NovaComm
Control Interface
(NCCI)
User Guide

Ver.1.6

June 20, 2012

Release Record

Version	Release Date	Comments
0.1	Nov 03, 2011	Draft
1.0	Dec 19, 2011	1. Add **PF** command and indicator to configure or report the profiles. 2. Change the usage of **FT** command and indicator. 3. Update the relevant command and indicator.
1.1	Feb 14, 2012	1. Add MM, SP, NC command and indicator.
1.2	Feb 27, 2012	1. Add RFCOMM commands and indicators.
1.3	Apr 23, 2012	1. Add AD, CD, IO, MT, SN, UM, UI, PK command and indicators. 2. Add default mode configuration of UART. 3. Correct some errors.
1.4	May 07, 2012	1. Add PT, SO commands and indicators for iAP. 2. Add an example for Iap. 3. Correct some errors.
1.5	June 05, 2012	1. Add PA command and indicator. 2. Add IO command to restore to factory.
1.6	June 20, 2012	1. Add commands and indicators for A2DP and AVRCP

Table of Contents

Table of Tables

Table of Figures

1. Introduction

The NovaComm Control Interface (NCCI) is a set of ASCII commands and indicators with which the user can control the NovaComm'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 NCCI.

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 NCCI command is shown as below:

BC:CMD[=Para1][,Para2][,RawData][,…]<CR><LF>

The general syntax of NCCI indicator is shown as below:

IDC[=Para1][,Para2][,RawData][,…]<CR><LF>

Description of each field:

BC: 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 NCCI commands and indicators.

Ex. 2.1

➔ BC:FT=01,00,00,0A<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

➔ BC: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 NCCI 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 NCCI Command List

Command	Short Description	Comments
General Commands		
PF	Query or configure the profiles of the Bluetooth Module.	
AD	Query the Bluetooth address of the module	
CD	Query or configure the Class of Device of the module	
FT	Query or configure the features of the Bluetooth 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 Bluetooth Module.	
NM	Query or change the local friendly name of the Bluetooth Module.	

BR	Query or change the UART baud rate.	
UM	Query or change the UART mode	
UI	Query or change the UART indicator output mode	
IQ	Inquiry the Bluetooth device.	
MD	Make the Bluetooth Module discoverable.	
PA	Query or change the state of pairing 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 bypass mode.	
SPP Commands		
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 package 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 iAP)		
PT	Query or change the protocol name of iAP application	
CR	Connect to the remote RFCOMM device	
DR	Disconnect with the remote RFCOMM device.	
RS	Query the RFCOMM state.	
RD	Send data package to remote RFCOMM device.	
A2DP Commands		
CM	Connect to the remote A2DP source device.	
DM	Disconnect with the remote A2DP source device.	
MS	Query the A2DP state.	
AVRCP Commands		
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.	
VU	Volume Up.	
VD	Volume Down.	

4. Indicator List

All the available NCCI 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 NCCI Indicator List

Indicator	Short Description	Comments
General Indicators		
OK	Indicates a command was adopted by the Bluetooth Module.	
ER	Indicates there is an error detected in the command sent by the host.	
AP	Indicates the state of Bluetooth Module as an application.	
AD	Report the Bluetooth address of the Module	
CD	Report the Class of Device of the Bluetooth Module	
PF	Reports the configured profiles of the Bluetooth Module.	
FT	Reports the configured features of the Bluetooth Module.	
MM	Reports the configuration of Man-In-The-Middle protection.	
IO	Reports the configuration of IO capability of local device.	
MT	Reports the configuration of force to be master feature.	
SN	Reports the configuration of sniff feature.	
SP	Reports the deep sleep state.	
PN	Reports the fixed pin code of the Bluetooth Module.	
NM	Reports the local friendly name of the Bluetooth Module.	
BR	Reports the UART baud rate.	
UM	Reports the configuration of UART mode	
UI	Reports the configuration of UART indicator output	
MD	Report the discoverable state.	
PA	Report the the state of pairing mode	
NC	Indicates the six digit number of numeric comparison.	
PK	Indicates there is a Passkey request	
IR	Indicates the inquiry result.	
FD	Indicates the address and name of found device.	
SPP Indicators		
SS	Indicates the SPP state.	
CS	Indicates the result of connect attempt to a remote SPP device.	
DT	Indicates a data package received from remote SPP device.	
HID Indicators		
IS	Indicates the HID state.	
CI	Indicates the result of connect attempt to a remote HID host.	
KR	Indicates a keyboard report received from remote HID host.	
RFCOMM Indicators(for Apple iAP)		
PT	Reports the protocol name of iAP application	
SO	Indicates the state of iAP data session	
RS	Indicates the RFCOMM state.	
CR	Indicates the result of connect attempt to a remote RFCOMM device.	
RD	Indicates a data package received from remote RFCOMM device.	

A2DP Indicators		
MS	Indicates the A2DP state.	
CM	Indicates the result of connect attempt to a remote A2DP source device.	
AVRCP Indicators		
VS	Indicates the AVRCP state.	
CV	Indicates the result of connect attempt to a remote AVRCP target device.	

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:
BC:PF[=SppCnt][,HidCnt][,RfcCnt]<CR><LF>

5.1.1.3.　Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
SppCnt	The maximum SPP instance count. Value: 00h—07h Default: 06	O	
HidCnt	The maximum HID instance count. Value: 00h—01h Default: 01	O	
RfcCnt	The maximum Rfc instance count. Value: 00h—01h Default: 01	O	

Notes:

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

2. *The total instance of SPP and HID should no more than 7 according to Bluetooth Spec.*

5.1.1.4.　Examples:

Ex. 5.1. To query current profile configuration of Bluetooth Module:

➔ BC:PF<CR><LF>　　　　　　　*← query current profile configuration.*

← PF=05,01,01<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:

➔ BC:PF=04,00,00<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.*

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:
BC:AD<CR><LF>

5.1.2.3.　Parameter Description:

None.

5.1.2.4.　Examples:

Ex. 5.3. To query the Bluetooth address of local module:

➔ BC: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.*

5.1.3. CD—Query or configure the Class of Device

5.1.3.1.　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.

5.1.3.2.　Syntax:

Synopsis:
BC:CD[=Cod]<CR><LF>

5.1.3.3. 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:

1. *The default COD has been configured properly by the Bluetooth firmware stack, so it is not necessary for user to configure it in general.*
2. *Some Bluetooth device will filter the devices by COD when searching for new device.*

5.1.3.4. Examples:

Ex. 5.4. To query current COD configuration of Bluetooth Module:

➔ BC:CD<CR><LF>   ← *query current COD configuration.*

← CD=001F00<CR><LF>   ← *report current COD configuration: 001F00.*

Ex. 5.5. To configure the COD of Bluetooth Module:

➔ BC:CD=000540<CR><LF>   ← *configure the module COD: 000540.*

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

5.1.4. FT—Query or configure the features

5.1.4.1. 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.

5.1.4.2. Syntax:

Synopsis:
BC:FT[=ATPowerOn,ACPaired,ATLinkLost,Interval]<CR><LF>

5.1.4.3. 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.	O	

		01-FE: The attempt times of auto connect after power on.		
		FF: The auto connect attempt will be performed permanently.		
		Default: FF (Permanent)		
ACPaired	Auto connects after paired with a device.	O	
	Value: 00 or 01		
	00: Disabled		
	01: Enabled		
	Default: 00 (Disabled)		
ATLinkLost	The attempt times of reconnect after link lost.	O	
	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)		
Interval	The interval between each reconnect attempt after link lost. The unit is second.	O	
	Value: 00h—FFh		
	Default: 0A (10 seconds)		

Notes:

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

5.1.4.4. Examples:

Ex. 5.6. To query current feature configuration of Bluetooth Module:

➜ BC:FT<CR><LF>   ← *query current feature configuration.*

← FT=FF,00,FF,0A<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.

Ex. 5.7. To configure the features of Bluetooth Module:

➜ BC: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.

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

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

5.1.5.1. 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.

5.1.5.2. Syntax:

Synopsis:
BC:MM[=State]<CR><LF>

5.1.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of Man-In-The-Middle protection. Value: 00h or 01h 00: Deactivated 01: Activated Default: 00 (Deactivated)	O	

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.

5.1.5.4. Examples:

Ex. 5.8. To query current Man-In-The-Middle protection state of the Bluetooth Module:

➜ BC: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. 5.9. To active Man-In-The-Middle protection feature:

➜ BC:MM=01<CR><LF>   ← *active Man-In-The-Middle protection feature.*

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

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

5.1.6.1.   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.

5.1.6.2.   Syntax:

Synopsis:
BC:IO[=IoCapability]<CR><LF>

5.1.6.3.   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 Default: 03 (No IO)	O	

Notes:

1. *The configuration of IO capability will take effect only when Man-In-The-Middle protection feature is enabled.*

2. *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 output a six digits numbers, the user should compare and confirm if the two numbers are the same or not. In case of "**Keyboard Only**", the remote device will output a six digits number as passkey, the user should input the same number at the Bluetooth Module side by command BC:PK (refer to section 5.1.19).*

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

5.1.6.4.   Examples:

Ex. 5.10. To query current IO capability configuration of local device:

➜ BC: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. 5.11. To configure the IO capability of local device as "Keyboard Only":

→ BC: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.*

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

5.1.7.1.   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.

5.1.7.2.   Syntax:

Synopsis:
BC:MT[=State]<CR><LF>

5.1.7.3.   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. *NovaComm'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.*

5.1.7.4.   Examples:

Ex. 5.12. To query current state of force to be master feature:

→ BC: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. 5.13. To active the force to be master feature:

→ BC:MT=01<CR><LF>   ← *active the force to be master feature.*

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

5.1.8. SN—Query or configure the sniff mode

5.1.8.1.   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.

5.1.8.2.   Syntax:

Synopsis:
BC:SN[=State]<CR><LF>

5.1.8.3.   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	

5.1.8.4.   Examples:

Ex. 5.14. To query current sniff mode of the Bluetooth Module:

➔ BC:SN<CR><LF>   ← *query the current state of sniff mode.*

← SN=00<CR><LF>   ← *report the sniff mode is deactivated currently.*

Ex. 5.15. To active the sniff mode:

➔ BC:SN=01<CR><LF>   ← *active the sniff mode.*

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

5.1.9. SP—Query or change the deep sleep mode

5.1.9.1.   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.

5.1.9.2.   Syntax:

Synopsis:
BC:SP[=State]<CR><LF>

5.1.9.3.   Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of deep sleep. Value: 00h or 01h 00: Deep sleep disabled 01: Deep sleep enabled Default: 00 (disabled)	O	

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 BC:SP=00 command to wake up the module.*

5.1.9.4.   Examples:

Ex. 5.16. To query the current deep sleep state of the Bluetooth Module:

➔ BC:SP<CR><LF>   ← *query the current deep sleep state.*

⬅ SP=00<CR><LF>   ← *report the deep sleep is disabled currently.*

Ex. 5.17. To enable the deep sleep mode of the Bluetooth Module:

➔ BC:SP=01<CR><LF>   ← *enable the deep sleep mode.*

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

Ex. 5.18. To wake up the Bluetooth Module from deep sleep:

➔ BC: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.*

5.1.10.   PN—Query or change the fixed pin code

5.1.10.1. 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.

5.1.10.2. Syntax:

Synopsis:
BC:PN[=PinCode]<CR><LF>

5.1.10.3. 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.

5.1.10.4. Examples:

Ex. 5.19. To query current fixed pin code of Bluetooth Module:

➔ BC:PN<CR><LF>   ← *query current fixed pin code.*

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

Ex. 5.20. To change the fixed pin code of Bluetooth Module:

➔ BC:PN=abcdef<CR><LF>   ← *change the fixed pin code to "abcdef"*

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

5.1.11. NM—Query or change the local friendly name

5.1.11.1. 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.

5.1.11.2. Syntax:

Synopsis:
BC:NM[=Name]<CR><LF>

5.1.11.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The new local friendly name of the Bluetooth Module. Length: 1—30 characters	O	

	Default: Per software version.		

5.1.11.4. Examples:

Ex. 5.21. To query current local friendly name of Bluetooth Module:

➔ BC:NM<CR><LF>   ← *query current local friendly name.*

⬅ NM=NVC_BT_DEVICE<CR><LF>   ← *report current local friendly name, it's "NVC_BT_DEVICE".*

Ex. 5.22. To change the local friendly name of Bluetooth Module:

➔ BC:NM=NVC_SPP<CR><LF>   ← *change the local friendly name to "NVC_SPP"*

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

5.1.12. BR—Query or change the UART baud rate

5.1.12.1. 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.

5.1.12.2. Syntax:

Synopsis:
BC:BR[=BaudRate]<CR><LF>

5.1.12.3. 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	O	The default baud rate may not be 9600 per software version.

	0F: 256000		
	10: 460800		
	11: 921600		
	12: 1382400		
	13: 1843200		
	14: 2764800		
	15: 3686400		
	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 reset it except for UART port.*

5.1.12.4. Examples:

Ex. 5.23. To query the baud rate of Bluetooth Module:

➔ BC:BR<CR><LF>   ← *query the baud rate.*

← BR=06<CR><LF>   ← *report the baud rate, it's 9600.*

Ex. 5.24. To change the baud rate of Bluetooth Module:

➔ BC: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.

5.1.13. UM—Query or configure the UART mode

5.1.13.1. 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. If the user wants to configure the UART mode, all of the parameters should be given together.

5.1.13.2. Syntax:

Synopsis:
BC:UM[=StopBits,Parity]<CR><LF>

5.1.13.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
StopBits	The stop bits of UART mode Value: 00h or 01h 00: 1 stop bit	O	

	01: 2 stop bits		
	Default: 00 (1 stop bit)		
Parity	The parity of UART mode	O	
	Value: 00h – 02h		
	00: No parity		
	01: Odd parity		
	02: Even parity		
	Default: 00 (No parity)		

5.1.13.4. Examples:

Ex. 5.25. To query the UART mode of Bluetooth Module:

➔ BC:UM<CR><LF>   ← *query the UART mode.*

← BR=00,00<CR><LF>   ← *report the UART mode, it's 1 stop bit and no parity.*

Ex. 5.26. To change the UART mode of Bluetooth Module:

➔ BC: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.*

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

5.1.14.1. 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.

5.1.14.2. Syntax:

Synopsis:
BC:UI[=State]<CR><LF>

5.1.14.3. Parameter Description:

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

5.1.14.4. Examples:

Ex. 5.27. To query current UART indicator output mode of the Bluetooth Module:

➔ BC:UI<CR><LF>   ← *query current UART indicator output mode.*

⬅ UI=01<CR><LF>   ← *report the UART indicator output is enabled currently.*

Notes:

    2.   *If the UART indicator output is disabled currently, the report will not be output.*

Ex. 5.28. To disable the UART indicator output:

➔ BC:UI=00<CR><LF>   ← *disable the UART indicator output*

                                      ← *no response output because the UART indicator output has been disabled*

Ex. 5.29. To enable the UART indicator output:

➔ BC:UI=01<CR><LF>   ← *enable the UART indicator output*

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

5.1.15. IQ—Inquiry the Bluetooth device

5.1.15.1. Description:

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

5.1.15.2. Syntax:

Synopsis:
BC:IQ<CR><LF>

5.1.15.3. Parameter Description:

None.

5.1.15.4. Examples:

Ex. 5.30. To inquire the Bluetooth device:

➔ BC: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,BT_DEV_3<CR><LF> ← *indicate the 3rd found device's address and name.*

⬅ FD=01,00189600000B<CR><LF>   ← *indicate the 2nd found device's address and the name is not gotten.*

⬅ FD=00,00189600000D,BT_DEV_1<CR><LF> ← *indicate the 1st found device's address and name.*

5.1.16. MD—Make the Bluetooth Module discoverable

5.1.16.1. Description:

This command can query or change the Bluetooth Module's discoverable state. 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 state by the Indicator MD.

5.1.16.2. Syntax:

Synopsis:
BC:MD[=State]<CR><LF>

5.1.16.3. Parameter Description:

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

5.1.16.4. Examples:

Ex. 5.31. To query the current discoverable state of the Bluetooth Module:

➔ BC:MD<CR><LF>   ← *query the current discoverable state.*

⬅ MD=00<CR><LF>   ← *report the Bluetooth Module is not discoverable currently.*

Ex. 5.32. To make Bluetooth Module discoverable:

➔ BC:MD=01<CR><LF>   ← *make Bluetooth Module discoverable.*

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

5.1.17. PA—Query of change the state of pairing mode

5.1.17.1. Description:

This command can query or change the Bluetooth Module's pairing mode state. 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 state of pairing mode by the Indicator PA.

5.1.17.2. Syntax:

Synopsis:
BC:PA[=State]<CR><LF>

5.1.17.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The new state of pairing mode.	O	

Value: 00h or 01h		
00: Pairing/Bonding disabled		
01: Pairing/Bonding enabled		
Default: 01 (Enabled)		

5.1.17.4. Examples:

Ex. 5.33. To query the current pairing mode state of the Bluetooth Module:

➔ BC:PA<CR><LF>　　　　　　　　　　*← query the current pairing mode state.*

← PA=01<CR><LF>　　　　　　　　　　*← report the pairing is enabled currently.*

Ex. 5.34. To disable the pairing mode of the Bluetooth Module:

➔ BC:PA=00<CR><LF>　　　　　　　　*← disable the pairing mode.*

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

5.1.18.　NC—Confirm or deny the numeric comparison

5.1.18.1. Description:

This command will confirm or deny the numeric comparison as a response of indicator NC when pairing.

About the passkey entry, please refer to 5.1.5. and 5.1.6.

5.1.18.2. Syntax:

Synopsis:
BC:NC{=Confirmation}<CR><LF>

5.1.18.3. 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	

5.1.18.4. Examples:

Ex. 5.35. To deny the numeric comparison:

← NC=012ABC<CR><LF>　　　　　　　*← indicate the number of numeric comparison with 012ABCh.*

➔ BC:NC=00<CR><LF>　　　　　　　　*← deny the numeric comparison.*

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

Ex. 5.36. To confirm the numeric comparison:

← NC=012ABC<CR><LF>　　　　　　　*← indicate the number of numeric comparison with 012ABCh.*

➔ BC:NC=01<CR><LF>　　　　　　　　*← confirm the numeric comparison.*

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

5.1.19. PK—Input the passkey when pairing

5.1.19.1. Description:

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

About the passkey entry, please refer to 5.1.5. and 5.1.6.

5.1.19.2. Syntax:

Synopsis:
BC:PK{=Number}<CR><LF>

5.1.19.3. Parameter Description:

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

5.1.19.4. Examples:

Ex. 5.37. To input the passkey number when pairing:

← PK<CR><LF>   *← indicates there is a Bluetooth device is passkey request*

→ BC:PK=012ABC<CR><LF>   *← input the passkey number: 012ABCh*

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

5.1.20. CP—Clear the paired Bluetooth device list

5.1.20.1. 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.

5.1.20.2. Syntax:

Synopsis:
BC:CP<CR><LF>

5.1.20.3. Parameter Description:

None.

5.1.20.4. Examples:

Ex. 5.38. To clear the paired device list:

→ BC:CP<CR><LF>   *← clear the paired device list.*

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

5.1.21. CT—Connect to remote Bluetooth device

5.1.21.1. 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.

5.1.21.2. Syntax:

Synopsis:
BC:CT[=BdAddr]<CR><LF>

5.1.21.3. Parameter Description:

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

5.1.21.4. Examples:

Ex. 5.39. To connect to the last connected device with the last connected profile:

➔ BC: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. 5.40. To connect to the specified device with the last connected profile:

➔ BC: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.*

5.1.22. DC—Disconnect with remote Bluetooth device

5.1.22.1. 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.

5.1.22.2. Syntax:

Synopsis:
BC:DC[=BdAddr]<CR><LF>

5.1.22.3. Parameter Description:

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

5.1.22.4. Examples:

Ex. 5.41. To disconnect with all of the connected devices:

→ BC: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. 5.42. To disconnect to the specified device:

→ BC: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.*

5.1.23. BP—Configure the Bypass mode

5.1.23.1. Description:

This command can configure the 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 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, 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 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(keyboard reports) to be sent the HID host.	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 [BypassChannel]
	HID channel	Raw data(keyboard reports) to be sent the HID host.	ASCII indicators	

Notes:

1. When the Bluetooth Module is configured to one of the Bypass modes, it does not mean the Bluetooth Module will working 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 6.4.2 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>   BC:BP=00,00<CR><LF>   <1 second idle on UART>

5.1.23.2. Syntax:

Synopsis:

BC:BP{=BypassMode,BypassChannel}<CR><LF>

5.1.23.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
BypassMode	The new Bypass 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	
BypassChannel	The new Bypass channel: Value: 00h—0xh, 07h, 10h—1xh (x is the maximum SPP instance count, refer to 5.1.1) 00—0x: the SPP channel ID. Up to 7 SPP channels available. 07: The HID channel. 08: The RFCOMM channel(for iAP) 10h—1xh: the SPP Name ID.	M	The parameter will only take effect when the Bypass mode is **Bypass to Specified Channel**

5.1.23.4. Examples:

Ex. 5.43. To change the bypass mode to **Proxy Mode** when working in one of the Bypass mode:

           ← *Keep the UART port idle for 1 second.*

➔ BC: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.*

Ex. 5.44. To change the bypass mode to **Bypass to HID ASCII Channel**:

➔ BC:BP=02,00<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. 5.45. To change the bypass mode to **Bypass to Specified Channel**:

➔ BC:BP=04,03<CR><LF>   ← *change the bypass mode to* **Bypass to Specified Channel**, *the channel is specified to SPP channel 03.*

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

5.2. SPP Commands

5.2.1. CS—Connect to the remote SPP device

5.2.1.1. 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.

5.2.1.2. Syntax:

Synopsis:
BC:CS[=BdAddr][,NameId]<CR><LF>

5.2.1.3. 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: 10h—1xh (x is the maximum SPP instance count, refer to 5.1.1)	O	This is only available when both of the two side 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).*

5.2.1.4. Examples:

Ex. 5.46. To connect to the last connected SPP device:

→ BC: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. 5.47. To connect to the specified device with the SPP profile:

→ BC: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. 5.48. To connect to the specified SPP device with the Name ID 13:

→ BC:CS=00189600000A,13<CR><LF>   ← *connect to the specified device 00:18:96:00:00:0A with the Name ID 13.*

← SS=01,00189600000A,13<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 13.*

← CS=00,00189600000A<CR><LF>   ← *connecting result: success.*

← SS=02,00189600000A,13<CR><LF>   ← *the Bluetooth Module is now connected to the specified SPP device, the Name ID is 13.*

5.2.2. DS—Disconnect with the remote SPP device

5.2.2.1.   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.

5.2.2.2.   Syntax:

Synopsis:
BC:DS[=BdAddr]<CR><LF>

5.2.2.3.   Parameter Description:

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

5.2.2.4.   Examples:

Ex. 5.49. To disconnect with all of the connected SPP devices:

➔ BC: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. 5.50. To disconnect to the specified device:

➔ BC: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.*

5.2.3. SS—Query the state of each SPP channel

5.2.3.1.   Description:

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

5.2.3.2.   Syntax:

Synopsis:
BC:SS<CR><LF>

5.2.3.3.   Parameter Description:

None.

5.2.3.4.   Examples:

Ex. 5.51. To query the state of each SPP channel:

➜ BC: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,13<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 13.*

← 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*

5.2.4. DT—Send data package to remote SPP device

5.2.4.1.   Description:

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

5.2.4.2.   Syntax:

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

5.2.4.3.   Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ChannelOrNameId	The SPP channel ID or Name ID which will be used to send the data package. Value: 00h—0xh or 10h—1xh (x is the maximum SPP instance count, refer to 5.1.1) 00—0x: the channel ID of SPP 10—1x: 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.(1)	M	

Notes:

1. *Since the <CR><LF> sequence is treated as the terminator of a command line, so it's the responsibility of the host to avoid sending this special sequence in raw data of this command. A way to solve this issue is to add an user defined escape character (e.g. FFh) between <CR> and <LF>, and the received side should remove this escape character accordingly.*

5.2.4.4.　Examples:

Ex. 5.52. To send data use SPP channel 0:

➜ BC:DT=00,0A,1234567890<CR><LF>　← *send a data package use SPP channel 0, the data length is 10(Dec).*

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

Ex. 5.53. To send data use SPP Name ID 13h:

➜ BC:DT=13,0A,1234567890<CR><LF>　← *send a data package use SPP Name ID 13h, 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:
BC: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. 5.54. To connect to the last connected HID host:

➜ BC: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. 5.55. To connect to the specified HID host:

➜ BC: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 host.

5.3.2.2. Syntax:

Synopsis:
BC: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. 5.56. To disconnect with all of the connected HID host:

➔ BC: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. 5.57. To disconnect to the specified device:

➔ BC: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:
BC:IS<CR><LF>

5.3.3.3. Parameter Description:

None.

5.3.3.4. Examples:

Ex. 5.58. To query the state of HID channel:

➔ BC: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 keyboard report to remote HID host

5.3.4.1.　Description:

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

5.3.4.2.　Syntax:

Synopsis:
BC:KR{=KeyboardReport}<CR><LF>

5.3.4.3.　Parameter Description:

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

Notes:

　　1.　The standard keyboard report should in the following format. All of the fields in the report should be given in ASCII characters.

　　　A1,01,Modifier,00,KeyCode1,KeyCode2,KeyCode3,KeyCode4,KeyCode5,KeyCode6

5.3.4.4.　Examples:

Ex. 5.59. To send a keyboard report to HID host:

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

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

Ex. 5.60. To send a *keyboard* report to HID host:

➔ BC:KR=A1,01,00,00,00,00,00,00,00,00<CR><LF>　*← send a keyboard report to the HID host. No key is pressed*

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

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:
BC: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 0Dh can be included in this parameter. The data out of range will be thrown away.	M	

5.3.5.4.   Examples:

Ex. 5.61. To send an ASCII string to HID host:

➔ BC:AS=Hello Bluetooth HID<CR><LF>　　　← *send an ASCII string to the HIS host.*

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

Ex. 5.62. To send a ASCII string to HID host:

➔ BC:AS=!@#$%^abcdef9876<CR><LF>　　　← *send a ASCII string to the HIS host.*

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

5.4. RFCOMM Commands (for iAP)

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

5.4.1. PT—Query or change the protocol name of iAP application

5.4.1.1.   Description:

This command can query or change the protocol name of iAP 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.

5.4.1.2.   Syntax:

Synopsis:
BC:PT [=ProtocolName]<CR><LF>

5.4.1.3.   Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
ProtocolName	The new protocol name of iAP application.	O	

	Length: 1—30 characters		
	Default: Per software version.		

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.novacomm.protocol").*

5.4.1.4. Examples:

Ex. 5.63. To query current protocol name of iAP application:

➔ BC: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. 5.64. To change the protocol name of iAP application:

➔ BC:PT=com.novacomm.protocol <CR><LF>   ← *change the local friendly name to "com.novacomm.protocol".*

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

5.4.2. CR—Connect to the remote RFCOMM device

5.4.2.1. 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

5.4.2.2. Syntax:

Synopsis:
BC:CR[=BdAddr]<CR><LF>

5.4.2.3. Parameter Description:

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

5.4.2.4. Examples:

Ex. 5.65. To connect to the last connected RFCOMM device:

➔ BC: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. 5.66. To connect to the specified RFCOMM device:

➔ BC: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.*

5.4.3. DR—Disconnect with the remote RFCOMM device

5.4.3.1.   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.

5.4.3.2.   Syntax:

Synopsis:
BC:DR[=BdAddr]<CR><LF>

5.4.3.3.   Parameter Description:

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

5.4.3.4.   Examples:

Ex. 5.67. To disconnect with all of the connected RFCOMM device:

→ BC: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. 5.68. To disconnect to the specified RFCOMM device:

→ BC: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.*

5.4.4. RS—Query the state of RFCOMM channel

5.4.4.1.   Description:

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

5.4.4.2.   Syntax:

Synopsis:
BC:RS<CR><LF>

5.4.4.3.   Parameter Description:

None.

5.4.4.4. Examples:

Ex. 5.69. To query the state of RFCOMM channel:

➔ BC: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.

5.4.5. RD—Send data package to remote RFCOMM device

5.4.5.1. Description:

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

5.4.5.2. Syntax:

Synopsis:
BC:RD{= DataLen,Data}\<CR>\<LF>

5.4.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DataLen	The length in bytes of the data to be sent. Value: 00h-F9h	M	
Data	The raw data.(1)	M	

Notes:

1. *Since the \<CR>\<LF> sequence is treated as the terminator of a command line, so it's the responsibility of the host to avoid sending this special sequence in raw data of this command. A way to solve this issue is to add an user defined escape character (e.g. FFh) between \<CR> and \<LF>, and the received side should remove this escape character accordingly.*

5.4.5.4. Examples:

Ex. 5.70. To send data to remote RFCOMM device:

➔ BC:DT=0A,1234567890\<CR>\<LF>   *← send a data package, the data length is 10(Dec).*

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

5.5. A2DP Commands

5.5.1. CM—Connect to the remote A2DP source device

5.5.1.1. Description:

This command will make the Bluetooth Module to connect to the remote Bluetooth A2DP source 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 A2DP source device.

5.5.1.2. Syntax:

Synopsis:
BC:CM[=BdAddr]<CR><LF>

5.5.1.3. Parameter Description:

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

5.5.1.4. Examples:

Ex. 5.71. To connect to the last connected A2DP source device:

➔ BC: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. 5.72. To connect to the specified A2DP source device:

➔ BC:CI=00189600000A<CR><LF>   ← *connect to the specified A2DP source device: 00:18:96:00:00:0A.*

← IS=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.*

← CI=00,00189600000A<CR><LF>   ← *connecting result: success.*

← IS=02,00189600000A<CR><LF>   ← *the Bluetooth Module is now connected to the specified A2DP source device.*

5.5.2. DM—Disconnect with the remote A2DP source device

5.5.2.1. 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.

5.5.2.2. Syntax:

Synopsis:
BC:DM[=BdAddr]<CR><LF>

5.5.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments

BdAddr	The Bluetooth address of the Bluetooth A2DP source device to disconnect.	O	

5.5.2.4. Examples:

Ex. 5.73. To disconnect with all of the connected A2DP source devices:

➔ BC: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. 5.74. To disconnect to the specified A2DP source device:

➔ BC:DM=00189600000A<CR><LF>   ← *disconnect with the specified A2DP source device: 00:18:96:00:00:0A.*

← MS=00<CR><LF>   ← th*e A2DP channel 0 of Bluetooth Module is now disconnected and is connectable.*

5.5.3. MS—Query the state of A2DP

5.5.3.1. Description:

This command is used to query the state of A2DP.

5.5.3.2. Syntax:

Synopsis:
BC:MS<CR><LF>

5.5.3.3. Parameter Description:

None.

5.5.3.4. Examples:

Ex. 5.75. To query the state of A2DP:

➔ BC: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.*

5.6. AVRCP Commands

5.6.1. CV—Connect to the remote AVRCP target device

5.6.1.1. 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.

5.6.1.2.　Syntax:

Synopsis:
BC:CV[=BdAddr]<CR><LF>

5.6.1.3.　Parameter Description:

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

5.6.1.4.　Examples:

Ex. 5.76. To connect to the last connected AVRCP target device:

➔ BC: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. 5.77. To connect to the specified AVRCP target device:

➔ BC: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.*

5.6.2. DV—Disconnect with the remote AVRCP target device

5.6.2.1.　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.

5.6.2.2.　Syntax:

Synopsis:
BC:DV[=BdAddr]<CR><LF>

5.6.2.3.　Parameter Description:

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

5.6.2.4.　Examples:

Ex. 5.78. To disconnect with all of the connected AVRCP target devices:

➔ BC: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. 5.79. To disconnect to the specified AVRCP target device:

➔ BC: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.*

5.6.3. VS—Query the state of AVRCP

5.6.3.1.　Description:

This command is used to query the state of AVRCP.

5.6.3.2.　Syntax:

Synopsis:
BC:VS<CR><LF>

5.6.3.3.　Parameter Description:

None.

5.6.3.4.　Examples:

Ex. 5.80. To query the state of AVRCP:

➔ BC: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.*

5.6.4. PL—Play or Pause

5.6.4.1.　Description:

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

5.6.4.2.　Syntax:

Synopsis:
BC:PL<CR><LF>

5.6.4.3.　Parameter Description:

None.

5.6.4.4.   Examples:

Ex. 5.81. To send a *Play* or *Pause* command to the connected AVRCP target device:

➔ BC:PL<CR><LF>   ← *send a Play or Pause command.*

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

5.6.5. ST—Stop play

5.6.5.1.   Description:

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

5.6.5.2.   Syntax:

Synopsis:
BC:ST<CR><LF>

5.6.5.3.   Parameter Description:

None.

5.6.5.4.   Examples:

Ex. 5.82. To send a *Stop* command to the connected AVRCP target device:

➔ BC:ST<CR><LF>   ← *send a Stop command.*

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

5.6.6. NX—Next

5.6.6.1.   Description:

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

5.6.6.2.   Syntax:

Synopsis:
BC:NX<CR><LF>

5.6.6.3.   Parameter Description:

None.

5.6.6.4.   Examples:

Ex. 5.83. To send a *Next* command to the connected AVRCP target device:

➔ BC:NX<CR><LF>   ← *send a Next command.*

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

5.6.7. PR—Previous

5.6.7.1. Description:

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

5.6.7.2. Syntax:

Synopsis:
BC:PR<CR><LF>

5.6.7.3. Parameter Description:

None.

5.6.7.4. Examples:

Ex. 5.84. To send a *Previous* command to the connected AVRCP target device:

➔ BC:PR<CR><LF>   *← send a Previous command.*

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

5.6.8. VU—Increase the Volume

5.6.8.1. Description:

This command is used to increase the volume by 1 step.

5.6.8.2. Syntax:

Synopsis:
BC:PL<CR><LF>

5.6.8.3. Parameter Description:

None.

5.6.8.4. Examples:

Ex. 5.85. To increase the volume:

➔ BC:VU<CR><LF>   *← increase the volume.*

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

5.6.9. VD—Decrease the volume

5.6.9.1. Description:

This command is used to decrease the volume by 1 step.

5.6.9.2. Syntax:

Synopsis:
BC:VD<CR><LF>

5.6.9.3. Parameter Description:

None.

5.6.9.4. Examples:

Ex. 5.86. To decrease the volume:

➔ BC:VD<CR><LF>   ← *decrease the volume.*

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

6. Description of ASCII Indicators

6.1. General Indicators

6.1.1. OK—Command was adopted by the Bluetooth 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:

➔ BC: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.

➔ BC: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.

➔ BC: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.

➔ BC: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.

➔ BC:XX<CR><LF>

← ER=03<CR><LF>　　　　　← *response from the module to indicate the command cannot be recognized.*

6.1.3. AP—State of Bluetooth 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 00: The Bluetooth Module is idle and ready. 01: The Bluetooth Module is now inquiring for Bluetooth device.	M	

6.1.3.4. Examples:

Refer to Ex. 5.30

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:
PF{=SppCnt,HidCnt,RfcCnt}<CR><LF>

6.1.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
SppCnt	The maximum SPP instance count. Value: 00h—07h Default: 06	M	
HidCnt	The maximum HID instance count. Value: 00h—01h Default: 01	M	
RfcCnt	The maximum Rfc instance count. Value: 00h—01h Default: 01	M	

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. CD—Class of Device of Bluetooth Module

6.1.6.1.　Description:

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

6.1.6.2.　Syntax:

Synopsis:
CD{=Cod}<CR><LF>

6.1.6.3.　Parameter Description:

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

6.1.6.4.　Examples:

Refer to the examples in section 5.1.3.4.

6.1.7. FT—Feature configuration

6.1.7.1.　Description:

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

6.1.7.2.　Syntax:

Synopsis:
FT{=ATPowerOn,ACPaired,ATLinkLost,Interval}<CR><LF>

6.1.7.3.   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	

Notes:

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

6.1.7.4.   Examples:

Refer to the examples in section 5.1.4.4.

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

6.1.8.1.   Description:

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

6.1.8.2.   Syntax:

Synopsis:
MM{=State}<CR><LF>

6.1.8.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of Man-In-The-Middle protection feature. Value: 00h or 01h 00: Deactivated 01: Activated	M	

6.1.8.4. Examples:

Refer to the examples in section 5.1.5.4.

6.1.9. IO—IO capability configuration

6.1.9.1. Description:

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

6.1.9.2. Syntax:

Synopsis:
IO{=IoCapability}<CR><LF>

6.1.9.3. 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 Default: 03 (No IO)	M	

6.1.9.4. Examples:

Refer to the examples in section 5.1.6.4.

6.1.10. MT—Force to be master state

6.1.10.1. Description:

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

6.1.10.2. Syntax:

Synopsis:
MT{=State}<CR><LF>

6.1.10.3. 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	

6.1.10.4. Examples:

Refer to the examples in section 5.1.7.4.

6.1.11. SN—Sniff mode state

6.1.11.1. Description:

This indicator will report current sniff mode state of the Bluetooth Module.

6.1.11.2. Syntax:

Synopsis:
SN{=State}<CR><LF>

6.1.11.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The state of Man-In-The-Middle protection 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. SP—Deep sleep state

6.1.12.1. Description:

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

6.1.12.2. Syntax:

Synopsis:
SP{=State}<CR><LF>

6.1.12.3. Parameter Description:

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

6.1.12.4. Examples:

Refer to the examples in section 5.1.9.4.

6.1.13. PN—Fixed pin code

6.1.13.1. Description:

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

6.1.13.2. Syntax:

Synopsis:
PN{=PinCode}<CR><LF>

6.1.13.3. 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.

6.1.13.4. Examples:

Refer to the examples in section 5.1.10.4

6.1.14. NM—Local friendly name

6.1.14.1. Description:

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

6.1.14.2. Syntax:

Synopsis:

NM[=Name]<CR><LF>

6.1.14.3. 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	

6.1.14.4. Examples:

Refer to the examples in section 5.1.11.4.

6.1.15. BR—UART baud rate

6.1.15.1. Description:

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

6.1.15.2. Syntax:

Synopsis:
BR{=BaudRate}<CR><LF>

6.1.15.3. 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	M	The default baud rate may not be 9600 per software version.

11: 921600			
12: 1382400			
13: 1843200			
14: 2764800			
15: 3686400			
Default: 06 (9600)			

6.1.15.4. Examples:

Refer to the examples in section 5.1.12.4.

6.1.16. UM—UART mode

6.1.16.1. Description:

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

6.1.16.2. Syntax:

Synopsis:
UM{=StopBits,Parity}<CR><LF>

6.1.16.3. 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)	M	
Parity	The parity of UART mode Value: 00h – 02h 00: No parity 01: Odd parity 02: Even parity Default: 00 (No parity)	M	

6.1.16.4. Examples:

Refer to the examples in section 5.1.13.4

6.1.17. UI—UART indicator output configuration

6.1.17.1. Description:

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

6.1.17.2. Syntax:

Synopsis:
UI{=State}<CR><LF>

6.1.17.3. 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 Default: 01 (Enabled)	M	

6.1.17.4. Examples:

Refer to the examples in section 5.1.5.4.

6.1.18. MD—Discoverable state

6.1.18.1. Description:

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

6.1.18.2. Syntax:

Synopsis:
MD{=State}<CR><LF>

6.1.18.3. Parameter Description:

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

6.1.18.4. Examples:

Refer to the examples in section 5.1.16.4.

6.1.19. PA—Pairing mode state

6.1.19.1. Description:

This indicator will report current state pairing mode of the Bluetooth Module.

6.1.19.2. Syntax:

Synopsis:
PA{=State}<CR><LF>

6.1.19.3. 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 Default: 01 (Enabled)	M	

6.1.19.4. Examples:

Refer to the examples in section 5.1.17.4.

6.1.20. NC—Number of numeric comparison

6.1.20.1. Description:

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

6.1.20.2. Syntax:

Synopsis:
NC{=Number }<CR><LF>

6.1.20.3. Parameter Description:

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

6.1.20.4. Examples:

Refer to the examples in section 5.1.18.4.

6.1.21. PK—Passkey request

6.1.21.1. Description:

This indicator indicates there is a passkey request.

6.1.21.2. Syntax:

Synopsis:
PK<CR><LF>

6.1.21.3. Parameter Description:

None.

6.1.21.4. Examples:

Refer to the examples in section 5.1.19.4.

6.1.22. IR—Inquiry result

6.1.22.1. Description:

This indicator indicates the result of last inquiring.

6.1.22.2. Syntax:

Synopsis:
IR{=DevCount}<CR><LF>

6.1.22.3. Parameter Description:

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

6.1.22.4. Examples:

Refer to the examples in section 5.1.15.4.

6.1.23. FD—Information of found devices

6.1.23.1. Description:

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

6.1.23.2. Syntax:

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

6.1.23.3. 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	
Name	The friendly name of the found device in the last inquiring	O	

Notes:

1. *The information of found device will be given in a countdown order. i.e. the last found device will be given first. So that the host can know there is no subsequent FD indicator expected if a FD indicator comes with Idx parameter as 00.*

6.1.23.4. Examples:

Refer to the examples in section 5.1.15.4.

6.2. SPP Indicators

6.2.1. SS—State of each SPP channel

6.2.1.1. Description:

This indicator reports the state of each SPP channel.

6.2.1.2. Syntax:

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

6.2.1.3. 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.	M	

	2: The SPP channel is connected with a remote SPP device		
BdAddr	The Bluetooth address of remote SPP device.	O	
NameId	The Name ID of the connection. Refer to 5.2	O	

6.2.1.4.   Examples:

Refer to the examples in section 5.2.1.4 and 5.2.3.4.

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

6.2.2.1.   Description:

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

6.2.2.2.   Syntax:

Synopsis:
cS{=RetCode,BdAddr}<CR><LF>

6.2.2.3.   Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RetCode	The result code of the connect attempt. The high nibble indicates the channel ID(0h—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	

6.2.2.4.   Examples:

Refer to the examples in section 5.2.1.4.

6.2.3. DT—Data package received from remote SPP device

6.2.3.1. Description:

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

6.2.3.2. Syntax:

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

6.2.3.3. 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 10h—1xh (x is the maximum SPP instance count, refer to 5.1.1) 00—0x: the channel ID of SPP 10—1x: the name ID of SPP	M	
DataLen	The length in bytes of the data received. Value: 00h-FFh	M	
Data	The raw data.	M	

6.2.3.4. Examples:

Ex. 6.6. A data package is received from the SPP channel 0:

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

Ex. 6.7. A data package is received from the SPP Name ID 13h:

← DT=13,0A,1234567890<CR><LF>   ← *a data package received from SPP Name ID 13h, 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 iAP)

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

6.4.1. PT—Protocol name

6.4.1.1. Description:

This indicator will report current protocol name of iAP application.

6.4.1.2. Syntax:

Synopsis:
PT[=ProtocolName]<CR><LF>

6.4.1.3. Parameter Description:

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

6.4.1.4. Examples:

Refer to the examples in section 5.4.1.4.

6.4.2. SO—State of iAP data session

6.4.2.1. 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.

6.4.2.2. Syntax:

Synopsis:
RS{=SessionOpen}<CR><LF>

6.4.2.3. 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	

6.4.2.4. 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.*

6.4.3. RS—State of RFCOMM channel

6.4.3.1. Description:

This indicator reports the state of RFCOMM channel.

6.4.3.2. Syntax:

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

6.4.3.3. 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	

6.4.3.4. Examples:

Refer to the examples in section 5.4.2.4 and 5.4.4.4.

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

6.4.4.1. Description:

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

6.4.4.2. Syntax:

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

6.4.4.3. 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.	M	

	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.		
BdAddr	The Bluetooth address of remote RFCOMM device.	M	

6.4.4.4.　Examples:

Refer to the examples in section 5.4.2.4.

6.4.5. RD—Data package received from remote RFCOMM device

6.4.5.1.　Description:

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

6.4.5.2.　Syntax:

Synopsis:
RD{= DataLen,Data}<CR><LF>

6.4.5.3.　Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
DataLen	The length in bytes of the data received. Value: 00h-F9h	M	
Data	The raw data.	M	

6.4.5.4.　Examples:

Ex. 6.12. A data package is received from the RFCOMM device:

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

6.5. A2DP Indicators

6.5.1. MS—State of A2DP

6.5.1.1.　Description:

This indicator reports the state of A2DP.

6.5.1.2.　Syntax:

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

6.5.1.3. 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	

6.5.1.4. Examples:

Refer to the examples in section 5.5.1.4 and 5.5.3.4.

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

6.5.2.1. Description:

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

6.5.2.2. Syntax:

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

6.5.2.3. 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 A2DP source device.	M	

6.5.2.4. Examples:

Refer to the examples in section 5.5.1.4.

6.6. AVRCP Indicators

6.6.1. VS—State of AVRCP

6.6.1.1. Description:

This indicator reports the state of AVRCP.

6.6.1.2. Syntax:

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

6.6.1.3. 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 device.	M	
BdAddr	The Bluetooth address of remote AVRCP target 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. CV—Result of connect attempt to a remote AVRCP target device

6.6.2.1. Description:

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

6.6.2.2. Syntax:

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

6.6.2.3.   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 AVRCP target device.	M	

6.6.2.4.   Examples:

Refer to the examples in section 5.6.1.4.

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
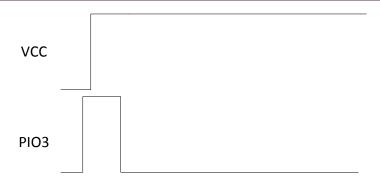
VCC

PIO3

Figure 7.1: Factory Restore Request

If the Bluetooth module detects logic 1 during power up, it will performer a factory restore.
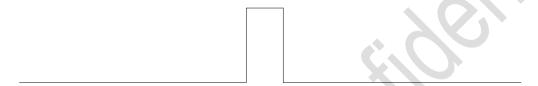
Ex. 7.2: Initiate a disconnect request

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.
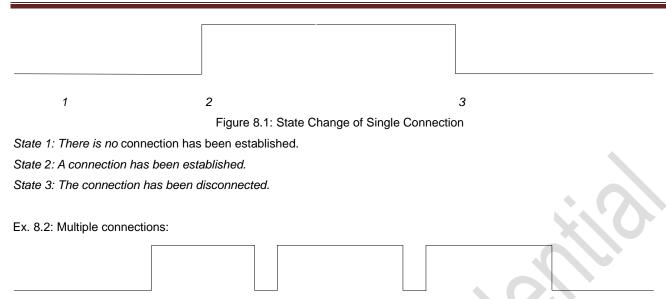
Ex. 8.2: Multiple connections:

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

Ex. 9.1:

➔ BC: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.*

➔ BC:FT=FF,00,FF,0A<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.
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.*

➔ BC: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,BT_SPP_DEV<CR><LF>   ← *indicate the 3rd found device's address and name.*

← FD=01,00189600000B<CR><LF>   ← *indicate the 2nd found device's address and the name is not gotten.*

← FD=00,00189600000D,BT_DEV_1<CR><LF>   ← *indicate the 1st found device's address and name.*

→ BC: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.*

→ BC: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.*

→ BC: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 Modules as a SPP Slave Device

Ex. 9.2:

→ BC: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.*

→ BC:FT=00,00,00,0A<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.
  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.*

→ BC: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 Modules as a HID Device to Send ASCII Characters to the HID Host

Ex. 9.3:

→ BC: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.*

→ BC:FT=FF,00,FF,0A<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.
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.*

→ BC:BP=02,00<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.*

→ BC: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.

……

……

                                        ← *Keep the UART port idle for 1 second.*

→ BC: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.*

→ BC: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 Modules as an Accessory of iOS Device (iAP)

Ex. 9.4:

➔ BC: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.*

➔ BC:FT=00,00,00,0A<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.
  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.*

➔ BC: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.*

➔ BC: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.*