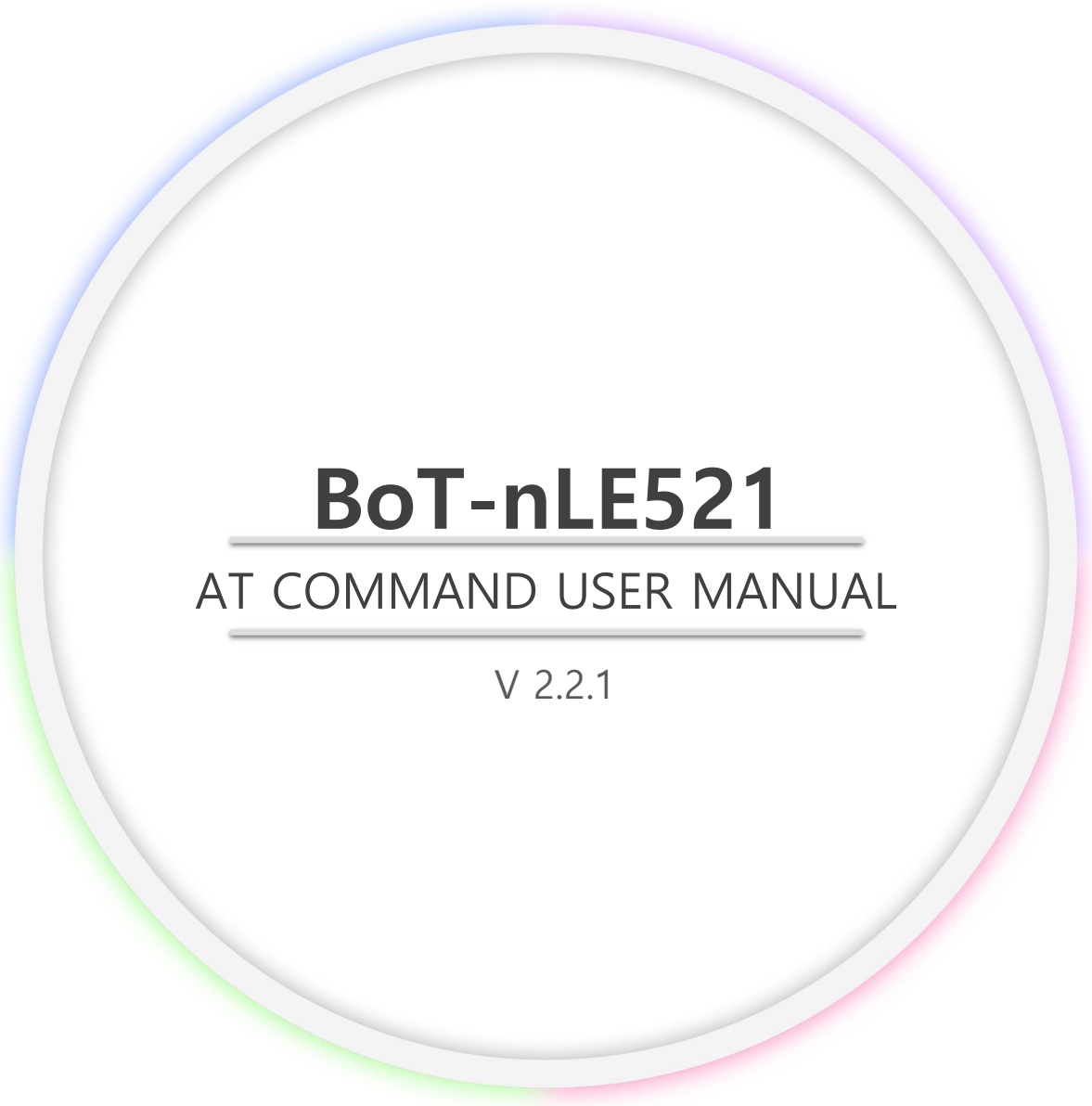




BoT-nLE521

AT COMMAND USER MANUAL

V 2.2.1

Table of contents

TABLE OF CONTENTS	오류! 책갈피가 정의되어 있지 않습니다.
1. DOCUMENT OVERVIEW	4
2. PRODUCT OVERVIEW	4
3. PROTOCOL BASIC RULE	4
3.1. PROTOCOL DEFAULT OPERATION.....	5
3.2. UART OPERATION MODE DESCRIPTION.....	6
3.2.1. <i>AT-COMMAND Mode</i>	6
3.2.2. <i>BYPASS Mode</i>	6
3.2.3. <i>PACKET Mode</i>	6
3.3. BOT STATE DESCRIPTION.....	7
3.3.1. <i>ADVERTISING</i>	7
3.3.2. <i>CONNECTED</i>	7
3.3.3. <i>STANDBY</i>	7
4. BOT CONTROL GPIO DESCRIPTION	8
4.1. BoT-nLE521 GPIO DEFINE.....	8
4.2. FUNCTION DESCRIPTION.....	8
4.2.1. <i>AT Command / BYPASS</i>	8
4.2.2. <i>Connection Status</i>	8
4.2.3. <i>Enter Sleep / Wakeup</i>	8
4.2.4. <i>Disconnect & Factory Reset</i>	8
4.2.5. <i>UART On/Off</i>	9
5. PROTOCOLS SUMMARY	10
6. ADVERTISING (DISCOVERABLE) DATA FORMAT	10
6.1. FORMAT.....	10
6.1.1. <i>Normal Advertising Mode</i>	10
6.1.2. <i>iBeacon Advertising Mode</i>	11
7. REQUEST PROTOCOL DETAIL	12
7.1. AT.....	12
7.2. ATZ.....	12
7.3. AT&F.....	12
7.4. AT+VER?.....	13

7.5.	AT+INFO?	14
7.6.	AT+UART=xxxx	14
7.7.	AT+FLOWCONTROL=xx.....	15
7.8.	AT+FLOWCONTROL?.....	15
7.9.	AT+MAJOR=xxxx.....	15
7.10.	AT+MAJOR?	16
7.11.	AT+MINOR=xxxx.....	16
7.12.	AT+MINOR?	17
7.13.	AT+IRSSI=xx	17
7.14.	AT+IRSSI?	18
7.15.	AT+ADVTYPE=x	18
7.16.	AT+MANUF=xxxx.....	19
7.17.	AT+MANUF?	19
7.18.	AT+TXPWR=x	19
7.19.	AT+TXPWR?.....	20
7.20.	AT+ADVDATA=xxxx	21
7.21.	AT+ADVDATA?	21
7.22.	AT+ADVINTERVAL=xxxx.....	21
7.23.	AT+ADVINTERVAL?	22
7.24.	AT+CONNINTERVAL=<MAX>	22
7.25.	AT+CONNINTERVAL=<MIN>,<MAX>.....	23
7.26.	AT+CONNINTERVAL?.....	24
7.27.	AT+DISCONNECT	24
7.28.	AT+NEGOCONN?	25
7.29.	AT+PACKETMODE=ON	25
7.30.	AT+PACKETMODE=OFF	26
7.31.	AT+PACKETUNIT=xxx	26
7.32.	AT+PACKETINFO?	27
7.33.	AT+INTPULLDOWN=ON.....	27
7.34.	AT+INTPULLDOWN=OFF	28
7.35.	AT+INTPULLDOWN?.....	28
7.36.	AT+ADVOFF	29
7.37.	AT+ADVON	29
7.38.	AT+CONNRSSI?.....	30
8.	GENERAL DATA TRANSMISSION	31
9.	SERVICE UUID INFORMATION.....	31

1. Document Overview

This document defines the communication protocol through serial port (UART) between the "CHIPSEN Bluetooth LE BoT-nLE521 Module " (hereinafter referred to as "BOT") and the customer's MCU (hereinafter referred to as "HOST") connected by UART interface.

2. Product Overview

- Operable Mode
 - **Normal Server:** Server mode can be searched, including some user settings before connecting to the Remote device, and can be connected through UUID with the service in use by BOT
 - **iBeacon Server:** Server mode can be searched for including the value of the previous iBeacon format, and can be connected through UUID with the service being used by BOT
- Communication Mode
 - **Normal Phy:** Wireless communication is possible with general devices that support Bluetooth Low Energy. (Long Range Phy is not supported)
- Certification
 - **KC with EMC**
 - **CE RED_DOC with ROHS**
 - **FCC**
 - **TELEC**
 - **Bluetooth SIG (QDID: 139090)**
 - **IC**

3. Protocol Basic Rule

- The transmission/ reception of data between HOST and BOT is based on the UART (serial port) interface.
- UART (serial port) Default Setting
 - Baud rate: 9600bps
 - Data bit: 8
 - Parity bit: none
 - Stop bit: 1.

- The above is the basic setting value, and if you want to change it, you can request a correction when writing the BOT firmware or use the appropriate AT command (AT+UART=xxxx).
- Communication direction
 - REQUEST (HOST→BOT): Transmit from HOST to BOT
 - NOTIFY (BOT → HOST): Message from BOT to HOST indicating the setup status of the BOT.
 - BOT.RESPONSE(BOT → HOST): Message from BOT to HOST, response to REQUEST
- Communication rule

All protocols consist of a combination of ascii values and announce the end of the command through a **CR (Carrage Return, 0x0D)**.

Ex) REQUEST – Current F/W Version: AT+VER?

Command	AT+VER?							
Command set	A	T	+	V	E	R	?	CR
Ascii set	0x41	0x54	0x2B	0x56	0x45	0x52	0x3F	0x0D

Ex) NOTIFY –When ready after power is applied: +READY.

Command	+READY						
Command set	+	R	E	A	D	Y	CR
Ascii set	0x2B	0x52	0x45	0x41	0x44	0x59	0x0D

Ex) RESPONSE – Operation failure for REQUEST: +ERROR

Command	+ERROR						
Command set	+	E	R	R	O	R	CR
Ascii set	0x2B	0x45	0x52	0x52	0x4F	0x52	0x0D

3.1. Protocol Default Operation

- BOT receives the REQUEST from HOST and transmits a corresponding RESPONSE. HOST can basically expect a RESPONSE of + OK or + ERROR or receive a specific RESPONSE corresponding to REQUEST.

3.2. UART Operation Mode Description

- BOT supports three modes For HOST's DATA delivered to UART: AT-COMMAND, BYPASS mode, PACKET mode.
- If BOT generates high speed/continuous data during operation, errors including data omission may occur due to Bluetooth radio quality, performance, and usage environment.

In this case, information on the use and test environment should be provided for technical support, and technical support should be requested to review the problem-solving method through optimization of the setting of the BOT. (Technical support inquiry: s1@chipsen.com)

3.2.1. AT-COMMAND Mode

- In HOST, you can change the device settings of BOT or command BOT to perform a specific operation, which is called AT-COMMAND mode.
- Default mode that is sent to the remote device operates in BYPASS mode unless set separately to the BOT.
- Most of this document describes the commands available in AT-COMMAND mode.

3.2.2. BYPASS Mode

- It is a default mode that is transmitting to the remote device and operates in BYPASS mode unless it is set separately in BOT.
- BOT delivers HOST's DATA as soon as possible when delivering the data generated by HOST to the remote device, and this is called the BYPASS mode.
- In the BYPASS mode, it is impossible to change the device setting of the BOT or perform a specific operation, and it is possible to read some setting values or terminate the connection through the AT command GPIO setting.
- HOST's DATA in BYPASS mode can be split transmission or delayed transmission depending on the Bluetooth radio quality of the use environment or the Baud rate at which data is transferred to UART.

3.2.3. PACKET Mode

- When BOT delivers HOST's DATA to the remote device, it designates data of a certain size and supports a mode that can transmit data, it called PACKET mode.
- In PACKET mode, it is impossible to change the device setting of BOT or perform a specific operation, and it is possible to read some setting values or

terminate the connection through the AT command GPIO setting.

- The transmission of DATA, which occurred in the HOST, and this range can be changed to 1~128 (bytes) range. Transmission is possible when the received data size is the same as the set transmission size.

* it should be possible to receive data of the size set by the remote device.

3.3. BOT state Description

- The BOT state is divided into **ADVERTISING**, **CONNECTED**, and **STANDBY** depending on the connection status of the remote device and the UART operation mode ("3.2. UART Operation Mode Description" check).

3.3.1. ADVERTISING

- Every case that are not connected to the Remote device and are scanned on the Remote device is Advertising state, and in this state UART operation mode always operates in AT-COMMAND mode.

3.3.2. CONNECTED

- It means that it has already been connecting to the Remote device.
- All data entered in HOST is sent to the Remote device.

3.3.3. STANDBY

- when the remote device is not connected and is in the Advertising Off state through AT command.
 - In the above situation, AT command setting is possible, and when the device is reset or advertising is restarted through AT command, it switches to the ADVERTISING state.
- ** When AT command is automatically reset in the Standby state, it automatically operates in the ADVERTISING state after reset

4. BOT Control GPIO Description

To make it easier for HOST to control the state or control of the BOT, assign five GPIOs separately.

4.1. BoT-nLE521 GPIO define.

Pin PAD#	Direction	Pin Name	Function
#24	Input	P0.18	AT Command / BYPASS
#25	Output	P0.20	Connection Status
#19	Input	P0.12	Enter Sleep / Wakeup
#18	Input	P0.10	Disconnect & Factory reset
#23	Input	P0.16	UART On/Off

4.2. Function Description

4.2.1. AT Command / BYPASS

It operates in ATCOMMAND mode when high (Rising Edge) is maintained in CONNECTED state and operates in BYPASS mode when low (Falling Edge) is maintained. However, only inquiry commands can be activated with ATCOMMAND in the CONNECTED state (ex- "AT+INFO?" and "AT+CONN?").

4.2.2. Connection Status

Maintain High when connected with Client Device.

Keep low when disconnected with client device (Remote device).

4.2.3. Enter Sleep / Wakeup

When a high level (Rising Edge) is detected, it enters a low-power mode, and when a low level (Falling Edge) is detected, it wakes up and the module automatically reboots.

**** During low power operation, leakage current is generated by pull-down resistance of BOT internal, and leakage current consumed can be changed by applying an appropriate pull-down resistance to the outside and AT+INTPULLDOWN command**

4.2.4. Disconnect & Factory Reset

When a high level (Rising Edge) is detected, if it is connected to the client device other side, terminates the connection.

If the high level (Rising Edge) is maintained for more than 4 seconds, return to the factory initialization state after a +OK response.

4.2.5. UART On/Off

When a high level (Rising Edge) is detected, the UART is turned off to reduce the standby current.

when a Falling Edge (Lowel) is detected, UART communicates normally.

** Even if UART is turned off at the high level, data from the Remote is automatically sent to HOST when data from the Remote is detected while it's connected to the Remote device, and if the Remote device does not make data more than 2 seconds, the UART automatically turns off again, reducing the standby power

** During low power operation, leakage current is generated by pull-down resistance of BOT internal, and leakage current consumed can be changed by applying an appropriate pull-down resistance to the outside and AT+INTPULLDOWN command

5. Protocols Summary

For commands stored in memory, the response defined in the protocol must be checked and the power must be reset.

If the power is reset before confirming the defined response, some or all setting values may be initialized, and the setting operation is not guaranteed after the command is automatically rebooted (reset).

<Default settings that can be changed through protocols >

1. UART: : Baud rate: 9600bps : Flow control: off	2. Manufacturer Name: CHIPSEN
3. Advertising interval: 1280	4. Advertising type: 0 (Normal)
5. MAJOR: 1111	6. MINOR: 2222
7. IRSSI: C3	8. role: Peripheral
9. Connection Interval : less than Firmware v0.8.0 - 20,40 : Firmware v0.8.0 or more - 15,15	10. Tx Power: 5 (-4dBm)
11. PACKET mode: OFF	

6. Advertising (Discoverable) Data format

Advertising Data: refers to data that is verified during a scan, even if the device operates to server mode and cannot be connected to the remote device.

6.1. Format

6.1.1. Normal Advertising Mode

Advertising data includes a changeable Data area that 22 bytes out of 31 bytes in total.

The data included in the advertising data consists of a Reserved Data (Length and Flag) area that cannot be changed by the user and a data area that can be changed by the user through UART. In the structure below, the yellow block is an area where the user can change to AT command.

■ Structure

The packet shape is as follows, and the index position of the packet may be variable according to the data length.

TX levels, device names, and user data settings respond to AT commands
 "AT+TXPWR", "AT+MANUF", and "AT+ADVDATA".

index	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Data	GAP Reserved			TX power Reserved		TX level User Data	Device name Reserved		Device Name User Data							

index	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Data	User data Reserved		Undefined User Data												

6.1.2. iBeacon Advertising Mode

It includes a 5-byte changeable data area out of 31 bytes of iBeacon data.

The data area that the user can change through UART is the area that can be changed to AT command only for Major/Minor and TX Power.

■ Structure

The packet type is as follows, and major/minor and TX power changes correspond to ATT commands "AT+MAJOR", "AT+MINOR", and "AT+IRSI", respectively.

index	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Data	GAP Reserved		iBeacon Reserved								UUID Reserved					

index	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Data	UUID Reserved									MAJOR		MINOR		TX power	x

7. REQUEST Protocol Detail

7.1. AT

Operation	HOST requests +OK to BOT
Response	+OK or +ERROR
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	Reply "+OK" when BOT successfully received a request and performed AT command.
Examples of Use	(HOST→BOT): AT (BOT→HOST): +OK

7.2. ATZ

Operation	HOST requests BOT to reset
Response	+OK or +ERROR
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	Reply "+OK" if BOT successfully received a request and performed AT command .and reset after 2sec. * AT Command mode is usable in connected state. * From 3ms after power application
Examples of Use	(HOST→BOT): ATZ (BOT→HOST): +OK ...after 2sec.. (BOT→HOST): +READY ➔ When you receive a +READY response, the reset is successful.

7.3. AT&F

Operation	When HOST Initialization changeable value of BOT
Response	+OK

Response Completed	Within 8ms of receiving the command (based on 9600 bps)																				
Description	When BOT have successfully received a request and performed the command, respond "+OK" and set the setting value. And reset after 2sec. * AT Command mode is usable in connected state. (Default setting) <table> <tr> <td>1. UART:</td><td>2. Manufacturer Name: CHIPSEN</td></tr> <tr> <td>: Baud rate: 9600bps</td><td></td></tr> <tr> <td>: Flow control: off</td><td></td></tr> <tr> <td>3. Advertising interval: 1280</td><td>4. Advertising type: 0(Normal)</td></tr> <tr> <td>5. MAJOR: 1111</td><td>6. MINOR: 2222</td></tr> <tr> <td>7. IRSSI: C3</td><td>8. role: Peripheral</td></tr> <tr> <td>9. Connection Interval</td><td>10. Tx Power: 5(-4dBm)</td></tr> <tr> <td>: firmware v0.8.0 or less - 20,40</td><td></td></tr> <tr> <td>: firmware v0.8.0 or more - 15,15</td><td></td></tr> <tr> <td>11. PACKET mode: OFF</td><td></td></tr> </table>	1. UART:	2. Manufacturer Name: CHIPSEN	: Baud rate: 9600bps		: Flow control: off		3. Advertising interval: 1280	4. Advertising type: 0 (Normal)	5. MAJOR: 1111	6. MINOR: 2222	7. IRSSI: C3	8. role: Peripheral	9. Connection Interval	10. Tx Power: 5 (-4dBm)	: firmware v0.8.0 or less - 20,40		: firmware v0.8.0 or more - 15,15		11. PACKET mode: OFF	
1. UART:	2. Manufacturer Name: CHIPSEN																				
: Baud rate: 9600bps																					
: Flow control: off																					
3. Advertising interval: 1280	4. Advertising type: 0 (Normal)																				
5. MAJOR: 1111	6. MINOR: 2222																				
7. IRSSI: C3	8. role: Peripheral																				
9. Connection Interval	10. Tx Power: 5 (-4dBm)																				
: firmware v0.8.0 or less - 20,40																					
: firmware v0.8.0 or more - 15,15																					
11. PACKET mode: OFF																					
Examples of Use	(HOST→BOT): AT&F (BOT→HOST): +OK ...after 2sec.. (BOT→HOST): +READY ➔ Reset after initializing the changeable value. When you receive a +READY response, the reset is successful.																				

7.4. AT+VER?

Operation	When you want to know the firmware version of BOT
Response	Version Information (v[major]. [minor]. [release])
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	firmware version is showed when command was performed by receiving request successfully. * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+VER? (BOT→HOST): v0.8.3. ➔ Reset after initializing the changeable value. When you receive a +READY response, the reset is successful.

7.5. AT+INFO?

Operation	When HOST want to know the address, Device Name, and Role status of the BOT
Response	BOT Address, Device Name, Mode [Role], State
Response Completed	Within 60ms of receiving the command (based on 9600 bps)
Description	Reply "+OK" if BOT successfully received a request and performed AT command and response Address as Device Name * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+INFO? (BOT→HOST): 5C:F2:86:A0:00:01,CHIPSEN,SERVER[PERIPHERAL],ADVERTISING → The HOST device can know the address of the BOT, the device name 'CHIPSEN', the Server Mode Peripheral Role, and the Advertising status information.

7.6. AT+UART=xxxx

Operation	When HOST change the baud-rate of the BOT
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	Reply "+OK" when BOT successfully received a request and performed AT command and Reset 2 seconds after changing the communication speed. If not, answer "+ERROR". Supported baud-rates are 2400, 4800, 9600, 14400, 28800, 19200, 38400, 57600, 115200, 230400, 460800.
Examples of Use	(HOST→BOT): AT+UART=9600 (BOT→HOST): +OK ... after 2sec.. (BOT→HOST): +READY

7.7. AT+FLOWCONTROL=xx

Operation	If HOST change the flow control status of the BOT
Response	+OK or +ERROR
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	Reply "+OK" when BOT successfully received a request and performed AT command and Reset 2 seconds after changing the flow control status. If not, answer "+ERROR". ** xx can be set to "ON" or "OFF". (Default setting) "OFF"
Examples of Use	(HOST→BOT): AT+FLOWCONTROL=ON (BOT→HOST): +OK ... after 2sec.. (BOT→HOST): +READY

7.8. AT+FLOWCONTROL?

Operation	When HOST wants to know whether flow control is used in communication with BOT
Response	Flow Control (ON or OFF)
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT successfully receives the request and perform the command, the BOT will determine whether to use flow control Reply in ON/OFF format. * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+FLOWCONTROL? (BOT→HOST): ON

7.9. AT+MAJOR=xxxx

Operation	When BOT is in iBeacon mode, HOST changes the Major setting value of the BOT
Response	+OK 또는 +ERROR

Response Completed	Within 8ms of receiving the command (Based on 9600 bps)
Description	Send the HEX value of the Major to be set to 4 characters. (ex) 0xFF50 => AT+MAJOR=FF50) When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR" **Hexadecimal range (0-9, A-F) must contain 4characters
Examples of Use	When BOT is in iBeacon mode (HOST→BOT): AT+MAJOR=FF50 (BOT→HOST): +OK (BOT→HOST): +ADVERTISING

7.10.AT+MAJOR?

Operation	When BOT is in iBeacon mode, use when HOST wants to know BOT's Major value
Response	Major response
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT received the request normally and performed the command, it responded with a Major value (HEX). If not, answer "+ERROR". * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+MAJOR? (BOT→HOST): F902

7.11.AT+MINOR=xxxx

Operation	When BOT is in iBeacon mode, HOST changes the Minor setting value of the BOT
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	Send the HEX value of the Minor to be set to 4 characters.

	(ex) 0xFF50 => AT+MINOR=FF50) When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR" **Hexadecimal range (0-9, A-F) must contain 4 characters
Examples of Use	When BOT is in iBeacon mode (HOST→BOT): AT+MINOR=FF50 (BOT→HOST): +OK (BOT→HOST): +ADVERTISING

7.12.AT+MINOR?

Operation	When BOT is in iBeacon mode, use when HOST wants to know BOT's Minor value
Response	Minor value
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT received the request normally and performed the command, it responded with a Minor value (HEX). If not, answer "+ERROR".. * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+MINOR? (BOT→HOST): 55FB

7.13.AT+IRSSI=xx

Operation	When BOT is in iBeacon mode, HOST changes the Rssi setting value of the BOT
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	Send the HEX value of RSSI be set to 2 characters. (ex) 0xF8 => AT+IRSSI=F8) When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR"

	**Hexadecimal range (0-9, A-F) must contain 4 characters
Examples of Use	When BOT is in iBeacon mode (HOST→BOT): AT+IRSSI=F8 (BOT→HOST): +OK (BOT→HOST): +ADVERTISING

7.14.AT+IRSSI?

Operation	When BOT is in iBeacon mode, use when HOST wants to know BOT's RSSI value
Response	Rossi value
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT received the request normally and performed the command, it responded with a RSSI (HEX). If not, answer "+ERROR".. * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+MINOR? (BOT→HOST): C5

7.15.AT+ADVTYPE=x

Operation	Change the BOT's mode type (Normal mode or iBeacon mode)
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR"
Examples of Use	(HOST→BOT): AT+ADVTYPE=N (BOT→HOST): +OK (BOT→HOST): +ADVERTISING ➔ BOT operates normal mode. (HOST→BOT): AT+ADVTYPE=I

	(BOT→HOST): +OK (BOT→HOST): + ADVERTISING → BOT operates iBeacon mode
--	---

7.16. AT+MANUF=xxxx

Operation	Change the BOT's Manufacture (Device Name)
Response	+OK 또는 +ERROR
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR", Max. 8characters
Examples of Use	(HOST→BOT): AT+MANUF=CHIPSEN (BOT→HOST): +OK (BOT→HOST): +ADVERTISING → Set BOT's manufacture (Device Name) with CHIPSEN. Information called CHIPSEN may be obtained from advertising data when searching from a client. When the command is performed during the advertisement, the name is applied, and the advertising starts again.

7.17. AT+MANUF?

Operation	when HOST wants to know BOT's Manufacturer (Device Name)
Response	Manufacturer (Device Name)
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered Manufacturer (Device Name) * * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+MANUF? (BOT→HOST): CHIPSEN

7.18. AT+TXPWR=x

Operation	Change the BOT's TX power
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR"
Examples of Use	(HOST→BOT): AT+TXPWR=xx (BOT→HOST): +OK (BOT→HOST): +ADVERTISING ** The range of xx can be changed from 0 to 7 0: -40dBm, 1 : -20dBm, 2 : -16dBm, 3 : -12dBm, 4 : -8dBm, 5 : -4dBm, 6 : 0dBm, 7 : 4dBm ➔ When the command is performed during the advertising, TX Power is applied, and the advertising starts again. ** If the firmware version is less than v1.0.0, it must be reset each time the power application is made, and if it is v1.0.0 or higher, the previous setting value is maintained even when the power application is made once.

7.19.AT+TXPWR?

Operation	when HOST wants to know BOT's TX Power
Response	TX Power (0~7)
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered TX Power Value (0: -40dBm, 1 : -20dBm, 2 : -16dBm, 3 : -12dBm, 4 : -8dBm, 5 : -4dBm, 6: 0dBm, 7 : 4 dBm) * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+TXPWR? (BOT→HOST): 5

7.20.AT+ADVDATA=xxxx

Operation	When BOT is in sever mode, HOST Change the BOT's Advertising User Data
Response	+OK or +ERROR
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR". * Up to 13 characters can be customized, with no setting as Default (NULL). No Memory Storage (Reset Initialize Value).
Examples of Use	(HOST→BOT): AT+ADVDATA=Beacon (BOT→HOST): +OK (BOT→HOST): +ADVERTISING When the command is performed during the advertising, User Data is applied, and the advertising starts again

7.21.AT+ADVDATA?

Operation	when HOST wants to know BOT's Advertising User Data
Response	Advertising User Data
Response Completed	Within 16ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered Advertising User Data * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+ADVDATA? (BOT→HOST): Beacon

7.22.AT+ADVINTERVAL=xxxx

Operation	Change the BOT's Advertising Interval
Response	+OK or +ERROR
Response	Within 8ms of receiving the command (based on 9600 bps)

Completed	
Description	When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR". Advertising Interval Range: 20~2560(ms) * Default Advertising Interval: 1280ms
Examples of Use	(HOST→BOT): AT+ADVINTERVAL=800 (BOT→HOST): +OK (BOT→HOST): +ADVERTISING *When the command is performed during the advertising, Advertising Interval is applied, and the advertising starts again.

7.23. AT+ADVINTERVAL?

Operation	when HOST wants to know BOT's Advertising Interval
Response	Advertising Interval (20~2560)
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered Advertising Interval Value * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+ADVINTERVAL? (BOT→HOST): 800

7.24. AT+CONNINTERVAL=<max>

Operation	Change the BOT's MAX. Connection Interval
Response	+OK 또는 +ERROR
Response Completed	Within 8ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR" Input values can be entered up to four characters (20 to 2000), and the Minimum connection interval is automatically set to half of the input value. At this time, depending on the counterpart device to be connected, the actual

	Connection Interval value that is negotiated may vary, and the Connection Parameter Update request may fail due to the rejection of the counterpart device. After setting the corresponding setting value, it is automatically reset, and advertising is restarted. **For firmware v0.7.2 or lower, it should be reset by manual after setting the value.
Examples of Use	(HOST→BOT): AT+CONNINTERVAL=30 (BOT→HOST): +OK ...after 2sec.. (BOT→HOST): +READY (BOT→HOST): +ADVERTISING

7.25. AT+CONNINTERVAL= <min>, <max>

Operation	Change the BOT's Connection Interval **Only works with firmware v0.8.0 or higher. In the case of the following version, only the max value is recognized, and the min value is automatically set.
Response	+OK or +ERROR
Response Completed	Within 20ms of receiving the command (based on 9600bps)
Description	When BOT successfully received the request and performed the command, BOT answered "+OK". If not, answer "+ERROR" Depending on the device being connected, the actual Connection Interval value that is negotiated may vary, and the Connection Parameter Update request may fail due to rejection from the device. <min> 8~2000 <max>8~2000 * If performed in the Advertising state, cancel the existing Advertising, apply the Connection Interval, and resume. * The applied connection interval must be supported by the counterpart device (smartphone) to be connected to operate normally. ** When operating with an iOS counterpart device, if changes are made except for the default values of BoT, min 15(ms) and max 15(ms), min and max should be twice as different, and the minimum value should be 15(ms).

	*After setting the corresponding setting value, it is automatically reset, and advertising is restarted. ** Firmware v0.7.2 and below shall be reset by manual after setting the value.
Examples of Use	(HOST→BOT): AT+CONNINTERVAL=30,50 (BOT→HOST): +OK ...after 2sec.. (BOT→HOST): +READY (BOT→HOST): +ADVERTISING When the command is performed during the advertising, Connection Interval is applied, and the advertising starts again.

7.26.AT+CONNINTERVAL?

Operation	when HOST wants to know BOT's Connection Interval
Response	Display Connection Interval
Response Completed	Within 5ms of receiving the command (based on 9600 bps)
Description	When BOT successfully received the request and performed the command, BOT answered Connection Interval Value * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+CONNINTERVAL? (BOT→HOST): 30

7.27.AT+DISCONNECT.

Operation	If you want to disconnect through ATCOMMAND while the BOT is connected to the Remote device
Response	+OK and .. +DISCONNECTED or +ERROR
Response Completed	Within 100ms of receiving the command (based on 9600bps)
Description	When BOT successfully received the request and performed the command, BOT answered "+OK" and "+DISCONNECTED" * AT Command mode is usable in Connect state
Examples	(HOST→BOT): AT+DISCONNECT

of Use	(BOT→HOST): +OK (BOT→HOST): +DISCONNECTED (BOT→HOST): +ADVERTISING
--------	--

7.28.AT+NEGOCONN?

Operation	If Host want to know the Connection Interval setting when BOT is negotiated with Remote Device
Response	Displays the current Negotiation value is shown to two places
Response Completed	Within 8ms of receiving the command
Description	When BOT successfully received the request and performed the command, BOT answered negotiated Connection Interval value. * AT Command mode is usable in Connect state
Examples of Use	(HOST→BOT): AT+NEGOCONN? (BOT→HOST): 18.75 **For firmware v0.7.2 and below, the response value multiplied by 1.25ms is the actual interval time.

7.29.AT+PACKETMODE=ON

Operation	When BOT connected with Remote device, used when you want to transfer data in a constant unit. Data transferred in PACKET mode
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command
Description	When BOT successfully received the request and performed the command, send data to the remote device when received at the data size set by the AT+PACKETUNIT command. In PACKET mode, the data transmission period must be maintained above 15 ms
Examples of Use	(HOST→BOT): AT+PACKETMODE=ON (BOT→HOST): +OK ➔ In PACKET mode, it must be received at the data size set by the

	AT+PACKETUNIT command. It is sent only if the sizes match exactly. Therefore, PACKET UNIT's drainage data must be configured and delivered. (TIP: If there is not enough data, it can be transmitted to match the size of PACKET UNIT including NULL data) **This command applies only to v0.8.0 and later versions
--	---

7.30. AT+PACKETMODE=OFF

Operation	When BOT connected with Remote device, use to transfer data in BYPASS mode
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command
Description	When BOT successfully received the request and performed the command, Data is then transferred to BYPASS mode. In BYPASS mode transmission, the size or delay of data may be different depending on the transmitted wireless environment.
Examples of Use	(HOST→BOT) : AT+PACKETMODE=OFF (BOT→HOST) : +OK ** This command applies only to v0.8.0 and later versions

7.31. AT+PACKETUNIT=xxx

Operation	When BOT connected with Remote device, Set the data size to be transmitted when operating in PACKET mode
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command
Description	When BOT successfully received the request and performed the command, In PACKET mode, it must be received at the data size set by the AT+PACKETUNIT command. *default size(bytes): 20 *range(bytes): 1 ~ 128
Examples	(HOST→BOT) : AT+ PACKETUNIT =64

of Use	(BOT→HOST) : +OK ➔ In PACKET mode, it must be received at the data size set by the AT+PACKETUNIT command. It is sent only if the sizes match exactly. Therefore, PACKET UNIT's drainage data must be configured and delivered. (TIP: If there is not enough data, it can be transmitted to match the size of PACKET UNIT including NULL data) **This command applies only to v0.8.0 and later versions
--------	---

7.32.AT+PACKETINFO?

Operation	when HOST wants to know BOT's PACKET mode setting value
Response	PACKETMODE state, PACKET size
Response Completed	Within 8ms of receiving the command
Description	When BOT successfully received the request and performed the command, BOT answered PACKETMODE state, PACKET size
Examples of Use	(HOST→BOT): AT+ PACKETINFO? (BOT→HOST): ON,64 **This command applies only to v0.8.0 and later versions

7.33.AT+INTPULLDOWN=ON

Operation	Set to enable for internal pull-down resistance inside the module in UART On/Off, Enter Sleep / Wakeup GPIO.
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command
Description	When BOT successfully received the request and performed the command, Use the internal pull-down resistor. In addition, when UART On/Off and Enter Sleep/Wakeup input signals are generated, leakage current is generated according to internal resistance (300 to 400 uA based on 3.3 V)
Examples of Use	(HOST→BOT): AT+ INTPULLDOWN =ON (BOT→HOST): +OK

	...after 2sec.. (BOT→HOST): +READY ➔ Default ON, reboot 2 seconds after receiving the command and the setting takes effect. **This command applies only to v0.8.0 and later versions
--	--

7.34. AT+INTPULLDOWN=OFF

Operation	Set to disable for internal pull-down resistance inside the module in UART On/Off, Enter Sleep / Wakeup GPIO.
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command
Description	When BOT successfully received the request and performed the command, don't use the internal pull-down resistor. UART On/Off 및 Enter Sleep / Wakeup When an input signal is generated, a leakage current is generated according to an external pull-down resistance value. * If the external pull-down resistance is not applied, the BOT may not operate normally, so in this case, the pull-down resistance must be applied to the outside.
Examples of Use	(HOST→BOT): AT+ INTPULLDOWN =OFF (BOT→HOST): +OK ...after 2sec.. (BOT→HOST): +READY **This command applies only to v0.8.0 and later versions

7.35. AT+INTPULLDOWN?

Operation	When HOST wants to Know internal pull-down resistance On/Off inside the module in UART On/Off, Enter Sleep / Wakeup GPIO.
Response	+OK or +ERROR
Response	Within 8ms of receiving the command

Completed	
Description	When BOT successfully received the request and performed the command, BOT answered Internal pull-down resistance On/Off
Examples of Use	(HOST→BOT): AT+ INTPULLDOWN? (BOT→HOST): ON **This command applies only to v0.8.0 and later versions

7.36.AT+ADVOFF

Operation	When BOT is not connected with Remote device, stop to Advertising
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command
Description	When BOT is not connected with Remote device, stop to Advertising
Examples of Use	(HOST→BOT): AT+ ADVOFF (BOT→HOST): +STANDBY ➔ The condition for re-advertising is to perform BOT reset or "AT+ADVON" command of the BOT **This command applies only to v0.8.0 and later versions

7.37.AT+ADVON

Operation	When BOT not connected with Remote device, When BOT doesn't advertise, start advertising again.
Response	+OK or +ERROR
Response Completed	Within 8ms of receiving the command
Description	When BOT not connected with Remote device, When BOT doesn't advertise, start advertising again.
Examples of Use	(HOST→BOT): AT+ ADVON (BOT→HOST): +ADVERTISING ➔ resume advertising.

	**This command applies only to v0.8.0 and later versions
--	--

7.38.AT+CONNRSSI?

Operation	When BOT is connected with Remote device, to check the RSSI of the BOT
Response	+OK or +ERROR
Response Completed	Within 100ms of receiving the command
Description	When BOT is connected with Remote device, upon receiving the corresponding command, it is possible to check the current communication channel and the sensitivity of the wireless reception signal. The reception sensitivity responds with a negative number, and the larger the value, the greater the signal strength. ** Because it is connected to the Remote device, you must set AT command/Bypass GPIO to High and change it to AT command mode.
Examples of Use	(HOST→BOT): AT+ CONNRSSI? (BOT→HOST): CH[13],RSSI[-55] ➔ Currently communicating on channel 13 and -55 dBm reception sensitivity is confirmed. The channel and reception sensitivity can be changed each time an AT command is receive. **This command applies only to v1.0.0 and later versions

8. General Data transmission

After SERVER and CLIENT BLE device are connected, data that received UART or data transmitted through BLE link from the other device are automatically transmitted without any form conversion.

States	SERVER and CLIENT Connected
Examples of Use	(HOST→BOT): ABCD<CR> (Remote device): ABCD<CR>

9. Service UUID Information

For connection with the remote device, a custom data transmission/reception service is provided through the service UUID. Therefore, if you connect BOT to a smartphone or other device, use it according to the table (UUID) below.

Service	UUID	Property
Custom Data service (Primary)	0xFFFF0	N/A
Data Out service	0xFFFF1	Notification
Data In service	0xFFFF2	Write / Write without Response **firmware v0.7.2 and earlier only support Write properties

*16bit UUID 0xFFFF0 to 128-bit UUID: 0000FFFF0-0000-1000-8000-00805F9B34FB

*16bit UUID 0xFFFF1 to 128-bit UUID: 0000FFFF1-0000-1000-8000-00805F9B34FB

*16bit UUID 0xFFFF2 to 128-bit UUID: 0000FFFF2-0000-1000-8000-00805F9B34FB

** Property Description

Notification: Used when data is sent from BoT-nLE521 to remote device

Write: It is used when data is transmitted from the remote device to BoT-nLE521, and after the data is transmitted, the completion of the transmission is confirmed through the response of BoT-nLE521.

Write without Response: It is used when data is transmitted from the remote device to BoT-nLE521, No separate response from BoT-nLE521 after data transfer.

Appendix 1. Limitations of Liability and Legal Notice

★ This product may cause radio confusion during use and may receive harmful interference from other devices. Therefore, there may be data transfer delays or losses, and users are advised to take this into consideration and use it after sufficient testing. Due to the nature of the radio, there is no guarantee of accuracy, reliability or completeness, nor is there any responsibility. In no event shall Chipsen Co., Ltd. or Seller's liability limit exceed the paid selling price of the product.

★ This product is not a "customized" product, but operates on the form, behavior, and software specified by Chipsen Co., Ltd. In other words, be aware that the product is not designed for your specific environment and use it. We recommend that you refer to the functions and commands provided in the manual. It is recommended that you use it after sufficient testing if you use it in a specific environment, and the use and application of this product is entirely with the user (customer) and Chipsen Co., Ltd. has no warranty and no liability.

★ The technical information provided in this document describes the representative characteristics of the product and examples of the application circuit. This does not mean allowing industrial property rights, intellectual property rights, or other rights.

★ This product and related everything are subject to change without prior notice to improve performance.

★ This publication, including photographs, drawings and software, is protected by international copyright law and has all rights. Nothing contained in this manual, as well as this manual, may be reproduced, modified or stolen without the author's written consent.

★ Chipsen and Chipsen logo is a trademark of Chipsen Co., Ltd. All other brands and product names mentioned in this User's Guide are trademarks of their respective owners.