delay.

Please note that a full piece of AT command, AT response or AT indication must be tailed with “\r” (0x0d).

3.1Command Format

```
<at-command-object>::={
        <at-command-header><SPACE>
        <at-command-body><SPACE>
        [<at-command-parameter>[COMMA]]*
        <CR>
}
```

<at-command-header>::=AT+B

<at-command-body>::='character set, upper case'

<at-command-parameter>::=' number set and character set, be separated by comma, the last parameter need not comma-tailed'

3.2 Response Format

<at-response-object>::={

<at-response-header><SPACE>

<at-response-body><SPACE>

[<at-response-parameter><COMMA>]*

<CR>

}

<at-response-header>::=AT-B

<at-response-body>::='character set, upper case'

<at-response-parameter>::='number set and character set, be separated by comma, the last parameter need not comma-tailed'

3.3 Indication Format

<at-indication-object>::={

<at-indication-header><SPACE>

<at-indication-body><SPACE>

[<at-indication-parameter><COMMA>]*

<CR>

}

<at-indication-header>::=AT-B

<at-indication-body>::='character set, upper case, length'

<at-indication-parameter>::=' number set and character set, be separated by comma, the last parameter need not comma-tailed'

4 Generic AT Command Definition

This chapter introduces the generic AT commands' definition, including a brief description of commands' syntax, responses and examples. All commands listed in this chapter are

profile-independent.

4.1GVER

The GVER command is used to get the version of the controller unit firmware.

Command	**AT+B GVER**	
Response	**AT-B GVER [ver]**	
Parameters	N/A	
Note	N/A	

4.2GLBD

The GLBD command is used to get the local *Bluetooth* device address.

Command	**AT+B GLBD**	
Response	Succeed: **AT-B GLBD 0,[bd]** Failed: **AT-B GLBD 1,0**	
Parameters	bd	Local *Bluetooth* device address.
Note	bd is comprised of 12 bytes hexadecimal characters.	

4.3GLDN

The GLDN command is used to get the local device name.

Command	**AT+B GLDN**	
Response	Succeed: **AT-B GLDN 0,[name]** Failed: **AT-B GLDN 1,**	
Parameter	name	Device name. Default device name is I50E_STD or I40E_MHF.
Note	N/A	

4.4SLDN

The SLDN command is used to set the local device name.

Command	**AT+B SLDN [name]**	
Response	Succeed: **AT-B SLDN 0** Failed: **AT-B SLDN 1**	
Parameter	name	Device name. UTF-8 format.
Note	The length of name can be up to 31 bytes at maximum.	

4.5GRDN

The GRDN command is used to get the specific remote device name.

Command	**AT+B GRDN [bd]**	
Response	Succeed: **AT-B GRDN 0,[bd],[name]** Failed: **AT-B GRDN 1,[bd],**	
Parameters	bd	Remote *Bluetooth* device address.
	name	Remote device name.
Note	bd is comprised of 12 bytes hexadecimal characters.	

4.6SPIN

The SPIN command is used to set the local PIN code.

Command	**AT+B SPIN [pin]**	
Response	Succeed: **AT-B SPIN 0** Failed: **AT-B SPIN 1**	
Parameters	pin	The new PIN code. The default PIN code is 0000.
Note	N/A	

4.7GPIN

The FPIN command is used to get the local PIN code.

Command	**AT+B GPIN**	
Response	Succeeded: **AT-B GPIN 0,[pin]** Failed: **AT-B GPIN 1,0**	
Parameters	pin	The local PIN code.
Note	N/A	

4.8GPRD

The GPRD command is used to get the paired record which stored in local BT module.

Command	**AT+B GPRD**	
Response	**AT-B GPRD [total],[index],[bd]** If no paired record found: **AT-B GPRD 0,0, 000000000000** If paired records found(n>=1): **AT-B GPRD n,0,bd** **AT-B GPRD n,1,bd** **……** **AT-B GPRD n,n-1,bd**	
Parameters	total	Total paired devices in the controller unit.
	index	Index of the total parameter
	bd	Remote *Bluetooth* device address.
Note	bd is comprised of 12 bytes hexadecimal characters.	

4.9DPRD

The DPRD command is used to delete the specified BD address paired record.

Command	**AT+B DPRD [bd]**	
Response	**AT-B DPRD [result],[bd]** If delete all paired device records: **AT-B DPRD 0,000000000000** If delete a paired device record with the specified *Bluetooth* device address: **AT-B DPRD 0,[bd]** If failed to delete a paired device record (For instance, not found in device paired list): **AT-B DPRD 1,[bd]**	
Parameter	bd	If the bd parameter in the command equals to"000000000000", all paired device records will be deleted; If the bd parameter in the command does not equals to "000000000000", the paired device record which *Bluetooth* address equals to bd will be deleted.
	result	0: succeeded; 1: failed.
Note	bd is comprised of 12 bytes hexadecimal characters.	

4.10 INQU

The INQU command will cause local device to discover other nearby *Bluetooth* devices.

Command	**AT+B INQU [op]**	
Response	If any nearby device was found: **AT-B INQR [bd],[class]** If the inquiry process finished: **AT-B INQC**	
Parameters	op	1: start searching nearby *Bluetooth* devices; 0: stop the inquiry procedure.
	bd	Remote *Bluetooth* device address.12 bytes hexadecimal characters
	class	Class of device

Note	Default inquiry time is 12.8s, default response number of device is 8, when either of conditions comes, the inquiry will terminate.

4.11 PAIR

The PAIR command is used to pair with remote device by BD address.

Command	**AT+B PAIR [bd]**	
Response	**AT-B PAIR [result],[bd]**	
Parameters	bd	Remote *Bluetooth* device address
	result	Pairing results, where 0: Authentication was successful; 1: Authentication timed out; 2: Authentication failed; 3: Authentication failed due to too many repeat attempts; 4: Authentication failed as remote device is not allowing pairing; 5: Authentication failed as unit keys are not supported; 6: Authentication failed as simple pairing is not supported; 7: Authentication failed as host is already busy pairing.
Note	bd is comprised of 12 bytes hexadecimal characters.	

4.12 SCAN

The SCAN command is used to set the scan mode.

Command	**AT+B SCAN [mode]**	
Response	Succeeded: **AT-B SCAN 0** Failed: **AT-B SCAN 1**	
Parameters	mode	Scan mode, where 0: No scans enabled;

		1: Enable Inquiry scan and Page scan disabled; 2: Enable page scan and Inquiry scan disabled; 3: Enable inquiry and page scan.
Note	Inquiry scan means the controller unit can be inquired by other *Bluetooth* devices. Page scan means the controller can be connected by other *Bluetooth* devices. Default settings is mode=3.	

4.13 EDFU

The EDFU command is used to make the module enter the DFU mode.

Command	**AT+B EDFU**
Response	**AT-B EDFU 0**
Parameters	N/A
Note	This command will force a warm reset and make the module enter the DFU mode.

4.14 UART

The UART command is used to set serial communication parameters of the module's UART controller.

Command	**AT+B UART [baud],[stop],[parity]**	
Response	Succeeded: **AT-B UART 0** Otherwise: **AT-B UART 1**	
Parameters	baud	Supported baud rate: 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600, and 1382400.
	stop	0: 1bit stop bit; 1: 2 bit stop bit.
	parity	0: No checksum; 1: Odd checksum;

		2: Even checksum.
Note	N/A	

4.15 SCOD

This command is used to set class of device of the local device to the supplied value.

Command	**AT+B SCOD [cod]**	
Response	Succeeded: **AT-B SCOD 0** Otherwise: **AT-B SCOD 1**	
Parameters	cod	Class of device, ASCII code characters.
Note	BlueSoleil firmware's default COD is not HID devices. Therefore, if HID profile is enabled, it needs to set the module's COD to Peripheral (mouse, joystick, keyboard, …) since some mobile phones may refuse to establish HID (keyboard or mouse) connections with *Bluetooth* devices which COD is not set to HID devices. COD (Major and minor device class fields) refers to the link: https://www.bluetooth.org/en-us/specification/assigned-numbers/baseband For instance, the major device class field (bit 12~bit8) should be set to 00101, the minor device class field (bit7~bit6) could be set to 01 (keyboard), a0(pointing device). HID Keyboard COD should be set to 0540, HID Mouse COD should be set to 0580.	

4.16 GCOD

This command is used to get the local class of device.

Command	**AT+B GCOD**	
Response	**AT-B GCOD [status],[cod]**	
Parameters	status	0: Succeeded; Else: Failed.
	cod	Class of device, ASCII code characters.
Note	N/A	

4.17 SPRO

Command	AT+B SPRO [profile_support_mask]\r	
Indication	Succeeded: **\r\nAT-B SPRO 0\r\n** Failed: **\r\nAT-B SPRO 1\r\n**	
Parameters	profile_support_mask	bit 0: mode, 0:sink mode; bit 1: HFP or AG, 0: disable, 1:enable; bit 2: A2DP, 0: disable, 1:enable; bit 3: AVRCP, 0: disable, 1:enable; bit 4: PBAP Client, 0: disable, 1:enable; bit 5: MAP Client, 0: disable, 1:enable; bit 6: OPP Client, 0: disable, 1:enable; bit 7: OPP Server, 0: disable, 1:enable; bit 8: SPP, 0: disable, 1:enable; bit 9: PBAP Server, 0: disable, 1:enable; bit 10: HID mouse, 0: disable, 1:enable; bit 11: HID keyboard, 0: disable, 1:enable.
Note	1. When both bit10 and bit11 equal to 1, i40e will only support HID keyboard. 2. For instance, 02 represents HFP is enabled; 112 represents HFP, PBAP and SPP are enabled; 512 represents HFP, PBAP, SPP and HID mouse are enabled; 912 represents HFP, PBAP, SPP and HID keyboard are enabled; 400 represents HID mouse is enabled; 800 represents HID keyboard is enabled. 3. If either HID mouse or HID keyboard is enabled, the module will support HID streaming method. Under this circumstance, it doesn't need to send Keyboard/mouse input reports that are formatted as AT+B command, but it can directly send report contents. See chapter 20.4 and 20.5.	

5 Generic Indication Definition

5.1 INIT

The INIT indication is used to inform the host unit if the *Bluetooth* initialization is successfully completed.

Indication	**AT-B INIT [status]**	
Parameters	status	0: succeeded; 1: failed.
Note	N/A	

5.2 ROLE

The ROLE indication is used to inform the host of current role in the specific connection.

Indication	**AT-B ROLE [role]**	
Parameters	role	0: master; 1: slave; 2: role doesn't care.
Note	N/A	

6 HFP AT Command Definition

This chapter introduces the HFP (HF Unit Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

6.1 HFP Status

This chapter introduces the defined HFP status.

6.1.1 HFP Lib Status

```
typedef enum
{
    hfp_success = 0,                        /*! Success.*/
    hfp_fail = 0x01,                        /*! Failure.*/
    hfp_ag_failure= 0x02,                   /*! Failure - AG failure.*/
    hfp_no_connection_to_phone= 0x03,       /*! Failure - No connection to phone.*/
    hfp_operation_not_allowed= 0x04,        /*! Failure - Operation not allowed.*/
    hfp_operation_not_supported= 0x05,      /*! Failure - Operation not supported.*/
    hfp_ph_sim_pin_required= 0x06,          /*! Failure - PH-SIM PIN required.*/
    hfp_sim_not_inserted= 0x07,             /*! Failure - SIM not inserted.*/
    hfp_sim_pin_required= 0x08,             /*! Failure - SIM PIN required.*/
    hfp_sim_puk_required= 0x09,             /*! Failure - SIM PUK required.*/
    hfp_sim_failure= 0x0a,                  /*! Failure - SIM failure.*/
    hfp_sim_busy= 0x0b,                     /*! Failure - SIM busy.*/
    hfp_incorrect_password= 0x0c,           /*! Failure - Incorrect password.*/
    hfp_sim_pin2_required= 0x0d,            /*! Failure - SIM PIN2 required.*/
    hfp_sim_puk2_required= 0x0e,            /*! Failure - SIM PUK2 required.*/
    hfp_memory_full= 0x0f,                  /*! Failure - Memory full.*/
    hfp_invalid_index= 0x10,                /*! Failure - Invalid index.*/
    hfp_memory_failure= 0x11,               /*! Failure - Memory failure.*/
    hfp_text_string_too_long= 0x12,         /*! Failure - Text string too long.*/
    hfp_invalid_chars_in_text_string= 0x13, /*! Failure - Invalid characters in text string.*/
    hfp_dial_string_too_long= 0x14,         /*! Failure - Dial string too long.*/
    hfp_invalid_chars_in_dial_string= 0x15, /*! Failure - Invalid characters in dial string.*/
    hfp_no_network_service= 0x16,           /*! Failure - No network service.*/
    hfp_network_not_allowed= 0x17,          /*! Failure - Network not allowed, emergency calls only.*/
    hfp_timeout=0x1d,                       /*! Failure - Timed out waiting for AG response */
    hfp_network_no_carrier,                 /*! Failure – No Carrier */
    hfp_network_busy,                       /*! Failure - BUSY */
    hfp_network_no_answer,                  /*! Failure – NO ANSWER */
    hfp_network_delayed,                    /*! Failure - DELAYED */
    hfp_network_blacklisted                 /*! Failure - BLACKLISTED */
} hfp_lib_status;
```

6.1.2 HFP Connect Status

```
typedef enum
{
    hfp_connect_success,                    /*! Successful connection.*/
```

```
    hfp_connect_sdp_fail,                   /*! Unsuccessful due to a service search failure.*/
    hfp_connect_slc_failed,                 /*! Unsuccessful due to a service level connection failure.*/
    hfp_connect_failed_busy,                /*! Unsuccessful due to service level connection already
                                            established.*/
    hfp_connect_failed,                     /*! Unsuccessful due to RFCOMM connection failing to be
                                            established.*/
    hfp_connect_server_channel_not_registered,  /*! Unsuccessful due to attempt to
                                                connect to unallocated server
                                                channel.*/
    hfp_connect_timeout,                    /*! Unsuccessful due to connection attempt timing out.*/
    hfp_connect_rejected,                   /*! Unsuccessful due to remote device rejecting
                                            connection.*/
    hfp_connect_normal_disconnect,          /*! Unsuccessful due to remote device terminating the
                                             connection.*/
    hfp_connect_abnormal_disconnect,        /*! Unsuccessful due to an abnormal disconnect while
                                             establishing an rfcomm connection.*/
    hfp_connect_fail_bad_params,            /*! Connection failed due to bad parameters supplied by the
                                             application. */
} hfp_connect_status;
```

6.1.3 HFP Disconnect Status

```
typedef enum
{
    hfp_disconnect_success,                 /*! Successful disconnection.*/
    hfp_disconnect_link_loss,               /*! Unsuccessful due to abnormal link loss.*/
    hfp_disconnect_no_slc,                  /*! Unsuccessful due to no current connection.*/
    hfp_disconnect_timeout,                 /*! Unsuccessful due to RFCOMM connection attempt
                                            timeout.*/
    hfp_disconnect_error,                   /*! Unsuccessful due to RFCOMM connection attempt
                                            error.*/
} hfp_disconnect_status;
```

6.2 HFCONN

The HFCONN command is used to create a HFP connection with the remote device.

Command	**AT+B HFCONN [bd]**
Response	**AT-B HFCONN [status],[bd],[profile]**

Parameters	status	Values in the chapter 6.1.2.
	bd	Remote *Bluetooth* device address.
	profile	Profile type, where 0: Not HSP/HFP; 1: Headset Profile; 2: Hands-free Profile.
Note	bd is comprised of 12 bytes hexadecimal characters.	

6.3HFDISC

The HFDISC command is used to disconnect the HFP connection with the remote device.

Command	**AT+B HFDISC**	
Response	**AT-B HFDISC [status],[bd]**	
Parameters	status	Values in the chapter 6.1.3.
	bd	Remote *Bluetooth* device address.
Note	N/A	

6.4HFANSW

The HFANSW command is used to answer the incoming call.

Command	**AT+B HFANSW**	
Response	**AT-B HFANSW [status]**	
Parameters	status	Values in the chapter 6.1.1.
Note	N/A	

6.5HFCHUP

The HFCHUP command is used to reject the incoming call, hang up the active call or cancel

the dialing out call.

Command	**AT+B HFCHUP**	
Response	**AT-B HFCHUP [status]**	
Parameters	status	Values in the chapter 6.1.1.
Note	N/A	

6.6HFDIAL

The HFDIAL command is used to dial a phone number, for Hands-Free profile only.

Command	**AT+B HFDIAL[type], [num]**	
Response	**AT-B HFDIAL [status],[type]**	
Parameters	type	Call type, where 0: dial the supplied number; 1: perform a last number redial.
	num	The dialed out number. The maximum length is 40 bytes.
	status	Values in the chapter 6.1.1.
Note	N/A	

6.7HFDTMF

The HFDTMF command is used to transmit a DTMF code to the AG, for Hands-Free profile only.

Command	**AT+B HFDTMF [key]**	
Response	**AT-B HFDTMF [status]**	
Parameters	key	DTMF key, including “0-9”, A, B, C, D, *, #.
	status	Values in the chapter 6.1.1.
Note	N/A	

6.8HFCTRS

The HFCTRS command is used to transfer audio from/to remote when a call is ongoing.

Command	**AT+B HFCTRS**	
Response	**AT-B HFCTRS [status]**	
Parameters	status	Values in the chapter 6.1.1.
Note	The host unit will receive the audio connection on/off indication when this command is successfully executed.	

6.9HFMCAL

The HFMCAL command is used to change three-way calling status (active or held). This command will be successfully executed when the HFP instance is in either hsActiveCall or hsTWCallWaiting status.

Command	**AT+B HFMCAL [op]**	
Response	**AT-B HFMCAL [status],[op]**	
Parameters	op	Operation code, where 0: MultipleCallsReleaseHeldOrRejectWaiting; 1: MultipleCallsReleaseActiveAcceptOther; 2: MultipleCallsHoldActiveAcceptOther.
	status	Values in the chapter 6.1.1
Note	N/A	

6.10 HFCLCC

The HFCLCC command is used to get current calls list of AG side, for Hands-free profile only.

Command	**AT+B HFCLCC**
Response	If one or more current calls found: **AT-B HFCCIN [status],[call_idx],[direction],**

	[mode],[multiparty],[number_type],[number] When the command finished: **AT-B HFCLCC [status]**	
Parameters	call_idx	Call index, defined by AG.
	direction	AG originated call indicator, where 0: Call from AG to network; 1: Call from network to AG.
	status	Call status, where 0: Call is currently active; 1: Call is currently held; 2: Call is being dialed - mobile originated only; 3: Call is alerting - mobile originated only; 4: Call is incoming - mobile terminated only; 5: Call is waiting - mobile terminated only.
	mode	Call mode, where 0: voice call; 1: data call; 2: fax call.
	multiparty	Call multiparty indicator, where 0: Call is not multiparty; 1: Call is multiparty.
	number_type	Number type, where 0: Type of number is unknown; 1: Number is an international number; 2: Number is a national number; 3: Number is a network specific number; 4: Number is a dedicated access, short code.
	number	Phone number

	status	Values in the chapter 6.1.1.
Note	N/A	

6.11 HFSVGS

The HFSVGS command is used to send speaker volume to AG side, for BT module part, the speaker gain is also changed.

Command	**AT+B HFSVGS [vol]**	
Response	**AT-B HFSVGS [status],[vol]**	
Parameters	vol	Speaker volume, where ranges from 0 to 15.
	status	Values in the chapter 6.1.1
Note	If the connection is a HSP SLC, this command can be sent when the audio connection is ongoing. If the connection is a HFP SLC, this command can be sent when the connection status equals to or greater than hsConnected.	

6.12 HFSVGM

The HFSVGM command is used to send microphone volume to AG side, for BT module part, the microphone gain is also changed.

Command	**AT+B HFSVGM [vol]**	
Response	**AT-B HFSVGM [status],[vol]**	
Parameters	vol	Microphone volume, where ranges from 0 to 15.
	status	Values in the chapter 6.1.1.
Note	If the connection is a HSP SLC, this command can be sent when the audio connection is ongoing. If the connection is a HFP SLC, this command can be sent when the connection status equals to or greater than hsConnected.	

6.13 HFMUTE

The HFMUTE command is used to mute or unmute the microphone when a call is ongoing. When an audio connection is established, the default setting is MIC muted.

Command	**AT+B HFMUTE [op]**	
Response	**AT-B HFMUTE [status]**	
Parameters	op	0: unmute; 1: mute.
	status	0: succeeded; 1: failed.
Note	N/A	

6.14 HFSCFG

The HFSCFG command is used to enable/disable the reconnect function, and enable/disable local ring tone.

Command	**AT+B HFSCFG [mask],[config]**	
Response	**AT-B HFSCFG [status]**	
Parameters	mask	1: enable/disable the reconnect function; 2: enable/disable local ring tone.
	config	When mask=1, if config=0: disable the reconnect function; if config=1, enable the reconnect function. When mask=2, if config=0: enable local ring tone; if config=1: disable local ring tone.
	status	0: succeeded; 1: failed.
Note	N/A	

6.15 HFGCFG

The HFGCFG command is used to query if the reconnect function is enabled or disabled, and if the local ring tone is used.

Command	**AT+B HFGCFG [mask]**	
Response	**Succeeded: AT-B HFGCFG 0,[config]** **Failed: AT-B HFGCFG 1,0**	
Parameters	mask	1: query if the reconnect function is enabled or disabled; 2: query if the local ring tone is used.
	config	When mask=0, if config=0: the reconnect function is disabled; if config=1, the reconnect function is enabled. When mask=1, if config=0: local ring tone is used; if config=1: local ring tone isn't used.
Note	N/A	

7 HFP Indication Definition

This chapter introduces the HFP (HF Unit Role) relevant indications' definition.

7.1 HFSTAT

The HFSTAT indication is used to inform the host unit when the HFP status is changed.

Indication	**AT-B HFSTAT [state]**	
Parameters	state	HFP status, where 1: HfpTLReady; 2: HfpTLSlcConnecting; 3: HfpTLSlcConnected; 4: HfpTLIncomingCallEstablish; 5: HfpTLOutgoingCallEstablish;

		6: HfpTLActiveCall; 7: HfpTLTWCalling.
Note	N/A	

7.2HFCONN

The HFCONN indication is used to inform the host unit when it initializes a HFP connection with the local device.

Indication	**AT-B HFCONN [status],[bd], [profile]**	
Parameters	status	Values in the chapter 6.1.2.
	bd	Remote *Bluetooth* device address
	profile	Profile type, where 0: Not HSP/HFP; 1: Headset Profile; 2: Hands-free Profile.
Note	N/A	

7.3HFDISC

The HFDISC indication happens when the remote device disconnect the HFP connection.

Indication	**AT-B HFDISC [status],[bd]**	
Parameters	status	Values in the chapter 6.1.3
	bd	Remote *Bluetooth* device address
Note	N/A	

7.4HFRING

The HFRING indication is used to inform the host unit when HFP ring comes. The host unit shall turn on the audio path when receives this indication, and turn off the audio path when the HFP status changes HfpTLSlcConnected.

Indication	**AT-B HFRING**
Parameters	N/A
Note	N/A

7.5HFIBRN

The HFIBRN indication is used to inform the host unit that HFP in-band ring feature turns on or off.

Indication	**AT-B HFIBRN [inbandring]**	
Parameters	inbandring	0: in-band ring tone is off; 1: in-band ring tone is on.
Note	N/A	

7.6HFAUDIO

The HFAUDIO indication is used to inform the host unit that the HFP audio connection is on or off.

Indication	**AT-B HFAUDIO [op]**	
Parameters	op	0: the HFP audio connection is off; 1: the HFP audio connection is on.
Note	N/A	

7.7HFCLIP

The HFCLIP indication is used to inform the host unit of the incoming call's caller ID.

Indication	**AT-B HFCLIP [callerid]**	
Parameters	callerid	Incoming call's caller ID. There is no limit to number length on HF device side, and it depends on mobile phone side, but we had better limit it to less than 40 bytes.

Note	N/A

7.8HFCCWA

The HFCCWA indication is used to inform the host unit when the second incoming call's caller ID.

Indication	**AT-B HFCCWA [callerid]**	
Parameters	callerid	Incoming call's caller ID. There is no limit to number length on HF device side, and it depends on mobile phone side, but we had better limit it to less than 40 bytes.
Note	N/A	

7.9HFNUML

The HFNUML indication is used to inform the host unit of the subscriber number of the AG side when the SLC connection is established.

Indication	**AT-B HFNUML [number]**	
Parameters	number	The subscriber number of the AG side. There is no limit to number length on HF device side, and it depends on mobile phone side, but we had better limit it to less than 40 bytes.
Note	N/A	

7.10 HFNUMC

The HFNUMC indication is used to inform the host that the query of subscriber number is completed.

Indication	**AT-B HFNUMC [status]**	
Parameters	status	Values in the chapter 6.1.1.
Note	N/A	

7.11 HFSGNL

The HFSGNL indication is used to inform the host unit of the signal strength of the AG side.

Indication	**AT-B HFSGNL [signal]**	
Parameters	signal	Signal strength indicator, where ranges from 0 to 5.
Note	N/A	

7.12 HFROAM

The HFROAM indication is used to inform the host unit of the roaming status of the AG side.

Indication	**AT-B HFROAM [roam]**	
Parameters	roam	Roaming status indicator, where: 0: roaming is not active; 1: roaming is active.
Note	N/A	

7.13 HFBATC

The HFBATC indication is used to inform the host unit of the battery charger status of the AG side.

Indication	**AT-B HFBATC [battchg]**	
Parameters	battchg	Battery charge indicator of AG, where ranges from 0 to 5.
Note	N/A	

7.14 HFVGSI

The HFVGSI indication is used to inform the host unit of the current speaker volume of the AG side.

Indication	**AT-B HFVGSI [spkvol]**

Parameters	spkvol	Speaker volume, where ranges from 0 to 15.
Note	N/A	

7.15 HFVGMI

The HFVGMI indication is used to inform the host unit of the current microphone volume of the AG side.

Indication	**AT-B HFVGMI [micvol]**	
Parameters	micvol	Microphone volume, where ranges from 0 to 15.
Note	N/A	

7.16 HFSRVC

This indication is used to inform the host unit of a change in the service indicator's status.

Indication	**AT-B HFSRVC [service]**	
Parameters	service	The new value of the service indicator.
Note	N/A	

7.17 HFCHLD

This indication is used to inform the host unit of the call held status of AG side.

Indication	**AT-B HFCHLD [callheld]**	
Parameters	callheld	*Bluetooth* proprietary call hold status indicator. Support for this indicator is mandatory for the AG, optional for the HF. Possible values are as follows: 0: No calls held; 1: Call is placed on hold or active/held calls swapped (The AG has both and active AND a held call); 2: Call on hold, no active call.

Note	N/A

8 A2DP Sink AT Command Definition

This chapter introduces A2DP (Sink Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

8.1A2DP Status

```
typedef enum
{
    a2dp_success,                   /*! The operation succeeded. */
    a2dp_invalid_parameters,        /*! Invalid parameters supplied by the client. */
    a2dp_sdp_fail,                  /*! SDP registration has failed. */
    a2dp_l2cap_fail,                /*! L2CAP registration has failed. */
     a2dp_operation_fail,           /*! The operation has failed. */
     a2dp_insufficient_memory,      /*! No memory to perform the required task. */
     a2dp_wrong_state,              /*! The library is in the wrong state to perform the operation. */
     a2dp_no_signalling_connection, /*! No signaling connection. */
     a2dp_no_media_connection,      /*! No media connection. */
     a2dp_rejected_by_remote_device,/*! Was rejected by the remote device. */
     a2dp_disconnect_link_loss,     /*! Link loss occurred. */
     a2dp_closed_by_remote_device,  /*! Closed by remote device. */
     a2dp_aborted                   /*! Connection was aborted. */
} a2dp_status_code;
```

8.2A2DPCONN

The A2DPCONN command is used to establish a A2DP connection with a remote device.

Command	**AT+B A2DPCONN [bd]**	
Response	**AT-B A2DPCONN [status],[bd]**	
Parameters	bd	Remote *Bluetooth* device address.
	status	Values in the chapter 8.1.

Note	bd is comprised of 12 bytes hexadecimal characters.

8.3A2DPDISC

The A2DPDISC command is used to release the A2DP connection with a remote device.

Command	**AT+B A2DPDISC**	
Response	**AT-B A2DPDISC [status],[bd]**	
Parameters	status	Values in the chapter 8.1
	bd	Remote *Bluetooth* device address
Note	N/A	

8.4A2DPSVGS

The A2DPSVGS command is used to change the speaker gain.

Command	**AT+B A2DPSVGS [gain]**	
Response	Succeeded: **AT-B A2DPSVGS 0** Failed: **AT-B A2DPSVGS 1**	
Parameters	gain	Speaker gain, where ranges from 0 to 15.
Note	N/A	

9 A2DP Sink Indication Definition

This chapter introduces the A2DP (Sink Role) relevant indications' definition.

9.1A2DPSTAT

The A2DPSTAT indication is used to inform the host unit when the A2DP sink's is changed.

Indication	**AT-B A2DPSTAT [state]**

Parameters	state	A2DP connection status, where 1: a2dpReady; 2: a2dpConnecting; 3: a2dpConnected; 4: a2dpStreaming.
Note	N/A	

9.2A2DPCONN

The A2DPCONN indication is used to inform the host unit when it initializes a A2DP connection with the local device.

Indication	**AT-B A2DPCONN [status], [bd]**	
Parameters	status	Values in the chapter 8.1.
	bd	Remote *Bluetooth* device address.
Note	bd is comprised of 12 bytes hexadecimal characters.	

9.3A2DPAUDIO

The A2DPAUDIO indication is used to inform the host unit that the A2DP audio connection is on or off.

Indication	**AT-B A2DPAUDIO [op]**	
Parameters	op	0: the A2DP audio connection is off; 1: the A2DP audio connection is on.
Note	N/A	

10 AVRCP Controller AT command Definition

This chapter introduces AVRCP (Controller Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

10.1 AVRCP Status

```
typedef enum
{
    avrcp_success = (0),                    /*! Operation was successful. */
    avrcp_fail,                             /*! Operation failed. */
    avrcp_no_resource,                      /*! Not enough resources. */
    avrcp_bad_state,                        /*! Request is not supported in the current state. */
    avrcp_timeout,                          /*! Operation timed out before completion. */
    avrcp_device_not_connected,             /*! Device specified is not connected. */
    avrcp_busy,                             /*! Operation is already in progress. */
    avrcp_unsupported,                      /*! Requested operation is not supported. */
    avrcp_invalid_sink,                     /*! Sink supplied was invalid. */
    avrcp_link_loss,                        /*! Link loss occurred. */
    avrcp_rejected=0x0A,                    /*! The operation was rejected. */
    avrcp_interim_success=0x0F,             /*! Operation was successful, but have only received an interim
                                            response.*/

/* Below status codes depends on the error status code received from the remote device. Retain the same
values while inserting new values or modifying this enum */
    avrcp_rejected_invalid_pdu = 0x80,      /*! The operation was rejected with reason - invalid PDU. */
    avrcp_rejected_invalid_param,           /*! The operation was rejected with reason - invalid parameter. */
    avrcp_rejected_invalid_content,         /*! The operation was rejected with reason - invalid content. */
    avrcp_rejected_internal_error,          /*! The operation was rejected with reason - internal error. */
    avrcp_rejected_uid_changed = 0x85,      /*! The operation was rejected with reason - UID Changed. */
    avrcp_rejected_invalid_direction = 0x87, /*! The command has been rejected with reason -Invalid
                                            Direction.*/
    avrcp_rejected_not_directory,           /*! The command has been rejected with reason -Not a
                                            Directory.*/
    avrcp_rejected_uid_not_exist,           /*! The command has been rejected with reason -Does not
                                            exist.*/
    avrcp_rejected_invalid_scope,           /*! The command has been rejected with reason -Invalid Scope.*/
    avrcp_rejected_out_of_bound,            /*! The command has been rejected with reason - Range Out of
                                            Bounds.*/
    avrcp_rejected_uid_directory,           /*! The command has been rejected with reason - UID is a
                                            Directory.*/
    avrcp_rejected_media_in_use,            /*! The command has been rejected with reason - Media in
                                            Use.*/
    avrcp_rejected_play_list_full,          /*! The command has been rejected with reason - Now Playing
                                            List Full.*/
    avrcp_rejected_search_not_supported,    /*! The command has been rejected with reason - Search Not
                                            Supported.*/
```

```
    avrcp_rejected_search_in_progress,      /*! The command has been rejected with reason - Search in
                                            Progress.*/
    avrcp_rejected_invalid_player_id,       /*! This command has been rejected with reason - Invalid Player
                                            ID.*/
    avrcp_rejected_player_not_browsable,    /*! This command has been rejected with reason - Player Not
                                            Browsable.*/
    avrcp_rejected_player_not_addressed,    /*! This command has been rejected with reason - Player Not
                                            Addressed.*/
    avrcp_rejected_no_valid_search_results, /*! This command has been rejected with reason - No valid
                                            Search Results.*/
    avrcp_rejected_no_available_players,    /*! This command has been rejected with reason - No available
                                            players.*/
    avrcp_rejected_addressed_player_changed,    /*! This command has been rejected with reason -
                                                Addressed Player Changed.*/
    avrcp_status_guard_reserverd = 0xFF     /* Dummy Place Holder */
} avrcp_status_code;
```

10.2 AVRCPPLAY

The AVRCPPLAY command is used to send “Play” command to start playing.

Command	**AT+B AVRCPPLAY**	
Response	**AT-B AVRCPPLAY [status]**	
Parameters	status	Values in the chapter 10.1.
Note	N/A	

10.3 AVRCPPAUSE

The AVRCPPAUSE command is used to send “Pause” command to pause playing track.

Command	**AT+B AVRCPPAUSE**	
Response	**AT-B AVRCPPAUSE [status]**	
Parameters	status	Values in the chapter 10.1.
Note	N/A	

10.4 AVRCPSTOP

The AVRCPSTOP command is used to send “Stop” command to stop playing.

Command	**AT+B AVRCPSTOP**	
Response	**AT-B AVRCPSTOP [status]**	
Parameters	status	Values in the chapter 10.1.
Note	N/A	

10.5 AVRCPFORWARD

The AVRCPFORWARD command is used to send “Forward” command to play the previous track.

Command	**AT+B AVRCPFORWARD**	
Response	**AT-B AVRCPFORWARD [status]**	
Parameters	status	Values in the chapter 10.1.
Note	N/A	

10.6 AVRCPBACKWARD

The AVRCPBACKWARD command is used to send “Backward” command to play the next track.

Command	**AT+B AVRCPBACKWARD**	
Response	**AT-B AVRCPBACKWARD [status]**	
Parameters	status	Values in the chapter 10.1.
Note	N/A	

10.7 AVRCPVOLUMEUP

The AVRCPVOLUMEUP command is used to send the Category 2 Pass through command of

volume-up.

Command	**AT+B AVRCPVOLUMEUP**	
Response	**AT-B AVRCPVOLUMEUP [status]**	
Parameters	status	Values in the chapter 10.1
Note	This command is only used for Category 2 device.	

10.8 AVRCPVOLUMEDOWN

The AVRCPBACKWARD command is used to end the Category 2 Pass through command of volume-down.

Command	**AT+B AVRCPVOLUMEDOWN**	
Response	**AT-B AVRCPVOLUMEDOWN [status]**	
Parameters	status	Values in the chapter 10.1
Note	This command is only used for Category 2 device.	

10.9 AVRCPSABSVOL

The AVRCPBACKWARD command is used by the CT (Category 2) to set the absolute volume at category 2 TG.

Command	**AT+B AVRCPSABSVOL [volume]**	
Response	**AT-B AVRCPSABSVOL [status]**	
Parameters	volume	Absolute volume, where ranges from 0 to 0x7F
	status	Values in the chapter 10.1
Note	This command is only used for Category 2 device.	

11 AVRCP Controller Indication Definition

This chapter introduces the AVRCP(Controller Role) relevant indications' definition.

11.1 AVRCPSTAT

The AVRCPSTAT indication is used to inform the host unit when the AVRCP Controller's is changed.

Indication	**AT-B AVRCPSTAT [state]**	
Parameters	status	AVRCP connection status, where, 1: avrcpReady; 2: avrcpConnecting; 3: avrcpConnected.
Note	N/A	

11.2 AVRCPCONN

The AVRCPCONN indication happens when local or remote device creates the A2DP connection.

Indication	**AT-B AVRCPCONN [status],[bd]**	
Parameters	status	Values in the chapter 10.1.
	bd	Remote *Bluetooth* device address.
Note	AVRCP connection will be established after A2DP connection has been created	

11.3 AVRCPDISC

The AVRCPDISC indication happens when the local or remote device disconnects the AVRCP connection.

Indication	**AT-B AVRCPDISC [status], [bd]**

Parameters	status	Values in the chapter 10.1.
	bd	Remote *Bluetooth* device address.
Note	AVRCP connection will be disconnected after A2DP connection has been disconnected	

11.4 AVRCPTITLE

The AVRCPTITLE indication is used to tell host the title of current playing media.

Indication	**AT-B AVRCPTITLE [title]**	
Parameters	title	Title of media, the maximum length is 128 bytes.
Note	N/A	

11.5 AVRCPARTIST

The AVRCPARTIST indication is used to tell host the artist of current playing media.

Indication	**AT-B AVRCPARTIST [artist]**	
Parameters	artist	Artist of media, the maximum length is 128 bytes.
Note	N/A	

11.6 PLAYSTATUS

The PLAYSTATUS indication is used to tell host the playback status has changed.

Indication	**AT-B PLAYSTATUS [status]**	
Parameters	**status**	0x00: play_status_stopped; 0x01: play_status_playing; 0x02: play_status_paused; 0x03: play_status_fwd_seek; 0x04: play_status_rev_seek; 0xFF: play_status_error.

Note	N/A

11.7 AVRCPFEATURE

The AVRCPFEATURE indication is used to tell host the features that the remote TG supports.

Indication	**AT-B AVRCPFEATURE [metadata],[feature]**	
Parameters	metadata	0: disable, AVRCP V1.0; 1: enable, AVRCP V1.3 or later.
	feature	The features supported by the remote TG, where, Bit 0: Category 1; Bit 1: Category 2; Bit 2: Category 3; Bit 3: Category 4; Bit 4: Player Application Settings. Bit 0 should be set for this bit to be set; Bit 5: Group Navigation. Bit 0 should be set for this bit to be set; Bit 6: Supports browsing; Bit 7: Supports multiple media player applications; Bit 8-15: Reserved for Future Additions; The bits for supported categories are set to 1. Others are set to 0.
Note	N/A	

12 PBAP Client AT Command Definition

This chapter introduces PBAP (Phone Book Client Equipment Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

12.1 PBAP Status

```
typedef enum
{
    pbapc_success,                          /*! Last operation was successful. */
    pbapc_failure,                          /*! Last operation failed. */
    pbapc_aborted,                          /*! Last operation was aborted. */
    pbapc_not_idle,   /*! Client is not idle, so cannot perform the current operation. */
    pbapc_wrong_state,      /*! Operation failed due to being in the wrong state.*/
    pbapc_sdp_failure_resource, /*! Unable to register the SDP record due to a lack of resources */
    pbapc_sdp_failure_bluestack,        /*! Unable to register the SDP record due to Bluestack */
    pbapc_remote_disconnect,    /*! Remote host has disconnected or the link has been lost. */
    pbapc_spb_unauthorised = 0x10,          /*! Not authorised to access this phonebook */
    pbapc_spb_no_repository,                /*! The server does not contain this repository */
    pbapc_spb_not_found,                    /*! Phonebook does not exist */
    pbapc_vcl_no_param_resources = 0x20,   /*! No resources to generate application specific parameters
                                            header for PullvCardList. */
    pbapc_vcl_no_pbook_folder,              /*! A phonebook folder was specified for PullvCardList where
                                            there are no sub-folders (i.e. in pb). */
     pbapc_vcl_invalid_pbook,                   /*! A phonebook folder was specified for PullvCardList
                                                which is invalid */
     pbapc_vce_no_param_resources = 0x30,               /*! No resources to generate application specific
                                                        parameters header for PullvCardEntry. */
     pbapc_vce_no_name_resources,                       /*! No resources to generate the vCard entry
                                                        name for PullvCardEntry. */
     pbapc_vce_invalid_entry,                           /*! Invalid entry for this phonebook for
                                                        PullvCardEntry. Only folder 'pb' can contain an
                                                        entry 0. */
     pbapc_ppb_no_param_resources = 0x40,               /*! No resources to generate application specific
                                                        parameters header for PullPhonebook. */
     pbapc_ppb_no_name_resources,                       /*! No resources to generate the phonebook
                                                        name for PullPhonebook. */
     pbapc_ppb_no_required_name,                        /*! No name for PullPhonebook when it is
                                                        required. e.g. server is not in a phonebook
                                                        directory */
     pbapc_ppb_no_repository,                   /*! The server does not contain this repository */
     pbapc_prop_sdp_error,                      /*! Request to get the server properties failed due to an
                                                SDP error */
     pbapc_end_of_status_list
} phone_book_status;
```

12.2 PBCCONN

The PBCCONN command is used to establish a PBAP connection with a remote device.

Command	**AT+B PBCCONN [bd]**	
Response	**AT-B PBCCONN [status],[bd]**	
Parameters	bd	Remote *Bluetooth* device address.
	status	Values in the chapter 12.1.
Note	HFP connection must have already been established before establishing PBAP connection.	

12.3 PBCDISC

The PBCDISC command is used to release the PBAP connection with a remote device.

Command	**AT+B PBCDISC [bd]**	
Response	**AT-B PBCDISC [status], [bd]**	
Parameters	bd	Remote *Bluetooth* device address.
	status	Values in the chapter 12.1.
Note	N/A	

12.4 PBCPULLPB

The PBCPULLPB command is used to start pulling the phonebook object from the remote device.

Command	**AT+B PBCPULLPB [repository],[folder],[maxList],[startOffset]**	
Response	Succeeded: **AT-B PBCPULLDATAIND [pbSize], [moreData], [length],[packet]0xFF** Failed: **AT-B PBCPULLPB 1**	
Parameters	repository	1: local;

		2: SIM card.
	folder	1: pb, main phone book; 2: ich,incoming calls; 3: och,outgoing calls; 4: mch,missed calls; 5: cch,combination of ich, och and mch.
	maxList	Maximum number of entries that PCE can handle.
	startOffset	Offset of first entry to pull
	pbSize	Number of entries interested
	moreData	More data to receive or not. More(TRUE) or not(FALSE)
	length	Length of the packet.
	packet	Data of packet.
Note	1. when maxList = 0, it can be used to get the maximum index that are actually used. In this case, all other parameters are ignored; 2. when maxList = 65535, it can be used to download all entries without knowing pbsize; 3. End of packet is 0xFF, not \r (0x0d).	
Example	**at+b pbcpullpb 1,1,0,0\CRAT-B PBCSTAT 4\CRAT-B PBCPULLDATAIND 40,0,0,\CRAT-B PBCSTAT 3\CR** **at+b pbcpullcmt\CRAT-B PBCPULLCMTIND** **at+b pbcpullpb 2,1,20,0\CRAT-B PBCSTAT 4\CRAT-B PBCPULLDATAIND 0,1,884……\CR** **at+b pbcpullcont\CRAT-B PBCSTAT 4\CRAT-B PBCPULLDATAIND 0,1,884……\CR** **……** **at+b pbcpullcont\CRAT-B PBCSTAT 4\CRAT-B PBCPULLDATAIND 0,0,420……\CR** **at+b pbcpullcmt\CRAT-B PBCPULLCMTIND\CR** **at+b pbcpullpb 2,1,20,20\CRAT-B PBCSTAT 4\CRAT-B PBCPULLDATAIND**	

	0,1,884……\CR …… at+b pbcpullcont\CRAT-B PBCSTAT 4\CRAT-B PBCPULLDATAIND 0,0,420……\CR at+b pbcpullcmt\CRAT-B PBCPULLCMTIND\CR

12.5 PBCPULLCONT

The PBCPULLCONT command is used to get more dada for the ongoing pull operation.

Command	**AT+B PBCPULLCONT**	
Response	Succeeded: **AT-B PBCPULLDATAIND [pbSize], [moreData], [length],[packet]0xFF** Failed: **AT-B PBCPULLCONT 1**	
Parameters	pbSize	Ignored
	moreData	More data to receive or not. More(TRUE) or not(FALSE)
	length	Length of the packet
	packet	Data of packet
Note	This command shall be used only if the [moreData] field of the "PBCPULLDATAIND" indication equals to one.	

12.6 PBCPULLCRT

The PBCPULLCRT command is used to get the current processed packet.

Command	**AT+B PBCPULLCRT**	
Response	Succeeded: **AT-B PBCPULLDATAIND [pbSize], [moreData], [length],[packet]0xFF** Failed: **AT-B PBCPULLCRT 1**	
Parameters	pbSize	Ignored
	moreData	More data to receive or not. More(TRUE) or not(FALSE)

	length	Length of the packet
	packet	Data of packet
Note	1. If the [length] field of the PBCPULLDATAIND indication is not equal to the real packet received by MCU, it is possible that packet loss has occurred on UART. In this situation, this command can be used to retransmit the packet. 2. End of packet is 0xFF, not \r (0x0d).	

12.7 PBCPULLCMT

The PBCPULLCMT command is used to get the current processed packet.

Command	**AT+B PBCPULLCMT**
Response	Succeeded: **AT-B PBCPULLCMTIND** Failed: **AT-B PBCPULLCMT 1**
Parameters	N/A
Note	N/A

13 PBAP Client Indication Definition

This chapter introduces the PBAP(Phone Book Client Equipment Role) relevant indications' definition.

13.1 PBCSTAT

The PBCSTAT indication is used to inform the host unit that the PBAP client's status is changed.

Indication	**AT-B PBCSTAT [state]**	
Parameters	state	Phonebook connection status, where, 1: pbapcReady; 2: pbapcConnecting;

		3: pbapcConnected; 4: pbapcDownloading; 5: pbapcDisconnecting.
Note	N/A	

13.2 PBPULLDATAIND

The PBPULLDATAIND indication is used to inform the host unit that packet pulled arrives.

Indication	**AT-B PBPULLDATAIND [pbSize], [moreData], [length],[packet]0xFF**	
Parameters	pbSize	Number of entries interested
	moreData	More data to receive or not. More(TRUE) or not(FALSE)
	length	Length of the packet
	packet	Data of packet
Note	End of packet is 0xFF, not \r (0x0d).	

13.3 PBCPULLCMTIND

The PBCPULLCMTIND indicates that the current pull operation has completed.

Indication	**AT-B PBCPULLCMTIND**
Parameters	N/A
Note	The client host shall use the “PBCPULLCMT” command to complete the current PULL operation every time when the [moreData] field of the PBCPULLDATATIND indication equals to zero.

14 MAP Client AT Command Definition

This chapter introduces MAP (Message Client Equipment Role) relevant AT commands’ definition, including a brief description of commands’ syntax, responses and examples.

14.1 MAP Status

This chapter introduces the defined MAP status.

14.1.1MAP Status

```
typedef enum mapc_status
{
    mapc_success,              /*!< The last operation was successful. */
    mapc_failure,              /*!< General failure */
    mapc_pending,              /*!< The operation is pending or in progress */
    mapc_connect_rejected,     /*!< The Connection has been rejected locally */
    mapc_aborted,              /*!< The operation has been aborted locally */
    mapc_invalid_state,        /*!< Command not acceptable in this state */
    mapc_mns_started,          /*!< The MNS Service was already started. */
    mapc_object_not_found,     /*!< Remote file or folder not found */
    mapc_object_protected,     /*!< Access denied to the remote object */
    mapc_command_rejected,     /*!< Remote rejected the command */
    mapc_invalid_parameter     /*!< Remote send corrupt or invalid response */
} MapcStatus;
```

14.1.2MAP Client Message Filter

```
typedef enum
{
    mapc_no_filtering = 0x00,          /*!< No filtering */
    mapc_filter_out_sms_gsm = 0x01,    /*!< Filter out GSM SMS */
    mapc_filter_out_sms_cdma= 0x02,    /*!< Filter out GSM CDMA */
    mapc_filter_out_email = 0x04,      /*!< Filter out EMAIL */
    mapc_filter_out_mms = 0x08,        /*!< Filter out    MMS */
    mapc_filter_unread = 0x10,         /*!< Get only the unread messages */
    mapc_filter_read = 0x20,           /*!< Get only the read messages */
    mapc_filter_params = 0x100         /*!< Use an auto filter for Param Mask*/
} MapcMessageFilter;
```

14.2 MAPCCONN

The MAPCCONN command is used to create an MAP connection with the remote MAP server.

Command	**AT+B MAPCCONN [bd]**	
Response	**AT-B MAPCCONN [status], [bd]**	
Parameters	bd	Remote *Bluetooth* device address
	status	Values in the chapter 14.1.1.
Note	N/A	

14.3 MAPCDISC

The MAPCDISC command is used to disconnect the existing MAP connection.

Command	**AT+B MAPCDISC**	
Response	**AT-B MAPCDISC [status],[bd]**	
Parameters	status	Values in the chapter 14.1.1
	bd	Remote *Bluetooth* device address
Note	If MAP client is in UPLOAD/DOWNLOAD operation, the MAPCCMT command shall be used to terminate the current operation before using MAPCDISC to disconnect the existing MAP connection.	

14.4 MAPCGETML

This command is used by the MAP Client to get message listing objects from the MAP Server.

Command	**AT+B MAPCGETML [folder],[maxList],[startOffset]**	
Response	Succeeded: **AT-B MAPCGETDATAIND [listSize],[moreData],[length],[packet]** Failed: **AT-B MAPCGETML 1**	
Parameters	folder	0: inbox; 1: outbox; 3: sent; 4: deleted; 5: draft.

	maxList	The maximum number of messages to be listed (0-65535).
	startOffset	Offset of the first entry to be listed.
	listSize	Size of available messages.
	moreData	More data to be received or sent. More(TRUE) or not(FALSE)
	length	The length of the packet field.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
	folder	0: inbox; 1: outbox; 3: sent; 4: deleted; 5: draft.
Note	If MAP client is in UPLOAD/DOWNLOAD operation, the MAPCCMT command shall be used to terminate the current operation before using the MAPCDISC command to disconnect the existing MAP connection.	

14.5 MAPCGETCONT

The MAPCGETCONT command is used to get more dada for the ongoing get operation.

Command	**AT+B MAPCGETCONT**	
Response	Succeeded: **AT-B MAPCGETDATAIND [listSize], [moreData], [length],[packet]** Failed: **AT-B PBCPULLCONT 1**	
Parameters	listSize	Size of available messages.
	moreData	More data to be received or sent. More(TRUE) or not(FALSE)
	length	The length of the packet field
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.

Note	This command shall be used only if the [moreData] field of the MAPCGETDATAIND indication equals to one.

14.6 MAPCGETMSG

This command is used by the MAP Client to get the packet of the message object with the [handle] field from the MAP Server.

Command	**AT+B MAPCGETMSG [handle]**	
Response	Succeeded: **AT-B MAPCGETDATAIND [moreData],[packetSize],[packet]** Failed: **AT-B MAPCGETMSG 1**	
Parameters	handle	You should get this parameter by parsing the Message Listing object
	moreData	More data to be received or sent. More(TRUE) or not(FALSE)
	length	The length of the packet field
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
Note	N/A	

14.7 MAPCGETCRT

The MAPCGETCRT command is used by the MAP client to get the previous indication when the MAP client has received the MAPCGETDATAIND indication.

Command	**AT+B MAPCGETCRT**	
Response	Succeeded: **AT-B MAPCGETCRTIND [moredata],[packetSize],[packet]** Failed: **AT-B MAPCGETCRT 1**	
Parameters	moreData	More data to be received or sent. More(TRUE) or not(FALSE)
	length	The length of the packet field
	packet	The partial or complete packet of an object, cannot be

		NULL, cannot include '\r'.
Note	1. If the [length] field of the MAPCGETDATAIND indication is not equal to the real packet received by MCU, it is possible that packet loss has occurred on UART. In this situation, this command can be used to retransmit the packet. 2. This command can be used to get the current packet before receiving the MAPCGETCMTIND indication, only for getting message listing function.	

14.8 MAPCPUSHMSG

The MAPCPUSHMSG command is used by the client host to put a message to remote MAP server.

Command	**AT+B MAPCPUTMSG [moreData],[packetSize],[packet]**	
Response	If this is the only packet to be sent, the response will be: **AT-B MAPCPUTCMTIND [status]** If there are more packets to be sent, the response will be: **AT-B MAPCPUTMSGIND** Failed: **AT-B MAPCPUTMSG 1**	
Parameters	moreData	More data to be received or sent. More(TRUE) or not(FALSE)
	length	The length of the packet field
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
Note	Maximum length of packet is 128 bytes.	

14.9 MAPCCMT

The MAPCCMT command is used to terminate the ongoing get/push operation.

Command	**AT+B MAPCCMT**
Response	If in get operation, the response is: **AT-B MAPCGETCMTIND** If in push operation, the response is: **AT-B MAPCPUSHCMTIND**

	Failed: **AT-B MAPCCMT 1**
Parameters	N/A
Note	Before receiving MAPCGETCMTIND or MAPCPUSHCMTIND indications, this command can be used to terminate the current operation.

15 MAP Client Indication Definition

This chapter introduces the MAP(Message Client Equipment Role) relevant indications' definition.

15.1 MAPCINIT

The MAPCINIT indication is used to inform the client host the result of the MAPC(Message Notification Service) initialization.

Indication	**AT-B MAPCINIT [status]**	
Parameters	status	Values in the chapter 14.1.1.
Note	N/A	

15.2 MAPCDISC

The MAPCDISC indication is used to inform the client host of the result of MAPC connection has been disconnected.

Indication	**AT-B MAPCDISC [status],[bd]**	
Parameters	status	Values in the chapter 14.1.1
	bd	Remote *Bluetooth* device address
Note	N/A	

15.3 MAPCGETDATAIND

The MAPCGETDATAIND indication is used to indicate the client host that the packet of a get

operation has arrived.

Indication	**AT-B MAPCGETDATAIND [listSize],[moreData],[length],[packet]**	
Parameters	listSize	Number of entries interested
	moreData	More data to be received or sent. More(TRUE) or not(FALSE)
	packetSize	The length of the packet field
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
Note	N/A	

15.4 MAPCGETCMTIND

The MAPCGETCMTIND indication is used to inform the client host that the current pull process is over.

Indication	**AT-B MAPCGETCMTIND**
Parameters	N/A
Note	The client host shall use the MAPCGETCMT command to complete the current UPLOAD operation every time when the [moreData] field of the MAPCGETDATATIND indication equals to zero.

15.5 MAPCPUSHCONTIND

The MAPCPUSHCONTIND indication is used to inform the client host to continue the push operation.

Indication	**AT-B MAPCPUSHCONTIND**
Parameters	N/A
Note	N/A

15.6 MAPCPUTCMTIND

The MAPCPUSHCMTIND indication is used to inform the client host that the push operation

is completed.

Indication	**AT-B MAPCPUSHCMTIND**
Parameters	N/A
Note	N/A

16 SPP AT Command Definition

This chapter introduces SPP relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

16.1 SPP Status

This chapter introduces the defined SPP status.

16.1.1SPP Connect Status

```
typedef enum
{
    spp_connect_success,        /*! Connect attempt succeeded. */
    spp_connect_sdp_fail,       /*! Service search failed. */
    spp_connect_slc_failed,     /*! Service level connection establishment failed. */
    spp_connect_failed_busy,    /*! Profile instance already connected. */
    spp_connect_failed,         /*! RFCOMM connection failed to be established. */
    spp_connect_server_channel_not_registered,    /*! Requested server channel not registered by this
                                                    profile instance. */
    spp_connect_timeout,        /*! Connection attempt timed out. */
    spp_connect_rejected,       /*! The remote device rejected the connection. */
    spp_connect_normal_disconnect, /*! The remote device terminated the connection. */
    spp_connect_abnormal_disconnect,            /*! Unsuccessful due to an abnormal disconnect while
                                                    establishing the RFCOMM connection. */
    spp_connect_rfcomm_channel_already_open,    /*! The connection attempt failed because there is
                                                    already a connection to that remote device on the
                                                    requested RFCOMM channel. */
    spp_connect_invalid_frame_size              /*! Connect failed due to invalid frame size request
                                                    from app. */
} spp_connect_status
```

16.1.2SPP Disconnect Status

```
typedef enum
{
    spp_disconnect_success,        /*! Successful disconnection. */
    spp_disconnect_link_loss,      /*! Disconnection ocurred due to link loss. */
    spp_disconnect_no_slc,         /*! Disconnect attempt failed, no service level connection. */
    spp_disconnect_timeout,        /*! Disconnect time out. */
    spp_disconnect_error,          /*! Unsuccessful for some other reason. */
} spp_disconnect_status;
```

16.2 SPPCONN

The SPPCONN command is used to establish a SPP connection with a remote device.

Command	**AT+B SPPCONN [bd]**	
Response	**AT-B SPPCONN [status], [bd]**	
Parameters	bd	Remote *Bluetooth* device address.
	status	Values in the chapter 16.1.1.
Note	N/A	

16.3 SPPDISC

The SPPDISC command is used to release the SPP connection with the remote device.

Command	**AT+B SPPDISC**	
Response	**AT-B SPPDISC [status],[bd]**	
Parameters	status	Values in the chapter 16.1.2.
	bd	Remote *Bluetooth* device address.
Note	N/A	

16.4 SPPDATA

The SPPDATA command is used to transfer data with the remote device.

Command	**AT+B SPPDATA [length],[data]**	
Response	Succeeded: **AT-B SPPDATA 0** Failed: **AT-B SPPDATA 1**	
Parameters	length	Length of data
	data	Data to be sent
Note	Before calling this command, the SPPCONN command must be called to establish a SPP connection with the remote *Bluetooth* device. A full piece of AT command must be tailed with "\r" (0x0d).	

17 SPP Indication Definition

This chapter introduces the SPP relevant indications' definition.

17.1 SPPSTAT

The SPPSTAT indication is used to inform the host unit when the local device's SPP status is changed.

Indication	**AT-B SPPSTAT [state]**	
Parameters	state	SPP connection status, where 1: sppReady; 2: sppConnecting; 3: sppConnected.
Note	N/A	

17.2 SPPDATAIND

The SPPDATAIND indication is used to inform the host unit that SPP data is received from the remote device.

Indication	**AT-B SPPDATAIND [length],[data]**	
Parameters	length	Length of received data
	data	Received data
Note	N/A	

18 OPP Client AT Command Definition

This chapter introduces OPP (Client Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

18.1 OPP Client Status

```
typedef enum
{
    oppc_success,                   /*! Last operation was successful.*/
    oppc_failure,                   /*! Last operation failed.*/
    oppc_not_idle,                  /*! A command has been attempted while another command is already
                                    running. */
    oppc_wrong_command,             /*! A command has been attempted while a different multi-packet
                                    command is running. */
    oppc_wrong_state,               /*! Operation failed due to being in the wrong state.*/
    oppc_badrequest,
    oppc_forbidden,
    oppc_notfound,
    oppc_aborted,
    oppc_connection_reject_key_missing,
    oppc_connection_reject_key_security,
    oppc_end_of_status_list
} oppc_lib_status;
```

18.2 OPPCCONN

The OPPCCONN command is used to create an OPP connection with the remote OPP server.

Command	**AT+B OPPCCONN [bd],[maxSize]**	
Response	**AT-B OPPCCONN [status], [bd], [maxSize]**	
Parameters	bd	Remote *Bluetooth* device address.
	maxSize	The maximum packet size transferred during a session. It cannot exceed 255.
	status	Values in the chapter 18.1
Note	N/A	

18.3 OPPCDISC

The OPPCDISC command is used to disconnect the existing OPP connection.

Command	**AT+B OPPCDISC**	
Response	**AT-B OPPCDISC [status],[bd]**	
Parameters	status	Values in the chapter 18.1
	bd	Remote *Bluetooth* device address.
Note	N/A	

18.4 OPPCPUSHST

The OPPCPUSHST command is used to push the first packet of an object.

Command	**AT+B OPPCPUSHST [nameLen],[name], [objectSizeQ],[objectSizeR],[packetSize],[packet]**	
Response	Succeeded: **AT-B OPPCPUSHNTIND [status]** Failed: **AT-B OPPCPUSHST 1**	
Parameters	nameLen	The length of the name field

	name	The name of the object.Unicode format, can include special characters:'nul' space '#' '*' '+' ':' '-' '_' '/' '\' '^', cannot be NULL.
	objectSizeQ	The result of objectSize/65536.
	objectSizeR	The result of objectSize%65536.
	packetSize	Package size, it shall not exceed MaxSize-nameLen-35.
	packet	The partial or complete packet of an object, cannot be NULL
Note	Both of objectSizeQ and objectSizeR shall not both be zero.	

18.5 OPPCPUSHNT

The OPPCPUSHNT command is used to push another packet of the object to the OPP server when the server host has received the "**OPPCPUSHNTIND**" indication.

Command	**AT+B OPPCPUSHNT [last],[packetSize],_[packet]**	
Response	Succeeded: **AT-B OPPCPUSHNTIND [status]** Failed: **AT-B OPPCPUSHNT 1**	
Parameters	last	Last to be sent. Last(TRUE) or not(FALSE)
	packetSize	The size of the packet. It shall not exceed MaxSize-11.
	packet	The partial or complete packet of an object, cannot be NULL
Note	A space should be inserted between the ',' and the [packet] field.	

18.6 OPPCPULLST

The OPPCPULLST command is used by the OPP client to pull the first packet of the default business card from the OPP server.

Command	**AT+B OPPCPULLST**
Response	Succeeded: **AT-B OPPCPULLSTIND [moreData,[nameLen],[name] ,[objectSize],[packetSize],[packet]**

	Failed: **AT-B OPPCPULLST 1**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	nameLen	The length of the name field
	name	The name of the object. Unicode format, can include special characters:'nul' space '#' '*' '+' '.' '-' '_' '/' '\' '^', cannot be NULL.
	objectSize	The size of an object.
	packetSize	The size of the packet.
	packet	The partial or complete packet of an object, cannot be NULL
Note	If the moreData field in the OPPCPULLSTIND indication equals to zero, the client host should also receive the **AT-B OPPCPULLCMTIND [status]** indication, where the [status] field's values are defined in the chapter 18.1.	

18.7 OPPCPULLNT

The OPPCPULLNT command is used by the OPP client to pull another packet from the OPP server when the [moreData] field of either OPPCPULLSTIND or OPPCPULLNTIND indication equals to one.

Command	**AT+B OPPCPULLNT**	
Response	Succeeded: **AT-B OPPCPULLNTIND [moreData],[packetSize],[packet]** Failed: **AT-B OPPCPULLNT 1**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	packetSize	The size of the packet.
	packet	The partial or complete packet of an object, cannot be NULL
Note	If the moreData field in the OPPCPULLSTIND indication equals to zero, the client host should also receive the AT-B OPPCPULLCMTIND [status] indication, where the [status] field's values are defined in the chapter 18.1.	

18.8 OPPCPULLCRTST

The OPPCPULLCRTST command is used by the OPP client to get the previous indication when the OPP client has received the OPPCPULLSTIND indication.

Command	**AT+B OPPCPULLCRTST**	
Response	Succeeded: **AT-B OPPCPULLSTIND [moreData],[nameLength],[name],[objectSize],[packetSize],[packet]** Failed: **AT-B OPPCPULLCRTST 1**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	nameLength	The length of the name field
	name	The name of the object.Unicode format, can include special characters:'nul' space '#' '*' '+' ':' '-' '_' '/' '\' '^', cannot be NULL.
	objectSize	The size of an object.
	packetSize	The size of a packet.
	packet	The partial or complete packet of an object, cannot be NULL
Note	N/A	

18.9 OPPCPULLCRTNT

The OPPCPULLCRTNT command is used by the OPP client to get the previous indication when the client host has received the OPPCPULLNTIND indication.

Command	**AT+B OPPCPULLCRTNT**	
Response	Succeeded: **AT-B OPPCPULLNTIND [moreData],[packetSize],[packet]** Failed: **AT-B OPPCPULLCRTNT 1**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	packetSize	The size of a packet.

	packet	The partial or complete packet of an object, cannot be NULL
Note	N/A	

18.10 OPPCPULLCMT

When the client host received the OPPCPULLCMTIND indication, the client host shall use this command to complete the current object PULL process. Before the client host sends this command, OPPCPULLCRTST or OPPCPULLCRTNT commands can be used to get the previous indication. If this command has been sent, OPPCPULLCRTST or OPPCPULLCRTNT commands shall not be used in the current PULL process.

Command	**AT+B OPPCPULLCMT**
Response	Succeeded: **AT-B OPPCPULLCMT 0** Failed: **AT-B OPPCPULLCMT 1**
Parameters	N/A
Note	N/A

18.11 OPPCABORT

The OPPCABORT command is used by the client host to abort an object either PULL or PUSH process when the current process includes multiple packets.

Command	**AT+B OPPCABORT**	
Response	Succeeded: **AT-B OPPCPULLCMTIND [status]** **AT-B OPPCPUSHCMTIND [status]** Failed: **AT-B OPPCABORT 1**	
Parameters	status	Values in the chapter 18.1
Note	N/A	

19 OPP Client Indication Definition

This chapter introduces the OPP (Client Role) relevant indications' definition.

19.1 OPPCCONN

The OPPCCONN indication is used to inform the client host of the result of connect request.

Indication	**AT-B OPPCCONN [status],[bd],[maxSize]**	
Parameters	status	Values in the chapter 18.1
	bd	Remote *Bluetooth* device address.
	maxSize	The maximum packet size transferred during a session. It cannot exceed 255
Note	If the [status] field equals to 0, it indicates that an OPP connection is successfully established, otherwise it means the connection establishment failed.	

19.2 OPPCPUSHNTIND

The OPPCPUSHNTIND indication is used to inform the client host to push another packet of an object. The client host can either use the OPPCPUSHNT command to push the next packet or use the OPPCABORT command to abort the PUSH process.

Indication	**AT-B OPPCPUSHNTIND**
Parameters	N/A
Note	N/A

19.3 OPPCPUSHCMTIND

The OPPCPUSHCMTIND indication is used to inform the client host that a complete object has been pushed and the current push process is finished.

Indication	**AT-B OPPCPUSHCMTIND [status]**	
Parameters	status	Values in the chapter 18.1

Note	N/A

19.4 OPPCPULLSTIND

The OPPCPULLSTIND indication is used to indicate the client host that the first packet of an object has arrived. The client host can use the OPPCPULLCRTST command to get the previous indication. The client host can use the OPPCPULLNT command to get another packet or use the OPPCABORT command to abort the current pull progress when the [moreData] field of the OPPCPULLSTIND indication equals to one. If the [moreData] field of the OPPCPULLSTIND indication equals to zero, the OPPCPULLCMTIND indication will be received.

Indication	**AT-B OPPCPULLSTIND [moreData],[nameLen],[name],[objectSize],[packetSize],[packet]**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	nameLen	The length of the name field
	name	The name of the object.Unicode format, can include special characters:'nul' space '#' '*' '+' '.' '-' '_' '/' '\' '^', cannot be NULL. If nameLen equals to zero, it will be NULL.
	objectSize	The size of an object.
	packetSize	The size of a packet.
	packet	The partial or complete packet of an object, cannot be NULL. If packetSize equals to zero, it will be NULL
Note	N/A	

19.5 OPPCPULLNTIND

The OPPCPULLNTIND indication is used to inform the client host that the next packet of an object has arrived. The client host can use the OPPCPULLCRTNT command to get the previous indication. The client host can use the OPPCPULLNT command to get another packet or use the OPPCABORT command to abort the current PULL progress when the [moreData] field of the OPPCPULLNTIND indication equals to one. If the [moreData] field of the OPPCPULLNTIND indication equals to zero, the OPPCPULLCMTIND indication will be received.

Indication	**AT-B OPPCPULLNTIND [moreData],[packetSize],[packet]**

Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	packetSize	The size of a packet.
	packet	The partial or complete packet of an object, cannot be NULL. If packetSize equals to zero, it will be NULL
Note	N/A	

19.6 OPPCPULLCMTIND

The OPPCPULLCMTIND indication is used to inform the client host that a complete object has been successfully pulled. The client host shall use the OPPCPULLCMT command to finish the current pull process every time when the client host received the OPPCPULLCMTIND indication. Before the client host sends this command, either OPPCPULLCRTST or OPPCPULLCRTNT command can be used to get the previous indication. After this command is sent, either OPPCPULLCRTST or OPPCPULLCRTNT command shall not be used in the current PULL process.

Indication	**AT-B OPPCPULLCMTIND [status]**	
Parameters	status	Values in the chapter 18.1
Note	N/A	

19.7 OPPCDISC

The OPPCDISC indication is used to indicate the client host that the current OPP connection has been disconnected.

Indication	**AT-B OPPCDISC [status],[bd]**	
Parameters	status	Values in the chapter 18.1
	bd	Remote *Bluetooth* device address.
Note	N/A	

20 HID Device AT Command Definition

20.1 HID Status

This chapter introduces the defined HID status.

20.1.1HID Connect Status

```
typedef enum
{
    hid_connect_success,            /*!< Successful connection.*/
    hid_connect_failed,             /*!< Connection failed. */
    hid_connect_out_of_resources,   /*!< Out of resource. */
    hid_connect_timeout,            /*!< Timeout waiting for connection. */
    hid_connect_disconnected,       /*!< Disconnected remotely during setup */
  hid_command_disallowed /* command disallowed. */
} hid_connect_status;
```

20.1.2HID Disconnect Status

```
typedef enum
{
    hid_disconnect_success,         /*!< Successful disconnection.*/
   hid_disconnect_link_loss,        /*!< Unsuccessful due to the link being lost.*/
   hid_disconnect_timeout,          /*!< Unsuccessful due to time out.*/
   hid_disconnect_violation,        /*!< Disconnection due to protocol violation */
   hid_disconnect_error,            /*!< Unsuccessful for some other reason.*/
    hid_disconnect_virtual_unplug,  /*!< Virtual unplug disconnection */
    hid_disconnect_command_disallowed       /* command disallowed. */
} hid_disconnect_status;
```

20.2 HIDCONN

The HIDCONN command is used to create an HID connection with the host.

Command	**AT+B HIDCONN [bd_addr]**	
Response	**AT-B HIDCONN [hid_connect_status],[bd_addr]**	
Parameters	hid_connect_status	Refer to the chapter 20.1.1.
	bd_addr	remote device address
Note	N/A	

20.3 HIDDISC

The HIDDISC command is used to disconnect the HID connection with the host.

Command	**AT+B HIDDISC**	
Response	**AT-B HIDDISC [disconnect_status]**	
Parameters	disconnect_staus	Refer to the chapter 20.1.2.
Note	N/A	

20.4 PINRES

The PINRES command is input paring PIN code or passkey when HID profile is enabled.

Command	**AT+B PINRES[pin_code]**	
Response	**AT-B PINRES [result_code]**	
Parameters	pin_code	PIN code
	result_code	0 : success; 1: command disallowed; 2: parameter error; 3: authencation failed;
Note	If HID keyboard is enabled, when the module is pairing with the HID host, it should call the PINRES command to input PIN code or passkey If HID mouse is enabled, when the module is pairing with *Bluetooth* 2.1+EDR HID host, it does not need to input PIN code; when the module is pairing with	

	Bluetooth 2.0 HID host, it needs to input the fixed PIN code 0000.

20.5 Keyboard Input/Output Report

The HIDIRPT command is used to send input report when the module is in the report mode.

<table>
<tr><td>Command</td><td colspan="2">AT+B HIDIRPT [len], [(raw_data)]</td></tr>
<tr><td>Indication</td><td colspan="2">Succeeded：AT-B HIDIRPT 0
Failed：AT-B HIDIRPT 1</td></tr>
<tr><td rowspan="2">Parameter</td><td>len</td><td>Data length</td></tr>
<tr><td>raw_data</td><td>Format definition as follows:
<table>
<tr><td>Report ID (Fixed as 0x01)</td><td>8-bit (Keycode-224-231)</td><td>8-bit (Reserved)</td><td>Key 1 (8-bit)</td><td>Key 2</td><td>Key 3</td><td>Key 4</td><td>Key 5</td><td>Key6 6</td></tr>
</table></td></tr>
<tr><td>Note</td><td colspan="2">1. When the module is in report mode, this command is used to send an input report to the host. The first field' length is 8 bit, every bit refers to Usage ID 224~231 statuses in HID Keyboard Usage Page. Key1~Key6 respectively represents the pressed key's Usage ID. The range of Usage ID is 0~101.
2. When only HID keyboard is enabled in i40e, i40e does not return the indication. When multiple profiles are enabled, the module will return the indication.
3. If using AT+B SPRO command to only enable HID keyboard function, it needs to send keyboard input as follows: \x0c\x00\xa1[(raw_data)]
Under this circumstance, the module will not return the indication.</td></tr>
</table>

The HIDORPT event is the output report sent by the host when the module is in the report mode.

Event	**AT-B HIDORPT [len], [(raw_data)]**

<table>
<tr><td rowspan="2">Parameter</td><td>len</td><td>Data length</td></tr>
<tr><td>raw_data</td><td>Format definition as follows:
<table>
<tr><td>Report ID (Fixed as 0x01)</td><td>5-bit (Num Lock --- Kana)</td><td>3-bt (Reserved)</td></tr>
</table>
</td></tr>
<tr><td>Note</td><td colspan="2">When the module is in report mode, this event represents the host sends an output report to the module. The output report’s length is 1 byte. The first 5 bits refer to Usage ID 1~5 statuses in HID LED Usage Page.</td></tr>
</table>

The HIDBIRPT command is used to send input report when the module is in the boot mode.

<table>
<tr><td>Command</td><td colspan="2">AT+B HIDBIRPT [len],[(raw_data)]</td></tr>
<tr><td>Indication</td><td colspan="2">Succeeded: AT-B HIDBIRPT 0
Failed: AT-B HIDBIRPT 1</td></tr>
<tr><td rowspan="2">Parameter</td><td>len</td><td>Data length</td></tr>
<tr><td>raw_data</td><td>Format definition as follows:
<table>
<tr><td>8-bit (Keycode-224-231)</td><td>8-bit (Reserved)</td><td>Key1 (8-bit)</td><td>Key 2</td><td>Key 3</td><td>Key 4</td><td>Key 5</td><td>Key 6</td></tr>
</table>
</td></tr>
<tr><td>Note</td><td colspan="2">1. When the module is in boot mode, this command is used to send an input report to the host. The first field’ length is 8 bit, every bit refers to Usage ID 224-331statuses in HID Keyboard Usage Page. Key1~Key6 respectively represents the pressed key’s Usage ID. The range of Usage ID is 0~255.
2. When only HID keyboard is enabled in i40e, i40e does not return the indication. When multiple profiles are enabled, the module will return the indication.
3. If using AT+B SPRO command to only enable HID keyboard function, it needs to send keyboard input as follows: \x0c\x00\xa1[(raw_data)]
Under this circumstance, the module will not return the indication.</td></tr>
</table>

The HIDBOPRT event is the output report sent by the host when the module is in the boot mode.

<table>
<tr><td>Event</td><td colspan="2">AT-B HIDBOPRT [len],[(raw_data)]</td></tr>
<tr><td rowspan="2">Parameter</td><td>len</td><td>Data length</td></tr>
<tr><td>raw_data</td><td>Format definition as follows:<table><tr><td>5-bit

(NumLock--Kana)</td><td>3-bt

(Reserved)</td></tr></table></td></tr>
<tr><td>Note</td><td colspan="2">When the module is boot mode, this event represents the host sends an output report to the module. The output report’s length is 1 byte. The first 5 bits refer to Usage ID 1~5 statuses in HID LED Usage Page.</td></tr>
</table>

20.4 Mouse Input/Output Report

The HIDIRPT command is used to send an input report sent when the module is in the report mode.

<table>
<tr><td>Command</td><td colspan="2">AT+B HIDIRPT [len], [(raw_data)]</td></tr>
<tr><td>Indication</td><td colspan="2">Succeeded: AT-B HIDIRPT 0
Failed: AT-B HIDIRPT 1</td></tr>
<tr><td rowspan="2">Parameter</td><td>len</td><td>Data length, equals to 6.</td></tr>
<tr><td>raw_data</td><td>Input X/Y coordinates, mouse wheel, press. Little endian.
Format definition as follows:<table><tr><td>Button1-3(bit0-2),(bit3-bit7 is 0)</td><td>X - 16bit</td><td>Y - 16bit</td><td>Wheel - 8bit</td></tr></table></td></tr>
<tr><td>Note</td><td colspan="2">1. When the module is in report mode, it uses this command to send mouse input information. The first field’s length is 1byte. Bit 0~2 represents mouse press status; bit 3~7 are 0. The second and third fields are 2 bytes, representing X/Y coordinate drift values. The forth field’s length is 1 byte, representing mouse wheel value. The X/Y values are between -2048 to 2047. For instance, “AT+B HIDIRPT 06,\x00\x50\x00\x50\x00\x00\r” represents the mouse position drift is 0x0050(X)/0x0050(Y).</td></tr>
</table>

	2. When only HID mouse is enabled in i40e, i40e does not return the indication. When multiple profiles are enabled, the module will return the indication. 3. If using AT+B SPRO command to only enable HID keyboard function, it needs to send keyboard input as follows: \x09\x00\xa1[(raw_data)] Under this circumstance, the module will not return the indication.

The HIDBIRPT command is used to send an input report sent when the module is in the boot mode.

Command	**AT+B HIDBIRPT [len], [(raw_data)]**	
Indication	Succeeded: **AT-B HIDBIRPT 0** Failed: **AT-B HIDBIRPT 1**	
Parameter	len	Data length, equals to 6.
	raw_data	Input X/Y coordinates, mouse wheel, press. Little endian. Format definition as follows: Button1-3(bit0-2),(bit3-bit7 is 0) | X - 8bit | Y - 8bit | Wheel - 8bit
Note	1. When the module is in boot mode, MCU uses this command to send mouse input information. 2. When only HID mouse is enabled in i40e, i40e does not return the indication. When multiple profiles are enabled, the module will return the indication. 3. If using AT+B SPRO command to only enable HID keyboard function, it needs to send keyboard input as follows: \x09\x00\xa1[(raw_data)] Under this circumstance, the module will not return the indication.	

21 HID Device Indication Definition

This chapter introduces the HID (HID Device Role) relevant indications' definition.

21.1 HIDINIT

This KEYBOARDINIT indication is used to indicate that the module supports HID keyboard.

Indication	**AT-B KEYBOARDINIT 0**
Parameters	N/A
Note	N/A

This MOUSEINIT indication is used to indicate that the module supports HID keyboard.

Indication	**AT-B MOUSEINIT 0**
Parameters	N/A
Note	N/A

21.2 PININD

The PININD indication is used to indicate the client host requires the module to input PIN code or passkey.

Indication	**AT-B PININD**
Parameters	N/A
Note	When HID keyboard is enabled, this event is returned to require calling the PINRES command to input PIN code or passkey.

21.3 HIDSUSPEND

The HIDSUSPEND indication is used to indicate the client host is suspended.

Indication	**AT-B HIDSUSPEND [suspend_status]**	
Parameters	suspend_status	Suspend status, where, 00: exit suspend; 01: suspend.
Note	N/A	

21.4 HIDPMODE

The HIDPMODE indication is used to indicate the client host when the protocol mode is updated.

Indication	**AT-B HIDPMODE [protocol_mode]**	
Parameters	protocol_mode	Protocol mode ,where, 00: root mode; 01: report protocol mode;
Note	After the HID connection is established, the default protocol mode is report protocol mode. the module will return this indication when the protocol mode is updated.	

22 HFPAG AT Command Definition

This chapter introduces the HFP (AG Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

22.1 HFP AG Status

This chapter introduces the defined HFP status.

22.1.1HFP AG Status

```
typedef enum
{
    aghfp_success,      /*!< Success. */
    aghfp_fail          /*!< Failure. */
} aghfp_lib_status;
```

22.1.2HFP AG Connect Status

```
typedef enum
{
    aghfp_connect_success,                  /*!< Successful connection. */
    aghfp_connect_sdp_fail,                 /*!< Unsuccessful due to a service search
```

```
                                                    failure.*/
    aghfp_connect_slc_failed,                       /*!< Unsuccessful due to a service level
                                                    connection failure. */
    aghfp_connect_failed_busy,                      /*!< Unsuccessful due to service level connection
                                                    already established. */
    aghfp_connect_failed,                           /*!< Unsuccessful due to RFCOMM connection
                                                    failing to be established. */
    aghfp_connect_server_channel_not_registered,    /*!< Unsuccessful due to attempt to connect to
                                                    unallocated server channel. */
    aghfp_connect_timeout,                          /*!< Unsuccessful due to connection attempt
                                                    timing out. */
    aghfp_connect_rejected,                         /*!< Unsuccessful due to remote device rejecting
                                                    connection. */
    aghfp_connect_normal_disconnect,                /*!< Unsuccessful due to remote device
                                                    terminating the connection. */
    aghfp_connect_abnormal_disconnect               /*!< Unsuccessful due to an abnormal disconnect
                                                    while establishing an rfcomm connection. */

    aghfp_connect_rejected_key_missing,
    aghfp_connect_rejected_key_security
} aghfp_connect_status;
```

22.1.3HFP AG Disconnect Status

```
typedef enum
{
    aghfp_disconnect_success,           /*!< Successful disconnection. */
    aghfp_disconnect_link_loss,         /*!< Unsuccessful due to abnormal linkloss. */
    aghfp_disconnect_no_slc,            /*!< Unsuccessful due to no current connection. */
    aghfp_disconnect_timeout,           /*!< Unsuccessful due to RFCOMM connection attempt
                                        timeout. */
    aghfp_disconnect_error              /*!< Unsuccessful due to RFCOMM connection attempt
                                        error. */
} aghfp_disconnect_status;
```

22.1.4HFP AG Audio Connect Status

```
typedef enum
{
    aghfp_audio_connect_success,        /*! Successful audio connection.*/
    aghfp_audio_connect_failure,        /*! Unsuccessful due to negotiation failure.*/
    aghfp_audio_connect_have_audio,     /*! Unsuccessful due to audio already being with device.*/
```

```
    aghfp_audio_connect_in_progress,        /*! Unsuccessful due to an audio connect already being
                                               attempted.*/
    aghfp_audio_connect_invalid_params,     /*! Unsuccessful due to one or more parameters specified being
                                               invalid.*/
    aghfp_audio_connect_call_manager_active,    /*! Unsuccessful due to Call Manager setting up/shutting
                                                   down a call (and hence audio).*/
    aghfp_audio_connect_error,              /*! Unsuccessful due to library being in incorrect state.*/
    aghfp_audio_connect_wbs_fail            /*! Unsuccessful due to a Wide Band Speech Error. */
} aghfp_audio_connect_status;
```

22.1.5HFP AG Audio Disconnect Status

```
typedef enum
{
    aghfp_audio_disconnect_success,         /*! Successful audio disconnection.*/
    aghfp_audio_disconnect_failure,         /*! Unsuccessful due to failure indication from firmware.*/
    aghfp_audio_disconnect_no_audio,        /*! Unsuccessful due to audio being with AG.*/
    aghfp_audio_disconnect_in_progress,     /*! Unsuccessful due to an audio disconnect already being
                                               attempted.*/
    aghfp_audio_disconnect_call_manager_active, /*! Unsuccessful due to Call Manager setting up/shutting
                                                   down a call (and hence audio).*/
    aghfp_audio_disconnect_error            /*! Unsuccessful due to library being in incorrect state.*/
} aghfp_audio_disconnect_status;
```

22.2 AGCONN

The AGCONN command is used to create a service level connection with the remote device.

Command	**AT+B AGCONN [bd]**	
Response	**AT-B AGCONN [status],[bd],[profile]**	
Parameters	status	Values in the chapter 22.1.2.
	bd	Remote *Bluetooth* device address
	profile	Profile type, where 0: Not HSP/HFP; 1: Headset Profile; 2: Hands-free Profile.

Note	bd is comprised of 12 bytes hexadecimal characters.

22.3 AGDISC

The AGDISC command is used to disconnect the service level connection with the remote device.

Command	**AT+B AGDISC**	
Response	**AT-B AGDISC [status],[bd]**	
Parameters	status	Values in the chapter 22.1.3.
	bd	Remote *Bluetooth* device address
Note	N/A	

22.4 AGAUDIOCONN

The AGAUDIOCONN command is used to create a SCO/eSCO connection with the remote device.

Command	**AT+B AGAUDIOCONN**	
Response	**AT-B AGAUDIOCONN [status]**	
Parameters	status	Values in the chapter 22.1.4.
Note	N/A	

22.5 AGAUDIODISC

The AGAUDIODISC command is used to disconnect the SCO/eSCO connection with the remote device.

Command	**AT+B AGAUDIODISC**	
Response	**AT-B AGAUDIODISC [status]**	
Parameters	status	Values in the chapter 22.1.5.

Note	N/A

22.6 AGAUDIOTRANS

The AGAUDIOTRANS command is used to transfer audio from/to remote device.

Command	**AT+B AGAUDIOTRANS**	
Response	If audio is transferred to HF, response is: **AT-B AGAUDIOCONN [status]** If audio is transferred to AG, response is: **AT-B AGAUDIODISC [status]**	
Parameters	status	Values in the chapter 22.1.1.
Note	N/A	

22.7 AGCIEVSVC

The AGCIEVSVC command is used to send a service indicator to the HF, only for HFP.

Command	**AT+B AGCIEVSVC [service]**	
Response	**AT-B AGCIEVSVC [status]**	
Parameters	service	0: no service; 1: presence of service.
	status	Values in the chapter 22.1.1
Note	N/A	

22.8 AGCIEVSIG

The AGCIEVSIG command is used to send a signal strength indicator to the HF, only for HFP.

Command	**AT+B AGCIEVSIG [signal]**
Response	**AT-B AGCIEVSIG [status]**

Parameters	signal	Ranges from 0 to 5.
	status	Values in the chapter 22.1.1.
Note	N/A	

22.9 AGCIEVBAT

The AGCIEVBAT command is used to send a battery charge indicator to HF, only for HFP.

Command	**AT+B AGCIEVBAT [battery]**	
Response	**AT-B AGCIEVBAT [status]**	
Parameters	battery	Ranges from 0 to 5.
	status	Values in the chapter 22.1.1
Note	N/A	

22.10 AGCIEVROAM

The AGCIEVROAM command is used to send a roaming status indicator to HF, only for HFP.

Command	**AT+B AGCIEVROAM [roam]**	
Response	**AT-B AGCIEVROAM [status]**	
Parameters	roam	0: roaming is not active; 1: roaming is active.
	status	Values in the chapter 22.1.1.
Note	N/A	

22.11 AGCALL

The AGCALL command is used to send a call indicator to HF, only for HFP.

Command	**AT+B AGCALL [call]**

Response	**AT-B AGCALL [status]**	
Parameters	call	0: there are no calls in progress; 1: at least one call in progress.
	status	Values in the chapter 22.1.1
Note	N/A	

22.12 AGCALLSETUP

The AGCALLSETUP command is used to send a call setup indicator to HF, only for HFP.

Command	**AT+B AGCALLSETUP [callsetup]**	
Response	**AT-B AGCALLSETUP [status]**	
Parameters	callsetup	0: not currently in call set up; 1: an incoming call process ongoing; 2: an outgoing call set up is ongoing; 3: remote party being alerted in an outgoing call.
	status	Values in the chapter 22.1.1.
Note	N/A	

22.13 AGCALLHELD

The AGCALLHELD command is used to send a call held indicator to HF, only for HFP.

Command	**AT+B AGCALLHELD [callheld]**	
Response	**AT-B AGCALLHELD [status]**	
Parameters	callheld	0:No calls held; 1: Call is placed on hold or active/held calls swapped (The AG has both an active AND a held call); 2: Call on hold, no active call.

	status	Values in the chapter 22.1.1.
Note	N/A	

22.14 AGCOPS

The AGCOPS command is used to send network operator to HF, only for HFP.

Command	**AT+B AGCOPS [mode],[operator]**	
Response	**AT-B AGCOPS [status]**	
Parameters	mode	The current mode and provides no information with regard to the name of the operator.
	operator	Specifies a quoted string in alphanumeric format representing the name of the network operator and this string shall not exceed 16 characters.
	status	Values in the chapter 22.1.1.
Note	N/A	

22.15 AGCMEERR

The AGCMEERR command is used to send extended error result code to HF, only for HFP.

Command	**AT+B AGCMEERR [errorcode]**	
Response	**AT-B AGCMEERR [status]**	
Parameters	errorcode	0: AG failure; 1: no connection to phone; 3: operation not allowed; 4: operation not supported; 5: PH-SIM pin required; 10: SIM not inserted; 11: SIM pin required; 12: SIM PUK required; 13: SIM failure; 14: SIM busy; 16: incorrect password = 16;

		17: SIM PIN2 required; 18: SIM PUK2 required; 20: memory full; 21: invalid index; 23: memory failure; 24: text string too long; 25: invalid chars in text string; 26: dial string too long; 27: invalid chars in dial string; 30: no network service; 32: network not allowed.
	status	Values in the chapter 22.1.1.
Note	This command can only be sent after HF enables the “Extended Audio Gateway Error Result Code” feature in the AG and returns the AGCMEEIND indication**.**	

22.16 AGCLIP

The AGCLIP command is used to send caller ID to HF, only for HFP.

Command	**AT+B AGCLIP [type],[number]**	
Response	**AT-B AGCLIP [status]**	
Parameters	type	Values 128-143: The phone number format may be a national or international format, and may contain prefix and/or escape digits. No changes on the number presentation are required. Values 144-159: The phone number format is an international number, including the country code prefix. If the plus sign ("+") is not included as part of the number and shall be added by the AG as needed. Values 160-175: National number. No prefix nor escape digits included.
	number	Phone number, a text string.
	status	Values in the chapter 22.1.1
Note	N/A	

22.17 AGSPKVOL

The AGSPKVOL command is used to set speaker volume on the HF or HS.

Command	**AT+B AGSPKVOL [volume]**	
Response	**AT-B AGSPKVOL [status]**	
Parameters	volume	The speaker volume, ranges from 0 to 15.
	status	Values in the chapter 22.1.1.
Note	N/A	

22.18 AGMICVOL

The AGMICVOL command is used to set microphone gain on the HF or HS.

Command	**AT+B AGMICVOL [gain]**	
Response	**AT-B AGMICVOL [status],**	
Parameters	gain	The microphone gain, ranges from 0 to 15.
	status	Values in the chapter 22.1.1
Note	N/A	

22.19 AGCNUM

The AGCNUM command is used to send subscriber number information to HF, only for HFP.

Command	**AT+B AGCNUM [type], [number], [service],[complete]**	
Response	**AT-B AGCNUM [status]**	
Parameters	type	Values 128-143: The phone number format may be a national or international format, and may contain prefix and/or escape digits. No changes on the number presentation are required. Values 144-159: The phone number format is an international number, including the country code prefix. If the plus sign ("+") is not included as part of the number and shall be added

		by the AG as needed. Values 160-175: National number. No prefix nor escape digits included.
	number	Phone number, a text string.
	Service	Indicates which service this phone number relates to, where, 4: voice; 5: fax.
	complete	whether the process is complete 1: complete; 0: not complete, more numbers to send.
	status	Values in the chapter 22.1.1.
Note	N/A	

22.20 AGCLCC

The AGCLCC command is used to send current calls list to HF, only for HFP.

Command	**AT+B AGCLCC [idx],[dir],[status],[mode],[mpty],[type],[number],[complete]**	
Response	**AT-B AGCLCC [respstatus]**	
Parameters	idx	The numbering (starting with 1) of the call given by the sequence of setting up or receiving the calls (active, held or waiting) as seen by the served subscriber. Calls hold their number until they are released. New calls takelowest available number.
	dir	0: outgoing; 1: incoming.
	status	0: Active; 1: Held; 2: Dialing (outgoing calls only); 3: Alerting (outgoing calls only);

		4: Incoming (incoming calls only); 5: Waiting (incoming calls only); 6: Call held by Response and Hold.
	mode	0: Voice; 1: Data; 2: FAX.
	mpty	0: this call is NOT a member of a multi-party (conference) cal; 1: this call IS a member of a multi-party (conference) call.
	type	Values 128-143: The phone number format may be a national or international format, and may contain prefix and/or escape digits. No changes on the number presentation are required. Values 144-159: The phone number format is an international number, including the country code prefix. If the plus sign ("+") is not included as part of the number and shall be added by the AG as needed. Values 160-175: National number. No prefix nor escape digits included.
	number	Phone number, a text string.
	Complete	whether the process is complete 1: complete; 0: not complete, more numbers to send.
	respstatus	Values in the chapter 22.1.1
Note	N/A	

22.21 AGRING

The AGRING command is used to send a ring alert to HF or HS.

Command	**AT+B AGRING**

Response	**AT-B AGRING [status],**	
Parameters	status	Values in the chapter 22.1.1.
Note	N/A	

22.22 AGCCWA

The AGCCWA command is used to tell the HF that an incoming call is waiting while another call is ongoing, only for HFP.

Command	**AT+B AGCCWA [type],[number]**	
Response	**AT-B AGCCWA [status]**	
Parameters	type	Values 128-143: The phone number format may be a national or international format, and may contain prefix and/or escape digits. No changes on the number presentation are required. Values 144-159: The phone number format is an international number, including the country code prefix. If the plus sign ("+") is not included as part of the number and shall be added by the AG as needed. Values 160-175: National number. No prefix nor escape digits included.
	number	Phone number, a text string.
	status	Values in the chapter 22.1.1.
Note	This command can only be sent after HF enables call waiting notification and returns the AGCCWAIND indication**.**	

22.23 AGMUTE

The AGMUTE command is used to mute on/off microphone or speaker when a call is ongoing.

Command	**AT+B AGMUTE [op]**
Response	**AT-B AGMUTE [status]**

Parameters	op	0: mute off; 1: mute microphone on, mute speaker off; 2: mute speaker on, mute microphone off; 3: mute both speaker and microphone on.
	status	Values in the chapter 22.1.1.
Note	N/A	

22.24 AGCIND

The AGCIND command is used to send the current status of the AG indicators.

Command	**AT+B AGCIND [service], [call],[callsetup],[callheld],[signal],[roam],[battery]**	
Response	**AT-B AGCIND [status]**	
Parameters	service	Whether or not a service is present. 0: no service; 1: presence of service.
	call	0: means there are no calls in progress; 1: means at least one call is in progress.
	callsetup	0: means not currently in call set up; 1: means an incoming call process ongoing; 2: means an outgoing call set up is ongoing; 3: means remote party being alerted in an outgoing call.
	callheld	0: No calls held; 1: Call is placed on hold or active/held calls swapped (The AG has both an active AND a held call); 2: Call on hold, no active call.
	signal	Ranges from 0 to 5.
	roam	0: roaming is not active; 1: roaming is active.
	battery	Ranges from 0 to 5.

	status	Values in the chapter 22.1.1.
Note	This command shall be sent after received the AGINDICATORSIND indication.	

22.25 AGOK

The AGOK command is used to send the OK command to the remote HF side.

Command	**AT+B AGOK**	
Response	**AT-B AGOK [status]**	
Parameters	status	Values in the chapter 22.1.1.
Note	N/A	

22.26 AGERROR

The AGERROR command is used to send the ERROR command to the remote HF side.

Command	**AT+B AGERROR**	
Response	**AT-B AGERROR [status]**	
Parameters	status	Values in the chapter 22.1.1.
Note	N/A	

23 HFP AG Indication Definition

This chapter introduces the HFP (AG Role) relevant indications' definition.

23.1 AGCONN

The AGCONN indication happens when remote device creates the service level connection with the module.

Indication	**AT-B AGCONN [status],[bd], [profile]**

Parameters	status	Values in the chapter 22.1.2.
	bd	Remote *Bluetooth* device address
	profile	Profile type, where 0: Not HSP/HFP; 1: Headset Profile; 2: Hands-free Profile.
Note	N/A	

23.2 AGDISC

The AGDISC indication happens when the remote device disconnect the service level connection with the module.

Indication	**AT-B AGDISC [status],[bd]**	
Parameters	status	Values in the chapter 22.1.3
	bd	Remote *Bluetooth* device address
Note	N/A	

23.3 AGDIALIND

The AGDIALIND indication is used to tell the host the HF has dialed a number, only for HFP.

Indication	**AT-B AGDIALIND [type],[number]**	
Parameters	type	0: normal dial; 1: memory dial; 2: last number redial.
	number	Phone number to dial, if type =2, this parameter is ignored.
Note	After received this indication, AT+B AGOK or AT+B AGERROR should be used as a response.	

23.4 AGCLIPIND

The AGCLIPIND indication is used to tell the host the HF wants to enable/disable caller ID notifications, only for HFP.

Indication	**AT-B AGCLIPIND [enable]**	
Parameters	enable	Whether to enable or disable caller ID notifications. 0: disable; 1: enable.
Note	N/A	

23.5 AGCCWAIND

The AGCCWAIND indication is used to tell the host the HF wants to enable/disable call waiting notifications, only for HFP.

Indication	**AT-B AGCCWAIND [enable]**	
Parameters	enable	Whether to enable or disable call waiting notifications. 0: disable; 1: enable.
Note	N/A	

23.6 AGDTMFIND

The AGDTMFIND indication is used to tell the host the HF has requested that a DTMF code be transmitted by the AG, only for HFP.

Indication	**AT-B AGDTMFIND [key]**	
Parameters	key	The single character DTMF code to transmit, may be 0-9, A-D, # or *
Note	N/A	

23.7 AGMICVOLIND

The AGMICVOLIND indication is used to tell the host the HF has sent a microphone gain synchronization message.

Indication	**AT-B AGMICVOLIND [gain]**	
Parameters	gain	The gain value received from the HF, ranges from 0 to 15.
Note	N/A	

23.8 AGSPKVOLIND

The AGSPKVOLIND indication is used to tell the host the HF has sent a speaker volume synchronization message.

Indication	**AT-B AGSPKVOLIND [volume]**	
Parameters	volume	The volume value received from the HF, ranges from 0 to 15.
Note	N/A	

23.9 AGCMEEIND

The AGCMEEIND indication is used to tell the host the HF wants to enable/disable Extended Audio Gateway Error result codes in the AG, only for HFP.

Indication	**AT-B AGCMEEIND [enable]**	
Parameters	enable	Whether to enable or disable extended error result code. 0: disable; 1: enable.
Note	N/A	

23.10 AGCNUMIND

The AGCNUMIND indication is used to tell the host the HF has sent a command to get subscriber number information, only for HFP.

Indication	**AT-B AGCNUMIND**
Parameters	N/A
Note	AT+B AGCNUM shall be sent as responses.

23.11 AGCLCCIND

The AGCLCCIND indication is used to tell the host the HF has sent a command to get current calls list of AG, only for HFP.

Indication	**AT-B AGCLCCIND**
Parameters	N/A
Note	AT+B AGCLCC shall be sent as responses.

23.12 AGBIAIND

The AGBIAIND indication is used to tell the host the HF wants to activate/deactivate indicators which sent by AG, only for HFP1.6.

Indication	**AT-B AGBIAIND [mask]**	
Parameters	mask	Mask indicating which indicators are active and can be sent to HF. 0x01: service indicator; 0x02: call indicator; 0x04: call setup indicator; 0x08: call held indicator; 0x10: signal strength indicator; 0x20: roaming status indicator; 0x40: battery charge indicator; 0x7f: all indicators;
Note	N/A	

23.13 AGANSWERIND

The AGANSWERIND indication is used to tell the host the HF has answered the call, only for HFP.

Indication	**AT-B AGANSWERIND**
Parameters	N/A
Note	N/A

23.14 AGHANGUPIND

The AGHANGUPIND indication is used to tell the host the HF has rejected or hang up the call, only for HFP.

Indication	**AT-B AGHANGUPIND**
Parameters	N/A
Note	N/A

23.15 AGCOPSIND

The AGCOPSIND indication is used to tell the host the HF has sent a request to get the currently selected operator, only for HFP.

Indication	**AT-B AGCOPSIND**
Parameters	N/A
Note	N/A

23.16 AGHSBUTTONIND

The AGHSBUTTONIND indication is used to tell the host the HS has pressed a button, only for HSP.

Indication	**AT-B AGHSBUTTONIND**
Parameters	N/A
Note	N/A

23.17 AGINDICATORSIND

The AGINDICATORSIND indication is used to tell the host the HF is requesting current status of the AG indicators.

Indication	**AT-B AGINDICATORSIND**
Parameters	N/A
Note	N/A

24 AVRCP Target AT Command Definition

This chapter introduces the AVRCP (Target Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples. Commands that are defined in this chapter is only used for Category 2 device.

24.1 AVRCPVOLCHANGEDRES

The AVRCPVOLCHANGEDRES command is used by the TG (Category 2) to notify CT with TG's volume change event if the AVRCPREGVOLCHANGEDIND indication is received.

Command	**AT+B AVRCPVOLCHANGEDRES [volume]**	
Response	**AT-B AVRCPVOLCHANGEDRES [status]**	
Parameters	volume	Absolute volume, where ranges from 0 to 0x7F
	status	Values in the chapter 10.1
Note	This command is only used for Category 2 device.	

25 AVRCP Target Indication Definition

This chapter introduces the AVRCP (Target Role) relevant indications' definition. Indication that are defined in this chapter is only used for category 2 device.

25.1 AVRCPREGVOLCHANGEDIND

The AVRCPREGVOLCHANGEDIND indication is used to tell the host the remote AVRCP CT has requested to be notified with the volume change event.

Indication	**AT-B AVRCPREGVOLCHANGEDIND**
Parameters	N/A
Note	N/A

25.2 AVRCPABSVOLIND

The AVRCPABSVOLIND indication is used to tell the host the remote AVRCP CT has sent a set_absolute_volume request to local AVRCP TG.

Indication	**AT-B AVRCPABSVOLIND [volume]**	
Parameters	volume	Absolute volume, where ranges from 0 to 0x7F, as a string
Note	N/A	

25.3 AVRCPVOLUMEUPIND

The AVRCPVOLUMEUPIND indication is used to tell host the remote AVRCP CT has sent a Pass through the volume up command.

Indication	**AT-B AVRCPVOLUMEUPIND**
Parameters	N/A
Note	N/A

25.4 AVRCPVOLUMEDOWNIND

The AVRCPVOLUMEDOWNIND indication is used to tell host the remote AVRCP CT has sent a Pass through the volume down command.

Indication	**AT-B AVRCPVOLUMEDOWNIND**

Parameters	N/A
Note	N/A

26 OPP Server AT Command Definition

This chapter introduces the OPP (Server Role) relevant AT commands' definition, including a brief description of commands' syntax, responses and examples.

26.1 OPP Server Status

```
typedef enum
{
    opps_success,           /*! Last operation was successful.*/
    opps_failure,           /*! Last operation failed.*/
    opps_maxsessions,       /*! Could not open a session due to having too many sessions open. */
    opps_not_idle,          /*! A command has been attempted while another command is already running. */
    opps_wrong_command,     /*! A command has been attempted while a different multi-packet
                                command is running. */
    opps_wrong_state,       /*! Operation failed due to being in the wrong state.*/
    opps_invalid_sdp,
    opps_badrequest,
    opps_forbidden,
    opps_notfound,
    opps_aborted,
    opps_end_of_status_list
} opps_lib_status;
```

26.2 OPPSCONNRES

The OPPSCONNRES command is used to accept or reject an OPP connect request when the OPPSCONNIND indication is received.

Command	**AT+B OPPSCONNRES [accept],[maxSize]**	
Response	**AT-B OPPSCONNCFM [status], [bd], [maxSize]**	
Parameters	accept	Accept the connection request. Accept (TRUE) or Reject

		(FALSE).
	bd	Remote *Bluetooth* device address
	maxSize	The maximum packet size transferred during a session. It cannot exceed 255
Note	N/A	

26.3 OPPPUSHCRTST

The OPPCPUSHCRTST command can be used by the OPP server to get the previous indication when the OPPSPUSHSTIND indication has been received and the OPPSPUSHCMTIND indication has not been received.

Command	**AT+B OPPCPUSHCRTST**	
Response	Succeeded: **AT-B OPPSPUSHSTIND [moreData],[nameLen],[name],[objectSize],[packetSize],[packet]** Failed: **AT-B OPPCPUSHCRTST 1**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	nameLen	The length of the name field
	name	The name of the object. Unicode format, can include special characters:'nul' space '#' '*' '+' ':' '-' '_' '/' '\' '^', cannot be NULL.
	objectSize	Optional, depends on if this field of the received command sent by the Server host is not empty
	packetSize	The length of the packet field. It shall not exceed maxSize.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
Note	N/A	

26.4 OPPSPUSHNT

The OPPSPUSHNT command can be used by the server host to get the next packet when the [moreData] field of the OPPSPUSHSTIND or OPPSPUSHNTIND indication equals to one.

Command	**AT+B OPPSPUSHNT**	
Response	Succeeded: **AT-B OPPSPUSHNTIND [moreData],[packetSize],[packet]** If the [moreData] field equals to zero, the server host will also receive the following indication: **AT-B OPPSPUSHCTIND [status]** Failed: **AT-B OPPSPUSHNT 1**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	packetSize	The length of the packet field. It shall not exceed maxSize.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
	status	Values in the chapter 26.1
Note	N/A	

26.5 OPPSPUSHCRTNT

The OPPSPUSHCRTNT command can be used by the OPP server to get the previous indication when the "**OPPSPUSHNTIND**" indication has been received and the "**OPPSPUSHCMTIND**" indication has not been received.

Command	**AT+B OPPSPUSHCRTNT**	
Response	Succeeded: **AT-B OPPSPUSHNTIND [moreData],[packetSize],[packet]** Failed: **AT-B OPPSPUSHCRTNT 1**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	packetSize	The length of the packet field. It shall not exceed maxSize.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
Note	N/A	

26.6 OPPSPUSHCMT

The OPPSPUSHCMT command is used to finish the current push process. Before the server

host sends this command, the OPPCPUSHCRTST" or "OPPCPUSHCRTNT command can be used to get the previous indication. After this command is sent, the OPPSPUSHCRTST or OPPSPUSHCRTNT command shall not be used in current push process.

Command	**AT+B OPPSPUSHCMT**	
Response	Succeeded: **AT-B OPPSPUSHCMTIND [status]** Failed: **AT-B OPPSPUSHCT 1**	
Parameters	status	Values in the chapter 26.1
Note	N/A	

26.7 OPPSPULLST

The OPPSPULLST command is used by the server host to push the first business card when the server host has received the OPPSPULLSTIND indication.

Command	**AT+B OPPSPULLST [nameLen],[name],[objectSizeQ],[objectSizeR],[packetSize],[packet]**	
Response	Succeeded: If this is the only packet to be sent, the response will be: **AT-B OPPSPULLCMTIND [status]** If there are more packets to be sent, the response will be: **AT-B OPPSPULLNTIND** Failed: **AT-B OPPSPULLST 1**	
Parameters	nameLen	The length of the name field
	name	The name of the object. Unicode format. It can include special characters:'nul' space '#' '*' '+' '.' '-' '_' '/' '\' '^', cannot be NULL
	objectSizeQ	The result of objectSize/65536
	objectSizeR	The result of objectSize%65536
	packetSize	The length of the packet field. It shall not exceed maxSize.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.

Note	N/A

26.8 OPPSPULLNT

The OPPSPULLNT command is used by the server host to push the next business card to the client host when the server host has received the OPPSPULLNTIND indication.

Command	**AT+B OPPSPULLNT [last],[packetSize],[packet]**	
Response	Succeeded: If there is the last packet to be sent, the response will be: **AT-B OPPSPULLCMTIND [status]** If there are no more packets to be sent, the response will be: **AT-B OPPSPULLNTIND** Failed: **AT-B OPPSPULLNT 1**	
Parameters	last	Last to be sent. Last(TRUE) or not(FALSE)
	packetSize	The length of the packet field. It shall not exceed maxSize.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
	status	Values in the chapter 26.1
Note	N/A	

26.9 OPPSABORT

The OPPSABORT command is used by the client host to abort an object PULL or PUSH process when this process includes multiple packets.

Command	**AT+B OPPSABORT**
Response	Succeeded: **AT-B OPPSPULLCMTIND [status]** **AT-B OPPSPUSHCMTIND [status]** Failed: **AT-B OPPSABORT 1**

Parameters	status	Values in the chapter 26.1
Note	N/A	

26.10 OPPSREJECT

The OPPSREJECT command can be used by the OPP server to finish the push process when the OPPSPUSHSTIND or the OPPSPUSHNTIND indication has been received.

Command	**AT+B OPPSREJECT**	
Response	Succeeded: **AT-B OPPSPUSHCTIND [status]** Failed: **AT-B OPPSREJECT 1**	
Parameters	status	Values in the chapter 26.1
Note	N/A	

27 OPP Server Indication Definition

This chapter introduces the OPP (Server Role) relevant indications' definition.

27.1 OPPSCONNIND

The OPPSCONNIND indication is used to inform the server host that an OPP connect request has been received. The server host shall use the OPPSCONNRES command to either accept or reject the connect request.

Indication	**AT-B OPPSCONNIND [bd],[maxSize]**	
Parameters	bd	Remote *Bluetooth* device address
	maxSize	The maximum packet size transferred during a session. It cannot exceed 255.
Note	N/A	

27.2 OPPSPUSHSTIND

The OPPSPUSHSTIND indication is used to inform the server host of the first packet of an object has arrived. The server host can use the OPPSPUSHCRTST command to get the previous indication. The server host can use the OPPSPUSHNT command to get another packet or use the OPPSABORT command to abort the current object pull progress when the [moreData] field of the OPPSPUSHSTIND indication equals to one. If he [moreData] field of the OPPSPUSHSTIND indication equals to zero, the OPPSPUSHCMT command shall be used to finish current push process.

Indication	**AT-B OPPSPUSHSTIND [moreData],[nameLen],[name],[objectSize],[packetSize],[packet]**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	nameLen	The length of the name field
	name	Refer to 错误!未找到引用源。 错误!未找到引用源。. Optional, depends on if this field of the command sent by the OPP client is not empty. If client doesn't write the value of this field, the [nameLength] will equal to zero and [name] will be NULL. The indication will be as follows: AT-B OPPSPUSHSTIND 1,0,,0,0,\r.
	objectSize	Refer to 错误!未找到引用源。 错误!未找到引用源。. Optional, depends on if this field of the command sent by the OPP client is not empty. It will equal to zero if client doesn't write the value of this field.
	packetSize	The length of the packet field. It shall not exceed maxSize.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
Note	N/A	
Example	**Received:** **AT-B OPPSCONNIND 38E7D83809D0,255\CR** **Sent:** **AT+B OPPSCONNRES 1,100\CR** **Received:** **AT-B OPPSCONNCFM 0,38E7D83809D0,100\CRAT-B OPPSPUSHSTIND 1,32,\NULW\NULe\NULi\NULH\NULo\NULn\NULg\NULf\NULe\NULn\NULg\N**	

	UL.\NULv\NULc\NULf\NUL\NUL,102,0,\CR **Sent:** AT+B OPPSPUSHNT\CR **Received:** AT-B OPPSPUSHNTIND 0,102,BEGIN:VCARD\CR\LFVERSION:2.1\CR\LFN:WeiHongfeng;;;;\CR\LFFN:WeiHongfeng\CR\LFTEL;CELL;PREF:1-355-252-9153\CR\LFEND:VCARD\CR\LF\CR **Sent:** AT+B OPPSPUSHCT\CR **Received:** AT-B OPPSPUSHCMTIND 0\CRAT-B OPPSDISCIND\CR

27.3 OPPSPUSHNTIND

The OPPSPUSHNTIND indication is used to inform the server host that another packet of an object has arrived. The server host can use the OPPSPUSHCRTNT command to get the previous indication. The server host can use the OPPSPUSHNT command to get another packet or use the OPPSABORT command to abort the current pull progress when the [moreData] field of the OPPSPUSHNTIND command equals to one. If he [moreData] field of the OPPSPUSHNTIND indication equals to zero, the OPPSPUSHCMT command shall be used to finish current push process.

Indication	**AT-B OPPSPUSHNTIND [moreData],[packetSize],[packet]**	
Parameters	moreData	More data to be received. More(TRUE) or not(FALSE)
	packetSize	The length of the packet field. It shall not exceed maxSize.
	packet	The partial or complete packet of an object, cannot be NULL, cannot include '\r'.
Note	N/A	

27.4 OPPSPUSHCMTIND

The OPPSPUSHCMTIND indication is used to inform the server host that the current object push process has completed. After the server host has received this indication, it shall not use the OPPSPUSHCRTST or OPPSPUSHCRTNT command to get the previous indication.

Indication	**AT-B OPPSPUSHCMTIND [status]**	
Parameters	status	Values in the chapter 26.1
Note	Ignore the [status] field	

27.5 OPPSPULLSTIND

The OPPSPULLSTIND indication is used to inform the server host that the client host wants to pull the first packet of a business card. The server host shall use the OPPSPULLST command to push the first packet to the client host or use the OPPSABORT command to abort the pull process.

Indication	**AT-B OPPSPULLSTIND**
Parameters	N/A
Note	N/A

27.6 OPPSPULLNTIND

The OPPSPULLNTIND indication is used to inform the server host that the client host wants to pull another packet of the business card. The server host shall use the OPPSPULLNT command to push another packet to the client host or use the OPPSABORT command to abort the pull process.

Indication	**AT-B OPPSPULLNTIND**
Parameters	N/A
Note	N/A

27.7 OPPSPULLCMTIND

The OPPSPULLCMTIND indication is used to inform the server host that the client host has received a complete business card and the current pull process is finished.

Indication	**AT-B OPPSPULLCMTIND [status]**	
Parameters	status	Values in the chapter 26.1

Note	N/A

27.8 OPPSDISCIND

The OPPSDISCIND indication is used to inform the server host that the current OPP connection has been disconnected.

Indication	**AT-B OPPSDISCIND [bd]**	
Parameters	bd	Remote *Bluetooth* device address
Note	N/A	

28 PIO Assignments

28.1 Wake-up PIOs (PIO3/4)

MCU uses PIO3 of *Bluetooth* chip to wake up *Bluetooth* chip, then MCU can send UART data to the *Bluetooth* Module and the *Bluetooth* Module will stay in the active state when PIO3 is pulled high. When MCU pulls PIO3 low, the *Bluetooth* Module will go into sleep state in 5 seconds.

Bluetooth chip pulls up its PIO4 to wake up MCU before sending data to MCU, after about 5 seconds, if no data needs to be sent and PIO3 is low, the *Bluetooth* Module will automatically go into sleep state.

28.2 PCM Loop Back PIO (PIO5)

When the *Bluetooth* module receives the Play Voice Message Command, it will pull down PIO5 to turn on PCM loop back mode. When Play Voice Message Command finishes, PIO5 will automatically pull up. PIO5 default is pulled up.

28.3 DFU/Production PIO (PIO9)

When *Bluetooth* receives the PIO9 is pulled up, it will reboot into DFU mode. DFU mode's host interface is BCSP and this mode also supports production trim and power table tuning.

28.4 Disconnect HID PIO (PIO10)

PIO10 is an input. If MCU pulls up PIO10 over 3s, the current HID connection will be released.

29 *Bluetooth* Technology Best Developed Corporation

IVT Corporation is one of *Bluetooth* technology BEST developed together which is authenticated by The Bluetooth SIG. See Figure 1 below.

Figure 1 IVT is One of *Bluetooth* Technology BEST Developed Together

30 Contact Information

Contacts: Mr. Zhu Yong

Mobile: +86 18910255873

Tel: +86 10 82898219

Fax: +86 10 62963059

Email: embedded@ivtcorporation.com

Address: IVT Corporation 5/F, Fa Zhan Building No.12, Shang Di Xin Xi Road, Beijing, 100085 P.R. China

Company Site: www.ivtcorporation.com

Support: support@ivtcorporation.com

31 Copyright