

BOT-nLE52x Protocol

V1.0.3

CHIPSEN BOT-nLE52x Protocol V1.0.3

■ History

버전	배포일자	내역	작성자
1.0.3	2010. 01. 13	- 속성 및 오탈자 수정	Ethan
1.0.2	2018. 07. 25	- 오탈자 수정	Ethan
1.0.1	2018. 06. 21	- 오탈자 수정	Ethan
1.0.0	2018. 06. 20	- BOT-nLE522 Protocol 기반 - BOT-nLE52x 공통 Protocol로 수정 및 적용	Ethan

목차

목차.....	3
1. 개요.....	5
2. PROTOCOL BASIC RULE.....	5
2.1. PROTOCOL 기본 동작.....	6
2.2. UART 동작 모드 설명.....	6
2.2.1. AT-COMMAND 모드.....	6
2.2.2. BYPASS 모드.....	6
2.3. BOT STATE 설명.....	7
2.3.1. ADVERTISING.....	7
2.3.2. CONNECTED.....	7
3. BOT 제어 GPIO 설명.....	8
3.1. BOT-nLE521 GPIO DEFINE.....	8
3.2. BOT-nLE522 GPIO DEFINE.....	8
3.3. FUNCTION 설명.....	8
3.3.1. AT Command / BYPASS.....	8
3.3.2. Connection Status.....	8
3.3.3. Enter Sleep / Wakeup.....	9
3.3.4. Disconnect & Factory Reset.....	9
4. PROTOCOLS SUMMARY.....	10
4.1. REQUEST (HOST→BOT) PROTOCOL SUMMARY.....	10
4.2. NOTIFY (BOT→HOST) PROTOCOL SUMMARY.....	11
4.3. GENERAL RESPONSE (BOT→HOST) PROTOCOL SUMMARY.....	11
5. ADVERTISING (DISCOVERABLE) DATA FORMAT.....	12
5.1. FORMAT.....	12
5.1.1. Normal Advertising Mode.....	12

CHIPSEN BOT-nLE52x Protocol V1.0.3

5.1.2.	<i>iBeacon Advertising Mode</i>	12
6.	REQUEST PROTOCOL DETAIL	13
6.1.	AT.....	13
6.2.	ATZ.....	13
6.3.	AT&F.....	14
6.4.	AT+VER?.....	14
6.5.	AT+INFO?.....	14
6.6.	AT+UART=xxxx.....	15
6.7.	AT+FLOWCONTROL=xx.....	15
6.8.	AT+FLOWCONTROL?.....	16
6.9.	AT+MAJOR=xxxx.....	16
6.10.	AT+MAJOR?.....	16
6.11.	AT+MINOR=xxxx.....	17
6.12.	AT+MINOR?.....	17
6.13.	AT+IRSSI=xx.....	17
6.14.	AT+IRSSI?.....	18
6.15.	AT+ADVTYPE=x.....	18
6.16.	AT+SERVER=ROLE.....	19
6.17.	AT+MANUF=xxxx.....	19
6.18.	AT+TXPWR=xx.....	20
6.19.	AT+TXPWR?.....	20
6.20.	AT+ADVDATA=xxxx.....	21
6.21.	AT+ADVDATA?.....	21
6.22.	AT+ADVINTERVAL=xxxx.....	22
6.23.	AT+ADVINTERVAL?.....	22
6.24.	AT+CONNINTERVAL=xxxx.....	22
6.25.	AT+CONNINTERVAL?.....	23
6.26.	AT+DISCONNECT.....	23
6.27.	AT+NEGOCONN?.....	24
7.	GENERAL DATA TRANSMISSION	25
8.	SERVICE UUID INFORMATION	25

1. 개요

본 문서는 “CHIPSEN Bluetooth LE BOT-nLE52x 모듈”과 (이하 “BOT”라 한다.)과 UART interface로 연결된 고객사의 MCU(이하 “HOST”라 한다.) 사이의 UART(serial port)를 통한 통신 규약을 정의한다.

2. Protocol Basic Rule

- HOST와 BOT 상호간 데이터 송/수신은 UART(serial port) 인터페이스를 기반으로 한다.
- UART(serial port) 기본 세팅값
 - Baud rate : 9600bps
 - Data bit : 8
 - Parity bit : none
 - Stop bit : 1
 - 위 사항은 기본적 세팅값이며, 변경을 원할 경우 BOT 펌웨어 작성시 수정 요청을 하거나 해당되는 AT command([AT+UART=xxxx](#))를 이용하여 수정한다.
- Communication direction
 - REQUEST (HOST→BOT): HOST에서 발생하여 BOT로 전달된다.
 - NOTIFY(BOT → HOST) : BOT 에서 발생하여 HOST으로 전달되는 메시지로, BOT의 기본적인 상태를 알려준다.
 - RESPONSE(BOT → HOST) : BOT에서 발생하여 HOST로 전달되는 메시지로, REQUEST에 대한 응답이다.
- Communication rule

모든 프로토콜은 ascii 값의 조합으로 구성되며 **CR(Carriage Return, 0x0D)** 을 통해 Command의 끝을 알린다.

Ex) REQUEST – 현재 장치 버전: 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 – 전원 인가 된 후 준비 완료 된 경우: +READY

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

Ex) RESPONSE – REQUEST에 대한 동작 실패 : +ERROR

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

2.1. Protocol 기본 동작

- BOT는 HOST로부터 REQUEST를 수신 후 대응되는 RESPONSE를 전송 한다. HOST는 기본적으로 +OK 또는 +ERROR의 RESPONSE를 기대할 수도 있고, REQUEST에 해당하는 특정 RESPONSE를 수신할 수 있다.

2.2. UART 동작 모드 설명

- BOT는 UART로 전달되는 HOST의 DATA에 대해 상대 장치와 연결 되지 않았을 경우의 AT-COMMAND와 상대 장치와 연결 되었을 경우인 BYPASS 모드의 두 가지 모드를 지원한다.

2.2.1. AT-COMMAND 모드

- HOST에서 BOT의 장치 설정을 변경 하거나, BOT가 특정 동작을 하도록 명령을 할 수 있는데, 이 상태를 AT-COMMAND 모드라고 한다.
- BOT는 Remote device와 연결이 이루어 지지 않을 경우 항상 AT-COMMAND 모드를 유지한다.
- 이 문서의 대부분은 AT-COMMAND 모드에서 사용 가능한 명령어를 설명하고 있다.

2.2.2. BYPASS 모드

- BOT가 Remote device와 연결되어 HOST에서 발생한 DATA를 Remote device로 전달이 가능한데, 이 상태를 BYPASS 모드라고 한다.
- BYPASS 모드 상태에서는 BOT의 장치 설정을 바꾸거나, 특정 동작을 수행하

는 것은 불가능하고, 일부 설정값을 읽거나 연결을 종료하는 동작은 가능하다.

2.3. BOT state 설명

- BOT state는 Remote device와 연결 상태 및 UART 동작 모드("1.3. UART 동작 모드" 참조)에 따라 **ADVERTISING, CONNECTED**로 나뉜다.

2.3.1. ADVERTISING

- Remote Client device와 연결되지 않고 Remote device에서 검색이 가능한 모 든 경우, Advertising 에 해당한다.
- Advertising 상태일 경우 UART 동작 모드는 언제나 AT-COMMAND 모드로 동작한다.

2.3.2. CONNECTED

- Remote device와 이미 연결이 이루어져 있는 경우에 해당한다.
- 이때 HOST에서 발생한 데이터는 모두 Remote device로 전송 된다.

3. BOT 제어 GPIO 설명

HOST에서 BOT의 상태 또는 컨트롤을 쉽게 하게 하기 위해 4개의 GPIO를 별도로 할당한다

3.1. 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

3.2. BOT-nLE522 GPIO define

Pin PAD#	Direction	Pin Name	Function
#17	Input	P0.02	AT Command / BYPASS
#14	Output	P0.03	Connection Status
#12	Input	P0.04	Enter Sleep / Wakeup
#8	Input	P0.12	Disconnect & Factory reset

3.3. Function 설명

3.3.1. AT Command / BYPASS

CONNECTED 상태에서 High(Rising Edge) 유지시 AT COMMAND 모드로 동작하며, Low(Falling Edge) 유지시 BYPASS 모드(Default mode)로 동작한다. 단, CONNECTED 상태에서 AT COMMAND로 조회 명령어들만 수행 가능하다(ex- "AT+INFO?", "AT+CONN?").

3.3.2. Connection Status

상대 Client 장치와 Connected 상태일 때 High를 유지한다.

상대 Client 장치와 Disconnected 상태일 때 Low를 유지한다.

3.3.3. Enter Sleep / Wakeup

High Level(Rising Edge) 이 감지되면 저전력 모드로 진입이 되며, 저전력 모드상태에서는 Low Level (Falling Edge) 이 감지되면 Wake Up 되고 모듈이 자동으로 재부팅한다.

3.3.4. Disconnect & Factory Reset

High Level(Rising Edge)이 감지되었을 때 상대 client 장치와 연결되어 있다면 연결을 종료한다.

High Level(Rising Edge)를 4초이상 유지 시 +OK 응답 후 공장초기화 상태로 복귀시킨다.

4. Protocols Summary

4.1. REQUEST (HOST→BOT) Protocol Summary

프로토콜	설명	비고
AT	+OK 요청	
ATZ	모듈 리셋 요청	
AT&F	현재 모듈의 변경가능한 값들을 공장초기화 후 리셋	
AT+VER?	현재 모듈에 적용된 펌웨어의 버전을 요청	
AT+INFO?	현재 모듈의 블루투스 주소, Device Name, Role정보 및 State 상태를 요청 ex) 5C:FE:86:00:00:01,CHIPSEN,SERVER[PERIPHERAL],ADVERTISING	
AT+UART=xxxx	UART Baudrate 을 변경할 경우 사용 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600, 115200, 230400, 460800	
AT+FLOWCONTROL=xx	UART Flow Control 활성화 여부를 변경할 경우 사용	
AT+FLOWCONTROL?	UART Flow Control 사용여부 요청	
AT+MAJOR=xxxx	BOT가 iBeacon mode일 때 HOST에서 BOT의 Major 설정 값을 변경할 경우	
AT+MAJOR?	HOST에서 BOT가 iBeacon으로 동작 시 설정된 Major 값을 알고자 할 경우 사용	
AT+MINOR=xxxx	BOT가 iBeacon mode일 때 HOST에서 BOT의 Minor 설정 값을 변경할 경우	
AT+MINOR?	HOST에서 BOT가 iBeacon으로 동작 시 설정된 Minor 값을 알고자 할 경우 사용	
AT+IRSSI=xx	BOT가 iBeacon mode일 때 HOST에서 BOT의 Rssi 설정 값을 변경할 경우	
AT+ADVTYPE=x	BOT의 Mode Type을 변경할 경우 사용 (Normal mode 또는 iBeacon mode)	
AT+SERVER=role	Server mode로 장치가 동작할 경우 role에 따라 동작모드가 결정된다. : role=B – Broadcaster (비연결성) : role=P – Peripheral (연결성)	
AT+MANUF=xxxxxxxx	Manufacturer(Device Name)를 변경할 경우 사용. 만약 Advertising 상태에서 이 명령을 사용하여 Manufacturer를 변경할 경우 Advertising Data에 즉시 반영이 된다. (MAX 8byte)	
AT+MANUF?	Manufacturer(Device Name)을 요청	

CHIPSEN BOT-nLE52x Protocol V1.0.3

AT+TXPWR=x	BOT의 TX power level을 조정할 경우 사용하고, 이 명령 사용 이후 power level이 자동 적용 된다. 입력범위 값 : 0~7(-18~8dBm)	
AT+TXPWR?	현재 BOT의 TX power level 값 요청	
AT+ADVDATA=xxxx	Server mode로 장치가 동작할 경우 상대 장치가 검색할때 보여질 수 있는 설정 가능한 Advertising user data 영역이다(MAX 13byte).	
AT+ADVDATA?	설정된 Advertising User data의 데이터를 알고자 할 때 요청	
AT+ADVINTERVAL=xxxxx	Advertising Interval을 변경하고자 할 때 사용. Range : 20~2560 (ms)	
AT+ADVINTERVAL?	Advertising Interval 값을 알고자 할 때 요청	
AT+CONNINTERVAL=xxxx	Connection Interval 값을 변경하고자 할 때 사용 Range : 10~1999ms	
AT+CONNINTERVAL?	Connection Interval 값을 알고자 할 때 요청	
AT+NEGOCONN?	현재 Negotiation된 Connection Interval 값을 알고자 할 때 요청	

4.2. NOTIFY (BOT→HOST) Protocol Summary

프로토콜	설명	비고
+READY	전원이 인가되었고 Server 동작이 가능한 상태	전원 인가 후 3ms 이내
+ADVERTISING	상대 장치에서 검색 및 연결 가능한 상태	
+BROADCASTING	상대 장치에서 검색만 가능한 상태	
+CONNECTED[xx:xx:xx:xx:xx:xx]	상대 장치와 연결 된 경우	[xx:xx:xx:xx:xx:xx] : Remote BT Addr
+DISCONNECTED	상대 장치와 연결이 해제된 경우	

4.3. General RESPONSE (BOT→HOST) Protocol Summary

프로토콜	설명	비고
+OK	REQUEST를 정상적으로 수행될 경우	
+ERROR	REQUEST가 정상적으로 수행되지 않을 경우	

5. Advertising (Discoverable) Data format

Advertising Data는 디바이스가 Server mode이고, 연결이 되지 않더라도 검색 시 확인이 가능한 데이터를 말한다.

5.1. Format

5.1.1. Normal Advertising Mode

Advertising data 총 31byte 중 22byte의 변경 가능한 Data 영역을 포함한다.

Advertising data에 포함되는 데이터는 사용자가 변경 불가능한 Reserved 데이터 (Length 및 Flag)영역과 사용자가 UART를 통해 변경 가능한 데이터 영역으로 구성된다. 아래의 구조내의 노란색 블록으로 되어 있는 FIELD는 사용자가 AT command로 변경 가능한 영역이다

■ 구조

패킷형태는 다음과 같으며, 패킷의 Index 위치는 데이터 길이에 따라 가변적일 수 있다.

TX 레벨 및 Device Name, User Data 설정은 AT Command "AT+TXPWR", "AT+MANUF", "AT+ADVDATA"에 각각 대응된다.

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												

5.1.2. iBeacon Advertising Mode

iBeacon data 총 31byte 중 5byte의 변경 가능한 Data 영역을 포함한다.

사용자가 UART를 통해 변경 가능한 데이터 영역은 Major/Minor 및 TX Power에 대하여만 AT command로 변경 가능한 영역이다

■ 구조

패킷형태는 다음과 같고, Major/Minor 및 TX Power 변경은 ATT Command

CHIPSEN BOT-nLE52x Protocol V1.0.3

"AT+MAJOR", "AT+MINOR", "AT+IRSSI"에 각각 대응된다.

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

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

6. REQUEST Protocol Detail

6.1. AT

동작	HOST에서 BOT에 +OK 요청
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 응답.
사용 예	(HOST→BOT) : AT (BOT→HOST) : +OK

6.2. ATZ

동작	HOST에서 BOT에 리셋을 요청
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 응답 후 2초 뒤 리셋. * CONNECTED 상태에서 AT COMMAND 모드로 수행가능 * 참고로 최초 전원인가후 약 3ms이후부터
사용 예	(HOST→BOT) : ATZ (BOT→HOST) : +OK ...after 2sec.. (BOT→HOST) : +READY ➔ 리셋 수행, +READY응답으로 리셋되었음을 확인할 수 있다.

6.3. AT&F

동작	HOST에서 BOT의 변경 가능한 값들을 공장초기화 할 경우												
응답	+OK												
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)												
설명	<p>정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 응답하고 설정값들을 초기화 한 후 2초 뒤 리셋.</p> <p>* CONNECTED 상태에서 AT COMMAND 모드로 수행가능 (초기값)</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%;">1. Baud Rate : 230400</td> <td style="width: 50%;">7. Advertising Type : Normal mode</td> </tr> <tr> <td>2. Device Name : Chipsen</td> <td>8. RSSI Value : C3</td> </tr> <tr> <td>3. Advertising Interval : 1280ms</td> <td>9. Major : 0000 , Minor : 0000</td> </tr> <tr> <td>4. Connection Interval : 20ms</td> <td>10. UART Flow control : OFF</td> </tr> <tr> <td>5. Server Role : Peripheral</td> <td>**Flow control은 하드웨어 적용 필요</td> </tr> <tr> <td>6. TX Power : 5(-4dBm)</td> <td></td> </tr> </table>	1. Baud Rate : 230400	7. Advertising Type : Normal mode	2. Device Name : Chipsen	8. RSSI Value : C3	3. Advertising Interval : 1280ms	9. Major : 0000 , Minor : 0000	4. Connection Interval : 20ms	10. UART Flow control : OFF	5. Server Role : Peripheral	**Flow control은 하드웨어 적용 필요	6. TX Power : 5(-4dBm)	
1. Baud Rate : 230400	7. Advertising Type : Normal mode												
2. Device Name : Chipsen	8. RSSI Value : C3												
3. Advertising Interval : 1280ms	9. Major : 0000 , Minor : 0000												
4. Connection Interval : 20ms	10. UART Flow control : OFF												
5. Server Role : Peripheral	**Flow control은 하드웨어 적용 필요												
6. TX Power : 5(-4dBm)													
사용 예	<p>(HOST→BOT) : AT&F</p> <p>(BOT→HOST) : +OK</p> <p>...after 2sec..</p> <p>(BOT→HOST) : +READY</p> <p>➔ 변경가능한 값들을 초기화 후 리셋을 수행, +READY응답으로 리셋되었음을 확인 할 수 있다.</p>												

6.4. AT+VER?

동작	HOST에서 BOT의 Firmware 버전을 알고 싶을 경우
응답	버전정보(v[major].[minor].[release])
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	<p>정상적으로 Request를 받아 명령을 수행하였으면 펌웨어 버전을 표시</p> <p>* CONNECTED 상태에서 AT COMMAND 모드로 수행가능</p>
사용 예	<p>(HOST→BOT) : AT+VER?</p> <p>(BOT→HOST) : v1.0.0</p> <p>➔ HOST 장치에서 BOT 펌웨어 버전 v1.0.0 이라는 정보를 얻을 수 있음.</p>

6.5. AT+INFO?

동작	HOST에서 BOT의 어드레스와 Device Name, Role 상태를 알고 싶을 경우
----	--

CHIPSEN BOT-nLE52x Protocol V1.0.3

응답	BOT Address,Device Name,Mode[Role],State
응답완료	커맨드 수신 후 60ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 응답 후 어드레스, DeviceName 형태로표시
사용 예	(HOST→BOT) : AT+INFO? (BOT→HOST) : 5C:FE:86:A0:00:01,CHIPSEN,SERVER[PERIPHERAL],ADVERTISING → HOST 장치에서 BOT의 어드레스와 'CHIPSEN' 이라는 디바이스 이름, Server 모드 Peripheral Role 이며 Advertising 상태정보를 알 수 있음.

6.6. AT+UART=xxxx

동작	HOST에서 BOT와의 통신 속도를 변경할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답하고 통신속도를 변경 후 2초후에 리셋을 하고 그렇지 않다면 "+ERROR" 를 응답. 지원가능한 통신 속도는 2400, 4800, 9600, 14400, 28800, 19200, 38400, 57600, 115200, 230400, 460800 등이 가능하다.
사용 예	(HOST→BOT) : AT+UART=9600 (BOT→HOST) : +OK ... after 2sec.. (BOT→HOST) : +READY

6.7. AT+FLOWCONTROL=xx

동작	HOST에서 BOT와의 UART 통신에서 흐름제어 사용여부를 변경할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답하고 흐름제어 여부를 변경 후 2초후에 리셋을 하고 그렇지 않다면 "+ERROR" 를 응답. ** xx 는 "ON" 이거나 "OFF" 설정 가능 (초기값) "OFF" 흐름제어 사용하지 않음
사용 예	(HOST→BOT) : AT+FLOWCONTROL=ON (BOT→HOST) : +OK ... after 2sec.. (BOT→HOST) : +READY

6.8. AT+FLOWCONTROL?

동작	HOST에서 BOT와의 통신에서 흐름제어 사용여부를 알고자 할 경우
응답	흐름제어 사용여부(ON or OFF)
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 흐름제어 사용여부를 ON/OFF 형식으로 응답. * CONNECTED 상태에서 AT COMMAND 모드로 수행가능
사용 예	(HOST→BOT) : AT+FLOWCONTROL? (BOT→HOST) : ON

6.9. AT+MAJOR=xxxx

동작	BOT가 iBeacon mode일 때 HOST에서 BOT의 Major 설정 값을 변경할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	설정할 Major의 HEX 값을 문자 4자리로 전송 (ex) 0xFF50 => AT+MAJOR=FF50 정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답. 그렇지 않다면 "+ERROR" 를 응답.
사용 예	BOT가 iBeacon mode로 동작 중일 경우, (HOST→BOT) : AT+MAJOR=FF50 (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING

6.10. AT+MAJOR?

동작	HOST에서 BOT가 iBeacon으로 동작 시 설정된 Major 값을 알고자 할 경우 사용
응답	Major 값 또는 +ERROR
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 Major 값을(HEX) 응답. 그렇지 않다면 "+ERROR" 를 응답.
사용 예	(HOST→BOT) : AT+MAJOR?

	(BOT→HOST) : F902
--	-------------------

6.11. AT+MINOR=xxxx

동작	BOT가 iBeacon mode일 때 HOST에서 BOT의 Minor 설정 값을 변경할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	설정할 Minor의 HEX 값을 문자 4자리로 전송 (ex) 0xFF50 => AT+MINOR=FF50 정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답. 그렇지 않다면 "+ERROR" 를 응답.
사용 예	BOT가 iBeacon mode로 동작 중일 경우, (HOST→BOT) : AT+MINOR=FF50 (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING

6.12. AT+MINOR?

동작	HOST에서 BOT가 iBeacon으로 동작 시 설정된 Minor 값을 알고자 할 경우 사용
응답	Minor 값 또는 +ERROR
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 Minor 값을(HEX) 응답. 그렇지 않다면 "+ERROR" 를 응답.
사용 예	(HOST→BOT) : AT+MINOR? (BOT→HOST) : 55FB

6.13. AT+IRSSI=xx

동작	BOT가 iBeacon mode일 때 HOST에서 BOT의 Rssi 설정 값을 변경할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	설정할 Rssi의 HEX 값을 문자 2자리로 전송 (ex) 0xF8 => AT+IRSSI=F8 정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답. 그렇지 않

	다면 "+ERROR" 를 응답.
사용 예	BOT가 iBeacon mode로 동작 중일 경우, (HOST→BOT) : AT+IRSSI=F8 (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING

6.14. AT+IRSSI?

동작	HOST에서 BOT가 iBeacon으로 동작 시 설정된 Rssi 값을 알고자 할 경우 사용
응답	Rssi 값 또는 +ERROR
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 Rssi 값을(HEX) 응답. 그렇지 않다면 "+ERROR" 를 응답.
사용 예	(HOST→BOT) : AT+MINOR? (BOT→HOST) : C5

6.15. AT+ADVTYPE=x

동작	BOT의 Mode Type을 변경할 경우 사용 (Normal mode 또는 iBeacon mode)
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답. 그렇지 않다면 "+ERROR" 를 응답.
사용 예	(HOST→BOT) : AT+ADVTYPE=N (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING → BOT가 Normal Server mode로 동작시작 (HOST→BOT) : AT+ADVTYPE=I (BOT→HOST) : +OK (BOT→HOST) : + ADVERTISING → BOT가 iBeacon 장치로 동작시작

6.16. AT+SERVER=ROLE

동작	BOT가 SERVER mode일 경우 HOST에서 BOT의 ROLE을 변경할 경우 사용
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답. 그렇지 않다면 "+ERROR" 를 응답.
사용 예	<p>(HOST→BOT) : AT+SERVER=P (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING → BOT가 Peripheral 장치로 동작시작</p> <p>(HOST→BOT) : AT+SERVER=B (BOT→HOST) : +OK (BOT→HOST) : +BROADCASTING → BOT가 Broadcaster 장치로 동작시작</p>

6.17. AT+MANUF=xxxx

동작	HOST에서 BOT의 Manufacture(Device Name) 명을 바꿀 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 그렇지 않다면 "+ERROR" 를 응답. 최대 8자리.
사용 예	<p>(HOST→BOT) : AT+MANUF=CHIPSEN (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING → HOST 장치에서 BOT의 MANUFACTURE를 CHIPSEN으로 설정. Client에서 검색시 Advertising Data에서 CHIPSEN이라는 정보를 얻을 수 있음. 만약 Advertising(혹은 Broadcasting)중에 해당 명령어를 수행하면 이름을 적용하여 Advertising(혹은 Broadcasting)을 재시작한다.</p>

6.18. AT+TXPWR=xx

동작	HOST에서 BOT의 TX power 를 변경할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK"를 응답하고 그렇지 않다면 "+ERROR"를 응답.
사용 예	(HOST→BOT) : AT+TXPWR=xx (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING ** xx 의 범위는 0 ~ 7 까지 변경 가능 0 : -40dBm, 1 : -20dBm, 2 : -16dBm, 3 : -12dBm, 4 : -8dBm, 5 : -4dBm, 6 : 0dBm, 7 : 4dBm → 만약 Advertising(혹은 Broadcasting)중에 해당 명령어를 수행하면 새로운 TX Power를 적용 뒤 Advertising(혹은 Broadcasting)을 재시작한다.

6.19. AT+TXPWR?

동작	HOST에서 BOT의 TX power 를 알고자 할 경우
응답	TX Power(0~7)
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 TX Power 값을 표시 (0 : -40dBm, 1 : -20dBm, 2 : -16dBm, 3 : -12dBm, 4 : -8dBm, 5 : -4dBm, 6 : 0dBm, 7 : 4 dBm) * CONNECTED 상태에서 AT COMMAND 모드로 수행가능
사용 예	(HOST→BOT) : AT+TXPWR? (BOT→HOST) : 5 → HOST 장비에서 BOT의 TX Power가 5(-4dBm) 라는 정보를 얻을 수 있음

6.20. AT+ADVDATA=xxxx

동작	BOT가 Server mode일 경우 HOST에서 BOT의 Advertising User Data를 변경할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK"를 응답하고 그렇지 않다면 "+ERROR"를 응답. 최대 13자 사용자 지정이 가능하며 Default로서 설정값은 없음(NULL). 메모리 저장기능없음(Reset시 값 초기화).
사용 예	(HOST→BOT) : AT+ADVDATA=Beacon (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING ➔ Remote 장치에서 BOT를 검색할 때 Advertising User Data 항목에서 Beacon 이라는 정보를 얻을 수 있음. 만약 Advertising(혹은 Broadcasting)중에 해당 명령어를 수행하면 새로운 User Data 를 적용 뒤 Advertising(혹은 Broadcasting)을 재시작한다.

6.21. AT+ADVDATA?

동작	HOST에서 BOT의 Advertising User Data를 알고자 할 경우
응답	Advertising User Data
응답완료	커맨드 수신 후 16ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 Advertising User Data를 표시 * CONNECTED 상태에서 AT COMMAND 모드로 수행가능
사용 예	(HOST→BOT) : AT+ADVDATA? (BOT→HOST) : Beacon ➔ HOST 장치에서 BOT의 Advertising User Data가 Beacon 이라는 정보를 얻을 수 있음

6.22. AT+ADVINTERVAL=xxxx

동작	HOST에서 BOT의 Advertising Interval을 변경하고자 할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK" 를 응답하고 그렇지 않다면 "+ERROR" 를 응답 Advertising Interval Range : 20~2560(ms) * Default Advertising Interval : 1280ms
사용 예	(HOST→BOT) : AT+ADVINTERVAL=800 (BOT→HOST) : +OK (BOT→HOST) : +ADVERTISING ➔ Remote 장치에서 BOT의 Advertising Interval을 800ms 로 설정 만약 Advertising(혹은 Broadcasting)중에 해당 명령어를 수행하면 새로운 Advertising Interval 을 적용 뒤 Advertising(혹은 Broadcasting)을 재시작한다.

6.23. AT+ADVINTERVAL?

동작	HOST에서 BOT의 Advertising Interval을 알고자 할 경우
응답	Advertising Interval 표시(20~2560)
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 Advertising Interval 표시 * CONNECTED 상태에서 AT COMMAND 모드로 수행가능
사용 예	(HOST→BOT) : AT+ADVINTERVAL? (BOT→HOST) : 800 ➔ HOST 장치에서 BOT의 Advertising Interval이 800ms 라는 정보 얻을 수 있음

6.24. AT+CONNINTERVAL=xxxx

동작	HOST에서 BOT의 (Maximum) Connection Interval 을 변경하고자 할 경우
응답	+OK 또는 +ERROR
응답완료	커맨드 수신 후 8ms 이내(9600bps 기준)
설명	정상적으로 Request를 받아 명령을 수행하였으면 "+OK"를 응답하고, 그렇지 않으면 "+ERROR"를 응답. 입력값은 네자리(10~1999)까지 입력받을 수 있고, 연결되는 상대방장치에 따라 실제 Negotiation되는 Connection Interval 값은 다를 수 있음

CHIPSEN BOT-nLE52x Protocol V1.0.3

	<p>며 상대장치의 거절로 인해 Connection Parameter Update 요청이 실패할 수도 있다.</p> <p>* Advertising상태에서 수행하면 기존 Advertising을 취소하고 Connection Interval을 적용 후 재개한다.</p>
사용 예	<p>(HOST→BOT) : AT+CONNINTERVAL=30</p> <p>(BOT→HOST) : +OK</p> <p>(BOT→HOST) : +ADVERTISING</p> <p>➔ Connection Interval 을 30ms로 요청</p> <p>만약 Advertising(혹은 Broadcasting)중에 해당 명령어를 수행하면 새로운 Connection Interval 을 적용 뒤 Advertising(혹은 Broadcasting)을 재시작한다.</p>

6.25. AT+CONNINTERVAL?

동작	HOST에서 BOT의 Connection Interval 설정값을 알고자 할 경우
응답	Connection Interval 표시(Max Conn Interval)
응답완료	커맨드 수신 후 5ms 이내(9600bps 기준)
설명	<p>정상적으로 Request를 받아 명령을 수행하였으면 설정된 Connection Interval 표시</p> <p>* CONNECTED 상태에서 AT COMMAND 모드로 수행가능</p>
사용 예	<p>(HOST→BOT) : AT+CONNINTERVAL?</p> <p>(BOT→HOST) : 30</p> <p>➔ HOST 장치에서 BOT가 현재 설정한 Connection Interval이 30ms라는 정보를 얻을 수 있음</p>

6.26. AT+DISCONNECT

동작	HOST에서 BOT가 상대장치와 연결된 상태에서 AT COMMAND를 통해 연결을 해제하고자 할 경우
응답	+OK 후 +DISCONNECT 또는 +ERROR
응답완료	커맨드 수신 후 100ms 이내(9600bps 기준)
설명	<p>정상적으로 Request를 받아 명령을 수행하였으면 +OK 응답하고, "+DISCONNECT"를 응답</p> <p>* CONNECTED 상태에서 AT COMMAND 모드로 수행가능</p>
사용 예	<p>(HOST→BOT) : AT+DISCONNECT</p> <p>(BOT→HOST) : +OK</p> <p>(BOT→HOST) : +DISCONNECT</p>

CHIPSEN BOT-nLE52x Protocol V1.0.3

	(BOT→HOST) : +ADVERTISING → HOST 장치에서 현재 연결된 BOT와 상대장치의 연결해제를 요청하고 연결이 정상적으로 해제
--	--

6.27. AT+NEGOCONN?

동작	HOST에서 BOT가 상대장치와 Negotiation 된 Connection Interval 설정값을 알고자 할 경우
응답	현재 Negotiation된 값(ms)을 소수점 두자리까지 표시
응답완료	커맨드 수신 후 8ms 이내
설명	정상적으로 Request를 받아 명령을 수행하였으면 Negotiation된 Connection Interval 표시 * CONNECTED 상태에서 AT COMMAND 모드로 수행가능
사용 예	(HOST→BOT) : AT+NEGOCONN? (BOT→HOST) : 18.75 → HOST 장치에서 BOT가 현재 상대장치와 Negotiation된 Connection Interval 이 18.75ms 라는 정보를 얻을 수 있음

7. General Data transmission

SERVER와 CLIENT BLE device가 연결 된 이후 UART를 전달 받은 데이터나, 상대방 장치에서 BLE link를 통해 전송된 데이터는 형태의 변환이 없이 자동으로 전송된다..

상태	SERVER와 CLIENT가 연결된 상태
사용 예	(HOST→BOT) : ABCD<CR> (상대방장치) : ABCD<CR>

8. Service UUID Information

상대 장치와 연결을 위하여 아래와 같은 service UUID를 통해 Custom Data 송수신 service를 제공한다. 따라서 스마트폰 또는 여타 장치에서 BOT를 연결하여 사용할 경우 아래의 UUID를 사용하도록 한다...

구분	UUID	속성
Custom Data service (Primary)	0xFFFF0	N/A
Data Out service	0xFFFF1	Notification
Data In service	0xFFFF2	Write

*16bit UUID 0xFFFF0 to 128 bit UUID : 0000FFF0-0000-1000-8000-00805F9B34FB

*16bit UUID 0xFFFF1 to 128 bit UUID : 0000FFF1-0000-1000-8000-00805F9B34FB

*16bit UUID 0xFFFF2 to 128 bit UUID : 0000FFF2-0000-1000-8000-00805F9B34FB