Ehong Control Interface (EHCI) User Guide

Ver.0.2
May 14, 2014

Release Record

Version	Release Date	Comments
0.1	Mar 03, 2014	Draft
0.2	May 6, 2014	1. Change the command and indicator for PIO assignment from PO to PM. 2. Correct some typo.

Table of Contents

1. Introduction

The Ehong Control Interface (EHCI) is a set of ASCII commands and indicators with which the user can control the Ehong's Bluetooth module via UART interface by a host (PC, MCU, etc.).
The commands are used to control the Bluetooth module sent by host. The indicators are output from the Bluetooth module to the host to indicate the status of the module.
In addition, there are some IO indicators available when the UART is used to transfer raw data (working in Bypass mode). As a complement of ASCII commands and indicators, the IO indicators are also a part of EHCI.

1.1. Default UART Configuration

The default configuration of UART is given below:
Baud rate: 9600
Data bits: 8
Stop bits: 1
Parity: None
Flow control: None

2. Command and Indicator Syntax

2.1. General Syntax

The general syntax of EHCI command is shown as below:
AT+CMD[=Para1][,Para2][,RawData][,…]<CR><LF>

The general syntax of EHCI indicator is shown as below:
IDC[=Para1][,Para2][,RawData][,…]<CR><LF>

Description of each field:
AT+ is the command line prefix.
CMD is the basic command. All of the commands are listed in section 3 .
IDC is the basic indicator. All of the indicators are listed in section 4 .
= is the separator between command/indicator and parameter. It's only needed if a parameter is presented.
Para1 is the first parameter. Not all of the commands have a parameter.
, is the separator between parameters. It's only needed if subsequent parameter is presented.
Para2 is the second parameter if available.
RawData is the raw data which will be sent by the command. Only parts of the commands have this field.
<CR><LF> is the terminator of the command line.
Notes:

1. *If a parameter is mandatory, it will be surrounded by { }. If a parameter is optional, it will be surrounded by [ ].*
2. *<CR> means Carriage Return, and <LF> means Line-Feed.*
3. *All of the parameters are composed of ASCII characters while the **RawData** field can composed of any data contents.*

2.2. Examples

Here is some examples show how to use the EHCI commands and indicators.
Ex. 2.1
➔ AT+FT=01,00,00,0A,01,0078<CR><LF> ← *configure the module features.*

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Notes:

1. For the examples in this document, the command sent to the Bluetooth module will be shown with "➔" at the beginning of the line, while the indicator output by Bluetooth module will be shown with "←" at the beginning of the line.

2. For the examples in this document, the comments will start with a "←" and be written in italic.

3. For the examples in this document, only the characters in grey background color are the real content of a command or indicator.

Ex. 2.2

➔ AT+CS=00189600ABCD<CR><LF> ← *establish SPP connection with the device which address is 00:18:96:00:AB:CD.*

← SS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the specified device which address is 00:18:96:00:AB:CD.*

← CS=00,00189600ABCD<CR><LF> ← *connecting result: success.*

← SS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the specified device which address is 00:18:96:00:AB:CD.*

3. Command List

All the available EHCI commands are listed and briefly described in the tables below. The detailed description of each command can be given in chapter 5 .

Table 3.1 EHCI Command List

Command	Short Description	Comments
General Commands		
PF	Query or configure the profiles of the module.	
AD	Query the Bluetooth address of the module.	
TP	Query or change the Tx Power of the module.	
CD	Query or configure the Class of Device of the module.	
FT	Query or configure the features of the module.	
MM	Query or configure Man-In-The-Middle protection feature.	
IO	Query or configure IO capability of local device.	
MT	Query or configure force to be master feature.	
SN	Query or configure the sniff mode.	
SP	Query or change the deep sleep mode.	
PN	Query or change the fixed pin code of the module.	
NM	Query or change the local friendly name of the module.	
IF	Query or change the host interface	
BR	Query or change the UART baud rate.	
UM	Query or change the UART mode.	
UI	Query or change the UART indicator output mode.	
RC	Query or change the remote control function.	
PM	Query or change the PIO assignment	
DB	Query or change the default bypass mode.	
PW	Power on or power off.	
IQ	Inquiry the Bluetooth device.	
MD	Query or change the state of discoverable mode.	
PA	Query or change the state of pairing mode.	
CA	Query or change the state of connectable mode.	
NC	Confirm or deny the numeric comparison.	
PK	Input the Passkey when pairing.	
CP	Clear the paired Bluetooth device list.	
CT	Connect to remote Bluetooth device.	
DC	Disconnect with remote Bluetooth device.	
BP	Configure the runtime bypass mode.	
PC	Read or write the PIO value.	
AC	Read voltage of AIO	
VU	Increase the Volume.	

VD	Decrease the Volume.	
LC	List connected devices	
FU	Make the module enter into DFU mode.	
TS	Make the module enter into Test mode.	
SPP Commands		
SM	Query or change the service name of SPP profile.	
CS	Connect to the remote SPP device.	
DS	Disconnect with the remote SPP device.	
SS	Query the SPP state of each SPP instance.	
DT	Send data packet to remote SPP device.	
HID Commands		
CI	Connect to the remote HID host.	
DI	Disconnect with the remote HID host.	
IS	Query the HID state.	
KR	Send keyboard report to remote HID host.	
AS	Send ASCII string to remote HID host.	
RFCOMM Commands (for Apple iOS devices)		
RM	Query or change the service name of RFCOMM profile.	
PT	Query or change the protocol name of iAP application.	
AH	Query the status of Apple authentication processor.	
CR	Connect to the remote RFCOMM device.	

DR	Disconnect with the remote RFCOMM device.	
RS	Query the RFCOMM state.	
RD	Send data packet to remote RFCOMM device.	
DUN Commands		
NN	Query or change the service name of DUN	
CN	Connect to a remote DUN DCE device	
DN	Disconnect with the remote DUN device	
NS	Query the DUN state	
ND	Send data packet to remote DUN device	
OPP Commands		
CO	Connect to a remote OPPS device	
DO	Disconnect with the remote OPP device	
OS	Query the OPP state	
OA	OPPC push object file name	
OY	OPPC push object file type	
OT	OPPC push object packet data	
BLE Commands		
GA	Query or change the GAP appearance characteristic.	
LP	Query or change the preferred connection parameters characteristic.	
AT	Query or change the advertising state.	
DL	Disconnect with remote BLE central device.	
LS	Query the state of BLE channel.	
LI	Query the RSSI of BLE connection.	
LD	Send data packet to remote BLE central device.	
HFP Commands		
CH	Connect to the remote HFP device.	
DH	Disconnect with the remote HFP device.	
HS	Query the state of HFP channel.	
AR	Accept or Reject call.	
HU	Hang Up call.	
TC	Transfer call.	
MU	Mute or unmute the microphone.	
LR	Last numbers redial.	

HV	Check or set the volume of HFP voice	
A2DP Commands		
OD	Query of change the optional decoder used by A2DP.	
CM	Connect to the remote A2DP source device.	
DM	Disconnect with the remote A2DP source device.	
MS	Query the state of A2DP.	
TM	Toggle the A2DP media source channel.	
MR	Switch the audio output route.	
MV	Check or set the volume of A2DP music.	
AVRCP Commands		

NP	Query or change the status of NowPlaying function.	
CV	Connect to the remote AVRCP target device.	
DV	Disconnect with the remote AVRCP target device.	
VS	Query the AVRCP state.	
PL	Send a Play/Pause command to remote AVRCP device.	
ST	Send a Stop command to remote AVRCP device.	
NX	Send a Next command to remote AVRCP device.	
PR	Send a Previous command to remote AVRCP device.	
FF	Send a Fast Forward command to remote AVRCP device.	
RW	Send a Rewind command to remote AVRCP device.	
TV	Toggle the AVRCP target channel.	

4. Indicator List

All the available EHCI indicators are listed and briefly described in the tables below. The detailed description of each command can be given in chapter 6 .

Table 4.1 EHCI Indicator List

Indicator	Short Description	Comments
General Indicators		
OK	Indicates a command was adopted by the module.	
ER	Indicates there is an error detected in the command sent by the host.	
AP	State of Bluetooth module as an application.	
AD	Bluetooth address of the module.	
TP	Tx Power of the module	
CD	Class of Device of the module.	
PF	Configuration of profiles of the module.	
FT	Features of the module.	
MM	States of Man-In-The-Middle protection.	
IO	Configuration of IO capability of local device.	
MT	Configuration of force to be master feature.	
SN	Configuration of sniff feature.	
SP	The deep sleep state.	
PN	Fixed pin code of the module.	
NM	Local friendly name of the module.	
IF	Host interface of the module	
BR	UART baud rate.	
UM	Configuration of UART mode.	
UI	Configuration of UART indicator output.	
RC	Configuration of remote control function.	
PM	Configuration of PIO assignment	
DB	Default configuration of bypass mode.	
MD	Discoverable state.	

PA	State of pairing mode.	

CA	State of connectable mode	
NC	Six digit number of numeric comparison.	
PK	Passkey request.	
PC	Status of PIO	
AC	Voltage of AIO	
IR	Inquiry result.	
FD	Address and name of found device.	
LC	List the connected devices	
SPP Indicators		
SM	Service name of the SPP profile.	
SS	State of SPP channel.	
CS	Result of connect attempt to a remote SPP device.	
DT	Data packet received from remote SPP device.	
HID Indicators		
IS	State of HID.	
CI	Result of connect attempt to a remote HID host.	
KR	Keyboard report received from remote HID host.	
RFCOMM Indicators(for Apple iOS devices)		
RM	Service name of the RFCOMM profile.	
PT	Protocol name of iAP application.	
AH	Status of Apple authentication processor.	
SO	State of iAP data session.	
RS	State of RFCOMM channel.	
CR	Result of connect attempt to a remote RFCOMM device.	
RD	Data packet received from remote RFCOMM device.	
DUN Indicators		
NN	Service name of the DUN profile.	
NS	State of DUN channel.	
CN	Result of connect attempt to a remote DUN device.	
ND	Data packet received from remote DUN device.	
OPP Indicators		
OS	State of OPP channel.	
CO	Result of connect attempt to a remote OPP device.	
OA	Object name pushed by an OPPC device	
OY	Object type pushed by an OPPC device	
OT	Object data packet pushed by an OPPC device	
BLE Indicators		
LA	Configuration of optional decoder used by A2DP.	
LN	BLE device name of the module	
GA	BLE GAP appearance characteristic	
LP	BLE preferred connection parameters characteristic	
AT	State of Advertising	
LS	State of BLE channel	

LI	RSSI of BLE connection	
LD	Data packet received from remote BLE central device	
HFP Indicators		
HS	State of HFP channel.	
CH	Result of connect attempt to a remote HFP device.	
CC	Call State	
HV	Volume of HFP voice	
A2DP Indicators		

OD	Configuration of optional decoder used by A2DP.	
MS	State of A2DP channel	
CM	Result of connect attempt to a remote A2DP source device.	
PL	Status of A2DP playing.	
DD	Decoder used by A2DP.	
SR	Simple rate of A2DP audio	
MR	Audio output route	
MV	Volume of A2DP music	
AVRCP Indicators		
NP	Status of NowPlaying function	
VS	State of AVRCP.	
CV	Result of connect attempt to a remote AVRCP target device.	
VC	Capabilities of AVRCP target device.	
TC	Track changed event.	
PO	Playback position changed event.	
MA	Media attributes.	

5. Description of ASCII Commands

5.1. General Commands

5.1.1. PF—Query or configure the profiles

5.1.1.1. Description:

This command can query or configure the profiles of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost. If the new configuration is adopted by the Bluetooth module, the module will perform a reboot, the non-memorable settings will return to their default value. Therefore, it is recommended to send this command first if necessary.
If the parameter is not presented, the Bluetooth module will report current profile configuration by the Indicator PF.

5.1.1.2. Syntax:

Synopsis:	Comments
AT+PF[=SppCnt][,HidCnt][,RfcCnt][,DunRole][OppRole]<CR><LF>	For MA41 and MA46
AT+PF[=SppCnt][,HidCnt][,RfcCnt][,HfpCnt][,A2dpCnt][,AvrcpCnt]<CR><LF>	For MB05, and MB18

5.1.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
SppCnt	The maximum SPP instance count. Value: 00h—07h Default: 05	O	For MB05, and MB18, the maximum SPP instance count is 01.

HidCnt	The maximum HID instance count. Value: 00h—01h Default: 01	O	
RfcCnt	The maximum RFCOMM instance count. Value: 00h—07h Default: 01	O	For MB05, and MB18, the maximum RFCOMM instance count is 01.
DunRole	The role of DUN profile Value: 00h—02h 00: Disable the DUN profile 01: The module works as DUN DCE(Modem) 02: The module works as DUN DTE(Data Terminal) Default: 00	O	Only available for MA41 and MA46.
OppRole	The role of OPP profile Value: 00h—03h 00: Disable the OPP profile 01: The module works as an OPP Client only(OPPC) 02: The module works as an OPP Server only(OPPS) 03: The module works as both OPP Client and Server(OPPC and OPPS). Default: 00	O	Only available for MA41 and MA46.
HfpCnt	The maximum HFP instance count. Value: 00h—02h Default: 00	O	Only available for MB05, and MB18.
A2dpCnt	The maximum A2DP instance count. Value: 00h—02h Default: 01	O	Only available for MB05, and MB18.
AvrcpCnt	The maximum AVRCP instance count. Value: 00h—02h Default: 01	O	Only available for MB05, and MB18.

Notes:

1. The default profile configuration may be different per software version.

2. The total instance of all profiles should no more than 7 according to Bluetooth Spec.

3. For HFP, A2DP and AVRCP profiles, if one of them is set to 02, then the other two profiles must be set to either 00

or 02.

5.1.1.4. Examples:

Ex. 5.1. To query current profile configuration of Bluetooth module (MA41 or MA46):

➔ AT+PF<CR><LF> ← *query current profile configuration.*

← PF=05,01,01,00,00<CR><LF> ← *report current profile configuration: 5 SPP instance,1 HID instance,1 RFCOMM instance.*

Ex. 5.2. To configure the features of Bluetooth module (MA41 or MA46):

➔ AT+PF=04,00,00,00,00<CR><LF> ← *configure the module profiles: 4 SPP instance and no HID and RFCOMM profile supported.*

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

← AP=00<CR><LF> ← *Indicate that the Bluetooth has performed a reboot and is ready now.*

Ex. 5.3. To configure the features of Bluetooth module (MB05 or):
➔ AT+PF=01,00,01,00,02,02<CR><LF> ← *configure the module profiles: 1 SPP instance, 1 RFCOMM instance, 2 A2DP instance and 2 AVRCP instance supported.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth has performed a reboot and is ready now.*

5.1.2. AD—Query the Bluetooth address

5.1.2.1. Description:

This command can query the Bluetooth address of local module. Once the Bluetooth module adopted this query request, it will report its Bluetooth address by the Indicator AD.

5.1.2.2. Syntax:

Synopsis:

AT+AD<CR><LF>

5.1.2.3. Parameter Description:

None.

5.1.2.4. Examples:

Ex. 5.4. To query the Bluetooth address of local module:
➔ AT+AD<CR><LF> ← *query the Bluetooth address of local module.*
← AD=00189600ABCD<CR><LF> ← *report the Bluetooth address is 00:18:96:00:AB:CD.*
← LA=00,00189680ABCD<CR><LF> ← *report the BLE address is 00:18:96:80:AB:CD, the address type is public. This is only available for MB18 module.*

5.1.3. TP—Query or change the Tx Power

5.1.3.1. Description:

This command can query or configure the transmit power of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current transmit power configuration by the Indicator TP.

5.1.3.2. Syntax:

Synopsis:

AT+TP[=DefaultTx][,MaximumTx]<CR><LF>

5.1.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

DefaultTx	Default TX power in dBm. The default TX power used for paging, inquiry, and their responses, and as the initial power for new ACL links. Value: a 8 digits signed number Default: per firmware version and Bluetooth power class.	O	The values of this parameter will always be rounded to the next available value in the radio power table.
MaximumTx	Maximum TX power in dBm. Bluetooth power control may raise the TX power up to this value. Value: a 8 digits signed number Default: per firmware version and Bluetooth power class.	O	The values of this parameter will always be rounded to the next available value in the radio power table.

Notes:

1. Please do NOT change the default configuration of Tx Power if not necessary.

5.1.3.4. Examples:

Ex. 5.5. To query current Tx Power configuration of Bluetooth module:
➔ AT+TP<CR><LF> ← *query current Tx Power configuration.*
← TP=04,04<CR><LF> ← *report current Tx Power configuration: default Tx Power is 4dBm and maximum Tx Power is 4dBm.*

Ex. 5.6. To configure the *Tx Power* of Bluetooth module:
➔ AT+TP=FC,00<CR><LF> ← *configure the Tx Power, set default Tx Power to -4dBm(FCh = -4), set the maximum Tx Power to 0dBm.*
← TP=FC,00<CR><LF> ← *response from the module to indicate the command is adopted and report the real Tx Power configuration after rounded.*

5.1.4. CD—Query or configure the Class of Device

5.1.4.1. Description:

This command can query or configure the Class of Device (COD) of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command or the module is rebooted. It means the Bluetooth module will not remember the configuration, and after the Bluetooth module has been powered off, the configuration will be lost.
If the parameter is not presented, the Bluetooth module will report current COD by the Indicator CD.

5.1.4.2. Syntax:

Synopsis:

AT+CD[=Cod]<CR><LF>

5.1.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Cod	The COD of the Bluetooth module Value: a 6 digits number Default: per firmware version	O	

Notes:

2. The default COD has been configured properly by the Bluetooth firmware stack, so it is not

necessary for user to configure it in general.

3. Some Bluetooth device will filter the devices by COD when searching for new device.

5.1.4.4. Examples:

Ex. 5.7. To query current COD configuration of Bluetooth module:
➔ AT+CD<CR><LF> ← *query current COD configuration.*
← CD=001F00<CR><LF> ← *report current COD configuration: 001F00.*

Ex. 5.8. To configure the COD of Bluetooth module:
➔ AT+CD=000540<CR><LF> ← *configure the module COD: 000540.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.5. FT—Query or configure the features

5.1.5.1. Description:

This command can query or configure the features of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current feature configuration by the Indicator FT. If the user wants to configure the features, all of the parameters should be given together.

5.1.5.2. Syntax:

Synopsis:
AT+FT[=ATPowerOn,ACPaired,ATLinkLost,Interval,DiscMode,DiscTimeout]<CR><LF>

5.1.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ATPowerOn	The attempt times of auto connect the last connected device after power on. Value: 00h—FFh 00: No auto connect attempt will be performed after power on. 01-FE: The attempt times of auto connect after power on. FF: The auto connect attempt will be performed permanently. Default: FF (Permanent)	O	
ACPaired	Auto connects after paired with a device. Value: 00 or 01 00: Disabled 01: Enabled Default: 00 (Disabled)	O	
ATLinkLost	The attempt times of reconnect after link lost. Value: 00h—FFh 00: No reconnect attempt will be performed after link lost. 01-FE: The attempt times of reconnect after link lost. FF: The reconnect attempt will be performed permanently. Default: FF (Permanent)	O	

Interval	The interval between each reconnect attempt after link lost. The unit is second. Value: 00h—FFh Default: 0A (10 seconds)	O	
DiscMode	The discoverable mode. Value: 00h—03h 00: The module will enter or quit discoverable mode just by the command AT+MD=xx. 01: The module will enter discoverable mode automatically when paired device list is empty. 02: The module will enter discoverable mode automatically when power on. 03: The module will enter discoverable mode automatically when there is no connection. Default: 01 (Auto discoverable when empty)	O	Even if the discoverable is set one of the auto mode (01 h—03h), it can also be controlled by the command AT+MD=xx.
DiscTimeout	The timeout of discoverable status. The unit is second. Value: 0000h—FFFFh 0000: No timeout for discoverable status.	O	
	0001-FFFF: The timeout in second of discoverable status.		

Notes:

1. The default feature configuration may be different per software version.

5.1.5.4. Examples:

Ex. 5.9. To query current feature configuration of Bluetooth module:
➔ AT+FT<CR><LF> ← *query current feature configuration.*
← FT=FF,00,FF,0A,01,0078<CR><LF> ← *report current feature configuration.*
The auto connection after power on has been enabled as permanent mode;
The auto connect after paired has been disabled;
The auto reconnect after link lost has been enabled as permanent mode;
The interval of auto reconnect has been set to 10s.
Set the discoverable mode as auto discoverable when empty.
The timeout of discoverable is 120s.

Ex. 5.10. To configure the features of Bluetooth module:
➔ AT+FT=14,00,00,0A<CR><LF> ← *configure the module features:*
Set the attempt time of auto connect after power on as 20 times;
Disable the auto connect after paired;
No reconnect attempt will be performed after link lost;
Set the interval of auto reconnect to 10s.
Keep the discoverable mode and timeout as it was.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.6. MM—Query or configure Man-In-The-Middle protection feature

5.1.6.1. Description:

This command can query or configure the Man-In-The-Middle protection feature of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost. If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator MM.

5.1.6.2. Syntax:

Synopsis:

AT+MM[=State]<CR><LF>

5.1.6.3. Parameter Description:

Parameter	Description	Mandatory	Comments
		or Optional	
State	The new state of Man-In-The-Middle protection. Value: 00h or 02h 00: Deactivated 01: Activated 02: Activated and auto confirm the numeric comparison. Default: 02 (Activated and auto confirm)	O	The default value may be different per software version.

Notes:

1. A man-in-the-middle (MITM) attack occurs when a user wants to connect two devices but instead of connecting directly with each other they unknowingly connect to a third (attacking) device that plays the role of the device they are attempting to pair with. The third device then relays information between the two devices giving the illusion that they are directly connected. The attacking device may even eavesdrop on communication between the two devices (known as active eavesdropping) and is able to insert and modify information on the connection. In this type of attack, all of the information exchanged between the two devices are compromised and the attacker may inject commands and information into each of the devices thus potentially damaging the function of the devices. Devices falling victim to the attack are capable of communicating only when the attacker is present. If the attacker is not active or out range, the two victim devices will not be able to communicate directly with each other and the user will notice it.

To prevent MITM attacks, Secure Simple Pairing offers two user assisted numeric methods: numerical comparison or passkey entry. If Secure Simple Pairing would use 16 decimal digit numbers, then the usability would be the same as using legacy pairing with 16 decimal digit PIN. The chance for a MITM to succeed inserting its own link keys in this case is a 1 in 1016 = 253 pairing instances, which is an unnecessarily low probability.
Secure Simple Pairing protects the user from MITM attacks with a goal of offering a 1 in 1,000,000 chance that a MITM could mount a successful attack. The strength of the MITM protections was selected to minimize the user impact by using a six digit number for numerical comparison and Passkey entry. This level of MITM protection was selected since, in most cases, users can be alerted to the potential presence of a MITM attacker when the connection process fails as a result of a failed MITM attack. While most users feel that provided that they have not compromised their passkey, a 4-digit key is sufficient for authentication (i.e. bank card PIN codes), the use of six digits allows Secure Simple Pairing to be FIPS compliant and this was deemed to have little perceivable usability impact.
If the Man-In-The-Middle protection feature is activated, the module may output the number for numeric comparison by indicator NC or a passkey request by indicator PK. About the command NC and PK, please refer to section 5.1.24 and 5.1.25.

*2. If the Man-In-The-Middle protection feature is activated, the IO capability can only be configured to "**Display Yes/No**"* or "***Keyboard Only***". About the IO capability, please refer to

section 5.1.7 .

3. When connect with some Android device by the SPP profile, it is required to active the Man-In-The-Middle protection.

5.1.6.4. Examples:

Ex. 5.11. To query current Man-In-The-Middle protection state of the Bluetooth module:
➔ AT+MM<CR><LF> ← *query the current Man-In-The-Middle protection state.*
← MM=00<CR><LF> ← *report the Man-In-The-Middle protection is deactivated currently.*

Ex. 5.12. To active Man-In-The-Middle protection feature:
➔ AT+MM=01<CR><LF> ← *active Man-In-The-Middle protection feature.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.13. To active Man-In-The-Middle protection feature, and let the module confirm the numeric comparison automatically:
➔ AT+MM=02<CR><LF> ← *active Man-In-The-Middle protection feature and auto confirm the numeric comparison. Thus, no NC indicator will be output by the module.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.7. IO—Query or configure the IO capability of local device

5.1.7.1. Description:

This command can query or configure the IO (input and output) capability of local device when pairing. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost. If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator IO.

5.1.7.2. Syntax:

Synopsis:
AT+IO[=IoCapability]<CR><LF>

5.1.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
IoCapability	The new IO capability of local device. Value: 00h – 03h 00: Display Only. The local device can only display 01: Display Yes/No. The local device can display and select Yes or No. 02: Keyboard Only. The local device can only input. 03: No IO. The local device has no IO capability Default: 01 (Display Yes/No)	O	The default value may be different per software version.

Notes:

*1. When the Man-In-The-Middle protection feature is enabled, different IO capability will cause different pairing procedure. In case of "**Display Yes/No**", both remote and local device will prompt a six digits numbers, the user should compare and confirm if the two numbers are the same or not, and then select **Yes** or **No** on the remote device accordingly, for the module side, the host MCU should send the command AT+NC=01 or AT+NC=00 to confirm or deny the numeric*

*comparison. In case of "**Keyboard Only**", the remote device will prompt a six digits number as passkey, the user should input the same number at the Bluetooth module side by command AT+PK (refer to section 5.1.25).*

*2. The "**Display Only**" and "**No IO**" are not allowed when the Man-In-The-Middle protection feature is enabled.*

5.1.7.4. Examples:

Ex. 5.14. To query current IO capability configuration of local device:
➔ AT+IO<CR><LF> ← *query current IO capability configuration of local device.*
← IO=03<CR><LF> ← *report current IO capability configuration of local device is "No IO".*

Ex. 5.15. To configure the IO capability of local device as "Keyboard Only":
➔ AT+IO=02<CR><LF> ← *configure the IO capability of local device as "Keyboard Only".*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.8. MT—Query or configure force to be master feature

5.1.8.1. Description:

This command can query or configure the force to be master feature of Bluetooth module. Once configured, the configuration will take effect at the next time when a Bluetooth connection is being established and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator MT.

5.1.8.2. Syntax:

Synopsis:
AT+MT[=State]<CR><LF>

5.1.8.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of force to be master feature. Value: 00h or 01h 00: Deactivated 01: Activated Default: 00 (Deactivated)	O	

Notes:

1. In general, the device which initiates the Bluetooth connection will act as the Master automatically. Only some special devices which cannot be a master device, in such cases, the user can use this command to make the Bluetooth module force to be master device.

2. Ehong's Bluetooth module can act as either Master or Slave device, i.e. it can either initiate a Bluetooth connection or accept a connection request.

5.1.8.4. Examples:

Ex. 5.16. To query current state of force to be master feature:
➔ AT+MT<CR><LF> ← *query current state of force to be master feature.*

← MT=00<CR><LF> ← *report the force to be master feature is deactivated currently.*

Ex. 5.17. To active the force to be master feature:
→ AT+MT=01<CR><LF> ← *active the force to be master feature.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.9. SN—Query or configure the sniff mode

5.1.9.1. Description:

This command can query or configure the sniff mode of Bluetooth module. Once configured, the configuration will take effect at the next time when a Bluetooth connection is being established and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator SN.

5.1.9.2. Syntax:

Synopsis:
AT+SN[=State][,MinInterval][,MaxInterval][,Attempt][,Timeout][,PassiveDuration]<CR><LF>

5.1.9.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of sniff mode. Value: 00h or 01h 00: Deactivated 01: Activated Default: 00 (Deactivated)	O	
MinInterval	Minimum acceptable interval in milliseconds Value: 0002h—FFFEh; only even values, up to max, are valid Time = MinInterval x 0.625 ms Time Range: 1.25 ms to 40959 ms	O	
MaxInterval	Maximum acceptable interval in milliseconds Value: 0004h—FFFEh; only even values, up to max, are valid Time = MaxInterval x 0.625 ms Time Range: 2.5 ms to 40959 ms	O	
Attempt	Number of slots the slave shall listen when the slave is not treating this as a scatternet link. Value: 0001h—7FFFh Time = Attempt x 1.25 ms Time Range: 1.25ms to 40959 ms	O	
Timeout	Number of additional slots the slave shall listen when the slave is not treating this as a scatternet link. Value: 0001h—7FFFh Time = Timeout x 1.25 ms Time Range: 1.25ms to 40959 ms	O	

PassiveDuration	The time in seconds that the module will keep in Passive mode Value: 0001—FFFFh Time range: 1 seconds to 65535 seconds	O	

5.1.9.4. Examples:

Ex. 5.18. To query current sniff mode of the Bluetooth module:
➔ AT+SN<CR><LF> ← *query the current state of sniff mode.*
← SN=00,0320,0320,0004,0001,0004<CR><LF> ← *report the sniff mode is deactivated currently,*
the MinInterval is 500ms(800x0.625ms),
the MaxInterval is 500ms(800x0.625ms),
the Attempt is 5ms,
the Timeout is 1.25ms,
the Passive duration is 4 seconds,

Ex. 5.19. To active the sniff mode:
➔ AT+SN=01<CR><LF> ← *active the sniff mode.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.20. To active the sniff mode with specified parameters:
➔ AT+SN=01,0320,0320,0004,0001,000A<CR><LF> ← *active the sniff mode with the specified parameters:*
the MinInterval is 500ms(800x0.625ms),
the MaxInterval is 500ms(800x0.625ms),
the Attempt is 5ms,
the Timeout is 1.25ms,
the Passive duration is 10 seconds,
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.10. SP—Query or change the deep sleep mode

5.1.10.1. Description:

This command can query or change the Bluetooth module's deep sleep mode.
If the parameter is not presented, the Bluetooth module will report current deep sleep state by the Indicator SP.

5.1.10.2. Syntax:

Synopsis:

AT+SP[=State]<CR><LF>

5.1.10.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of deep sleep. Value: 00h or 01h	O	

00: Deep sleep disabled
01: Deep sleep enabled
Default: 00 (disabled)

Notes:

1. In deep sleep mode, the Bluetooth module will save power because the firmware will not

need to calibrate the slow clock against the standard 20 ppm clock after the frequency of the slow clock has been determined at boot.

2. Deep sleep may be entered when an ACL connection is in hold, sniff or park mode; hence accuracy may be lost when using the internal low power slow speed clock. In certain situations, where power saving is not a major priority but maintaining accuracy of the clock is (for example, a network access point which has a large number of parked connections), it may be advantageous to disable deep sleep mode with this command.

3. In deep sleep mode, the data sent to UART port of the Bluetooth module maybe lost. So, it is required to wake up the module first. Therefore, the firmware is designed that any command can and can only wake up the module when it is in deep sleep mode. But, to avoid any ambiguous, it is recommended to use AT+SP=00 command to wake up the module.

5.1.10.4. Examples:

Ex. 5.21. To query the current deep sleep state of the Bluetooth module:
➔ AT+SP<CR><LF> ← *query the current deep sleep state.*
← SP=00<CR><LF> ← *report the deep sleep is disabled currently.*

Ex. 5.22. To enable the deep sleep mode of the Bluetooth module:
➔ AT+SP=01<CR><LF> ← *enable the deep sleep mode.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.23. To wake up the Bluetooth module from deep sleep:
➔ AT+SP=00<CR><LF> ← *wake up the Bluetooth module from deep sleep.*
← SP=00<CR><LF> ← *response from the module to indicate the module is waked up and the deep sleep mode has been disabled.*

5.1.11. PN—Query or change the fixed pin code

5.1.11.1. Description:

This command can query or change the fixed pin code of Bluetooth module. Once changed, the new pin code will take effect at next pairing procedure and until the next time the pin code is changed by this command. It means the Bluetooth module will remember the pin code, and even if the Bluetooth module has been powered off, the pin code will not be lost.
If the parameter is not presented, the Bluetooth module will report current pin code by the Indicator PN.

5.1.11.2. Syntax:

Synopsis:
AT+PN[=PinCode]<CR><LF>

5.1.11.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
PinCode	The new fixed pin code of the Bluetooth module. Length: 1—16 characters Default: 0000	O	The default pin code may not be “0000” per software version.

5.1.11.4. Examples:

Ex. 5.24. To query current fixed pin code of Bluetooth module:
➔ AT+PN<CR><LF> ← *query current fixed pin code.*

← PN=0000<CR><LF> ← *report current fixed pin code, it's "0000".*

Ex. 5.25. To change the fixed pin code of Bluetooth module:
→ AT+PN=abcdef<CR><LF> ← *change the fixed pin code to "abcdef"*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.12. NM—Query or change the local friendly name

5.1.12.1. Description:

This command can query or change the local friendly name of Bluetooth module. Once changed, the new friendly name will take effect at next time the remote device get local name and until the next time the friendly name is changed by this command. It means the Bluetooth module will remember the friendly name, and even if the Bluetooth module has been powered off, the friendly name will not be lost.
If the parameter is not presented, the Bluetooth module will report current friendly name by the Indicator NM.

5.1.12.2. Syntax:

Synopsis:
AT+NM[=Name]<CR><LF>

5.1.12.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The new local friendly name of the Bluetooth module. Length: 1—30 characters Default: Per software version.	O	

5.1.12.4. Examples:

Ex. 5.26. To query current local friendly name of Bluetooth module:
→ AT+NM<CR><LF> ← *query current local friendly name.*
← NM=NVC_BT_DEVICE<CR><LF> ← *report current local friendly name, it's "NVC_BT_DEVICE".*
← LN=NVC_BLE_DEVICE<CR><LF> ← *report current local friendly name of the BLE channel, it's "NVC_BLE_DEVICE". This is only available for MB18 module.*

Ex. 5.27. To change the local friendly name of Bluetooth module:
→ AT+NM=MY_BT_DEVICE<CR><LF> ← *change the local friendly name to "MY_BT_DEVICE"*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
Notes:
1. For MB18 module, the module will append "_L" at the end of given name as the BLE device name. That means, in the above example, the BLE device name will be changed to "MY_BT_DEVICE_L".

5.1.13. IF—Query or configure the host interface

5.1.13.1. Description:

This command can query or configure host interface of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator IF.

5.1.13.2. Syntax:

Synopsis:
AT+IF[=HostInterface]<CR><LF>

5.1.13.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
HostInterface	The new host interface Value: 00h—03h 00: UART running EHCI 01: USB CDC running EHCI 02: UART running H4 03: USB HCI Default: 00 (UART)	O	The interface 02 and 03 is only available for MA41 and MA46.

5.1.13.4. Examples:

Ex. 5.28. To query current host interface of the Bluetooth module:
➔ AT+IF<CR><LF> ← *query current host interface.*
← IF=00<CR><LF> ← *report the current host interface is UART.*
Notes:
1. If the UART indicator output is disabled currently, the report will not be output.

Ex. 5.29. To change the host interface to USB CDC:
➔ AT+IF=01<CR><LF> ← *change the host interface to USB CDC.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
Notes:
1. Once the host interface has been changed, the communication between the module and the host(a PC or a MCU) will be handled by the new interface. That means, if you change the host interface to USB CDC, you have to use the USB communication to control the module.

2. Ehong provides an USB CDC driver for Windows PC. After installed this driver, a virtual COM port will be presented in the Device Manager, the user can use a Serial Tool to open this COM port for sending and receive data between PC and module.

5.1.14. BR—Query or change the UART baud rate

5.1.14.1. Description:

This command can query or change the UART baud rate of Bluetooth module. Once changed, the new baud rate will take effect immediately and until the next time the baud rate is changed by this command. It means the Bluetooth module will remember the baud rate, and even if the Bluetooth module has been powered off, the baud rate will not be lost.
If the parameter is not presented, the Bluetooth module will report current baud rate by the Indicator BR.

5.1.14.2. Syntax:

Synopsis:

AT+BR[=BaudRate]<CR><LF>

5.1.14.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BaudRate	The new baud rate of the Bluetooth module. Value: 01h—15h 01: 1200 02: 1800 03: 2400 04: 4800 05: 7200 06: 9600 07: 14400 08: 19200 09: 38400 0A: 56000 0B: 57600 0C: 115200 0D: 128000 0E: 230400 0F: 256000 10: 460800 11: 921600 12: 1382400 13: 1843200 14: 2764800 15: 3686400	O	The default baud rate may not be 9600 per software version.

Default: 06 (9600)

Warning:

1. Please do NOT try to change to a new baud rate if you don't have a host which can work in that baud rate, for there is no other way to restore it except for UART port.

5.1.14.4. Examples:

Ex. 5.30. To query the baud rate of Bluetooth module:
➔ AT+BR<CR><LF> ← *query the baud rate.*
← BR=06<CR><LF> ← *report the baud rate, it's 9600.*

Ex. 5.31. To change the baud rate of Bluetooth module:
➔ AT+BR=0C<CR><LF> ← *change the baud rate to 115200.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
Notes:
1. The response will be sent in current baud rate.

5.1.15. UM—Query or configure the UART mode

5.1.15.1. Description:

This command can query or configure the UART mode of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator UM.

5.1.15.2. Syntax:

Synopsis:
AT+UM[=StopBits,Parity][,Latency]<CR><LF>

5.1.15.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
StopBits	The stop bits of UART mode Value: 00h or 01h 00: 1 stop bit 01: 2 stop bits Default: 00 (1 stop bit)	O	
Parity	The parity of UART mode Value: 00h – 02h 00: No parity 01: Odd parity 02: Even parity Default: 00 (No parity)	O	
Latency	The latency mode Value: 00h or 01h 00: Throughput priority 01: Low latency priority Default: 01(Low latency priority)	O	

5.1.15.4. Examples:

Ex. 5.32. To query the UART mode of Bluetooth module:
➔ AT+UM<CR><LF> ← *query the UART mode.*
← UM=00,00,01<CR><LF> ← *report the UART mode, it's 1 stop bit, no parity and low latency priority.*

Ex. 5.33. To change the UART mode of Bluetooth module:
➔ AT+UM=01,01<CR><LF> ← *change the UART mode to 2 stop bits and odd parity.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
Notes:
1. The response will be sent in current UART mode.

Ex. 5.34. To change the UART mode to throughput priority:
➔ AT+UM=00,00,00<CR><LF> ← *change the UART mode to 1 stop bits, no parity and throughput priority.*

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.16. UI—Query or configure the UART indicator output mode

5.1.16.1. Description:

This command can query or configure (disable or enable) the UART indicator output mode of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator UI.

5.1.16.2. Syntax:

Synopsis:

AT+UI[=State]<CR><LF>

5.1.16.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of UART indicator output mode Value: 00h or 01h 00: Disabled 01: Enabled	O	

Default: 01 (Enabled)

5.1.16.4. Examples:

Ex. 5.35. To query current UART indicator output mode of the Bluetooth module:
➔ AT+UI<CR><LF> ← *query current UART indicator output mode.*
← UI=01<CR><LF> ← *report the UART indicator output is enabled currently.*
Notes:
1. If the UART indicator output is disabled currently, the report will not be output.

Ex. 5.36. To disable the UART indicator output:
➔ AT+UI=00<CR><LF> ← *disable the UART indicator output*
← *no response output because the UART indicator output has been disabled*
Ex. 5.37. To enable the UART indicator output:
➔ AT+UI=01<CR><LF> ← *enable the UART indicator output*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.17. RC—Query or configure the remote control function

5.1.17.1. Description:

This command can query or configure (disable or enable) the remote control of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.

Once the remote control function is enabled, the Bluetooth module can be controlled by remote device via SPP or iAP. That means, you can send the EHCI commands from the remote SPP or iAP device to control the Bluetooth module. In this case, the Bluetooth module will try to recognize the data received on SPP or iAP as EHCI command first, if the data comply with the rule of EHCI command (i.e. start with AT+ and end by <CR><LF>), the data will be treated as a command, otherwise, the data will be treated as raw data.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator RC.

5.1.17.2. Syntax:

Synopsis:

AT+RC[=State]<CR><LF>

5.1.17.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of remote control function Value: 00h or 01h 00: Disabled 01: Enabled	O	

Default: 00 (Disabled)

5.1.17.4. Examples:

Ex. 5.38. To query current configuration of remote control function of the Bluetooth module:
➔ AT+RC<CR><LF> ← *query current configuration of remote control function.*
← RC=00<CR><LF> ← *report the remote control function is currently disabled.*

Ex. 5.39. To enable the remote control function:
➔ AT+RC=01<CR><LF> ← *enable the remote control function.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.18. PM—Query or configure PIO assignment

5.1.18.1. Description:

This command can query or configure the PIO assignment. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost. If the new configuration is adopted by the Bluetooth module, the module will perform a reboot, the non-memorable settings will return to their default value. Therefore, it is recommended to send this command first if necessary.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator PM.

5.1.18.2. Syntax:

Synopsis:

AT+PM[=DSR,DTR,RI,DCD]<CR><LF>

5.1.18.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

DSR	The PIO number assigned for DSR.	O
DTR	The PIO number assigned for DTR.	O
RI	The PIO number assigned for RI.	O
DCD	The PIO number assigned for DCD.	O

5.1.18.4. Examples:

Ex. 5.40. To query current configuration of PIO assignment:
➔ AT+PM<CR><LF> ← *query current configuration of PIO assignment.*
← PM=08,09,0A,0B<CR><LF> ← *report the current PIO assignment: DSR=PIO8, DTR=PIO9, RI=PIO10, DCD=PIO11.*

Ex. 5.41. To configure the PIO assignment:
➔ AT+PM=0B,0A,09,08<CR><LF> ← *configure the PIO assignment.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.19. DB—Query or configure the default Bypass mode

5.1.19.1. Description:

This command can query or configure the default bypass mode of Bluetooth module. Once configured, the configuration will take effect at the next time the module is power on. It means the Bluetooth module will remember the configuration and even if the Bluetooth module has been powered off, the configuration will not be lost, but the configuration will NOT take effect immediately. To change the runtime bypass mode, use the command **BP** instead.
For more information of bypass mode, refer to the section 5.1.29 .
If all of the parameters are not presented, the Bluetooth module will report current configuration by the Indicator DB.

5.1.19.2. Syntax:

Synopsis:

AT+DB[=ChannelMode][,Channel][,SpeedMode]<CR><LF>

5.1.19.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelMode	The new Bypass channel mode: Value: 00h—04h 00: Proxy mode 01: Bypass to First Connected Channel 02: Bypass to HID ASCII Channel 03: Bypass to All SPP 04: Bypass to Specified Channel Default: 01(Bypass to First Connected Channel)	O	

Channel	The new Bypass channel: Value: 00h—10h, 40h—46h 00—06: the SPP channel ID. Up to 7 SPP channels available. 07: The HID channel. 08—0E: The RFCOMM(for iAP) channel ID. Up to 7 RFCOMM channels available. 0F: The DUN channel 10: The BLE channel. Only available for and MB18. 40h—46h: the SPP Name ID.	O	
SpeedMode	The new Bypass speed mode: Value: 00h or 01h 00: Normal speed 01: High speed	O	For MA41 and MA46, the high speed mode is only available for SPP and DUN channel. For MB05, and MB18, the high speed mode is available for SPP and RFCOMM(for iAP) channel.

5.1.19.4. Examples:

Ex. 5.42. To query current configuration of default Bypass mode of the Bluetooth module:
➔ AT+DB<CR><LF> ← *query current configuration of default Bypass mode.*
← DB=01,00,00<CR><LF> ← *report the configuration of default Bypass mode.*

Ex. 5.43. To configure the default Bypass mode:
➔ AT+DB=01,00,01<CR><LF> ← *configure the default Bypass mode.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.20. IQ—Inquiry the Bluetooth device

5.1.20.1. Description:

This command makes the Bluetooth module to inquiry the Bluetooth device in its visible range.

5.1.20.2. Syntax:

Synopsis:

AT+IQ<CR><LF>

5.1.20.3. Parameter Description:

None.

5.1.20.4. Examples:

Ex. 5.44. To inquire the Bluetooth device:
➔ AT+IQ<CR><LF> ← *inquire the Bluetooth device.*
← AP=01<CR><LF> ← *indicate the Bluetooth module is now inquiring.*
← IR=03<CR><LF> ← *indicate there are 3 Bluetooth devices found.*
← AP=00<CR><LF> ← *indicate Bluetooth module is now in idle.*
← FD=02,00189600000C,FFC6,BT_DEV_3<CR><LF> ← *indicate the 3rd found device's address, the RSSI is -58dBm, the device name is BT_DEV_3.*
← FD=01,00189600000B,FFC7<CR><LF> ← *indicate the 2nd found device's address, the RSSI is -57dBm, the device name is not gotten.*
← FD=00,00189600000D,FFC8,BT_DEV_1<CR><LF> ← *indicate the 1st found device's address, the RSSI is -56dBm, the device name is BT_DEV_1.*

5.1.21. MD—Make the Bluetooth module discoverable

5.1.21.1. Description:

This command can query or change the Bluetooth module's discoverable status. Only when the Bluetooth module is discoverable, it can be found by other Bluetooth device.
If the parameter is not presented, the Bluetooth module will report current discover status by the Indicator MD.

5.1.21.2. Syntax:

Synopsis:

AT+MD[=Status]<CR><LF>

5.1.21.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Status	The new status of discoverable. Value: 00h or 01h 00: Not discoverable 01: Discoverable Default: 00 (Not discoverable)	O	

5.1.21.4. Examples:

Ex. 5.45. To query the current discoverable status of the Bluetooth module:
➔ AT+MD<CR><LF> ← *query the current discoverable status.*
← MD=00<CR><LF> ← *report the Bluetooth module is not discoverable currently.*

Ex. 5.46. To make Bluetooth module discoverable:
➔ AT+MD=01<CR><LF> ← *make Bluetooth module discoverable.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.22. PA—Query of change the status of pairing mode

5.1.22.1. Description:

This command can query or change the Bluetooth module's pairing mode status. Only when the pairing mode is enabled, it can be paired/bonded with other Bluetooth device.

If the parameter is not presented, the Bluetooth module will report current status of pairing mode by the Indicator PA.

5.1.22.2. Syntax:

Synopsis:

AT+PA[=Status]<CR><LF>

5.1.22.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Status	The new status of pairing mode. Value: 00h or 01h 00: Pairing/Bonding disabled 01: Pairing/Bonding enabled Default: 01 (Enabled)	O	

5.1.22.4. Examples:

Ex. 5.47. To query the current pairing mode status of the Bluetooth module:
➔ AT+PA<CR><LF> ← *query the current pairing mode status.*
← PA=01<CR><LF> ← *report the pairing is enabled currently.*

Ex. 5.48. To disable the pairing mode of the Bluetooth module:
➔ AT+PA=00<CR><LF> ← *disable the pairing mode.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.23. CA—Query of change the state of connectable mode

5.1.23.1. Description:

This command can query or change the Bluetooth module's connectable mode state. Only when the connectable mode is enabled, it can be connected with other Bluetooth device.
If the parameter is not presented, the Bluetooth module will report current state of connectable mode by the Indicator CA.

5.1.23.2. Syntax:

Synopsis:

AT+CA[=State]<CR><LF>

5.1.23.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of connectable mode. Value: 00h or 01h 00: connect disabled 01: connect enabled Default: 01 (Enabled)	O	

5.1.23.4. Examples:

Ex. 5.49. To query the current connectable mode state of the Bluetooth module:
➔ AT+CA<CR><LF> ← *query the current connectable mode state.*
← CA=01<CR><LF> ← *report the* connectable mode *is enabled currently.*

Ex. 5.50. To disable the connectable mode of the Bluetooth module:
➔ AT+CA=00<CR><LF> ← *disable the connectable mode.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.24. NC—Confirm or deny the numeric comparison

5.1.24.1. Description:

If the Man-In-The-Middle protection is activated and the IO capability is configured as *"**Display Yes/No**"*, the module may output the number for numeric comparison by the indicator NC. The command NC is used to confirm or deny the numeric comparison as a response of indicator NC when pairing.
About the numeric comparison, please refer to 5.1.6 . and 5.1.7 .

5.1.24.2. Syntax:

Synopsis:

AT+NC{=Confirmation}<CR><LF>

5.1.24.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Confirmation	The confirmation of numeric comparison. Value: 00h or 01h 00: deny the numeric comparison. 01: confirm the numeric comparison.	M	

5.1.24.4. Examples:

Ex. 5.51. To deny the numeric comparison:
← NC=012ABC<CR><LF> ← *indicate the number of numeric comparison with 012ABCh.*
➔ AT+NC=00<CR><LF> ← *deny the numeric comparison.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.52. To confirm the numeric comparison:
← NC=012ABC<CR><LF> ← *indicate the number of numeric comparison with 012ABCh.*
➔ AT+NC=01<CR><LF> ← *confirm the numeric comparison.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.25. PK—Input the passkey when pairing

5.1.25.1. Description:

This command is used to input the passkey being displayed on the remote device when pairing.

About the passkey entry, please refer to 5.1.6 . and 5.1.7 .

5.1.25.2. Syntax:

Synopsis:

AT+PK{=Number}<CR><LF>

5.1.25.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Number	The passkey number Value: 000000h – 0F423Fh	M	

5.1.25.4. Examples:

Ex. 5.53. To input the passkey number when pairing:
← PK<CR><LF> ← *indicates there is a Bluetooth device is passkey request*
→ AT+PK=012ABC<CR><LF> ← *input the passkey number: 012ABCh*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.26. CP—Clear the paired Bluetooth device list

5.1.26.1. Description:

This command can clear the paired device list stored in the Bluetooth module. If there is some Bluetooth device is connected with the Bluetooth module, it will perform a disconnection before clear the paired device list.

5.1.26.2. Syntax:

Synopsis:

AT+CP<CR><LF>

5.1.26.3. Parameter Description:

None.

5.1.26.4. Examples:

Ex. 5.54. To clear the paired device list:
→ AT+CP<CR><LF> ← *clear the paired device list.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.27. CT—Connect to remote Bluetooth device

5.1.27.1. Description:

This command will make Bluetooth module to connect to the remote Bluetooth device. If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected device. If the specified Bluetooth device has never connected with the Bluetooth module, it will attempt to connect to the specified device with all profiles supported by the Bluetooth module, otherwise, it will attempt to connect with the last connected profile.

5.1.27.2. Syntax:

Synopsis:

AT+CT[=BdAddr]<CR><LF>

5.1.27.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth device to connect.	O	

5.1.27.4. Examples:

Ex. 5.55. To connect to the last connected device with the last connected profile:
➔ AT+CT<CR><LF> ← *connect to the last connected device with the last connected profile.*
← SS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected device which address is 00:18:96:00:AB:CD, and the last connected profile is SPP.*
← CS=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← SS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected device.*

Ex. 5.56. To connect to the specified device with the last connected profile:
➔ AT+CT=00189600000A<CR><LF> ← *connect to the specified device 00:18:96:00:00:0A with the last connected profile.*
← IS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified device which address is 00:18:96:00:00:0A, and the last connected profile is HID.*
← CI=00,00189600000A<CR><LF> ← *connecting result: success.*
← IS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified device.*

5.1.28. DC—Disconnect with remote Bluetooth device

5.1.28.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected devices.

5.1.28.2. Syntax:

Synopsis:

AT+DC[=BdAddr]<CR><LF>

5.1.28.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth device to disconnect.	O	

5.1.28.4. Examples:

Ex. 5.57. To disconnect with all of the connected devices:
➔ AT+DC<CR><LF> ← *disconnect with all of the connected devices. Assume it's a SPP device.*
← SS=00<CR><LF> ← *the SPP channel 0 of Bluetooth module is now disconnected and is connectable.*

Ex. 5.58. To disconnect to the specified device:

➔ AT+DC=00189600000A<CR><LF> ← *disconnect with the specified device 00:18:96:00:00:0A. Assume it's a HID device.*
← IS=00<CR><LF> ← *the HID profile of Bluetooth module is now disconnected and is connectable.*

5.1.29. BP—Configure the runtime Bypass mode

5.1.29.1. Description:

This command can configure the runtime Bypass mode of the Bluetooth module. Once configured, the new configuration will take effect immediately until the Bluetooth module reboot. It means this command is a non-memorable command, the configuration will be lost (i.e. return to its default value) after reboot.
There are 5 different channel mode explained below:

0. Proxy mode

When working in this mode, the content sent to the Bluetooth module via UART port will be treated as ASCII command. And the content sent from the Bluetooth module should be treated ASCII indicator.
When there is not any connection has been established, the Bluetooth module is working in this mode.

1. Bypass to First Connected Channel

When working in this mode, the content sent to the Bluetooth module via UART port will be treated as raw data, and will be transparent transfer to Bypass channel. At the same time, if the Bypass channel is connected with a SPP device, the content received from the remote SPP device will be output by the Bluetooth module via UART port. So, when working in this mode the local host should treat the content output from the Bluetooth module as raw data. But, if the Bypass channel is connected with a HID device, the content output from the module should be treated as ASCII Indicators.
In this mode, if the bypass channel is connected with a SPP device, the content received from a non-Bypass channel will be thrown away.
In this mode, the Bypass channel will be automatically selected to the first connected channel.
If there are more than one connections have been established, the host cannot configure the Bluetooth module to this Bypass mode since it's hard to determine which is the first connected channel.

2. Bypass to HID ASCII Channel

When working in this mode, the content sent to the Bluetooth module via UART port can only be ASCII characters 20h—7Eh and 0Dh, the Bluetooth module will send these ASCII characters to remote Bluetooth host directly.
In this mode, the content sent from the Bluetooth module should be treated as ASCII indicator.
In this mode, the Bypass channel will be automatically selected to HID channel.

3. Bypass to All SPP Channel(Mixture)

When working in this mode, the content sent to the Bluetooth module via UART port will be treated as raw data, and will be transparent transfer to all of the connected SPP devices. At the same time, the content received from each of the remote SPP device will be transparent output by the Bluetooth module via UART port without a channel identifier to indicate the data source.

4. Bypass to Specified Channel

Like the **Bypass to First Connected Channel** mode, but the Bypass channel will be specified by the parameter [BypassChannel]

Table 5.1 Bypass Channel Mode

Mode	Channel	Content Sent to UART	Content Output from UART	Comments
Proxy mode	N/A	ASCII commands	ASCII indicators	
Bypass to First Connected Channel	SPP channel	Raw data to be sent to remote SPP device	Raw data received from remote SPP device.	The Bypass channel will be selected to the first connected channel.
	HID channel	Raw data(HID reports) to be sent the HID host. About the HID report, please refer to section 5.3.4.4.	ASCII indicators	
Bypass to HID ASCII Channel	HID channel	ASCII characters(20h—7Eh) to be sent to HID host	ASCII indicators	
Bypass to All SPP Channel	Connected SPP channel	Raw data to be sent to all connected SPP device	Raw data received from all connected SPP device	

Bypass to Specified Channel	SPP channel	Raw data to be sent to remote SPP device	Raw data received from remote SPP device.	the Bypass channel will be specified by the parameter [Channel]
	HID channel	Raw data(HID reports) to be sent the HID host. About the HID report, please refer to section 5.3.4.4.	ASCII indicators	

There are 2 different speed mode explained below:

0. Normal speed mode

When working in this mode, the Bluetooth module will try to parse the content received from UART to find if there is a **BP** command, so the speed is affected accordingly.

1. High speed mode

When working in this mode, the Bluetooth module will transfer the content received from UART to bypass channel directly. In this case the Bluetooth module will not parse the content, so the host cannot change the bypass mode by **BP** command.

Notes:

1. When the Bluetooth module is configured to one of the Bypass modes, it does NOT mean the Bluetooth module will work in the specified Bypass mode immediately. Only when the configured Bypass channel is connected with a remote Bluetooth device, the Bluetooth module will work in Bypass mode automatically. But, there is an exception, for iAP application, only when the data session has been opened by the application on iOS device(see 6.4.3 for data session open state), the Bluetooth module will work in Bypass mode.

2. When the Bluetooth module is working in Bypass mode, the Bluetooth module will quit Bypass mode automatically once the Bypass channel is disconnected with the remote Bluetooth device.

*3. When the Bluetooth module is working in Bypass mode, only this command is available to send to the Bluetooth module via UART port. If the user/host wants to send any other command to the module, the user/host has to change the working mode to **Proxy Mode** by this command first.*

*4. To send this command to the Bluetooth module when working in Bypass mode, a **Change Bypass Sequence** condition must be matched, the **Change Bypass Sequence** is shown as below:*

<1 second idle on UART> AT+BP=00,00<CR><LF> <1 second idle on UART>

5.1.29.2. Syntax:

Synopsis:

AT+BP{=ChannelMode,Channel}[,SpeedMode]<CR><LF>

5.1.29.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

ChannelMode	The new Bypass channel mode: Value: 00h—04h 00: Proxy mode 01: Bypass to First Connected Channel 02: Bypass to HID ASCII Channel 03: Bypass to All SPP 04: Bypass to Specified Channel Default: 01(Bypass to First Connected Channel)	M	
Channel	The new Bypass channel: Value: 00h—10h, 40h—46h 00—06: the SPP channel ID. Up to 7 SPP channels available. 07: The HID channel. 08—0E: The RFCOMM(for iAP) channel ID. Up to 7 RFCOMM channels available. 0F: The DUN channel 10: The BLE channel. Only available for and MB18. 40h—46h: the SPP Name ID.	M	The parameter will only take effect when the Bypass channel mode is **Bypass to Specified Channel**
SpeedMode	The new Bypass speed mode: Value: 00h or 01h 00: Normal speed 01: High speed	O	For MA41 and MA46, the high speed mode is only available for SPP channel. For MB05, and MB18, the high speed mode is available for SPP and RFCOMM channel.

5.1.29.4. Examples:

Ex. 5.59. To change the bypass mode to **Proxy Mode** when working in one of the Bypass mode:
← *Keep the UART port idle for 1 second.*
➔ AT+BP=00,00,00<CR><LF> ← *change the bypass mode to **Proxy Mode**.*
← *Keep the UART port idle for 1 second.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.60. To change the bypass mode to **Bypass to HID ASCII Channel**:
➔ AT+BP=02,00,00<CR><LF> ← *change the bypass mode to **Bypass to HID ASCII Channel**.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.61. To change the bypass mode to **Bypass to Specified Channel**:
➔ AT+BP=04,03,00<CR><LF> ← *change the bypass mode to **Bypass to Specified Channel**, the channel is specified to SPP channel 03, **Normal Speed**.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.62. To change the bypass mode to **Bypass to Specified Channel, High Speed**:
➔ AT+BP=04,03,01<CR><LF> ← *change the bypass mode to **Bypass to Specified Channel**, the channel is specified to SPP channel 03, **High Speed**.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.30. PC—Read and write the PIO value

5.1.30.1. Description:

This command can read and write the value of a specified PIO.
If the parameter [value] is not presented, the Bluetooth module will read and report the value of specified PIO by the Indicator PC.

5.1.30.2. Syntax:

Synopsis:

AT+PC{=Pio}[,Val]<CR><LF>

5.1.30.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Pio	The PIO number need to read or write: Value: 00h-0Fh or 00h-1Fh per module type	M	
Val	The value to write to the PIO Value: 00h – 01h 00: Logic Low 01: Logic High	O	

Notes:

1. Different module has different number of PIOs available, and parts of the PIOs are assigned for internal function usage by the Bluetooth module. See the datasheet of the Bluetooth module for more information.

5.1.30.4. Examples:

Ex. 5.63. To read the status of PIO 2 of Bluetooth module:
➔ AT+PC=02<CR><LF> ← *read the status of PIO 2.*
← PC=02,01<CR><LF> ← *report the PIO 2 is currently logic high.*

Ex. 5.64. To write the PIO 3 of Bluetooth module to logic high:
➔ AT+PC=03,01<CR><LF> ← *write the PIO 3 of Bluetooth module to logic high.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.31. AC—Read voltage of AIO

5.1.31.1. Description:

This command can read the voltage of a specified AIO pin.
The Bluetooth module will read and report the voltage of specified AIO by the Indicator AC.

5.1.31.2. Syntax:

Synopsis:

AT+AC{=Aio}<CR><LF>

5.1.31.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Aio	The AIO number need to read: Value: 00h or 01h	M	

5.1.31.4. Examples:

Ex. 5.65. To read the voltage on AIO 0 of Bluetooth module:
➔ AT+AC=00<CR><LF> ← *read the voltage on AIO0.*
← AC=00,0022<CR><LF> ← *report the voltage on AIO0 is 34mV.*

Ex. 5.66. To read the voltage on AIO 1 of Bluetooth module:
➔ AT+AC=01<CR><LF> ← *read the voltage on AIO0.*
← AC=01,05F6<CR><LF> ← *report the voltage on AIO0 is 1526mV.*

5.1.32. VU—Increase the Volume

5.1.32.1. Description:

This command is used to increase the volume of HFP voice or A2DP music by 1 step.
Only the volume of currently output will be increased, that means if it is in a phone call, the volume of HFP voice will be increased while the volume of A2DP music will not be affected. Otherwise, if it is playing A2DP music, the volume of A2DP music will be increased while the volume of HFP voice will not be affected.

5.1.32.2. Syntax:

Synopsis:

AT+VU<CR><LF>

5.1.32.3. Parameter Description:

None.

5.1.32.4. Examples:

Ex. 5.67. To increase the volume:
➔ AT+VU<CR><LF> ← *increase the volume.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.33. VD—Decrease the Volume

5.1.33.1. Description:

This command is used to decrease the volume of HFP voice or A2DP music by 1 step.
Only the volume of currently output will be decreased, that means if it is in a phone call, the volume of HFP voice will be decreased while the volume of A2DP music will not be affected. Otherwise, if it is playing A2DP music, the volume of A2DP music will be decreased while the volume of HFP voice will not be affected.

5.1.33.2. Syntax:

Synopsis:

AT+VD<CR><LF>

5.1.33.3. Parameter Description:

None.

5.1.33.4. Examples:

Ex. 5.68. To increase the volume:
➔ AT+VD<CR><LF> ← *decrease the volume.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.1.34. LC—List connected devices

5.1.34.1. Description:

This command is used to list the connected devices.

5.1.34.2. Syntax:

Synopsis:

AT+LC<CR><LF>

5.1.34.3. Parameter Description:

None.

5.1.34.4. Examples:

Ex. 5.69. To list the connected devices:
➔ AT+LC<CR><LF> ← *list the connected devices.*
← LC=03,00123456ABCD <CR><LF> ← *report the device which name is unknown is connected with SPP profile, the device address is 00:12:34:56:AB:CD.*
← LC=04,00123456CDEF, iPhone<CR><LF> ← *report the device "iPhone" is connected with RFCOMM profile(for data transmission), the device address is 00:12:34:56:CD:EF.*
← LC=04,00123456CDEF, iPhone<CR><LF> ← *report the device "iPhone" is connected with HFP profile, the device address is 00:12:34:56:CD:EF.*
← LC=05,00123456CDEF, iPhone<CR><LF> ← *report the device "iPhone" is connected with A2DP profile, the device address is 00:12:34:56:CD:EF.*
← LC=06,00123456CDEF, iPhone<CR><LF> ← *report the device "iPhone" is connected with AVRCP profile, the device address is 00:12:34:56:CD:EF.*

5.1.35. FU—Make the Bluetooth module enter into DFU mode

5.1.35.1. Description:

This command is used to make the module enter into DFU(Device Firmware Upgrade) mode. In DFU mode, the user can upgrade the firmware via USB port.

5.1.35.2. Syntax:

Synopsis:

AT+FU<CR><LF>

5.1.35.3. Parameter Description:

None.

5.1.35.4. Examples:

Ex. 5.70. To make the module enter into DFU mode:
➔ AT+FU<CR><LF> ← *make the module enter into DFU mode.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
Notes:
1. Once enter into DFU mode, the module will not response any command.
2. If the module has been rebooted before the DFU process start, it will quit DFU mode. This is the only way to quit DFU mode. The user can resent the command AT+FU to make the module enter into DFU mode.
3. In case of a failure of DFU, the module will stay in DFU mode, the user can perform a DFU process again.
4. Please refer to the DFU guide document for more detailed description of DFU operation.

5.1.36. TS—Make the Bluetooth module enter into Test mode

5.1.36.1. Description:

This command is used to make the module enter into Test mode. In Test mode, the user can do the radio test of BQB.

5.1.36.2. Syntax:

Synopsis:

AT+TS{=CmdId}[,LocalFreq][,Gain][,ModulateFreq]<CR><LF>

5.1.36.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
CmdId	The command ID of Test mode: Value: 01h-04h 01: Radio Test TxStart 02: Radio Test Tx Data1 03: Radio Test RxStart 04: DUT mode	M	
LocalFreq	The local frequency in MHz Value: 0962h – 09B0h	O	
Gain	The gain to use. The lower 8 bits are the internal gain, the upper 8 bits are the external gain. For Class1 module, the external gain shall be 1, while for Class2 module, the external gain shall be 0. This parameter is only available for command ID 01 and command ID 02.	O	

ModulateFreq	the modulate frequency. This parameter is only available for command ID 01.	O

5.1.36.4. Examples:

Ex. 5.71. To make the module enter into Radio Test mode:TxStart:
➔ AT+TS=01,0989,013F,0000<CR><LF> ← *Radio Test TxStart, Local Frequency: 2441MHz, internal gain: 63, external gain: 1, modulate frequency 0Hz.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.72. To make the module enter into Radio Test mode: TxData1:
➔ AT+TS=02,0989,013F<CR><LF> ← *Radio Test TxData1, Local Frequency: 2441MHz, internal gain: 63, external gain: 1.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.73. To make the module enter into Radio Test mode:RxStart:
➔ AT+TS=03,0989<CR><LF> ← *Radio Test RxStart, Local Frequency: 2441MHz.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.74. To make the module enter into DUT mode:
➔ AT+TS=04<CR><LF> ← *Enter into DUT mode*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Notes:

1. *Once enter into Test mode, the module will not response any command.*
2. *If the module has been rebooted, it will quit Test mode. This is the only way to quit Test mode.*

5.2. SPP Commands

5.2.1. SM—Query or change the service name of SPP

5.2.1.1. Description:

This command can query or change the service name of SPP profile. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost. If the new configuration is adopted by the Bluetooth module, the module will perform a reboot, the non-memorable settings will return to their default value. Therefore, it is recommended to send this command first if necessary.
If the parameter is not presented, the Bluetooth module will report current profile configuration by the Indicator SM.

5.2.1.2. Syntax:

Synopsis:

AT+SM[=Name]<CR><LF>

5.2.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The new service name of SPP profile. Length: 1—16 characters Default: SPP Dev.	O	

5.2.1.4. Examples:

Ex. 5.75. To query current service name of SPP profile:
➔ AT+SM<CR><LF> ← *query current service name.*
← SM=SPP Dev<CR><LF> ← *report current service name, it's "SPP Dev".*

Ex. 5.76. To change the service name of SPP profile:
➔ AT+SM=GPS Dev<CR><LF> ← *change the service name to "GPS Dev"*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.2.2. CS—Connect to the remote SPP device

5.2.2.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth SPP device. If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected SPP device.

5.2.2.2. Syntax:

Synopsis:

AT+CS[=BdAddr][,NameId]<CR><LF>

5.2.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth SPP device to connect.	O	
NameId	The name ID of this connection. Once connected, the host can use the name ID to identify the source or destination. Value: 40h—4xh (x is the maximum SPP instance count, refer to 5.1.1)	O	This is only available when both of the two sides are NVC's software.

Notes:

1. If either local or remote device has already established a SPP connection with some other device use the same Name ID, the remote device will disconnect with local device immediately.

2. Once the connection with a specified Name ID has been successfully established, the Bluetooth module will remember the Name ID and use this Name ID to auto connect after power on and auto reconnect after link lost(if these features are enabled).

5.2.2.4. Examples:

Ex. 5.77. To connect to the last connected SPP device:
➔ AT+CS<CR><LF> ← *connect to the last connected device with the SPP profile.*
← SS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected device which address is 00:18:96:00:AB:CD.*
← CS=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← SS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected device.*

Ex. 5.78. To connect to the specified device with the SPP profile:
➔ AT+CS=00189600000A<CR><LF> ← *connect to the specified device 00:18:96:00:00:0A with the SPP profile.*
← SS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified SPP device which address is*

00:18:96:00:00:0A.
← CS=00,00189600000A<CR><LF> ← *connecting result: success.*
← SS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified SPP device.*

Ex. 5.79. To connect to the specified SPP device with the Name ID 13:
→ AT+CS=00189600000A,43<CR><LF> ← *connect to the specified device 00:18:96:00:00:0A with the Name ID 43.*
← SS=01,00189600000A,43<CR><LF> ← *the Bluetooth module is now connecting to the specified SPP device which address is 00:18:96:00:00:0A,use Name ID 43.*
← CS=00,00189600000A<CR><LF> ← *connecting result: success.*
← SS=02,00189600000A,43<CR><LF> ← *the Bluetooth module is now connected to the specified SPP device, the Name ID is 43.*

5.2.3. DS—Disconnect with the remote SPP device

5.2.3.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth SPP device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected SPP devices.

5.2.3.2. Syntax:

Synopsis:

AT+DS[=BdAddr]<CR><LF>

5.2.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth SPP device to disconnect.	O	

5.2.3.4. Examples:

Ex. 5.80. To disconnect with all of the connected SPP devices:
→ AT+DS<CR><LF> ← *disconnect with all of the connected SPP devices.*
← SS=00<CR><LF> ← *the SPP channel 0 of Bluetooth module is now disconnected and is connectable.*
← SS=10<CR><LF> ← *the SPP channel 1 of Bluetooth module is now disconnected and is connectable.*
← SS=30<CR><LF> ← *the SPP channel 3 of Bluetooth module is now disconnected and is connectable.*

Ex. 5.81. To disconnect to the specified device:
→ AT+DS=00189600000A<CR><LF> ← *disconnect with the specified device 00:18:96:00:00:0A.*
← SS=00<CR><LF> ← *the SPP channel 0 of Bluetooth module is now disconnected and is connectable.*

5.2.4. SS—Query the state of each SPP channel

5.2.4.1. Description:

This command is used to query the state of each SPP channel.

5.2.4.2. Syntax:

Synopsis:

AT+SS<CR><LF>

5.2.4.3. Parameter Description:

None.

5.2.4.4. Examples:

Ex. 5.82. To query the state of each SPP channel:
➔ AT+SS<CR><LF> ← *query the state of each SPP channel.*
← SS=01,00189600ABCD<CR><LF> ← *the SPP channel 0 of Bluetooth module is now connecting to the remote device which address is 00:18:96:00:AB:CD.*
← SS=12,00189601ABCD<CR><LF> ← *the SPP channel 1 of Bluetooth module is now connected with the remote device which address is 00:18:96:01:AB:CD.*
← SS=22,00189603ABCD,43<CR><LF> ← *the SPP channel 2 of Bluetooth module is now connected with the remote device which address is 00:18:96:02:AB:CD, the name ID is 43.*
← SS=30<CR><LF> ← *the SPP channel 3 of Bluetooth module is now connectable*
← SS=40<CR><LF> ← *the SPP channel 4 of Bluetooth module is now connectable*
← SS=50<CR><LF> ← *the SPP channel 5 of Bluetooth module is now connectable*

5.2.5. DT—Send data packet to remote SPP device

5.2.5.1. Description:

This command is used to send a data packet to the remote SPP device.

5.2.5.2. Syntax:

Synopsis:

AT+DT{=ChannelOrNameId,DataLen,Data}<CR><LF>

5.2.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelOrNameId	The SPP channel ID or Name ID which will be used to send the data packet. Value: 00h—0xh or 40h—4xh (x is the maximum SPP instance count, refer to 5.1.1) 00—0x: the channel ID of SPP 40—4x: the name ID of SPP	M	
DataLen	The length in bytes of the data to be sent. Value: 00h-FFh	M	
Data	The raw data.	M	

5.2.5.4. Examples:

Ex. 5.83. To send data use SPP channel 0:
➔ AT+DT=00,0A,1234567890<CR><LF> ← *send a data packet use SPP channel 0, the data length is 10(Dec).*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.84. To send data use SPP Name ID 13h:
➔ AT+DT=43,0A,1234567890<CR><LF> ← *send a data packet use SPP Name ID 43h, the data length is 10*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.3. HID Commands

5.3.1. CI—Connect to the remote HID host

5.3.1.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth HID host. If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected HID host.

5.3.1.2. Syntax:

Synopsis:

AT+CI[=BdAddr]<CR><LF>

5.3.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth HID host to connect.	O	

5.3.1.4. Examples:

Ex. 5.85. To connect to the last connected HID host:
➔ AT+CI<CR><LF> ← *connect to the last connected HID host.*
← IS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected HID host which address is 00:18:96:00:AB:CD.*
← CI=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← IS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected HID host.*

Ex. 5.86. To connect to the specified HID host:
➔ AT+CI=00189600000A<CR><LF> ← *connect to the specified HID host: 00:18:96:00:00:0A.*
← IS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified HID host which address is 00:18:96:00:00:0A.*
← CI=00,00189600000A<CR><LF> ← *connecting result: success.*
← IS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified HID host.*

5.3.2. DI—Disconnect with the remote HID host

5.3.2.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth HID host. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected HID hosts.

5.3.2.2. Syntax:

Synopsis:

AT+DI[=BdAddr]<CR><LF>

5.3.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth HID host to disconnect.	O	

5.3.2.4. Examples:

Ex. 5.87. To disconnect with all of the connected HID host:
➔ AT+DI<CR><LF> ← *disconnect with all of the connected HID host.*
← IS=00<CR><LF> ← *the HID channel of Bluetooth module is now disconnected and is connectable.*

Ex. 5.88. To disconnect to the specified device:
➔ AT+DI=00189600000A<CR><LF> ← *disconnect with the specified HID host: 00:18:96:00:00:0A.*
← IS=00<CR><LF> ← *the HID channel of Bluetooth module is now disconnected and is connectable.*

5.3.3. IS—Query the state of HID channel

5.3.3.1. Description:

This command is used to query the state of HID channel.

5.3.3.2. Syntax:

Synopsis:

AT+IS<CR><LF>

5.3.3.3. Parameter Description:

None.

5.3.3.4. Examples:

Ex. 5.89. To query the state of HID channel:
➔ AT+IS<CR><LF> ← query the state of HID channel.
← IS=01,00189600ABCD<CR><LF> ← *the HID channel of Bluetooth module is now connecting to the remote HID host which address is 00:18:96:00:AB:CD.*

5.3.4. KR—Send HID report to remote HID host

5.3.4.1. Description:

This command is used to send a HID report to the remote HID host.

5.3.4.2. Syntax:

Synopsis:

AT+KR{=HidReport}<CR><LF>

5.3.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
HidReport	The HID report needs to be sent to HID host.	M	

5.3.4.4. Report Format and Examples:

Start (1Byte)	Report Id (1Byte)	Data (8 Bytes for Keyboard Report and Joystick/Gamepad, 2 Bytes for Consumer Report, 5 Bytes for Mouse)

Notes:

*1. In **Proxy mode**, all of the data field in HID report should be given in ASCII characters and separated by comma, while in **Bypass mode**, all of the data field should be given in raw data (binary) and no separator is needed. About the Proxy mode and Bypass mode, please refer to section 5.1.19 and 5.1.29.*

1. Keyboard Report:

A1 01 Modifier 00 ScanCode1 ScanCode2 ScanCode3 ScanCode4 ScanCode5 ScanCode6

The **Modifier** byte is a bit mask interpreted as shown in Table 5.2 . For example, you can use 02h or 20h to turn a lower case 'a' into an upper case 'A'.

Table 5.2 Bit Mask of Modifier Byte in Keyboard Report

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Right GUI	Right Alt	Right Shift	Right Ctrl	Left GUI	Left Alt	Left Shift	Left Ctrl

The **ScanCode** is defined by the USB HID Spec.

Ex. 5.90. If the key A and the Right Shift are pressed, the keyboard report should be:

A1 01 20 00 04 00 00 00 00 00

Ex. 5.91. If all of the pressed keys have been released, the keyboard report should be:

A1 01 00 00 00 00 00 00 00 00

Ex. 5.92. To send a keyboard report to HID host:

➔ AT+KR=A1,01,00,00,04,00,00,00,00,00<CR><LF> ← *send a keyboard report to the HID host. The key A is pressed.*

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

➔ AT+KR=A1,01,00,00,00,00,00,00,00,00<CR><LF> ← *send a keyboard report to the HID host. The pressed key is released.*

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

2. Consumer Key Report:

A1 02 LowByte HighByte

The **LowByte** and **HighByte** are bit mask interpreted as shown in Table 5.3 :

Table 5.3 Consumer Key Function

Consumer Key Function	LowByte	HighByte
AC Home	01	00
AL Email Reader	02	00
AC Search	04	00
AL Keyboard Layout (Virtual Apple Keyboard Toggle)	08	00
Volume Up	10	00
Volume Down	20	00
Mute	40	00
Play/Pause	80	00
Scan Next Track	00	01
Scan Previous Track	00	02
Stop	00	04
Eject	00	08
Fast Forward	00	10
Rewind	00	20
Stop/Eject	00	40
AL Internet Browser	00	80

Ex. 5.93. To increase the volume, the consumer key report should be:
A1 02 10 00

Ex. 5.94. To release the consumer key, the consumer key report should be:
A1 02 00 00

Ex. 5.95. To send a consumer key report to HID host:
➔ AT+KR=A1,02,10,00<CR><LF> ← *send a consumer key report to the HID host. The Volume Up key is pressed.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,02,00,00<CR><LF> ← *send a consumer key report to the HID host. The pressed key is released.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,02,08,00<CR><LF> ← *send a consumer key report to the HID host to popup the Virtual Apple Keyboard.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,02,00,00<CR><LF> ← *send a consumer key report to the HID host. The pressed key is released.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

3. Mouse Report:

A1 03 Buttons XmXl YlXh YhYm Wheel

The **Buttons** is a bit mask interpreted as shown in Table 5.4 :

Table 5.4 Bit Mask of Buttons Byte in Mouse Report

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Button 8	Button 7	Button 6	Button 5	Button 4	Button 3	Button 2	Button 1

The **XhXmXl** compose the movement on X axis. The range is from -2048(800h) to +2047(7FFh).
The **YhYmYl** compose the movement on Y axis. The range is from -2048(800h) to +2047(7FFh).
The **Wheel** is the movement of wheel. The range is from -127(81h) to +127(7Fh).
Ex. 5.96. To press the left button of the mouse, the mouse report should be:
A1 03 01 00 00 00 00

Ex. 5.97. To move the mouse towards top-right(X:3 pixel, Y:-4 pixel), the mouse report should be:
A1 03 00 03 C0 FF 00

XhXmXl = 003h = 3 (Decimal)
YhYmYl = FFCh = -4 (Decimal)

Ex. 5.98. To scroll up for 1 line, the mouse report should be:
A1 03 00 00 00 00 FF

Wheel = FFh = -1 (Decimal)

Ex. 5.99. To send a mouse report to HID host:
➔ AT+KR=A1,03,01,00,00,00,00<CR><LF> ← *send a mouse report to the HID host. The Button1(Left Button) is pressed.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,03,00,00,00,00,00<CR><LF> ← *send a mouse report to the HID host. The pressed key is released.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,03,00,03,C0,FF,00<CR><LF> ← *send a mouse report to the HID host. The mouse is move to upper-right.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,03,00,00,00,00,00<CR><LF> ← *send a mouse report to the HID host. The mouse is stopped.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,03,00,00,00,00,FF<CR><LF> ← *send a mouse report to the HID host. The mouse wheel scroll up for one line.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,03,00,00,00,00,00<CR><LF> ← *send a mouse report to the HID host. The mouse is stopped.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

4. Joystick/Gamepad:

A1 04 Throttle X Y Z Rz HatSwitch Buttons1 Buttons2

The **Throttle** is the throttle value. The range is from -127(81h) to +127(7Fh).
The **X** is the position of X axis of left stick. The range is from -127(81h) to +127(7Fh).
The **Y** is the position of Y axis of left stick. The range is from -127(81h) to +127(7Fh).
The **Z** is the position of Z axis (generally, it is used as X axis of right stick). The range is from -127(81h) to +127(7Fh).
The **Rz** is the rotation of Z axis (generally, it is used as Y axis of right stick). The range is from -127(81h) to +127(7Fh).
The **HatSwitch** is the direction of hat switch. The range is from 00h to 07h, represents Top(00h), Top-right(01h), Right(02h), Bottom-right(03h), Bottom(04h), Bottom-left(05h), Left(06h), Top-Left(07h). The value out of range is invalid, and the hat switch will not move.
The **Buttons1** is a bit mask of first 8 buttons(Button1—Button8). Each bit represents one button.
The **Buttons2** is a bit mask of second 8 buttons(Button9—Button16). Each bit represents one button.
Ex. 5.100. To set the Throttle to 5, Left X to -2, Left Y to 3, Right X to 2, Right Y to -5, the joystick report should be:
A1 04 05 FE 03 02 FB 08 00 00

Throttle = 05h = 5 *(Decimal)*
X = Left X = FEh = -2 *(Decimal)*
Y = Left Y = 03h = 3 *(Decimal)*
Z = Right X = 02h = 2 *(Decimal)*
Rz = Right Y = FBh = -5 *(Decimal)*
HatSwitch = 08 = No movement
Buttons1 = 00h = No button be pressed.
Buttons2 = 00h = No button be pressed

Ex. 5.101. To set the Throttle to -10, Button2, Button3 and Button 15 pressed, Hat Switch to Bottom-left, the joystick report should be:
A1 04 F6 00 00 00 00 05 06 40

Throttle = F6h = -10 *(Decimal)*
X = Left X = 00h = 00 *(Decimal)*
Y = Left Y = 00h = 00 *(Decimal)*
Z = Right X = 00h = 00 *(Decimal)*
Rz = Right Y = 00h = 00 *(Decimal)*
HatSwitch = 05 = Bottom-left
Buttons1 = 06h = Button2 and Button3 be pressed.
Buttons2 = 40h = Button15 be pressed

Ex. 5.102. To send a joystick/gamepad report to HID host:
➔ AT+KR=A1,04,05,FE,03,02,FB,08,00,00<CR><LF> ← *send a* joystick/gamepad *report to the HID host.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+KR=A1,04,F6,00,00,00,00,05,06,40<CR><LF> ← send a joystick/gamepad report to the HID host.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

➔ AT+KR=A1,04,00,00,00,00,00,08,00,00<CR><LF> ← *send a* joystick/gamepad *report to the HID host.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5. User Defined Report:

A1 FF Length D1 D2 D3 D4 …… …… Dn

The **Length** is the length of report data in byte.
The **D1, D2, D3, D4,……,Dn** are the report data, here **n** equal to the **Length**.
Ex. 5.103. If a user defined report has 10 report data, the report should be:
A1 FF 0A 01 02 03 04 05 06 07 08 09 0A

Length = 0Ah = 10 *(Decimal)*
Report data = 01 02 03 04 05 06 07 08 09 0A
Note:
*1. For Bypass mode, the **Length** can up to FFh(255 in decimal), while for Proxy mode, the maximum **Length** is 55h(85 in decimal).*

5.3.5. AS—Send ASCII string to remote HID host

5.3.5.1. Description:

This command is used to send an ASCII string to the remote HID host.

5.3.5.2. Syntax:

Synopsis:

AT+AS{=AsciiStr}<CR><LF>

5.3.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
AsciiStr	The ASCII string needs to be sent to HID host. Only the ASCII character in the range of 20h—7Eh and 08h, 09h, 0Dh can be included in this parameter. The data out of range will be thrown away.	M	

Notes:
1. An escape character ('\') is available like which has been widely used in C/C++ language. In this case, "\r" or "\R" represents Enter(0Dh), "\b" or "\B" represents Backspace(08h), "\t" or "\T" represents Tab(09h), and "\\" represents '\' character.

5.3.5.4. Examples:

Ex. 5.104. To send an ASCII string to HID host:
➔ AT+AS=Hello Bluetooth HID<CR><LF> ← *send an ASCII string to the HID host.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.105. To send a ASCII string to HID host:

➔ AT+AS=!@#$%^abcdef9876<CR><LF> ← *send an ASCII string to the HID host.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

Ex. 5.106. To send a ASCII string to HID host:
➔ AT+AS=Hello\tWorld!\b\rThis is \\Ehong\\<CR><LF> ← *send an ASCII string to the HID host.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
The HID host will receive and display as below:
Hello World
This is \Ehong\

5.4. RFCOMM Commands (for Apple iOS devices)

The RFCOMM commands are mainly used to connect and communicate with an iOS device, such as iPod, iPhone and iPad.

5.4.1. RM—Query or change the service name of RFCOMM

5.4.1.1. Description:

This command can query or change the service name of RFCOMM profile. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost. If the new configuration is adopted by the Bluetooth module, the module will perform a reboot, the non-memorable settings will return to their default value. Therefore, it is recommended to send this command first if necessary.
If the parameter is not presented, the Bluetooth module will report current profile configuration by the Indicator RM.

5.4.1.2. Syntax:

Synopsis:

AT+RM[=Name]<CR><LF>

5.4.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The new service name of RFCOMM profile. Length: 1—16 characters Default: iAP Dev.	O	

5.4.1.4. Examples:

Ex. 5.107. To query current service name of RFCOMM profile:
➔ AT+RM<CR><LF> ← *query current service name.*
← RM=iAP Dev<CR><LF> ← *report current service name, it's "iAP Dev".*

Ex. 5.108. To change the service name of RFCOMM profile:
➔ AT+RM=GPS Dev<CR><LF> ← *change the service name to "GPS Dev"*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.4.2. PT—Query or change the protocol name of MFi application

5.4.2.1. Description:

This command can query or change the protocol name of MFi application. Once changed, the new protocol name will take effect at next time the Bluetooth module connect with an iOS device and until the next time the protocol name is changed by this command. It means the Bluetooth module will remember the protocol name, and even if the Bluetooth module has been powered off, the friendly name will not be lost.
If the parameter is not presented, the Bluetooth module will report current protocol name by the Indicator PT.

5.4.2.2. Syntax:

Synopsis:

AT+PT [=ProtocolName]<CR><LF>

5.4.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ProtocolName	The new protocol name of MFi application. Length: 1—30 characters Default: Per software version.	O	

Notes:

1. The protocol name string will be compared (without considering case) to strings presented by applications on the iOS device. The recommended format of protocol name is revise-DNS string (e.g. "com.Ehong.protocol").

5.4.2.4. Examples:

Ex. 5.109. To query current protocol name of iAP application:
➔ AT+PT<CR><LF> ← *query current protocol name.*
← PT=com.nvc.bt.iap <CR><LF> ← *report current protocol name, it's "com.nvc.bt.iap".*

Ex. 5.110. To change the protocol name of iAP application:
➔ AT+PT=com.Ehong.protocol <CR><LF> ← *change the local friendly name to "com.Ehong.protocol".*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.4.3. AH—Query the status of Apple authentication processor

5.4.3.1. Description:

This command is used to query the status of Apple authentication processor.

5.4.3.2. Syntax:

Synopsis:

AT+AH<CR><LF>

5.4.3.3. Parameter Description:

None.

5.4.3.4. Examples:

Ex. 5.111. To query the status of Apple authentication processor:
➔ AT+AH<CR><LF> ← *query the status of Apple authentication processor.*

← AH=01<CR><LF> ← *Report the Apple authentication processor works normally.*

5.4.4. CR—Connect to the remote RFCOMM device

5.4.4.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth RFCOMM device. If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected RFCOMM device.

5.4.4.2. Syntax:

Synopsis:

AT+CR[=BdAddr]<CR><LF>

5.4.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth RFCOMM device to connect.	O	

5.4.4.4. Examples:

Ex. 5.112. To connect to the last connected RFCOMM device:
→ AT+CR<CR><LF> ← *connect to the last connected RFCOMM device.*
← RS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected RFCOMM device which address is 00:18:96:00:AB:CD.*
← CR=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← RS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected RFCOMM device.*

Ex. 5.113. To connect to the specified RFCOMM device:
→ AT+CR=00189600000A<CR><LF> ← *connect to the specified RFCOMM device: 00:18:96:00:00:0A.*
← RS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified RFCOMM device which address is 00:18:96:00:00:0A.*
← CR=00,00189600000A<CR><LF> ← *connecting result: success.*
← RS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified RFCOMM device.*

5.4.5. DR—Disconnect with the remote RFCOMM device

5.4.5.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth RFCOMM device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected RFCOMM devices.

5.4.5.2. Syntax:

Synopsis:

AT+DR[=BdAddr]<CR><LF>

5.4.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth RFCOMM device to disconnect.	O	

5.4.5.4. Examples:

Ex. 5.114. To disconnect with all of the connected RFCOMM device:
➔ AT+DR<CR><LF> ← *disconnect with all of the connected RFCOMM device.*
← RS=00<CR><LF> ← *the RFCOMM channel of Bluetooth module is now disconnected and is connectable.*

Ex. 5.115. To disconnect to the specified RFCOMM device:
➔ AT+DR=00189600000A<CR><LF> ← *disconnect with the specified* RFCOMM device*: 00:18:96:00:00:0A.*
← RS=00<CR><LF> ← *the* RFCOMM *channel of Bluetooth module is now disconnected and is connectable.*

5.4.6. RS—Query the state of RFCOMM channel

5.4.6.1. Description:

This command is used to query the state of RFCOMM channel.

5.4.6.2. Syntax:

Synopsis:

AT+RS<CR><LF>

5.4.6.3. Parameter Description:

None.

5.4.6.4. Examples:

Ex. 5.116. To query the state of RFCOMM channel:
➔ AT+RS<CR><LF> ← *query the state of RFCOMM channel.*
← RS=01,00189600ABCD<CR><LF> ← *the RFCOMM channel of Bluetooth module is now connecting to the remote RFCOMM device which address is 00:18:96:00:AB:CD.*

5.4.7. RD—Send data packet to remote RFCOMM device

5.4.7.1. Description:

This command is used to send a data packet to the remote RFCOMM device.

5.4.7.2. Syntax:

Synopsis:	Comments
AT+RD{=ChannelId,DataLen,Data}<CR><LF>	For MA41 and MA46
AT+RD{=DataLen,Data}<CR><LF>	For MB05, and MB18

5.4.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelId	The RFCOMM channel ID which will be used to send the data packet. Value: 00h—0xh(x is the maximum RFCOMM instance count, refer to 5.1.1)	M	Only available for MA41 and MA46
DataLen	The length in bytes of the data to be sent. Value: 00h-F9h	M	
Data	The raw data.	M	

5.4.7.4. Examples:

Ex. 5.117. To send data to remote RFCOMM device (for MA41 and MA46):
➔ AT+RD=00,0A,1234567890<CR><LF> ← *send a data packet, the data length is 10(Dec).*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.5. DUN commands

The DUN commands are only available for MA41 and MA46.

5.5.1. NN—Query or change the service name of DUN

5.5.1.1. Description:

This command can query or change the service name of DUN profile. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost. If the new configuration is adopted by the Bluetooth module, the module will perform a reboot, the non-memorable settings will return to their default value. Therefore, it is recommended to send this command first if necessary.
If the parameter is not presented, the Bluetooth module will report current profile configuration by the Indicator NN.

5.5.1.2. Syntax:

Synopsis:

AT+NN[=Name]<CR><LF>

5.5.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The new service name of DUN profile. Length: 1—16 characters Default: DUN Dev.	O	

5.5.1.4. Examples:

Ex. 5.118. To query current service name of DUN profile:
➔ AT+NN<CR><LF> ← *query current service name.*

← NN=DUN Dev<CR><LF> ← *report current service name, it's "DUN Dev".*

Ex. 5.119. To change the service name of DUN profile:
→ AT+NN=Modem<CR><LF> ← *change the service name to "Modem"*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.5.2. CN—Connect to a remote DUN DCE device

5.5.2.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth DUN DCE device. If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected DUN DCE device.
This command is only available when the module works as a DUN DTE(Data Terminal) device.

5.5.2.2. Syntax:

Synopsis:

AT+CN[=BdAddr]<CR><LF>

5.5.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth DUN DCE device to connect.	O	

5.5.2.4. Examples:

Ex. 5.120. To connect to the last connected DUN DCE device:
→ AT+CN<CR><LF> ← *connect to the last connected DUN DCE device.*
← NS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected DUN DCE device which address is 00:18:96:00:AB:CD.*
← CN=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← NS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected DUN DCE device.*

Ex. 5.121. To connect to the specified DUN DCE device:
→ AT+CN=00189600000A<CR><LF> ← *connect to the specified DUN DCE device: 00:18:96:00:00:0A.*
← NS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified DUN DCE device which address is 00:18:96:00:00:0A.*
← CN=00,00189600000A<CR><LF> ← *connecting result: success.*
← NS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified DUN DCE device.*

5.5.3. DN—Disconnect with the remote DUN device

5.5.3.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth DUN device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected DUN devices.

5.5.3.2. Syntax:

Synopsis:

AT+DN[=BdAddr]<CR><LF>

5.5.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth DUN device to disconnect.	O	

5.5.3.4. Examples:

Ex. 5.122. To disconnect with all of the connected DUN devices:
➔ AT+DN<CR><LF> ← *disconnect with all of the connected DUN devices.*
← NS=00<CR><LF> ← *the DUN channel of Bluetooth module is now disconnected and is connectable.*

Ex. 5.123. To disconnect to the specified DUN device:
➔ AT+DN=00189600000A<CR><LF> ← *disconnect with the specified device 00:18:96:00:00:0A.*
← NS=00<CR><LF> ← *the DUN channel of Bluetooth module is now disconnected and is connectable.*

5.5.4. NS—Query state of DUN channel

5.5.4.1. Description:

This command is used to query the state of DUN channel.

5.5.4.2. Syntax:

Synopsis:

AT+NS<CR><LF>

5.5.4.3. Parameter Description:

None.

5.5.4.4. Examples:

Ex. 5.124. To query the state of DUN channel:
➔ AT+NS<CR><LF> ← *query the state of DUN channel.*
← NS=02,00189603ABCD<CR><LF> ← *the DUN channel is now connected with the remote device which address is 00:18:96:02:AB:CD.*

5.5.5. ND—Send data packet to remote DUN device

5.5.5.1. Description:

This command is used to send a data packet to the remote DUN device.

5.5.5.2. Syntax:

Synopsis:

AT+ND{=Channel,DataLen,Data}<CR><LF>

5.5.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelId	The DUN channel ID which will be used to send the data packet. Value: 00h (Only one DUN channel is available)	M	
DataLen	The length in bytes of the data to be sent. Value: 00h-FFh	M	
Data	The raw data.	M	

5.5.5.4. Examples:

Ex. 5.125. To send data to remote DUN device:
➔ AT+ND=00,04,AT<CR><LF><CR><LF> ← *send a data packet, the data length is 04(Dec), the data to be sent is: "**AT<CR><LF>**".*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.6. OPP Commands

The OPP commands are only available for the module/firmware which supports the OPP profile.

5.6.1. CO—Connect to a remote OPPS device

5.6.1.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth OPPS(OPP Server) device. If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected OPPS device.
This command is only available when OPPC(OPP Client) function is enabled.

5.6.1.2. Syntax:

Synopsis:

AT+CO[=BdAddr]<CR><LF>

5.6.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth OPPS device to connect.	O	

5.6.1.4. Examples:

Ex. 5.126. To connect to the last connected OPPS device:
➔ AT+CO<CR><LF> ← *connect to the last connected OPPS device.*
← OS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected OPPS device which address is 00:18:96:00:AB:CD.*
← CO=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← OS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected OPPS device.*

Ex. 5.127. To connect to the specified OPPS device:
➔ AT+CN=00189600000A<CR><LF> ← *connect to the specified OPPS device: 00:18:96:00:00:0A.*
← NS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified OPPS device which address is 00:18:96:00:00:0A.*
← CN=00,00189600000A<CR><LF> ← *connecting result: success.*
← NS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified OPPS device.*

5.6.2. DO—Disconnected with the remote OPP device

5.6.2.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth OPP device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected OPP devices.

5.6.2.2. Syntax:

Synopsis:

AT+DO[=BdAddr]<CR><LF>

5.6.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth OPP device to disconnect.	O	

5.6.2.4. Examples:

Ex. 5.128. To disconnect with all of the connected OPP devices:
➔ AT+DO<CR><LF> ← *disconnect with all of the connected OPP devices.*
← OS=00<CR><LF> ← *the OPP channel of Bluetooth module is now disconnected and is connectable.*

Ex. 5.129. To disconnect to the specified OPP device:
➔ AT+DO=00189600000A<CR><LF> ← *disconnect with the specified device 00:18:96:00:00:0A.*
← OS=00<CR><LF> ← *the OPP channel of Bluetooth module is now disconnected and is connectable.*

5.6.3. OS—Query state of OPP channel

5.6.3.1. Description:

This command is used to query the state of OPP channel.

5.6.3.2. Syntax:

Synopsis:

AT+OS<CR><LF>

5.6.3.3. Parameter Description:

None.

5.6.3.4. Examples:

Ex. 5.130. To query the state of OPP channel:
➔ AT+OS<CR><LF> ← *query the state of OPP channel.*
← OS=02,00189602ABCD<CR><LF> ← *the OPP channel is now connected with the remote device which address is 00:18:96:02:AB:CD.*

5.6.4. OA—OPPC push object file name

5.6.4.1. Description:

This command is used to send the file size and file name of the object which will be pushed to the remote OPPS device.
This command is only available when OPPC(OPP Client) function is enabled and a remote OPPS device is connected.

5.6.4.2. Syntax:

Synopsis:

AT+OA{=ObjSize,NameLen,FileName}<CR><LF>

5.6.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ObjSize	The size in bytes of the object which will be pushed. Value: 00000000h—FFFFFFFFh	M	
NameLen	The length in bytes of the object name. The terminator 0000 is NOT included. The name length must be even, because the name is encoded in UTF-16. Value: 02h—FEh	M	
FileName	The file name in UTF-16 of the object. The terminator 0000 should NOT be included.	M	

5.6.4.4. Examples:

Ex. 5.131. To send the file size and file name of the object, the example is shown in HEX mode:
➔42 43 3A 4F 41 3D 30 30 30 30 30 30 33 30 2C 31 30 2C 00 43 00 61 00 72 00 64 00 2E 00 76 00 63 00 66 0D 0A
← *the object size is 48 Bytes(30 30 30 30 30 30 33 30=00000030h).*
the name length is 16 Bytes(31 30 = 10h).
*The file name is "**Card.vcf**" (00 43 = 'C', 00 61 = 'a', 00 72 = 'r', 00 64 = 'd', 00 2E = '.', 00 76 = 'v', 00 63 = 'c', 00 66 = 'f').*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.6.5. OY—OPPC push object file type

5.6.5.1. Description:

This command is used to send the file type of the object which will be pushed to the remote OPPS device.
This command is only available when OPPC(OPP Client) function is enabled and a remote OPPS device is connected.

5.6.5.2. Syntax:

Synopsis:

AT+OY{=TypeLen,Type}<CR><LF>

5.6.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
TypeLen	The length in bytes of the object type. The terminator 00 is NOT included. Value: 01h—FFh	M	
Type	The type of the object. The terminator 00 should NOT be included.	M	

5.6.5.4. Examples:

Ex. 5.132. To send the file type of the object:
➔ AT+OY=0C,text/x-vcard<CR><LF> ← *the type length is 12(0Ch).*
*the object type is "**text/x-vcard**".*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.6.6. OT—OPPC push object data packet

5.6.6.1. Description:

This command is used to send a data packet of the object which will be pushed to the remote OPPS device.
This command is only available when OPPC(OPP Client) function is enabled and a remote OPPS device is connected.

5.6.6.2. Syntax:

Synopsis:

AT+OT{=Final,PacketLen,Pakcet}<CR><LF>

5.6.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Final	To indicate if this is the final or only packet of this object. Value: 00h or 01h 00: This packet is not the final packet of this object 01: This packet is the final packet of this object.	M	
PacketLen	The length in bytes of the packet. Value: 01h—FFh	M	
Packet	The raw data of the packet.	M	

5.6.6.4. Examples:

Ex. 5.133. To send a data packet of the object, the example is shown in HEX mode:
➔ 42 43 3A 4F 54 3D 30 31 2C 33 30 2C 42 45 47 49 4E 3A 56 43 41 52 44 0D 0A 56 45 52 53 49 4F 4E 3A 32 2E 31 0D 0A 4E 3A 4D 69 63 68 61 65 6C 0D 0A 45 4E 44 3A 56 43 41 52 44 0D 0A 0D 0A
← *this is the final/only packet of this object (30 31 = 01h)*
the length of this packet is 48 Bytes (33 30 = 30h).
the raw data of the packet is displayed in red(42 45 47…… 52 44 0D 0A).

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.7. BLE commands

The BLE commands are only available for MB18 which is a Bluetooth 2.1 and Bluetooth Low Energy(BLE) Combo module.

5.7.1. GA—Query or change the GAP appearance characteristic

5.7.1.1. Description:

This command can query or configure the GAP device appearance character value of Bluetooth module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command or the module is rebooted. It means the Bluetooth module will not remember the configuration, and after the Bluetooth module has been powered off, the configuration will be lost.
If the parameter is not presented, the Bluetooth module will report current appearance by the Indicator GA.
The appearance is also used in the advertising. When updated, the value will be used in next time the module start advertising.
Standard definitions of appearance could be found in Table 5.5 .
Table 5.5 Code of GAP Appearance

Value(DEC)	Device Type	Category
0	Unknown	None
64	Generic Phone	Generic category
128	Generic Computer	Generic category
192	Generic Watch	Generic category
193	Watch: Sports Watch	Watch subtype
256	Generic Clock	Generic category
320	Generic Display	Generic category
384	Generic Remote Control	Generic category
448	Generic Eye-glasses	Generic category
512	Generic Tag	Generic category
576	Generic Keyring	Generic category
640	Generic Media Player	Generic category
704	Generic Barcode Scanner	Generic category
768	Generic Thermometer	Generic category
769	Thermometer: Ear	Thermometer subtype
832	Generic Heart rate Sensor	Generic category
833	Heart Rate Sensor: Heart Rate Belt	Heart Rate Sensor subtype
896	Generic Blood Pressure	Generic category
897	Blood Pressure: Arm	Blood Pressure subtype
898	Blood Pressure: Wrist	Blood Pressure subtype
960	Human Interface Device (HID)	HID Generic
961	Keyboard	HID subtype
962	Mouse	HID subtype
963	Joystick	HID subtype
964	Gamepad	HID subtype
965	Digitizer Tablet	HID subtype
966	Card Reader	HID subtype
967	Digital Pen	HID subtype
968	Barcode Scanner	HID subtype
1024	Generic Glucose Meter	Generic category
1088	Generic: Running Walking Sensor	Generic category
1089	Running Walking Sensor: In-Shoe	Running Walking Sensor subtype
1090	Running Walking Sensor: On-Shoe	Running Walking Sensor subtype
1091	Running Walking Sensor: On-Hip	Running Walking Sensor subtype
1152	Generic: Cycling	Generic category
1153	Cycling: Cycling Computer	Cycling subtype
1154	Cycling: Speed Sensor	Cycling subtype

1155	Cycling: Cadence Sensor	Cycling subtype
1156	Cycling: Power Sensor	Cycling subtype
1157	Cycling: Speed and Cadence Sensor	Cycling subtype

5.7.1.2. Syntax:

Synopsis:

AT+GA[=Appearance]<CR><LF>

5.7.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Appearance	The GAP appearance characteristic of the Bluetooth module. Value: 0000h—FFFFh as defined by Bluetooth Assigned Number Default: per firmware version	O	

5.7.1.4. Examples:

Ex. 5.134. To query current appearance of Bluetooth module:
➔ AT+GA<CR><LF> ← *query current appearance value.*
← GA=03C1<CR><LF> ← *report current appearance value: 03C1 as a keyboard.*

Ex. 5.135. To configure the appearance of Bluetooth module:
➔ AT+GA=0301<CR><LF> ← *configure the module appearance value: 0301.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.7.2. LP—Query or change the preferred connection parameters characteristic

5.7.2.1. Description:

This command can query or configure the GAP preferred connection parameter characteristic of Bluetooth module. Once configured, the configuration will take effect at the next time when a Bluetooth connection is being established and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current configuration by the Indicator LP.

5.7.2.2. Syntax:

Synopsis:

AT+LP[=MinInterval,MaxInterval,Latecy,SupervisionTimeout]<CR><LF>

5.7.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

MinInterval	The minimum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means no specific define) Default depends on firmware	M
MaxInterval	The maximum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means no specific define) Default depends on firmware	M
Latecy	The slave latency for the connection in number of connection events. Hex Value: xxxx (0000 to 03E8) Default depends on firmware	M
SupervisionTimeout	The connection supervisor timeout nultiplier as a nultple of 10ms. Hex Value: xxxx (0006 to 0C80, FFFF means no specific define) Default depends on firmware	M

5.7.2.4. Examples:

Ex. 5.136. To query current preferred connection parameters of the Bluetooth module:
➔ AT+LP<CR><LF> ← *query the current preferred connection parameters.*
← LP=0010,0050,0004,0258<CR><LF> ← *report the current preferred connection parameters.*

Ex. 5.137. To set the preferred connection parameters of the Bluetooth module:
➔ AT+LP=0010,0050,0004,0258<CR><LF> ← *set new connection parameters.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.7.3. AT—Query or change the advertising state

5.7.3.1. Description:

This command can query or change the advertising state of BLE channel. Only when the Bluetooth module is advertising, it can be found and connected by a BLE Central device.
If the parameter is not presented, the Bluetooth module will report current advertising state by the Indicator AT.

5.7.3.2. Syntax:

Synopsis:

AT+AT[=State]<CR><LF>

5.7.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

State	The new state of advertising. Value: 00h or 01h 00: Not advertising. 01: Advertising to a known BLE central. 02: Advertising to white list (if bonded) or uncertain BLE central (if not bonded).	M

5.7.3.4. Examples:

Ex. 5.138. To query the current advertising state of the Bluetooth module:
➔ AT+AT<CR><LF> ← *query the current advertising state.*
← AT=00<CR><LF> ← *report the Bluetooth module is not advertising currently.*

Ex. 5.139. To make Bluetooth module advertising:
➔ AT+AT=01<CR><LF> ← *make Bluetooth module advertising.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.7.4. DL—Disconnect with remote BLE central device

5.7.4.1. Description:

This command will make Bluetooth module to disconnect with the remote BLE central device.

5.7.4.2. Syntax:

Synopsis:

AT+DL<CR><LF>

5.7.4.3. Parameter Description:

None.

5.7.4.4. Examples:

Ex. 5.140. To disconnect with the connected devices:
➔ AT+DL<CR><LF> ← *disconnect with current connected device.*
← LS=00<CR><LF> ← *the Bluetooth module is now disconnected and connectable.*

5.7.5. LS—Query the state of BLE channel

5.7.5.1. Description:

This command is used to query the status of BLE channel

5.7.5.2. Syntax:

Synopsis:

AT+LS<CR><LF>

5.7.5.3. Parameter Description:

None.

5.7.5.4. Examples:

Ex. 5.141. To query the state of BLE channel:
➔ AT+LS<CR><LF> ← *query the state of BLE connection.*
← LS=02,00189600ABCD,01<CR><LF> ← *the Bluetooth module is now connected to the BLE central device whose random address is 00:18:96:00:AB:CD.*

5.7.6. LI—Query the RSSI of BLE connection

5.7.6.1. Description:

This command is used to query the RSSI (radio signal strength indication) of the current BLE connection.

5.7.6.2. Syntax:

Synopsis:

AT+LI<CR><LF>

5.7.6.3. Parameter Description:

None.

5.7.6.4. Examples:

Ex. 5.142. To query the RSSI of current connection:
➔ AT+LI<CR><LF> ← *query the RSSI of current BLE connection.*
← LI=BB<CR><LF> ← *the RSSI of the current BLE connection is -69dBm.*

5.7.7. LD—Send data packet to remote BLE central

This command is used to send a data packet to the remote BLE central device.

5.7.7.1. Syntax:

Synopsis:

AT+LD{=DataLen,Data}<CR><LF>

5.7.7.2. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DataLen	The length in bytes of the data to be sent. Value: 00h-FFh	M	
Data	The raw data.	M	

5.7.7.3. Examples:

Ex. 5.143. To send data to remote BLE central device:

➔ AT+LD=0A,1234567890<CR><LF> ← *send a data packet, the data length is 10(Dec).*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.8. HFP Commands

The HFP commands are only available for MB05, and MB18 module.

5.8.1. CH—Connect to the remote HFP device

5.8.1.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth HFP device (generally, it's a mobile phone, tablet or laptop, etc.). If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected HFP device.

5.8.1.2. Syntax:

Synopsis:

AT+CH[=BdAddr]<CR><LF>

5.8.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth HFP device to connect.	O	

5.8.1.4. Examples:

Ex. 5.144. To connect to the last connected HFP device:
➔ AT+CH<CR><LF> ← *connect to the last connected HFP device.*
← HS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected HFP device which address is 00:18:96:00:AB:CD.*
← CH=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← HS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected HFP device.*

Ex. 5.145. To connect to the specified HFP device:
➔ AT+CH=00189600000A<CR><LF> ← *connect to the specified HFP device: 00:18:96:00:00:0A.*
← HS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified HFP device which address is 00:18:96:00:00:0A.*
← CH=00,00189600000A<CR><LF> ← *connecting result: success.*
← HS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified HFP device.*

5.8.2. DH—Disconnect with the remote HFP device

5.8.2.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth HFP device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected HFP devices.

5.8.2.2. Syntax:

Synopsis:

AT+DH[=BdAddr]<CR><LF>

5.8.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth HFP device to disconnect.	O	

5.8.2.4. Examples:

Ex. 5.146. To disconnect with all of the connected HFP devices:
➔ AT+DH<CR><LF> ← *disconnect with all of the connected HFP devices.*
← HS=00<CR><LF> ← *the HFP channel 0 of Bluetooth module is now disconnected and is connectable.*

Ex. 5.147. To disconnect to the specified A2DP source device:
➔ AT+DH=00189600000A<CR><LF> ← *disconnect with the specified HFP device: 00:18:96:00:00:0A.*
← HS=00<CR><LF> ← *the HFP channel 0 of Bluetooth module is now disconnected and is connectable.*

5.8.3. HS—Query the state of HFP channel

5.8.3.1. Description:

This command is used to query the state of HFP channel.

5.8.3.2. Syntax:

Synopsis:

AT+HS<CR><LF>

5.8.3.3. Parameter Description:

None.

5.8.3.4. Examples:

Ex. 5.148. To query the state of HFP channel:
➔ AT+HS<CR><LF> ← *query the state of HFP channel.*
← HS=01,00189600ABCD<CR><LF> ← *the HFP channel of Bluetooth module is now connecting to the remote HFP device which address is 00:18:96:00:AB:CD.*

5.8.4. AR—Answer or reject an incoming call

5.8.4.1. Description:

This command is used to accept or reject an incoming call.

5.8.4.2. Syntax:

Synopsis:

AT+AR{=AcpOrRej}<CR><LF>

5.8.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
AcpOrRej	Accept or Reject. Value: 00h or 01h 00h: Reject an incoming call 01h: Accept an incoming call	M	

5.8.4.4. Examples:

Ex. 5.149. To accept an incoming call:

← CC=01,01<CR><LF> ← *indicate there is an incoming call, and remote device support in-band ring.*
→ AT+AR=01<CR><LF> ← *accept the incoming call.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← CC=04<CR><LF> ← *indicate the call state is activated.*

Ex. 5.150. To reject an incoming call:

← CC=01,01<CR><LF> ← *indicate there is an incoming call, and remote device support in-band ring.*
→ AT+AR=00<CR><LF> ← *reject the incoming call.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← CC=00<CR><LF> ← *indicate the call is rejected and the call state is idle.*

5.8.5. HU—Hang up an active call

5.8.5.1. Description:

This command is used to hang up an active call.

5.8.5.2. Syntax:

Synopsis:

AT+HU<CR><LF>

5.8.5.3. Parameter Description:

None.

5.8.5.4. Examples:

Ex. 5.151. To hang up an active call:

← CC=04<CR><LF> ← *indicate the call state is activated.*
→ AT+HU<CR><LF> ← *hang up the active call.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← CC=00<CR><LF> ← *indicate the call state is activated.*

5.8.6. TC—Transfer a call

5.8.6.1. Description:

This command is used to transfer a call between the Bluetooth module and remote HFP device (the mobile phone).

5.8.6.2. Syntax:

Synopsis:

AT+TC<CR><LF>

5.8.6.3. Parameter Description:

None.

5.8.6.4. Examples:

Ex. 5.152. To transfer the active call between the Bluetooth module and mobile phone:

➔ AT+TC<CR><LF> ← *transfer the active call to the mobile phone.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+TC<CR><LF> ← *transfer the active call to the Bluetooth mobile.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.8.7. MU—Mute or unmute the microphone

5.8.7.1. Description:

This command is used to mute or unmute the microphone.

5.8.7.2. Syntax:

Synopsis:

AT+MU<CR><LF>

5.8.7.3. Parameter Description:

None.

5.8.7.4. Examples:

Ex. 5.153. To mute or unmute the microphone:

➔ AT+MU<CR><LF> ← *mute the microphone.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+MU<CR><LF> ← *unmute the microphone.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.8.8. LR—Last number redial

5.8.8.1. Description:

This command is used to redial the last number.

5.8.8.2. Syntax:

Synopsis:

AT+LR<CR><LF>

5.8.8.3. Parameter Description:

None.

5.8.8.4. Examples:

Ex. 5.154. To redial the last number:
➔ AT+LR<CR><LF> ← *redial the last number.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← CC=03<CR><LF> ← *indicate the call state is outgoing (dialing).*

5.8.9. HV—Check or set the volume of HFP voice

5.8.9.1. Description:

This command is used to check or set the volume level of specified HFP channel.
If the parameter is not presented, the module will report the current volume level of each available HFP channel.

5.8.9.2. Syntax:

Synopsis:

AT+HV{=ChVol}<CR><LF>

5.8.9.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChVol	The channel and volume level of HV to set The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the volume level(0h-Fh).	O	

5.8.9.4. Examples:

Ex. 5.155. To check current volume level of each HFP channel:
➔ AT+HV<CR><LF> ← *check the volume of HFP level.*
← HV=0F<CR><LF> ← *report the volume level of HFP channel 0 is 15(Dec).*
← HV=1A<CR><LF> ← *report the volume level of HFP channel 1 is 10(Dec).*

Ex. 5.156. To set the volume of HFP channel 0 to level 10:
➔ AT+HV=0A<CR><LF> ← *set the volume of HFP channel 0 to level 10.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.9. A2DP Commands

The A2DP commands are only available for MB05, and MB18 module.

5.9.1. OD—Query or change the optional decoder used by A2DP

5.9.1.1. Description:

This command can query or change the configuration of optional decoder used by A2DP. Once configured, the configuration will take effect at the next boot and until the next time the module is configured by this command. Therefore, a manually reboot is needed to make the new configuration take effect and the Bluetooth module will remember the configuration, and
even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current profile configuration by the Indicator OD.

5.9.1.2. Syntax:

Synopsis:

AT+OD[=Mp3,Aac,FastStream,AptX,AptXLL]<CR><LF>

5.9.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Mp3	The new status of MP3 decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	O	
Aac	The new status of AAC decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	O	
FastStream	The new status of Fast Stream decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disable	O	
AptX	The new status of Apt-X decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	O	
AptXLL	The new status of Apt-X LL decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	O	

Notes:

1. Even if the module has implemented the function of optional decoder, it is illegal to use it and/or may not work in your product without the corresponding license from its owner. It is the

customer's responsibility to get the license from the owner of the decoder.

5.9.1.4. Examples:

Ex. 5.157. To query current configuration of optional decoder:
➔ AT+OD<CR><LF> ← *query current configuration of optional decoder.*
← OD=00,00,00,00,00<CR><LF> ← *report current configuration of optional decoder: all optional decoder is disabled*

Ex. 5.158. To change the configuration of optional decoder:
➔ AT+OD=00,01,00,01,00<CR><LF> ← *change the configuration of optional decoder: enable the AAC and Apt-X decoder.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.9.2. CM—Connect to the remote A2DP source device

5.9.2.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth A2DP source device (generally, it's a mobile phone, tablet or laptop, etc.). If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected A2DP source device.

5.9.2.2. Syntax:

Synopsis:

AT+CM[=BdAddr]<CR><LF>

5.9.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth A2DP source device to connect.	O	

5.9.2.4. Examples:

Ex. 5.159. To connect to the last connected A2DP source device:
➔ AT+CM<CR><LF> ← *connect to the last connected A2DP source device.*
← MS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected A2DP source device which address is 00:18:96:00:AB:CD.*
← CM=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← MS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected A2DP source device.*

Ex. 5.160. To connect to the specified A2DP source device:
➔ AT+CM=00189600000A<CR><LF> ← *connect to the specified A2DP source device: 00:18:96:00:00:0A.*
← MS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified A2DP source device which address is 00:18:96:00:00:0A.*
← CM=00,00189600000A<CR><LF> ← *connecting result: success.*
← MS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified A2DP source device.*

5.9.3. DM—Disconnect with the remote A2DP source device

5.9.3.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth A2DP source device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected A2DP source devices.

5.9.3.2. Syntax:

Synopsis:

AT+DM[=BdAddr]<CR><LF>

5.9.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth A2DP source device to disconnect.	O	

5.9.3.4. Examples:

Ex. 5.161. To disconnect with all of the connected A2DP source devices:
➔ AT+DM<CR><LF> ← *disconnect with all of the connected A2DP source devices.*
← MS=00<CR><LF> ← *the A2DP channel 0 of Bluetooth module is now disconnected and is connectable.*

Ex. 5.162. To disconnect to the specified A2DP source device:
➔ AT+DM=00189600000A<CR><LF> ← *disconnect with the specified A2DP source device: 00:18:96:00:00:0A.*
← MS=00<CR><LF> ← *the A2DP channel 0 of Bluetooth module is now disconnected and is connectable.*

5.9.4. MS—Query the state of A2DP

5.9.4.1. Description:

This command is used to query the state of A2DP.

5.9.4.2. Syntax:

Synopsis:

AT+MS<CR><LF>

5.9.4.3. Parameter Description:

None.

5.9.4.4. Examples:

Ex. 5.163. To query the state of A2DP:
➔ AT+MS<CR><LF> ← query the state of A2DP.
← MS=01,00189600ABCD<CR><LF> ← *the A2DP channel 0 of Bluetooth module is now connecting to the remote A2DP source device which address is 00:18:96:00:AB:CD.*
← MS=10<CR><LF> ← *the A2DP channel 1 of Bluetooth module is now connectable.*

5.9.5. TM—Toggle the A2DP media source channel

5.9.5.1. Description:

This command is used to toggle the A2DP media source channel in double link. When the module is connected with two A2DP

source, only one media stream which is the active source can be played on the module side, in such case, the host can send this command to toggle between the two media source.

5.9.5.2. Syntax:

Synopsis:

AT+TM<CR><LF>

5.9.5.3. Parameter Description:

None.

5.9.5.4. Examples:

Ex. 5.164. Toggle the A2DP media source channel:
➔ AT+TM<CR><LF> ← *toggle the A2DP media source channel.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
Notes:
1. Only when both connected A2DP media sources are playing, the toggle operation can be adopted by the module, otherwise, an ER=01 will output by the module.
2. When the active A2DP source has been Paused, Stopped or disconnected, the other playing A2DP source will be set as the active source automatically.
3. By default, the first playing device will be set as the active source.

5.9.6. MR—Switch the audio output route

5.9.6.1. Description:

This command is used to switch the audio output route between Analog output, I2S digital output and SPDIF digital output. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command. It means the Bluetooth module will remember the configuration, and even if the Bluetooth module has been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current profile configuration by the Indicator MR.

5.9.6.2. Syntax:

Synopsis:

AT+MR[=Route]<CR><LF>

5.9.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Route	The new output route. Value: 00h—02h 00h: Analog output 01h: I2S digital output 02h: SPDIF digital output Default: 00h(Analog output).	O	

5.9.6.4. Examples:

Ex. 5.165. Query the audio route configuration:
➔ AT+MR<CR><LF> ← *query the audio route configuration.*
← MR=00<CR><LF> ← *report the current audio route is Analog output.*

Ex. 5.166. Switch the audio route to I2S digital output:
➔ AT+MR=01<CR><LF> ← *switch the audio route to I2S digital output.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.9.7. MV—Check or set the volume of A2DP music

5.9.7.1. Description:

This command is used to check or set the volume level of specified A2DP channel.
If the parameter is not presented, the module will report the current volume level of each available A2DP channel.

5.9.7.2. Syntax:

Synopsis:

AT+MV{=ChVol}<CR><LF>

5.9.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChVol	The channel and volume level of A2DP to set The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the volume level(0h-Fh).	O	

5.9.7.4. Examples:

Ex. 5.167. To check current volume level of each A2DP channel:
➔ AT+MV<CR><LF> ← *check the volume of A2DP level.*
← MV=0F<CR><LF> ← *report the volume level of A2DP channel 0 is 15(Dec).*
← MV=1A<CR><LF> ← *report the volume level of A2DP channel 1 is 10(Dec).*

Ex. 5.168. To set the volume of A2DP channel 0 to level 10:
➔ AT+MV=0A<CR><LF> ← *set the volume of A2DP channel 0 to level 10.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10. AVRCP Commands

The AVRCP commands are only available for MB05, and MB18 module.

5.10.1. NP— Query or change the status of NowPlaying function

5.10.1.1. Description:

This command can query or change the status of NowPlaying function of AVRCP. Once configured, the configuration will take effect at the next boot and until the next time the module is configured by this command. Therefore, a manually reboot is needed to make the new configuration take effect and the Bluetooth module will remember the configuration, and even if the Bluetooth module has

been powered off, the configuration will not be lost.
If the parameter is not presented, the Bluetooth module will report current profile configuration by the Indicator NP.

5.10.1.2. Syntax:

Synopsis:

AT+NP[=NowPlayingStatus]<CR><LF>

5.10.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
NowPlayingStatus	The new status of NowPlaying function. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	O	

5.10.1.4. Examples:

Ex. 5.169. To query current configuration of NowPlaying function:
➔ AT+NP<CR><LF> ← *query current configuration of NowPlaying function.*
← NP=00<CR><LF> ← *report that the NowPlaying function has been disabled currently*

Ex. 5.170. To enable the NowPlaying function:
➔ AT+NP=01<CR><LF> ← *enable the NowPlaying function*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10.2. CV—Connect to the remote AVRCP target device

5.10.2.1. Description:

This command will make the Bluetooth module to connect to the remote Bluetooth AVRCP target device (generally, it's a mobile phone or laptop, etc.). If the Bluetooth address parameter is not presented, the Bluetooth module will attempt to connect to the last connected AVRCP target device.

5.10.2.2. Syntax:

Synopsis:

AT+CV[=BdAddr]<CR><LF>

5.10.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth AVRCP target device to connect.	O	

5.10.2.4. Examples:

Ex. 5.171. To connect to the last connected AVRCP target device:

➔ AT+CV<CR><LF> ← *connect to the last connected AVRCP target device.*
← VS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected AVRCP target device which address is 00:18:96:00:AB:CD.*
← CV=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← VS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected AVRCP target device.*

Ex. 5.172. To connect to the specified AVRCP target device:
➔ AT+CV=00189600000A<CR><LF> ← *connect to the specified AVRCP target device: 00:18:96:00:00:0A.*
← VS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified AVRCP target device which address is 00:18:96:00:00:0A.*
← CV=00,00189600000A<CR><LF> ← *connecting result: success.*
← VS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified AVRCP target device.*

5.10.3. DV—Disconnect with the remote AVRCP target device

5.10.3.1. Description:

This command will make Bluetooth module to disconnect with the remote Bluetooth AVRCP target device. If the Bluetooth address parameter is not presented, the Bluetooth module will disconnect with all of the connected AVRCP target devices.

5.10.3.2. Syntax:

Synopsis:

AT+DV[=BdAddr]<CR><LF>

5.10.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BdAddr	The Bluetooth address of the Bluetooth AVRCP target device to disconnect.	O	

5.10.3.4. Examples:

Ex. 5.173. To disconnect with all of the connected AVRCP target devices:
➔ AT+DV<CR><LF> ← *disconnect with all of the connected AVRCP target devices.*
← VS=00<CR><LF> ← *the AVRCP channel 0 of Bluetooth module is now disconnected and is connectable.*

Ex. 5.174. To disconnect to the specified AVRCP target device:
➔ AT+DV=00189600000A<CR><LF> ← *disconnect with the specified AVRCP target device: 00:18:96:00:00:0A.*
← VS=00<CR><LF> ← *the AVRCP channel 0 of Bluetooth module is now disconnected and is connectable.*

5.10.4. VS—Query the state of AVRCP

5.10.4.1. Description:

This command is used to query the state of AVRCP.

5.10.4.2. Syntax:

Synopsis:

AT+VS<CR><LF>

5.10.4.3. Parameter Description:

None.

5.10.4.4. Examples:

Ex. 5.175. To query the state of AVRCP:
➔ AT+VS<CR><LF> ← query the state of AVRCP.
← VS=01,00189600ABCD<CR><LF> ← *the AVRCP channel 0 of Bluetooth module is now connecting to the remote AVRCP target device which address is 00:18:96:00:AB:CD.*
← VS=10<CR><LF> ← *the AVRCP channel 1 of Bluetooth module is now connectable.*

5.10.5. PL—Play or Pause

5.10.5.1. Description:

This command is used to send a *Play* or *Pause* command to the connected/active AVRCP target device.

5.10.5.2. Syntax:

Synopsis:

AT+PL<CR><LF>

5.10.5.3. Parameter Description:

None.

5.10.5.4. Examples:

Ex. 5.176. To send a *Play* or *Pause* command to the connected/active AVRCP target device:
➔ AT+PL<CR><LF> ← *send a Play or Pause command.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10.6. ST—Stop play

5.10.6.1. Description:

This command is used to send a *Stop* command to the connected/active AVRCP target device.

5.10.6.2. Syntax:

Synopsis:

AT+ST<CR><LF>

5.10.6.3. Parameter Description:

None.

5.10.6.4. Examples:

Ex. 5.177. To send a *Stop* command to the connected/active AVRCP target device:
➔ AT+ST<CR><LF> ← *send a Stop command.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10.7. NX—Next

5.10.7.1. Description:

This command is used to send a *Next* command to the connected/active AVRCP target device.

5.10.7.2. Syntax:

Synopsis:

AT+NX<CR><LF>

5.10.7.3. Parameter Description:

None.

5.10.7.4. Examples:

Ex. 5.178. To send a *Next* command to the connected/active AVRCP target device:

➔ AT+NX<CR><LF> ← *send a Next command.*

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10.8. PR—Previous

5.10.8.1. Description:

This command is used to send a *Previous* command to the connected/active AVRCP target device.

5.10.8.2. Syntax:

Synopsis:

AT+PR<CR><LF>

5.10.8.3. Parameter Description:

None.

5.10.8.4. Examples:

Ex. 5.179. To send a *Previous* command to the connected/active AVRCP target device:

➔ AT+PR<CR><LF> ← *send a Previous command.*

← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10.9. FF—Fast Forward

5.10.9.1. Description:

This command is used to send a *Fast Forward* command to the connected/active AVRCP target device.

5.10.9.2. Syntax:

Synopsis:

AT+FF{=State}<CR><LF>

5.10.9.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of Fast Forward. Value: 00h or 01h 00h: Fast Forward released 01h: Fast Forward pressed	M	

5.10.9.4. Examples:

Ex. 5.180. To send a *Fast Forward* command to the connected/active AVRCP target device:

➔ AT+FF=01<CR><LF> ← *send a Fast Forward pressed command.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+FF=00<CR><LF> ← *send a Fast Forward released command.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10.10. RW—Rewind

5.10.10.1. Description:

This command is used to send a *Rewind* command to the connected/active AVRCP target device.

5.10.10.2. Syntax:

Synopsis:

AT+RW{=State}<CR><LF>

5.10.10.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of Rewind. Value: 00h or 01h 00h: Rewind released 01h: Rewind pressed	M	

5.10.10.4. Examples:

Ex. 5.181. To send a *Rewind* command to the connected/active AVRCP target device:

➔ AT+RW=01<CR><LF> ← *send a Rewind pressed command.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+RW=00<CR><LF> ← *send a Rewind released command.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*

5.10.11. TV—Toggle the AVRCP target channel

5.10.11.1. Description:

This command is used to toggle the AVRCP target channel in double link. When the module is connected with two AVRCP target, only one AVRCP target which is the active target can be controlled by the module, in such case, the host can send this command to toggle between the two AVRCP targets.

5.10.11.2. Syntax:

Synopsis:

AT+TV<CR><LF>

5.10.11.3. Parameter Description:

None.

5.10.11.4. Examples:

Ex. 5.182. Toggle the AVRCP target channel:
➔ AT+TV<CR><LF> ← *toggle the AVRCP target channel.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
Notes:
1. Only when there are 2 AVRCP target devices are connected with the module, the toggle operation can be adopted by the module, otherwise, an ER=01 will output by the module.
2. When the active device of A2DP source has been changed for some reason, the active AVRCP target will be changed correspondingly.

6. Description of ASCII Indicators

6.1. General Indicators

6.1.1. OK—Command was adopted by the module

6.1.1.1. Description:

This indicator indicates a command was adopted by the Bluetooth module successfully.

6.1.1.2. Syntax:

Synopsis:

OK<CR><LF>

6.1.1.3. Parameter Description:

None.

6.1.1.4. Examples:

Ex. 6.1. To make Bluetooth module discoverable:

➔ AT+MD=01<CR><LF> ← *make Bluetooth module discoverable.*
➔ OK<CR><LF> ← *response from the module to indicate the command is adopted.*

6.1.2. ER—Error detected in the command sent by the host

6.1.2.1. Description:

This indicator indicates there is an error detected in the command sent by the host.

6.1.2.2. Syntax:

Synopsis:

ER{=ErrCode}<CR><LF>

6.1.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ErrCode	The error code to give the reason of an error. Value: 01h—03h 01: The command is not allowed in current state. 02: The command is not given in proper format. 03: The command cannot be recognized.	M	

6.1.2.4. Examples:

Ex. 6.2.
➔ AT+MD=01<CR><LF> ← *make Bluetooth module discoverable.*
➔ ER=01<CR><LF> ← *response from the module to indicate the command is not allowed in current state since the Bluetooth module is already discoverable.*

Ex. 6.3.
➔ AT+CS=00189600ABCD<CR><LF> ← *connect to the SPP device*
➔ ER=01<CR><LF> ← *response from the module to indicate the command is not allowed in current state, the possible reason can be the SPP device has already connected or there is another connect attempting is being performed.*

Ex. 6.4.
➔ AT+CS=00189600ABCD,03<CR><LF> ← *connect to the SPP device use the name ID 03h*
➔ ER=02<CR><LF> ← *response from the module to indicate the command is not given in proper format because the name ID is out of range.*

Ex. 6.5.
➔ AT+XX<CR><LF>
➔ ER=03<CR><LF> ← *response from the module to indicate the command cannot be recognized.*

6.1.3. AP—State of module

6.1.3.1. Description:

This indicator indicates the state of the Bluetooth module.

6.1.3.2. Syntax:

Synopsis:

AP{=StateCode}<CR><LF>

6.1.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
StateCode	The state of the Bluetooth module. Value: 00h—01h, FFh 00: The Bluetooth module is idle and ready. 01: The Bluetooth module is now inquiring for Bluetooth device. FF: The Bluetooth module is now initializing or power off.	M	

6.1.3.4. Examples:

Refer to the examples in section 5.1.20.4 .

6.1.4. PF—Profile configuration

6.1.4.1. Description:

This indicator will report current feature configuration of the Bluetooth module.

6.1.4.2. Syntax:

Synopsis:	Comments
PF{=SppCnt,HidCnt,RfcCnt,DunRole}<CR><LF>	For MA41 and MA46
PF{=SppCnt,HidCnt,RfcCnt,HfpCnt,A2dpCnt,AvrcpCnt}<CR><LF>	For MB05, and MB18

6.1.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
SppCnt	The maximum SPP instance count. Value: 00h—07h Default: 05	M	For MB05, and MB18, the maximum SPP instance count is 01.
HidCnt	The maximum HID instance count. Value: 00h—01h Default: 01	M	
RfcCnt	The maximum RFCOMM instance count. Value: 00h—07h Default: 01	M	For MB05, and MB18, the maximum RFCOMM instance count is 01.

DunRole	The role of DUN profile Value: 00h—02h 00: The DUN profile is disabled 01: The module works as DUN DCE(Modem) 02: The module works as DUN DTE(Data Terminal) Default: 00	M	Only available for MA41 and MA46.
HfpCnt	The maximum HFP instance count. Value: 00h—02h Default: 01	M	Only available for MB05, and MB18.
A2dpCnt	The maximum A2DP instance count. Value: 00h—02h Default: 01	M	Only available for MB05, and MB18.
AvrcpCnt	The maximum AVRCP instance count. Value: 00h—02h Default: 01	M	Only available for MB05, and MB18.

Notes:

1. The default profile configuration may be different per software version.

6.1.4.4. Examples:

Refer to the examples in section 5.1.1.4 .

6.1.5. AD—Bluetooth address of the module

6.1.5.1. Description:

This indicator reports the Bluetooth address of the module.

6.1.5.2. Syntax:

Synopsis:

AD{=BtAddr}<CR><LF>

6.1.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BtAddr	The Bluetooth address of the module. Value: 12 digits number	M	

6.1.5.4. Examples:

Refer to the examples in section 5.1.2.4 .

6.1.6. TP—Tx Power of the module

6.1.6.1. Description:

This indicator reports the transmit power of the module.

6.1.6.2. Syntax:

Synopsis:

AT+TP[=DefaultTx][,MaximumTx]<CR><LF>

6.1.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DefaultTx	Default TX power in dBm. The default TX power used for paging, inquiry, and their responses, and as the initial power for new ACL links. Value: a 8 digits signed number	M	
MaximumTx	Maximum TX power in dBm. Bluetooth power control may raise the TX power up to this value. Value: a 8 digits signed number	M	

6.1.6.4. Examples:

Refer to the examples in section 5.1.3.4 .

6.1.7. CD—Class of Device of module

6.1.7.1. Description:

This indicator reports the COD (Class of Device) of the Bluetooth module.

6.1.7.2. Syntax:

Synopsis:

CD{=Cod}<CR><LF>

6.1.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Cod	The COD of the Bluetooth module. Value: 6 digits number	M	

6.1.7.4. Examples:

Refer to the examples in section 5.1.4.4 .

6.1.8. FT—Feature configuration

6.1.8.1. Description:

This indicator will report current feature configuration of the Bluetooth module.

6.1.8.2. Syntax:

Synopsis:

FT{=ATPowerOn,ACPaired,ATLinkLost,Interval,DiscMode,DiscTimeout}<CR><LF>

6.1.8.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ATPowerOn	The attempt times of auto connect the last connected device after power on. Value: 00h—FFh 00: No auto connect attempt will be performed after power on. 01-FE: The attempt times of auto connect after power on. FF: The auto connect attempt will be performed permanently. Default: FF (Permanent)	M	
ACPaired	Auto connects after paired with a device. Value: 00 or 01 00: Disabled 01: Enabled Default: 00 (Disabled)	M	
ATLinkLost	The attempt times of reconnect after link lost. Value: 00h—FFh 00: No reconnect attempt will be performed after link lost. 01-FE: The attempt times of reconnect after link lost. FF: The reconnect attempt will be performed permanently. Default: FF (Permanent)	M	
Interval	The interval between each reconnect attempt after link lost. The unit is second. Value: 00h—FFh Default: 0A (10 seconds)	M	
DiscMode	The discoverable mode. Value: 00h—03h 00: The module will enter or quit discoverable mode just by the command AT+MD=xx. 01: The module will enter discoverable mode automatically when paired device list is empty. 02: The module will enter discoverable mode automatically when power on. 03: The module will enter discoverable mode automatically when	M	

	there is no connection. Default: 01 (Auto discoverable when empty)	
DiscTimeout	The timeout of discoverable status. The unit is second. Value: 0000h—FFFFh 0000: No timeout for discoverable status. 0001-FFFF: The timeout in second of discoverable status.	M

Notes:

1. The default feature configuration may be different per software version.

6.1.8.4. Examples:

Refer to the examples in section 5.1.5.4 .

6.1.9. MM—Man-In-The-Middle protection state

6.1.9.1. Description:

This indicator will report current Man-In-The-Middle protection state of the module.

6.1.9.2. Syntax:

Synopsis:

MM{=State}<CR><LF>

6.1.9.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of Man-In-The-Middle protection feature. Value: 00h or 02h 00: Deactivated 01: Activated 02: Activated and auto confirm the numeric comparison.	M	

6.1.9.4. Examples:

Refer to the examples in section 5.1.6.4 .

6.1.10. IO—IO capability configuration

6.1.10.1. Description:

This indicator will report current IO capability configuration of the Bluetooth module.

6.1.10.2. Syntax:

Synopsis:

IO{=IoCapability}<CR><LF>

6.1.10.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
IoCapability	The new IO capability of local device. Value: 00h – 03h 00: Display Only. The local device can only display 01: Display Yes/No. The local device can only display Yes or No. 02: Keyboard Only. The local device can only input. 03: No IO. The local device has no IO capability	M	

6.1.10.4. Examples:

Refer to the examples in section 5.1.7.4 .

6.1.11. MT—Force to be master state

6.1.11.1. Description:

This indicator will report current force to be master feature state of the Bluetooth module.

6.1.11.2. Syntax:

Synopsis:

MT{=State}<CR><LF>

6.1.11.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of force to be master feature. Value: 00h or 01h 00: Deactivated 01: Activated	M	

6.1.11.4. Examples:

Refer to the examples in section 5.1.8.4 .

6.1.12. SN—Sniff mode state

6.1.12.1. Description:

This indicator will report current sniff mode state and parameters of the Bluetooth module.

6.1.12.2. Syntax:

Synopsis:

SN{=State,MinInterval,MaxInterval,Attempt,Timeout,PassiveDuration}<CR><LF>

6.1.12.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of sniff mode. Value: 00h or 01h 00: Deactivated 01: Activated	M	
MinInterval	Minimum acceptable interval in milliseconds Value: 0002h—FFFEh; only even values, up to max, are valid Time = MinInterval x 0.625 ms Time Range: 1.25 ms to 40959 ms	M	
MaxInterval	Maximum acceptable interval in milliseconds Value: 0004h—FFFEh; only even values, up to max, are valid Time = MaxInterval x 0.625 ms Time Range: 2.5 ms to 40959 ms	M	
Attempt	Number of slots the slave shall listen when the slave is not treating this as a scatternet link. Value: 0001h—7FFFh Time = Attempt x 1.25 ms Time Range: 1.25ms to 40959 ms	M	
Timeout	Number of additional slots the slave shall listen when the slave is not treating this as a scatternet link. Value: 0001h—7FFFh Time = Timeout x 1.25 ms Time Range: 1.25ms to 40959 ms	M	
PassiveDuration	The time in seconds that the module will keep in Passive mode Value: 0001—FFFFh Time range: 1 seconds to 65535 seconds	M	

6.1.12.4. Examples:

Refer to the examples in section 5.1.9.4

6.1.13. SP—Deep sleep state

6.1.13.1. Description:

This indicator will report current deep sleep state of the Bluetooth module.

6.1.13.2. Syntax:

Synopsis:

SP{=State}<CR><LF>

6.1.13.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of deep sleep mode. Value: 00h or 01h 00: Disabled 01: Enabled	M	

6.1.13.4. Examples:

Refer to the examples in section 5.1.10.4 .

6.1.14. PN—Fixed pin code

6.1.14.1. Description:

This indicator will report current fixed pin code of the Bluetooth module.

6.1.14.2. Syntax:

Synopsis:

PN{=PinCode}<CR><LF>

6.1.14.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
PinCode	The fixed pin code of the Bluetooth module. Length: 1—16 characters Default: 0000	M	The default pin code may not be “0000” per software version.

6.1.14.4. Examples:

Refer to the examples in section 5.1.11.4

6.1.15. NM—Local friendly name

6.1.15.1. Description:

This indicator will report current local friendly name of the Bluetooth module.

6.1.15.2. Syntax:

Synopsis:

NM{=Name}<CR><LF>

6.1.15.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The local friendly name of the Bluetooth module. Length: 1—30 characters Default: Per software version.	M	

6.1.15.4. Examples:

Refer to the examples in section 5.1.12.4 .

6.1.16. IF—Host interface

6.1.16.1. Description:

This indicator will report current host interface of the module.

6.1.16.2. Syntax:

Synopsis:

IF{=HostInterface}<CR><LF>

6.1.16.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
HostInterface	The host interface of the module Value: 00h or 01h 00: UART running EHCI 01: USB CDC running EHCI	M	

6.1.16.4. Examples:

Refer to the examples in section 5.1.13.4 .

6.1.17. BR—UART baud rate

6.1.17.1. Description:

This indicator will report current UART baud of the Bluetooth module.

6.1.17.2. Syntax:

Synopsis:

BR{=BaudRate}<CR><LF>

6.1.17.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BaudRate	The baud rate of the Bluetooth module. Value: 01h—15h 01: 1200 02: 1800 03: 2400 04: 4800 05: 7200 06: 9600 07: 14400 08: 19200 09: 38400 0A: 56000 0B: 57600 0C: 115200 0D: 128000 0E: 230400 0F: 256000 10: 460800 11: 921600 12: 1382400 13: 1843200 14: 2764800 15: 3686400 Default: 06 (9600)	M	The default baud rate may not be 9600 per software version.

6.1.17.4. Examples:

Refer to the examples in section 5.1.14.4 .

6.1.18. UM—UART mode

6.1.18.1. Description:

This indicator will report current UART mode of the Bluetooth module.

6.1.18.2. Syntax:

Synopsis:

UM{=StopBits,Parity,Latency}<CR><LF>

6.1.18.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
StopBits	The stop bits of UART mode Value: 00h or 01h 00: 1 stop bit 01: 2 stop bits	M	
Parity	The parity of UART mode Value: 00h – 02h 00: No parity 01: Odd parity 02: Even parity	M	
Latency	The latency mode Value: 00h or 01h 00: Throughput priority 01: Low latency priority	M	

6.1.18.4. Examples:

Refer to the examples in section 5.1.15.4

6.1.19. UI—UART indicator output configuration

6.1.19.1. Description:

This indicator will report current UART indicator output configuration (disable or enable) of the Bluetooth module.

6.1.19.2. Syntax:

Synopsis:

UI{=State}<CR><LF>

6.1.19.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of UART indicator output mode Value: 00h or 01h 00: Disabled 01: Enabled Default: 01 (Enabled)	M	

6.1.19.4. Examples:

Refer to the examples in section 5.1.6.4 .

6.1.20. RC—Remote control function status

6.1.20.1. Description:

This indicator will report current remote control function configuration (disable or enable) of the Bluetooth module.

6.1.20.2. Syntax:

Synopsis:

RC{=State}<CR><LF>

6.1.20.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The status of remote control function Value: 00h or 01h 00: Disabled 01: Enabled Default: 00 (Disabled)	M	

6.1.20.4. Examples:

Refer to the examples in section 5.1.17.4 .

6.1.21. PM—Configuration of PIO assignment

6.1.21.1. Description:

This indicator will report current PIO assignment of the Bluetooth module.

6.1.21.2. Syntax:

Synopsis:

PM{=DSR,DTR,RI,DCD}<CR><LF>

6.1.21.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DSR	The PIO number assigned for DSR.	M	
DTR	The PIO number assigned for DTR.	M	
RI	The PIO number assigned for RI.	M	

DCD	The PIO number assigned for DCD.	M

6.1.21.4. Examples:

Refer to the examples in section 5.1.18.4 .

6.1.22. DB—Default Bypass mode

6.1.22.1. Description:

This indicator will report current configuration of default Bypass mode Bluetooth module.

6.1.22.2. Syntax:

Synopsis:

DB{=ChannelMode,Channel,SpeedMode}<CR><LF>

6.1.22.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelMode	The Bypass channel mode: Value: 00h—04h 00: Proxy mode 01: Bypass to First Connected Channel 02: Bypass to HID ASCII Channel 03: Bypass to All SPP 04: Bypass to Specified Channel Default: 01(Bypass to First Connected Channel)	M	
Channel	The new Bypass channel: Value: 00h—10h, 40h—46h 00—06: the SPP channel ID. Up to 7 SPP channels available. 07: The HID channel. 08—0E: The RFCOMM(for iAP) channel ID. Up to 7 RFCOMM channels available. 0F: The DUN channel 10: The BLE channel. Only available for and MB18. 40h—46h: the SPP Name ID.	M	

SpeedMode	The Bypass speed mode: Value: 00h or 01h 00: Normal speed 01: High speed	M	For MA41 and MA46, the high speed mode is only available for SPP channel. For MB05, and MB18, the high speed mode is available for SPP and RFCOMM channel.

6.1.22.4. Examples:

Refer to the examples in section 5.1.19.4 .

6.1.23. MD—Discoverable state

6.1.23.1. Description:

This indicator will report current discoverable state of the Bluetooth module.

6.1.23.2. Syntax:

Synopsis:

MD{=State}<CR><LF>

6.1.23.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of discoverable. Value: 00h or 01h 00: Not discoverable 01: Discoverable	M	

6.1.23.4. Examples:

Refer to the examples in section 5.1.21.4 .

6.1.24. PA—State of pairing mode

6.1.24.1. Description:

This indicator will report current state of pairing mode of the module.

6.1.24.2. Syntax:

Synopsis:

PA{=State}<CR><LF>

6.1.24.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of pairing mode. Value: 00h or 01h 00: Pairing/Bonding disabled 01: Pairing/Bonding enabled	M	

6.1.24.4. Examples:

Refer to the examples in section 5.1.22.4 .

6.1.25. CA—Connectable state

6.1.25.1. Description:

This indicator will report current connectable state of the module.

6.1.25.2. Syntax:

Synopsis:

CA{=State}<CR><LF>

6.1.25.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of connectable mode. Value: 00h or 01h 00: connect disabled 01: connect enabled	M	

6.1.25.4. Examples:

Refer to the examples in section 5.1.23.4 .

6.1.26. NC—Number of numeric comparison

6.1.26.1. Description:

This indicator will indicate the six digit number of numeric comparison.

6.1.26.2. Syntax:

Synopsis:

NC{=Number }<CR><LF>

6.1.26.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Number	The six digit number of numeric comparison. Value: 000000h – 0F423Fh	M	

6.1.26.4. Examples:

Refer to the examples in section 5.1.24.4 .

6.1.27. PK—Passkey request

6.1.27.1. Description:

This indicator indicates there is a passkey request.

6.1.27.2. Syntax:

Synopsis:

PK<CR><LF>

6.1.27.3. Parameter Description:

None.

6.1.27.4. Examples:

Refer to the examples in section 5.1.25.4 .

6.1.28. PC—PIO status

6.1.28.1. Description:

This indicator reports the status of specified PIO.

6.1.28.2. Syntax:

Synopsis:

PC{=Pio,Val}<CR><LF>

6.1.28.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Pio	The PIO number has been read. Value: 00h-0Fh or 00h-1Fh per module type	M	
Val	The value of the PIO Value: 00h – 01h 00: Logic Low 01: Logic High	M	

6.1.28.4. Examples:

Refer to the examples in section 5.1.30.4 .

6.1.29. AC—AIO voltage

6.1.29.1. Description:

This indicator reports the voltage on a specified PIO.

6.1.29.2. Syntax:

Synopsis:

AC{=Aio,Vol}<CR><LF>

6.1.29.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Aio	The AIO number has been read. Value: 00h or 01h	M	
Vol	The voltage on the AIO in mV. Value: 0000h – xxxxh	M	

6.1.29.4. Examples:

Refer to the examples in section 5.1.31.4 .

6.1.30. IR—Inquiry result

6.1.30.1. Description:

This indicator indicates the result of last inquiring.

6.1.30.2. Syntax:

Synopsis:

IR{=DevCount}<CR><LF>

6.1.30.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DevCount	The device count found in the last inquiring. Value: 00h—0Ch	M	

6.1.30.4. Examples:

Refer to the examples in section 5.1.20.4 .

6.1.31. FD—Information of found devices

6.1.31.1. Description:

This indicator indicates the Bluetooth address, RSSI and friendly name of found devices in last inquiring.

6.1.31.2. Syntax:

Synopsis:

FD{=Idx,BdAddr,Rssi}[,Name]<CR><LF>

6.1.31.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Idx	The index of found device in the last inquiring. Value: 00h—0Bh	M	
BdAddr	The Bluetooth address of the found device in the last inquiring.	M	
Rssi	The RSSI(Receive Signal Strength Indicator) of the found device in the last inquiring. Value: FFA6—FFFF or 7FFF RSSI Range: -90dBm — -1dBm 7FFF means the RSSI is unknown	M	
Name	The friendly name of the found device in the last inquiring	O	

6.1.31.4. Examples:

Refer to the examples in section 5.1.20.4 .

6.1.32. LC—List the connected devices

6.1.32.1. Description:

This indicator reports the connected devices.

6.1.32.2. Syntax:

Synopsis:	Comments
LC{=Profile,BdAddr}[,Name]<CR><LF>	This indicator is not available for MA41 and MA46 currently.

6.1.32.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Profile	The profile type which is connected. Value: 01h—06h 02h: SPP profile 03h: HID profile 03h: RFCOMM profile 04h: HFP profile 05h: A2DP profile 06h: AVRCP profile	M	
BdAddr	The Bluetooth address of the connected device.	M	
Name	The device name of the connected device	O	If get the remote name failed, this parameter will not be presented.

6.1.32.4. Examples:

Refer to the examples in section 5.1.34.4 .

6.2. SPP Indicators

6.2.1. SM—Service name of SPP profile

6.2.1.1. Description:

This indicator will report current service name of the SPP profile.

6.2.1.2. Syntax:

Synopsis:

SM{=Name}<CR><LF>

6.2.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The service name of the SPP profile Length: 1—16 characters	M	

6.2.1.4. Examples:

Refer to the examples in section 5.2.1.4 .

6.2.2. SS—State of each SPP channel

6.2.2.1. Description:

This indicator reports the state of each SPP channel.

6.2.2.2. Syntax:

Synopsis:

SS{=SppState}[,BdAddr][,NameId]<CR><LF>

6.2.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
SppState	The state of each SPP channel. The high nibble indicates the channel ID(0h-xh, x is the maximum SPP instance count, refer to 5.1.1) and the low nibble indicates the state(0h-2h). State Value: 0h-2h 0: the SPP channel is idle and connectable. 1: The SPP channel is connecting to a remote SPP device. 2: The SPP channel is connected with a remote SPP device	M	
BdAddr	The Bluetooth address of remote SPP device.	O	
NameId	The Name ID of the connection. Refer to 5.2	O	

6.2.2.4. Examples:

Refer to the examples in section 5.2.2.4 and 5.2.4.4 .

6.2.3. CS—Result of connect attempt to a remote SPP device

6.2.3.1. Description:

This indicator indicates the result of connect attempt to a remote SPP device.

6.2.3.2. Syntax:

Synopsis:

CS{=RetCode,BdAddr}<CR><LF>

6.2.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. The high nibble indicates the channel ID(0h—xh, x is the maximum SPP instance count, refer to 5.1.1) and the low nibble indicates the result code (0h—Bh). Result code value: 0h—Bh 0: Connect attempt succeeded. 1: Service search failed. 2: Service level connection establishment failed. 3: Profile instance already connected. 4: RFCOMM connection failed to be established. 5: Requested server channel not registered by this profile instance. 6: Connection attempt timed out. 7: The remote device rejected the connection. 8: The remote device terminated the connection. 9: Unsuccessful due to an abnormal disconnect while establishing the RFCOMM connection. A: The connection attempt failed because there is already a connection to that remote device on the requested RFCOMM channel. B: Connect failed due to invalid frame size request from app.	M	
BdAddr	The Bluetooth address of remote SPP device.	M	

6.2.3.4. Examples:

Refer to the examples in section 5.2.2.4 .

6.2.4. DT—Data packet received from remote SPP device

6.2.4.1. Description:

This indicator indicates there is a data packet received from a remote SPP device.

6.2.4.2. Syntax:

Synopsis:

DT{=ChannelOrNameId,DataLen,Data}<CR><LF>

6.2.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelOrNameId	The SPP channel ID or Name ID from which the data is received. Value: 00h—0xh or 40h—4xh (x is the maximum SPP instance count, refer to 5.1.1) 00—0x: the channel ID of SPP 40—4x: the name ID of SPP	M	
DataLen	The length in bytes of the data received. Value: 00h-FFh	M	
Data	The raw data.	M	

6.2.4.4. Examples:

Ex. 6.6. A data packet is received from the SPP channel 0:
← DT=00,0A,1234567890<CR><LF> ← *a data packet received from SPP channel 0, the data length is 10(Dec).*

Ex. 6.7. A data packet is received from the SPP Name ID 13h:
← DT=43,0A,1234567890<CR><LF> ← *a data packet received from SPP Name ID 43h, the data length is 10(Dec).*

6.3. HID Indicators

6.3.1. IS—State of HID channel

6.3.1.1. Description:

This indicator reports the state of HID channel.

6.3.1.2. Syntax:

Synopsis:

IS{=HidState}[,BdAddr] <CR><LF>

6.3.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
HidState	The state of HID channel. State Value: 00h-02h 00: the HID channel is idle and connectable. 01: The HID channel is connecting to a remote HID host. 02: The HID channel is connected with a remote HID host.	M	
BdAddr	The Bluetooth address of remote HID host.	O	

6.3.1.4. Examples:

Refer to the examples in section 5.3.1.4 and 5.3.3.4 .

6.3.2. CI—Result of connect attempt to a remote HID host

6.3.2.1. Description:

This indicator indicates the result of connect attempt to a remote HID host.

6.3.2.2. Syntax:

Synopsis:

CI{=RetCode,BdAddr}<CR><LF>

6.3.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. Value: 00h—04h 00: Connect attempt succeeded. 01: Connection failed. 02: Out of resource. 03: Timeout waiting for connection. 04: Disconnected remotely during setup.	M	
BdAddr	The Bluetooth address of remote HID host.	M	

6.3.2.4. Examples:

Refer to the examples in section 5.3.1.4 .

6.3.3. KR—Keyboard report from remote HID host

6.3.3.1. Description:

This indicator indicates a keyboard report is received from the remote HID host.

6.3.3.2. Syntax:

Synopsis:

KR{=IdleRate,NumLock,CapsLock,ScrollLock}<CR><LF>

6.3.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
IdleRate	The idle rate set by the remote HID host.	M	
NumLock	The Number Lock state. Value: 00h or 01h: 00: The Number Lock is deactivated. 01: The Number Lock is activated.	M	
CapsLock	The Caps Lock state. Value: 00h or 01h: 00: The Caps Lock is deactivated. 01: The Caps Lock is activated.	M	
ScrollLock	The Scroll Lock state. Value: 00h or 01h: 00: The Scroll Lock is deactivated. 01: The Scroll Lock is activated.	M	

6.3.3.4. Examples:

Ex. 6.8:
➔ KR=00,00,01,00<CR><LF> ← *a keyboard report is received from the HID host to indicate the idle rate is 0, the NumLock is deactivated, the CapsLock is activated and the ScrollLock is deactivated.*

Ex. 6.9:
➔ KR=00,01,00,00<CR><LF> ← *a keyboard report is received from the HID host to indicate the idle rate is 0, the NumLock is activated, the CapsLock is deactivated and the ScrollLock is deactivated.*

6.4. RFCOMM Indicators (for Apple iOS devices)

The RFCOMM indicators are mainly used to connect and communicate with an iOS device, such as iPod, iPhone and iPad.

6.4.1. RM—Service name of RFCOMM profile

6.4.1.1. Description:

This indicator will report current service name of the RFCOMM profile.

6.4.1.2. Syntax:

Synopsis:

RM{=Name}<CR><LF>

6.4.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The service name of the RFCOMM profile Length: 1—16 characters	M	

6.4.1.4. Examples:

Refer to the examples in section 5.4.1.4

6.4.2. PT—Protocol name

6.4.2.1. Description:

This indicator will report current protocol name of iAP application.

6.4.2.2. Syntax:

Synopsis:

PT[=ProtocolName]<CR><LF>

6.4.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ProtocolName	The protocol name of iAP application. Length: 1—30 characters Default: Per software version.	M	

6.4.2.4. Examples:

Refer to the examples in section 5.4.2.4 .

6.4.3. AH—Apple authentication processor status

6.4.3.1. Description:

This indicator will report the status of Apple authentication processor.

6.4.3.2. Syntax:

Synopsis:

AH{=Status}<CR><LF>

6.4.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

Stauts	The status of Apple authentication processor. Value: 00h or 01h 00h: The Apple authentication processor works abnormally or not found. 01h: The Apple authentication processor works normally.	M

6.4.3.4. Examples:

Refer to the examples in section 5.4.3.4 .

6.4.4. SO—State of iAP data session

6.4.4.1. Description:

This indicator indicates the iAP data session has been opened or closed.
A data session (data stream) must be opened between the Bluetooth module and an application on iOS device before they can exchange data.

6.4.4.2. Syntax:

Synopsis:

SO{=SessionOpen}<CR><LF>

6.4.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
SessionOpen	The state of iAP data session. State Value: 00h or 01h 00: the iAP data session is closed. 01: the iAP data session is opened.	M	

6.4.4.4. Examples:

Ex. 6.10. To indicate the iAP data session has been opened:
➔ SO=01<CR><LF> ← *indicates the data session has been opened.*

Ex. 6.11. To indicate the iAP data session has been closed:
➔ SO=00<CR><LF> ← *indicates the data session has been closed.*

6.4.5. RS—State of RFCOMM channel

6.4.5.1. Description:

This indicator reports the state of RFCOMM channel.

6.4.5.2. Syntax:

Synopsis:

RS{=RfcState}[,BdAddr]<CR><LF>

6.4.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RfcState	The state of RFCOMM channel. State Value: 00h-02h 00: the RFCOMM channel is idle and connectable. 01: The RFCOMM channel is connecting to a remote RFCOMM device. 02: The RFCOMM channel is connected with a remote RFCOMM device	M	
BdAddr	The Bluetooth address of remote RFCOMM device.	O	

6.4.5.4. Examples:

Refer to the examples in section 5.4.4.4 and 5.4.6.4 .

6.4.6. CR—Result of connect attempt to a remote RFCOMM device

6.4.6.1. Description:

This indicator indicates the result of connect attempt to a remote RFCOMM device.

6.4.6.2. Syntax:

Synopsis:

CR{=RetCode,BdAddr}<CR><LF>

6.4.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. Result code value: 00h—07h 00: Connect attempt succeeded. 01: Connect attempt failed. 02: The server channel was not registered.. 03: The connection timed out. 04: The connection was rejected. 05: The connection was disconnected normally. 06: The connection was disconnected abnormally. 07: The client has attempted to connect to a server channel that has already been connected to.	M	
BdAddr	The Bluetooth address of remote RFCOMM device.	M	

6.4.6.4. Examples:

Refer to the examples in section 5.4.4.4 .

6.4.7. RD—Data packet received from remote RFCOMM device

6.4.7.1. Description:

This indicator indicates there is a data packet received from a remote RFCOMM device.

6.4.7.2. Syntax:

Synopsis:	Comments
RD{=ChannelId,DataLen,Data}<CR><LF>	For MA41 and MA46
RD{=DataLen,Data}<CR><LF>	For MB05, and MB18

6.4.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelId	The RFCOMM channel ID from which the data is received. Value: 00h—0xh(x is the maximum RFCOMM instance count, refer to 5.1.1)	M	Only available for MA41 and MA46
DataLen	The length in bytes of the data received. Value: 00h-F9h	M	
Data	The raw data.	M	

6.4.7.4. Examples:

Ex. 6.12. A data packet is received from the RFCOMM device:
← RD=0A,1234567890<CR><LF> ← *a data packet received from RFCOMM device, the data length is 10(Dec).*

6.5. DUN Indicators

The DUN indicators are only available for MA41 and MA46.

6.5.1. NN—Service name of DUN profile

6.5.1.1. Description:

This indicator will report current service name of the DUN profile.

6.5.1.2. Syntax:

Synopsis:

NN{=Name}<CR><LF>

6.5.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The service name of the DUN profile Length: 1—16 characters	M	

6.5.1.4. Examples:

Refer to the examples in section 5.5.1.4

6.5.2. NS—State of DUN channel

6.5.2.1. Description:

This indicator reports the state of DUN channel.

6.5.2.2. Syntax:

Synopsis:

NS{=DunState}[,BdAddr]<CR><LF>

6.5.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DunState	The state of DUN channel. State Value: 00h-02h 00: the DUN channel is idle and connectable. 01: The DUN channel is connecting to a remote DUN device. 02: The DUN channel is connected with a remote DUN device	M	
BdAddr	The Bluetooth address of remote DUN device.	O	

6.5.2.4. Examples:

Refer to the examples in section 5.5.2.4 and 5.5.4.4 .

6.5.3. CN—Result of connect attempt to a remote DUN DCE device

6.5.3.1. Description:

This indicator indicates the result of connect attempt to a remote DUN DCE device.

6.5.3.2. Syntax:

Synopsis:

CN{=RetCode,BdAddr}<CR><LF>

6.5.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. Result code value: 00h—07h 00: Connect attempt succeeded. 01: Unsuccessful due to a service search failure. 02: Unsuccessful due to a service level connection failure. 03: Unsuccessful due to service level connection already established. 04: Unsuccessful due to RFCOMM connection failing to be established. 05: Unsuccessful due to attempt to connect to unallocated server channel. 06: Unsuccessful due to connection attempt timing out. 07: Unsuccessful due to remote device rejecting connection. 08: Unsuccessful due to remote device terminating the connection. 09: Unsuccessful due to an abnormal disconnect while establishing a RFCOMM connection. 0A: Unsuccessful due to being an invalid request. DUN DCE cannot initial connection.	M	
BdAddr	The Bluetooth address of remote DUN device.	M	

6.5.3.4. Examples:

Refer to the examples in section 5.5.2.4 .

6.5.4. ND—Data packet received from remote DUN device

6.5.4.1. Description:

This indicator indicates there is a data packet received from a remote DUN device.

6.5.4.2. Syntax:

Synopsis:

ND{=ChannelId,DataLen,Data}<CR><LF>

6.5.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelId	The DUN channel ID from which the data is received. Value: 00h (Only one DUN channel available)	M	
DataLen	The length in bytes of the data received. Value: 00h-FFh	M	
Data	The raw data.	M	

6.5.4.4. Examples:

Ex. 6.13. A data packet is received from the DUN device:
← ND=04,OK<CR><LF><CR><LF> ← *a data packet received from remote DUN device, the data length is 4(Dec), the data is: "**OK<CR><LF>**".*

6.6. OPP Indicators

The OPP indicators are only available for the module/firmware which supports the OPP profile.

6.6.1. OS—State of OPP channel

6.6.1.1. Description:

This indicator reports the state of OPP channel.

6.6.1.2. Syntax:

Synopsis:

OS{=OppState}[,BdAddr]<CR><LF>

6.6.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
OppState	The state of OPP channel. State Value: 00h-02h 00: the OPP channel is idle and connectable. 01: The OPP channel is connecting to a remote OPP device. 02: The OPP channel is connected with a remote OPP device	M	
BdAddr	The Bluetooth address of remote OPP device.	O	

6.6.1.4. Examples:

Refer to the examples in section 5.6.1.4 and 5.6.3.4 .

6.6.2. CO—Result of the connect attempt to a remote OPPS device

6.6.2.1. Description:

This indicator indicates the result of the connect attempt to a remote OPPS device.

6.6.2.2. Syntax:

Synopsis:

CO{=RetCode,BdAddr}<CR><LF>

6.6.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. Result code value: 00h—08h 00: Connect attempt succeeded. 01—08: Connect attempt failed	M	
BdAddr	The Bluetooth address of remote OPP device.	M	

6.6.2.4. Examples:

Refer to the examples in section 5.6.1.4 .

6.6.3. OA—File name of an object pushed by an OPPC device

6.6.3.1. Description:

This indicator indicates the file size and file name of an object pushed by a remote OPPC device.

6.6.3.2. Syntax:

Synopsis:

OA{=ObjSize,NameLen,FileName}<CR><LF>

6.6.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ObjSize	The size in bytes of the object which will be pushed. Value: 00000000h—FFFFFFFFh	M	
NameLen	The length in bytes of the object name. The terminator 0000 is NOT included. Value: 0002h—FFFEh	M	

FileName	The file name in UTF-16 of the object. The terminator 0000 is NOT included.	M

6.6.3.4. Examples:

Ex. 6.14. a remote OPPC device is starting to push an object, the file size and file name is given as(the example is shown in HEX mode):
←4F 41 3D 30 30 30 30 30 30 33 30 2C 31 30 2C 00 43 00 61 00 72 00 64 00 2E 00 76 00 63 00 66 0D 0A
← *the object size is 48 Bytes(30 30 30 30 30 30 33 30=00000030h).*
the name length is 16 Bytes(31 30 = 10h).
*the file name is "**Card.vcf**" (00 43 = 'C', 00 61 = 'a', 00 72 = 'r', 00 64 = 'd', 00 2E = '.', 00 76 = 'v', 00 63 = 'c', 00 66 = 'f').*

6.6.4. OY—File type of an object pushed by an OPPC device

6.6.4.1. Description:

This indicator indicates the file type of an object pushed by a remote OPPC device.

6.6.4.2. Syntax:

Synopsis:

OY{=TypeLen,Type}<CR><LF>

6.6.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
TypeLen	The length in bytes of the object type. The terminator 00 is NOT included. Value: 0001h—FFFFh	M	
Type	The type of the object. The terminator 00 is NOT included.	M	

6.6.4.4. Examples:

Ex. 6.15. To send the type of the object:
← OY=000C,text/x-vcard<CR><LF> ← *the type length is 12(000Ch).*
*the object type is "**text/x-vcard**".*

6.6.5. OT—Data packet of an object pushed by an OPPC device

6.6.5.1. Description:

This indicator indicates a data packet of an object pushed by a remote OPPC device is received.

6.6.5.2. Syntax:

Synopsis:

OT{=Final,PacketLen,Pakcet}<CR><LF>

6.6.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Final	To indicate if this is the final or only packet of this object. Value: 00h or 01h 00: This packet is not the final packet of this object 01: This packet is the final packet of this object.	M	
PacketLen	The length in bytes of the packet. Value: 0001h—FFFFh	M	
Packet	The raw data of the packet.	M	

6.6.5.4. Examples:

Ex. 6.16. To send a data packet of the object, the example is shown in HEX mode:
← 4F 54 3D 30 31 2C 33 30 2C 42 45 47 49 4E 3A 56 43 41 52 44 0D 0A 56 45 52 53 49 4F 4E 3A 32 2E 31 0D 0A 4E 3A 4D 69 63 68 61 65 6C 0D 0A 45 4E 44 3A 56 43 41 52 44 0D 0A 0D 0A
← this is the final/only packet of this object (30 31 = 01h)
the length of this packet is 48 Bytes (33 30 = 30h).
the raw data of the packet is displayed in red(42 45 47…… 52 44 0D 0A).

6.7. BLE Indicators

The BLE indicators are only available for MB18 which is a Bluetooth 2.1 and Bluetooth Low Energy Combo module.

6.7.1. LA—BLE address of the module

6.7.1.1. Description:

This indicator reports the BLE address of the module.

6.7.1.2. Syntax:

Synopsis:

LA{=AddType,BtAddr}<CR><LF>

6.7.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
AddType	The type of BLE address Value: 00h-01h 00: Public device address 01: Random device address	M	
BtAddr	The BLE address of the module. Value: 12 digits number	M	

6.7.1.4. Examples:

Refer to the examples in section 5.1.2.4 .

6.7.2. LN—BLE device name of the module

6.7.2.1. Description:

This indicator will report current BLE local friendly name of the module.

6.7.2.2. Syntax:

Synopsis:

LN{=Name}<CR><LF>

6.7.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The local friendly name of the Bluetooth module. Length: 1—19 characters Default: Per software version.	M	

6.7.2.4. Examples:

Refer to the examples in section 5.1.12.4 .

6.7.3. GA—BLE GAP appearance characteristic

6.7.3.1. Description:

This indicator indicates the GAP appearance characteristic.

6.7.3.2. Syntax:

Synopsis:

GA{=Appearance }<CR><LF>

6.7.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Appearance	The GAP appearance characteristic of the Bluetooth module. Value: 0000h—FFFFh as defined by Bluetooth Assigned Number	M	

6.7.3.4. Examples:

Refer to the examples in section 5.7.1.4 .

6.7.4. LP—BLE preferred connection parameters characteristic

6.7.4.1. Description:

This indicator report the GAP preferred connection parameter characteristic of Bluetooth module.

6.7.4.2. Syntax:

Synopsis:

LP[=MinInterval,MaxInterval,Latency,SupervisionTimeout]<CR><LF>

6.7.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
MinInterval	The minimum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means no specific define) Default depends on firmware	M	
MaxInterval	The maximum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means no specific define) Default depends on firmware	M	
Latency	The slave latency for the connection in number of connection events. Hex Value: xxxx (0000 to 03E8) Default depends on firmware	M	
SupervisionTimeout	The connection supervisor timeout nultiplier as a nultple of 10ms. Hex Value: xxxx (0006 to 0C80, FFFF means no specific define) Default depends on firmware	M	

6.7.4.4. Examples:

Refer to the examples in section 5.7.2.4 .

6.7.5. AT—State of Advertising

6.7.5.1. Description:

This indicator will report current advertising state of the BLE channel.

6.7.5.2. Syntax:

Synopsis:

AT{=State}<CR><LF>

6.7.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of advertising. Value: 00h or 01h 00: Not advertising. 01: Advertising to a known BLE central. 02: Advertising to white list (if bonded) or uncertain BLE central (if not bonded).	M	

6.7.5.4. Examples:

Refer to the examples in section 5.7.3.4 .

6.7.6. LS—State of BLE channel

6.7.6.1. Description:

This indicator reports the state of BLE channel.

6.7.6.2. Syntax:

Synopsis:

LS{=State}[,BdAddr,AddType]<CR><LF>

6.7.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of BLE channel. State Value: 00h-02h 00: the BLE channel is idle and connectable. 01: The BLE channel is connecting to a remote BLE central device. 02: The BLE channel is connected with a remote BLE central device	M	
BdAddr	The Bluetooth address of remote BLE central device.	O	
AddType	The type of Bluetooth address of remote BLE central device. Value: 00h-01h 00: Public device address 01: Random device address	O	

6.7.6.4. Examples:

Refer to the examples in section 5.7.5.4 .

6.7.7. LI—RSSI of BLE connection

6.7.7.1. Description:

This indicator reports the RSSI of the current BLE connection.

6.7.7.2. Syntax:

Synopsis:

LI[=RSSI]<CR><LF>

6.7.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RSSI	The RSSI of the current BLE connection. The value is in dBm and the accuracy is 6dB. The value is in two's complement signed format. So the 2 hex characters have a range of +127 to -128.	M	

6.7.7.4. Examples:

Refer to the examples in section 5.7.6.4 .

6.7.8. LD—Data packet received from remote BLE central device

6.7.8.1. Description:

This indicator indicates there is a data packet received from a remote BLE central device.

6.7.8.2. Syntax:

Synopsis:

LD{=DataLen,Data}<CR><LF>

6.7.8.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DataLen	The length in bytes of the data received. Value: 00h-FFh	M	
Data	The raw data.	M	

6.7.8.4. Examples:

Ex. 6.17. A data packet is received from the BLE central device:
← LD=0A,1234567890<CR><LF> ← *a data packet received from BLE central device, the data length is 10(Dec).*

6.8. HFP Indicators

The HFP indicators are only available for MB05, and MB18 module.

6.8.1. HS—State of HFP channel

6.8.1.1. Description:

This indicator reports the state of each HFP channel.

6.8.1.2. Syntax:

Synopsis:

HS{=State}[,Bdaddr]<CR><LF>

6.8.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of each HFP channel. The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the state(0h-2h). State Value: 0h-2h 0: the A2DP channel is idle and connectable. 1: The A2DP channel is connecting to a remote HFP device. 2: The A2DP channel is connected with a remote HFP device.	M	
Bdaddr	The Bluetooth address of remote HFP device.	O	

6.8.1.4. Examples:

Refer to the examples in section 5.8.1.4 and 5.8.3.4 .

6.8.2. CH—Result of connect attempt to a remote HFP device

6.8.2.1. Description:

This indicator indicates the result of connect attempt to a remote HFP device.

6.8.2.2. Syntax:

Synopsis:

CH{=RetCode,BdAddr}<CR><LF>

6.8.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. The high nibble indicates the channel ID(0h—1h) and the low nibble indicates the result code (0h—xh). Result code value: 0h—xh Value: 0h—xh 0: Connect attempt succeeded. X: A non-zero code indicates the connect attempt failed.	M	
BdAddr	The Bluetooth address of remote HFP device.	M	

6.8.2.4. Examples:

Refer to the examples in section 5.8.1.4 .

6.8.3. CC—Call State

6.8.3.1. Description:

This indicator indicates the call state of HFP.

6.8.3.2. Syntax:

Synopsis:

CC{=CallState}[,InbandRing]<CR><LF>

6.8.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
CallState	The call state of HFP. Value: 00h—09h 00h: Idle 01h: Incoming Call. In this case, the second parameter will be available to indicate whether an in-band is supported. 02h: Incoming Held (Three Way Call) 03h: Outgoing Call 04h: Active Call 05h: Call Waiting (Three Way Call) 06h: Outgoing Call (Three Way Call) 07h: Held Active (Three Way Call) 08h: Held Remaining (Three Way Call) 09h: Conference Call (Three Way Call)	M	

InbandRing	The in-band ring state of the incoming call Value: 00h or 01h 00h: The remote HFP device does not support in-band ring. 01h: The remote HFP device supports in-band ring.	O	This parameter is only presented when the call state is incoming call.

6.8.3.4. Examples:

Refer to the examples in section 5.8.4.4 and 5.8.5.4 .

6.8.4. HV—Volume of HFP voice

6.8.4.1. Description:

This indicator reports the volume level of each available HFP channel.

6.8.4.2. Syntax:

Synopsis:

HV{=ChVol}<CR><LF>

6.8.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChVol	The channel and volume level of HFP music. The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the volume level(0h-Fh).	M	

6.8.4.4. Examples:

Refer to the examples in section 5.8.9.4 .

6.9. A2DP Indicators

The A2DP indicators are only available for MB05, and MB18 module.

6.9.1. OD—Configuration of optional decoder used by A2DP

6.9.1.1. Description:

This indicator reports the configuration of optional decoder used by A2DP.

6.9.1.2. Syntax:

Synopsis:

OD{=Mp3,Aac,FastStream,AptX,AptXLL}<CR><LF>

6.9.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Mp3	The status of MP3 decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	M	
Aac	The status of AAC decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	M	
FastStream	The status of Fast Stream decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disable	M	
AptX	The status of Apt-X decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	M	
AptXLL	The status of Apt-X LL decoder. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	M	

6.9.1.4. Examples:

Refer to the examples in section 5.9.1.4 .

6.9.2. MS—State of A2DP channel

6.9.2.1. Description:

This indicator reports the state of A2DP.

6.9.2.2. Syntax:

Synopsis:

MS{=A2dpState}[,BdAddr] <CR><LF>

6.9.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
A2dpState	The state of each A2DP channel. The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the state(0h-2h). State Value: 0h-2h 0: the A2DP channel is idle and connectable. 1: The A2DP channel is connecting to a remote A2DP source device. 2: The A2DP channel is connected with a remote A2DP source device.	M	
BdAddr	The Bluetooth address of remote A2DP source device.	O	

6.9.2.4. Examples:

Refer to the examples in section 5.9.2.4 and 5.9.4.4 .

6.9.3. CM—Result of connect attempt to a remote A2DP source device

6.9.3.1. Description:

This indicator indicates the result of connect attempt to a remote A2DP source device.

6.9.3.2. Syntax:

Synopsis:

CM{=RetCode,BdAddr}<CR><LF>

6.9.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. The high nibble indicates the channel ID(0h—1h) and the low nibble indicates the result code (0h—xh). Result code value: 0h—xh Value: 0h—xh 0: Connect attempt succeeded. X: A non-zero code indicates the connect attempt failed.	M	
BdAddr	The Bluetooth address of remote A2DP source device.	M	

6.9.3.4. Examples:

Refer to the examples in section 5.9.2.4 .

6.9.4. PL—State of A2DP playing

6.9.4.1. Description:

This indicator reports the state of A2DP playing.

6.9.4.2. Syntax:

Synopsis:

PL{=PlayingState}<CR><LF>

6.9.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
PlayingState	The state of A2DP playing: The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the state(0h-1h). State Value: 0h-1h 0h: The A2DP device is now suspended or stopped. 1h: The A2DP device is now playing.	M	

6.9.4.4. Examples:

Ex. 6.18. The A2DP playing state:

← PL=01<CR><LF>	← *the A2DP device on channel 0 is not playing.*
← PL=11<CR><LF>	← *the A2DP device on channel 1 is not playing.*
← PL=00<CR><LF>	← *the A2DP device on channel 0 is not suspended or stopped.*
← PL=10<CR><LF>	← *the A2DP device on channel 1 is not suspended or stopped.*

6.9.5. DD—Decoder used by A2DP

6.9.5.1. Description:

This indicator reports the decoder used by A2DP.

6.9.5.2. Syntax:

Synopsis:

DD{=Decoder}<CR><LF>

6.9.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Decoder	The decoder used by A2DP: The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the decoder(1h-6h).	M	

Decoder Value: 1h-6h
1h: SBC decoder.
2h: MP3 decoder
3h: AAC decoder.
4h: Fast Stream decoder
5h: APT-X decoder.
6h: APT-X LL decoder

6.9.5.4. Examples:

Ex. 6.19. The decoder used by A2DP:

← DD=01<CR><LF>	← *the decoder used by A2DP of channel 0 is SBC decoder*
← DD=11<CR><LF>	← *the decoder used by A2DP of channel 1 is SBC decoder*
← DD=03<CR><LF>	← *the decoder used by A2DP of channel 0 is AAC decoder*
← DD=15<CR><LF>	← *the decoder used by A2DP of channel 1 is APT-X decoder.*

6.9.6. SR—Simple rate of A2DP audio

6.9.6.1. Description:

This indicator reports the simple rate of A2DP audio.

6.9.6.2. Syntax:

Synopsis:

SR{=SimpleRate}<CR><LF>

6.9.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
SimpleRate	The Simple Rate used by A2DP: The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the simple rate(0h or 1h). Simple Rate Value: 0h-1h 0h: 44100Hz. 1h: 48000Hz.	M	

6.9.6.4. Examples:

Ex. 6.20. The decoder used by A2DP:

← SR=00<CR><LF>	← *the simple rate used by A2DP of channel 0 is 44100Hz*
← SR=11<CR><LF>	← *the simple rate used by A2DP of channel 1 is 48000Hz*

6.9.7. MR—Audio output route

6.9.7.1. Description:

This indicator reports the audio output route.

6.9.7.2. Syntax:

Synopsis:

MR{=Route}<CR><LF>

6.9.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Route	The audio output route: Value: 00h—02h 00: Analog output 01: I2S digital output 02: SPDIF digital output	M	

6.9.7.4. Examples:

Refer to the examples in section 5.9.6.4 .

6.9.8. MV—Volume of A2DP music

6.9.8.1. Description:

This indicator reports the volume level of each available A2DP channel.

6.9.8.2. Syntax:

Synopsis:

MV{=ChVol}<CR><LF>

6.9.8.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChVol	The channel and volume level of A2DP music. The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the volume level(0h-Fh).	M	

6.9.8.4. Examples:

Refer to the examples in section 5.9.7.4 .

6.10. AVRCP Indicators

The AVRCP indicators are only available for MB05, and MB18 module.

6.10.1. NP—Status of NowPlaying function

6.10.1.1. Description:

This indicator reports the status of NowPlaying function.

6.10.1.2. Syntax:

Synopsis:

NP{=NowPlayingStatus}<CR><LF>

6.10.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
NowPlayingStatus	The status of NowPlaying status. Value: 00h or 01h 00h: Disabled 01h: Enabled Default: Disabled	M	

6.10.1.4. Examples:

Refer to the examples in section 5.10.1.4 .

6.10.2. VS—State of AVRCP channel

6.10.2.1. Description:

This indicator reports the state of AVRCP.

6.10.2.2. Syntax:

Synopsis:

VS{=AvrcpState}[,BdAddr] <CR><LF>

6.10.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

AvrcpState	The state of each AVRCP channel. The high nibble indicates the channel ID(0h-1h) and the low nibble indicates the state(0h-2h). State Value: 0h-2h 0: the AVRCP channel is idle and connectable. 1: The AVRCP channel is connecting to a remote AVRCP target device. 2: The AVRCP channel is connected with a remote AVRCP target	M
	device.	
BdAddr	The Bluetooth address of remote AVRCP target device.	O

6.10.2.4. Examples:

Refer to the examples in section 5.10.2.4 and 5.10.4.4 .

6.10.3. CV—Result of connect attempt to a remote AVRCP target device

6.10.3.1. Description:

This indicator indicates the result of connect attempt to a remote AVRCP target device.

6.10.3.2. Syntax:

Synopsis:

CV{=RetCode,BdAddr}<CR><LF>

6.10.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. The high nibble indicates the channel ID(0h—1h) and the low nibble indicates the result code (0h—xh). Result code value: 0h—xh Value: 0h—xh 0: Connect attempt succeeded. X: A non-zero code indicates the connect attempt failed.	M	
BdAddr	The Bluetooth address of remote AVRCP target device.	M	

6.10.3.4. Examples:

Refer to the examples in section 5.10.2.4 .

6.10.4. VC—Capabilities of AVRCP target

6.10.4.1. Description:

This indicator indicates the event capabilities of AVRCP target.

6.10.4.2. Syntax:

Synopsis:

VC{=Capabilities}<CR><LF>

6.10.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Capabilities	The event capabilities of AVRCP target in a 16 bit digits. This is a bit mask, each bit represent an event, if a bit is 1, means this event is supported by the AVRCP target device while 0 means this event is not supported by the AVRCP target device. The events of each bit are as follows: Bit0: Reserved. Bit1: Playback Status Changed Bit2: Track Changed Bit3: Track Reached End Bit4: Track Reached Start Bit5: Playback Pos Changed Bit6: Battery Status Changed Bit7: System Status Changed Bit8: Player App Setting Changed Bit9: Now Playing Content Changed Bit10: Available Players Changed Bit11: Addressed Player Changed Bit12: UIDs Changed Bit13: Volume Changed Bit14: Reserved Bit15: Reserved	M	

6.10.4.4. Examples:

Ex. 6.21. The event capabilities of AVRCP target device:

← VC=1F06<CR><LF> ← *the target device supports these events: Playback Status Changed, Track Changed, Player App Setting Changed, Now Playing Content Changed, Available Players Changed, Addressed Player Changed, UIDs Changed.*

6.10.5. TC—Track changed event

6.10.5.1. Description:

This indicator indicates the track changed event.
Only when the NowPlaying function has been enabled, the TC indicator is available.

6.10.5.2. Syntax:

Synopsis:

TC{=TrackIdHighWord,TrackIdLowWord}<CR><LF>

6.10.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
TrackIdHighWord	The high 32bit of new Track ID.	M	
TrackIdLowWord	The low 32bit of new Track ID.	M	

6.10.5.4. Examples:

Ex. 6.22. The track changed event:
← TC=2345ABCD,CDEF6789<CR><LF> ← *the track has been changed, and the track ID of currently played is: 2,541,626,465,132,177,289 (in Decimal)*

6.10.6. PO—Playback position changed event

6.10.6.1. Description:

This indicator indicates the playback position changed event.
Only when the NowPlaying function has been enabled, the PO indicator is available.

6.10.6.2. Syntax:

Synopsis:

PO{=Position}<CR><LF>

6.10.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Position	Current playback position of this track in millisecond (32bit).	M	

6.10.6.4. Examples:

Ex. 6.23. The track changed event:
← PO=0002AB78<CR><LF> ← *the current playback position is 174,968 ms*

6.10.7. MA—Media attributes

6.10.7.1. Description:

This indicator indicates the media attributes of current playing.
Only when the NowPlaying function has been enabled, the MA indicator is available.

6.10.7.2. Syntax:

Synopsis:

MA{=AttrId,CharSetId,AttrLen,AttrData}<CR><LF>

6.10.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
AttrId	The Attribute ID of this attribute.(8bit) Value: 01h—07h 01: Title of the media 02: Name of the Artist 03: Name of the Album 04: Number of the Media(e.g. Track number of the CD) 05: Total Number of the Media(e.g. Total track number of the CD) 06: Genre 07: Playing Time	M	
CharSetId	The character set ID of this attribute.(16bit) The character set ID is defined in IANA character set document. refer to: http://www.iana.org/assignments/character-sets	M	
AttrLen	The length in byte of this attribute. (16bit)	M	
AttrData	The raw data of the attribute's content.	M	

6.10.7.4. Examples:

Ex. 6.24. The media attribute:
← MA=01,006A,0E,Heal The World<CR><LF> ← *the Title of the media is "Heal The World"*

7. Description of IO Commands

The direction of command IO is input.
Note: The number of PIO may be different per the software version.

7.1. PIO3—Restore Factory and Disconnect Request IO

7.1.1. Description:

This IO is used to initiate a factory restore or disconnect request. During power up, it works as a factory restore IO, i.e. If the Bluetooth module detects a logic 1 on this PIO, it will performance a factory restore, otherwise, it works as a disconnect IO.

Low: Normal
High: Initiate a factory restore request (during power up)
Plus Pulse: Initiate a disconnect request
This IO can be used to initiate a disconnect request if the host does not have a UART port to control the Bluetooth module.

7.1.2. Examples:

Ex. 7.1: Initiate a factory restore request during power up

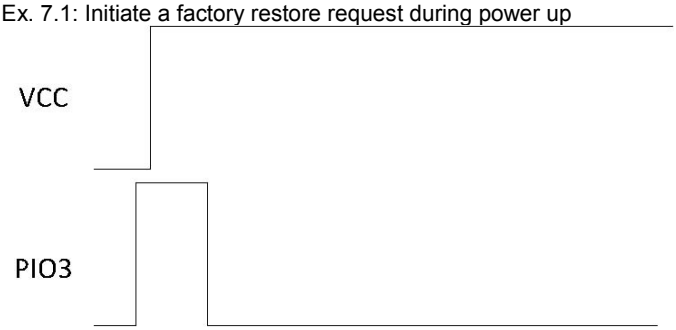

Figure 7.1: Factory Restore Request
If the Bluetooth module detects logic 1 during power up and keep at least for 300ms, it will performer a factory restore.

Ex. 7.2: Initiate a disconnect request

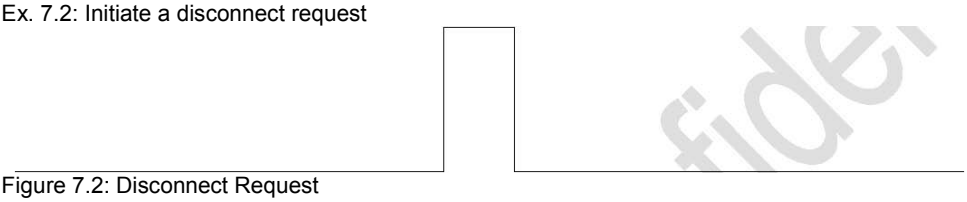

Figure 7.2: Disconnect Request
When the Bluetooth module detects the plus pulse on PIO3, it will disconnect all of the connected devices.

8. Description of IO Indicators

The direction of command IO is output.
Note: The number of PIO may be different per the software version.

8.1. PIO2—State Change Trigger IO

8.1.1. Description:

This IO indicates a state change.
Low: There is no connection has been established.
High: There is one or more connection has been established.
Minus Pulse: There is a new connection has been established or a connection has been disconnected.
The state change trigger IO is very useful when the Bluetooth module is working in Bypass mode and the Bypass channel is a SPP channel, since there is no indicator will be output from the UART port. In this case, the host can detect the state of the trigger IO to know if there is a state change occurs, and then change to Proxy Mode to inquiry the SPP and HID state.

8.1.2. Examples:

Ex. 8.1: Single connection:

Figure 8.1: State Change of Single Connection
State 1: There is no connection has been established.
State 2: A connection has been established.
State 3: The connection has been disconnected.

Ex. 8.2: Multiple connections:

Figure 8.2: State Change of Multiple Connections

State 1: There is no connection has been established.
State 2: A connection has been established.
State 3: A new connection has been established.
State 4: A connection has been disconnected.
State 5: All of the connection has been disconnected.

9. Appendix A: Examples

9.1. Use the Bluetooth module as a SPP Master Device

This example shows how to use the Bluetooth module as a SPP Master device. A SPP master device means the device who will initiate the SPP connection to a remote SPP Slave device.

Ex. 9.1:

➔ AT+PF=01,00,00<CR><LF> ← *configure the module profiles: 1 SPP channel, no HID and RFCOMM profile supported. This command is only needed when the first time use this Bluetooth module.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth module has performed a reboot and is ready now.*
➔ AT+FT=FF,00,FF,0A,01,0078<CR><LF> ← *configure the module features:*
enable the auto connection after power on as permanent mode;
disable the auto connect after paired;
enable auto reconnect after link lost as permanent mode;
set the interval of auto reconnect to 10s.
configure the discover mode as 01: auto discoverable when empty.
configure the timeout of discoverable as 120 seconds.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+IQ<CR><LF> ← *inquire the Bluetooth device. This command is not necessary if the host knows the Bluetooth address of remote device.*
← AP=01<CR><LF> ← *indicate the Bluetooth module is now inquiring.*
← IR=03<CR><LF> ← *indicate there are 3 Bluetooth devices found.*
← AP=00<CR><LF> ← *indicate Bluetooth module is now in idle.*
← FD=02,00189600000A,FFC6,SPP_DEV<CR><LF> ← *indicate the 3rd found device's address, RSSI and name.*
← FD=01,00189600000B,FFC7<CR><LF> ← *indicate the 2nd found device's address and the name is not gotten.*
← FD=00,00189600000D,FFC8,BT_DEV_1<CR><LF> ← *indicate the 1st found device's address and name.*
➔ AT+CS=00189600000A<CR><LF> ← *connect to the specified device 00:18:96:00:00:0A with the SPP profile.*
← SS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified SPP device which address is 00:18:96:00:00:0A.*
← CS=00,00189600000A<CR><LF> ← *connecting result: success.*
← SS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified SPP device and working in Bypass mode, the Bypass channel is the SPP channel 0. The host can now exchange the raw data with the remote SPP device by sending and receiving data via the UART port.*
……
……
← *Keep the UART port idle for 1 second.*
➔ AT+BP=00,00,00<CR><LF> ← *change the bypass mode to* ***Proxy Mode****.*
← *Keep the UART port idle for 1 second.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+DS=00189600000A<CR><LF> ← *disconnect with the specified device 00:18:96:00:00:0A.*
← SS=00<CR><LF> ← *the SPP channel 0 of Bluetooth module is now disconnected and is connectable.*

9.2. Use the Bluetooth Module as a SPP Slave Device

This example shows how to use the Bluetooth module as a SPP Slave device. A SPP Slave device means the device who is waiting for and will accept the SPP connection request from a remote SPP Master device.

Ex. 9.2:

➔ AT+PF=04,00,00<CR><LF> ← *configure the module profiles: 4 SPP channels, no HID and RFCOMM profile supported.*

This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth module has performed a reboot and is ready now.*
→ AT+FT=00,00,00,0A,01,0078<CR><LF> ← *configure the module features:*
disable the auto connection after power on;
disable the auto connect after paired;
disable auto reconnect after link lost;
set the interval of auto reconnect to 10s.
configure the discover mode as 01: auto discoverable when empty.
configure the timeout of discoverable as 120 seconds.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+MD=01<CR><LF> ← *make Bluetooth module discoverable. This command is not necessary if the remote device know the Bluetooth address of the Bluetooth module.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← SS=01,00189600000B<CR><LF> ← *a remote SPP device is connecting to the Bluetooth module. Its Bluetooth address is 00:18:96:00:00:0B.*
← CS=00,00189600000B<CR><LF> ← *connecting result: success.*
← SS=02,00189600000B<CR><LF> ← *the Bluetooth module is now connected to the remote SPP device and working in Bypass mode, the Bypass channel is the SPP channel 0. The host can now exchange the raw data with the remote SPP device by sending and receiving data via the UART port.*
……
……
← SS=00<CR><LF> ← *the remote SPP device has disconnected with the Bluetooth module. The Bluetooth module is now disconnected and is connectable.*

9.3. Use the Bluetooth Module as a HID Device to Send ASCII Characters to the HID Host

This example shows how to use the Bluetooth module as a HID Device(a keyboard) to send the ASCII characters to the HID Host(a computer, tablet, mobile phone, etc.).
Ex. 9.3:
→ AT+PF=00,01,00<CR><LF> ← *configure the module profiles: 1 HID channel, no SPP and RFCOMM profile supported. This command is only needed when the first time use this Bluetooth module.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth module has performed a reboot and is ready now.*
→ AT+FT=FF,00,FF,0A,01,0078<CR><LF> ← *configure the module features:*
enable the auto connection after power on as permanent mode;
disable the auto connect after paired;
enable auto reconnect after link lost as permanent mode;
set the interval of auto reconnect to 10s.
configure the discover mode as 01: auto discoverable when empty.
configure the timeout of discoverable as 120 seconds.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+BP=02,00,00<CR><LF> ← *change the bypass mode to **Bypass to HID ASCII Channel**.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+CI<CR><LF> ← *connect to the last connected HID host.*
← IS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected HID host which address is 00:18:96:00:AB:CD.*
← CI=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← IS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected HID host. The host can now send ASCII characters to the HID host by sending data to the Bluetooth module via UART port.*
→ Hello, this is Ehong! ← *the ASCII characters need to send to remote HID device*
……
……
← *Keep the UART port idle for 1 second.*
→ AT+BP=00,00<CR><LF> ← *change the bypass mode to **Proxy Mode**.*
← *Keep the UART port idle for 1 second.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+DI<CR><LF> ← *disconnect the HID channel.*
← IS=00<CR><LF> ← *the HID channel of the Bluetooth module is now disconnected and is connectable.*

9.4. Use the Bluetooth Module as a HID Device to Send HID report to the HID Host

This example shows how to use the Bluetooth module as a HID Device(a keyboard, mouse, gamepad/joystick) to send the HID report to the HID Host(a computer, tablet, mobile phone, etc.).
Ex. 9.4:
➔ AT+PF=00,01,00<CR><LF> ← *configure the module profiles: 1 HID channel, no SPP and RFCOMM profile supported. This command is only needed when the first time use this Bluetooth module.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth module has performed a reboot and is ready now.*
➔ AT+FT=FF,00,FF,0A,01,0078<CR><LF> ← *configure the module features:*
enable the auto connection after power on as permanent mode;
disable the auto connect after paired;
enable auto reconnect after link lost as permanent mode;
set the interval of auto reconnect to 10s.
configure the discover mode as 01: auto discoverable when empty.
configure the timeout of discoverable as 120 seconds.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+CI<CR><LF> ← *connect to the last connected HID host.*
← IS=01,00189600ABCD<CR><LF> ← *the Bluetooth module is now connecting to the last connected HID host which address is 00:18:96:00:AB:CD.*
← CI=00,00189600ABCD<CR><LF> ← *connecting result: success.*
← IS=02,00189600ABCD<CR><LF> ← *the Bluetooth module is now connected to the last connected HID host. The host can now send ASCII characters to the HID host by sending data to the Bluetooth module via UART port.*
➔ A1 01 00 00 04 00 00 00 00 00 ← *(shown in HEX mode)the keyboard report in raw data (binary), the key A is pressed*
➔ A1 01 00 00 00 00 00 00 00 00 ← *(shown in HEX mode)the keyboard report in raw data (binary), the pressed key is released*
➔ A1 02 10 00 ← *(shown in HEX mode)the consumer key report in raw data (binary), the Volume Up key is pressed*
➔ A1 02 00 00 ← *(shown in HEX mode)the consumer key report in raw data (binary), the pressed key is released*
➔ A1 03 01 00 00 00 00 ← *(shown in HEX mode)the mouse report in raw data (binary), the left button is pressed*
➔ A1 03 00 03 C0 FF 00 ← *(shown in HEX mode)the mouse report in raw data (binary), the mouse is move to upper-right, X = 3, Y = -4.*
➔ A1 03 00 00 00 00 FF ← *(shown in HEX mode)the mouse report in raw data (binary), the wheel is scroll up.*
➔ A1 03 00 00 00 00 00 ← *(shown in HEX mode)the mouse report in raw data (binary), the mouse is stopped, pressed button is released*
➔ A1 04 05 FE 03 02 FB 08 00 00 ← *(shown in HEX mode)the joystick/gamepad report in raw data (binary), the Throttle = 5, Left X = -2, Left Y = 3, Right X = 2, Right Y = -5, no movement on Hat Switch, no button is pressed.*
➔ A1 04 F6 00 00 00 00 05 06 40 ← *(shown in HEX mode)the joystick/gamepad report in raw data (binary), the Throttle = -10, Left X = 0, Left Y = 0, Right X = 0, Right Y = 0, Hat Switch is move towards bottom-left, Button2 and Button3 is pressed, Button15 is pressed.*
➔ A1 04 00 00 00 00 00 08 00 00 ← *(shown in HEX mode)the joystick/gamepad report in raw data (binary), the Throttle = 0, no movement on Stick, no movement on Hat Switch, no Button is pressed.*
……
……
← *Keep the UART port idle for 1 second.*
➔ AT+BP=00,00<CR><LF> ← *change the bypass mode to **Proxy Mode**.*
← *Keep the UART port idle for 1 second.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
➔ AT+DI<CR><LF> ← *disconnect the HID channel.*
← IS=00<CR><LF> ← *the HID channel of the Bluetooth module is now disconnected and is connectable.*

9.5. Use the Bluetooth Module as an Accessory of iOS Device (iAP)

This example shows how to use the Bluetooth module as an Bluetooth accessory of an iOS device(iPod, iPhone or iPad) to exchange data with the iOS device.
Ex. 9.5:
➔ AT+PF=00,00,01<CR><LF> ← *configure the module profiles: 1 RFCOMM channel, no SPP and HID profile supported. This command is only needed when the first time use this Bluetooth module.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth module has performed a reboot and is ready now.*
➔ AT+FT=00,00,00,0A,01,0078<CR><LF> ← *configure the module features:*
disable the auto connection after power on;
disable the auto connect after paired;

disable auto reconnect after link lost;
set the interval of auto reconnect to 10s.
configure the discover mode as 01: auto discoverable when empty.
configure the timeout of discoverable as 120 seconds.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+PT=com.nvc.protocol<CR><LF> ← *configure the iAP protocol name: com.nvc.protocol. This command is only needed when the first time use this Bluetooth module.*
→ AT+MD=01<CR><LF> ← *make Bluetooth module discoverable. This command is not necessary if the Bluetooth module has been paired with the iOS device already.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← RS=01,90840D00000B<CR><LF> ← *a remote iOS device is connecting to the Bluetooth module. Its Bluetooth address is 90:84:0D:00:00:0B.*
← CR=00,90840D00000B<CR><LF> ← *connecting result: success.*
← RS=02,90840D00000B<CR><LF> ← *the Bluetooth module is now connected to the remote iOS device and passed the the authentication.*
← SO=01<CR><LF> ← *the application on the iOS device has opened the data session for data transfer. Now the Bluetooth module is working in Bypass mode, the Bypass channel is the RFCOMM channel. The host can now exchange the raw data with the remote iOS device by sending and receiving data via the UART port.*
……
……
← SO=00<CR><LF> ← *the application on the iOS device has closed the data session.*
← RS=00<CR><LF> ← *the remote iOS device has disconnected with the Bluetooth module. The Bluetooth module is now disconnected and is connectable.*

9.6. Use the Bluetooth Module as an OPPC Device

This example shows how to use the Bluetooth module as an OPPC device to push object to an remote OPPS device.
Ex. 9.6:
→ AT+PF=00,00,00,00,01<CR><LF> ← *configure the module profiles: OPPC is enabled, no SPP, HID, RFCOMM or DUN profile is supported. This command is only needed when the first time use this Bluetooth module.*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth module has performed a reboot and is ready now.*
→ AT+FT=00,00,00,0A,01,0078<CR><LF> ← *configure the module features:*
disable the auto connection after power on;
disable the auto connect after paired;
disable auto reconnect after link lost;
set the interval of auto reconnect to 10s.
configure the discover mode as 01: auto discoverable when empty.
configure the timeout of discoverable as 120 seconds.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+IQ<CR><LF> ← *inquire the Bluetooth device. This command is not necessary if the host knows the Bluetooth address of remote device.*
← AP=01<CR><LF> ← *indicate the Bluetooth module is now inquiring.*
← IR=03<CR><LF> ← *indicate there are 3 Bluetooth devices found.*
← AP=00<CR><LF> ← *indicate Bluetooth module is now in idle.*
← FD=02,00189600000A,FFC6,OPPS_DEV<CR><LF> ← *indicate the 3rd found device's address, RSSI and name.*
← FD=01,00189600000B,FFC7<CR><LF> ← *indicate the 2nd found device's address and the name is not gotten.*
← FD=00,00189600000D,FFC8,BT_DEV_1<CR><LF> ← *indicate the 1st found device's address and name.*
→ AT+CO=00189600000A<CR><LF> ← *connect to the specified device 00:18:96:00:00:0A with the OPP profile.*
← OS=01,00189600000A<CR><LF> ← *the Bluetooth module is now connecting to the specified OPPS device which address is 00:18:96:00:00:0A.*
← CS=00,00189600000A<CR><LF> ← *connecting result: success.*
← OS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected to the specified OPPS device. The host can now start to push object to the remote OPPS device*
→42 43 3A 4F 41 3D 30 30 30 30 30 30 33 30 2C 31 30 2C 00 43 00 61 00 72 00 64 00 2E 00 76 00 63 00 66 0D 0A
← *(shown in HEX mode)*
the object size is 48 Bytes(30 30 30 30 30 30 33 30=00000030h).
the name length is 16 Bytes(31 30 = 10h).
*the file name is "**Card.vcf**" (00 43 = 'C', 00 61 = 'a', 00 72 = 'r', 00 64 = 'd', 00 2E = '.', 00 76 = 'v', 00 63 = 'c', 00 66 = 'f')*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+OY=0C,text/x-vcard<CR><LF> ← *the type length is 12(0Ch).*
*the object type is "**text/x-vcard**".*
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ 42 43 3A 4F 54 3D 30 31 2C 33 30 2C 42 45 47 49 4E 3A 56 43 41 52 44 0D 0A 56 45 52 53 49 4F 4E 3A 32 2E 31 0D 0A 4E 3A 4D 69 63 68 61 65 6C 0D 0A 45 4E 44 3A 56 43 41 52 44 0D 0A 0D 0A

← *(shown in HEX mode)*
this is the final/only packet of this object (30 31 = 01h).
the length of this packet is 48 Bytes (33 30 = 30h).
the raw data of the packet is displayed in red(42 45 47…… 52 44 0D 0A).
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
→ AT+DO=00189600000A<CR><LF> ← *disconnect with the specified device 00:18:96:00:00:0A with the OPP profile as we have already finished the object push.*
← OS=00<CR><LF> ← *the module is disconnected with the remote OPPS device.*

9.7. Use the Bluetooth Module as an OPPS Device

This example shows how to use the Bluetooth module as an OPPS device to receive object pushed by an remote OPPC device.
Ex. 9.7:
→ AT+PF=00,00,00,00,03<CR><LF> ← *configure the module profiles: both OPPC and OPPS is enabled.*
no SPP, HID, RFCOMM or DUN profile is supported.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← AP=00<CR><LF> ← *Indicate that the Bluetooth module has performed a reboot and is ready now.*
→ AT+FT=00,00,00,0A,03,0000<CR><LF> ← *configure the module features:*
disable the auto connection after power on;
disable the auto connect after paired;
disable auto reconnect after link lost;
set the interval of auto reconnect to 10s.
configure the discover mode as 03: auto discoverable when there is no connection, and the discoverable state will never timeout.
This command is only needed when the first time use this Bluetooth module.
← OK<CR><LF> ← *response from the module to indicate the command is adopted.*
← OS=01,00189600000A<CR><LF> ← *a remote OPPC device is initiating an OPP connection request, its address is 00:18:96:00:00:0A.*
← CS=00,00189600000A<CR><LF> ← *connecting result: success.*
← OS=02,00189600000A<CR><LF> ← *the Bluetooth module is now connected with the remote OPPC device.*
←4F 41 3D 30 30 30 30 30 30 33 30 2C 31 30 2C 00 43 00 61 00 72 00 64 00 2E 00 76 00 63 00 66 0D 0A
← *(shown in HEX mode)*
the remote OPPC device is started to push an object.
the object size is 48 Bytes(30 30 30 30 30 30 33 30=00000030h).
the name length is 16 Bytes(31 30 = 10h).
*the file name is "**Card.vcf**" (00 43 = 'C', 00 61 = 'a', 00 72 = 'r', 00 64 = 'd', 00 2E = '.', 00 76 = 'v', 00 63 = 'c', 00 66 = 'f').*
← OY=0C,text/x-vcard<CR><LF> ← *the object type length is 12(0Ch).*
*the object type is "**text/x-vcard**".*
← 4F 54 3D 30 31 2C 33 30 2C 42 45 47 49 4E 3A 56 43 41 52 44 0D 0A 56 45 52 53 49 4F 4E 3A 32 2E 31 0D 0A 4E 3A 4D 69 63 68 61 65 6C 0D 0A 45 4E 44 3A 56 43 41 52 44 0D 0A 0D 0A
← *(shown in HEX mode)*
this is the final/only packet of this object (30 31 = 01h).
the length of this packet is 48 Bytes (33 30 = 30h).
the raw data of the packet is displayed in red(42 45 47…… 52 44 0D 0A).
← OS=00<CR><LF> ← *the module is disconnected with the remote OPPC device.*