

LOM102A 사용자 매뉴얼

와이솔

March 17, 2017

목차

하드웨어	2
Sleep mode의 wake-up 방식	3
주기전송 ON모드의 주기전송 데이터	5
Tx power와 Atten(antenna gain)	6
Application 프로세서 진행 시 CLI Command 운용	7
테스트 프로그램	7
Join Retransmission(재전송) back-off Time 기능	12
Debug message 가이드	13
Time Sync 기능	21

하드웨어

LOM102A

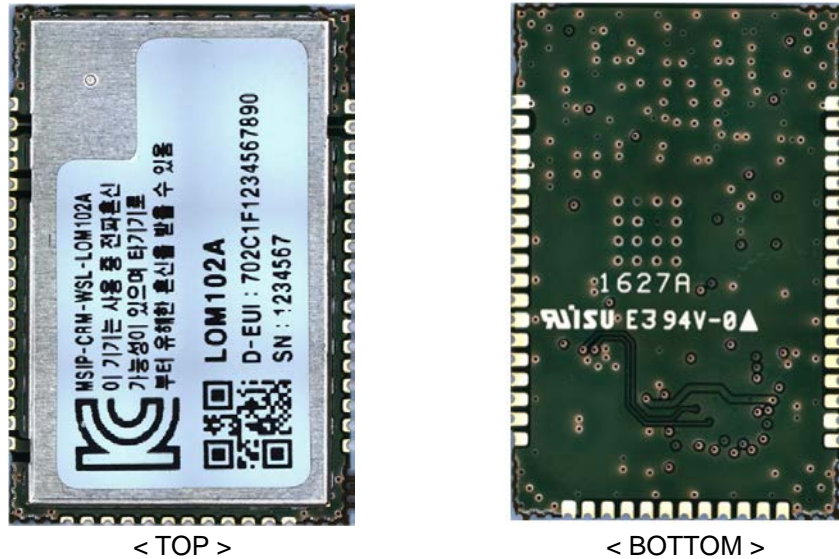


Figure 1-1: Picture of LOM102A

모듈 핀 설명

Pin No.	Pin name	MCU pin name		Description
7	P2	PB0	ADC IN	Battery Level(9V criterion) 12bit (TBD)
12	MOSI	PB15	GPIO OUT	sleep state: 0, Normal(wake-up) state:1
16	P7	PA9	UART1 TX	UART TX
19	P8	PA11	GPIO IN	Payload data bit 1
22	SWDIO	PA13	SWDIO	SWDIO
23	SWCLK	PA14	SWCLK	SWCLK
24	P11	PA15	GPIO IN	Payload data bit 0
29	P15	PB4	GPIO IN	Payload data bit 3
34	P18	PB7	GPIO IN	Boot Loader Enable(High Active)
35	P19	PA10	UART1 RX	UART RX
38	NRST	NRST	RST	Reset
39	WKUP1	PA0_WKUP1	INT IN	Wake Up: Rising Edge, Payload data bit 2
43	P24	PA3	ADC IN	Payload data bit 4 ~ 15

Firmware download 관련 Hardware 구성

모듈 펌웨어 버전 1.04이하 버전의 경우 (1.01, 102, 103, 1.04) 디바이스 설계 시 모듈의 SWD PIN (PIN22, PIN23)을 디바이스 외부에 구성하여 hex file 형태로 다운로드 할 수 있도록 해야 한다. 펌웨어 버전 1.04_PLUS부터는 디바이스 외부에 별도의 SWD PIN을 구성 할 필요가 없다.

Sleep mode의 wake-up 방식

Sleep mode wake-up 관련 Pin number

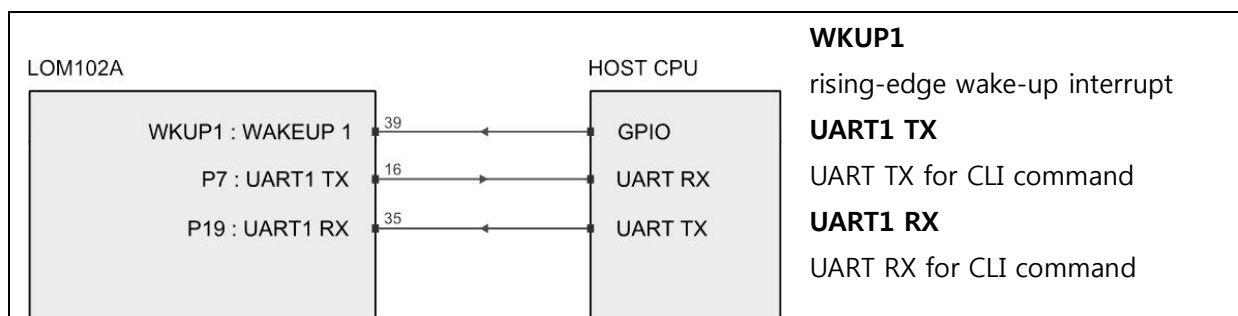
Pin No.	Pin name	MCU pin name		Description
12	MOSI	PB15	GPIO OUT	sleep state: 0, Normal:1
39	WKUP1	PA0_WKUP1	INT IN	Wake Up: Rising Edge, Payload data bit 2
35	P19	PA10	UART1 RX	UART RX

1. Pin 12 : sleep 또는 wake-up 상태를 확인 할 수 있다.
 - Low : sleep 상태
 - High : wake-up 상태
2. Pin 39 : sleep mode를 wake-up 시킨다.
 - Rising edge : sleep mode wake-up

※ Class C는 적용되지 않는다. (Class C는 sleep mode에 진입하지 않는다)

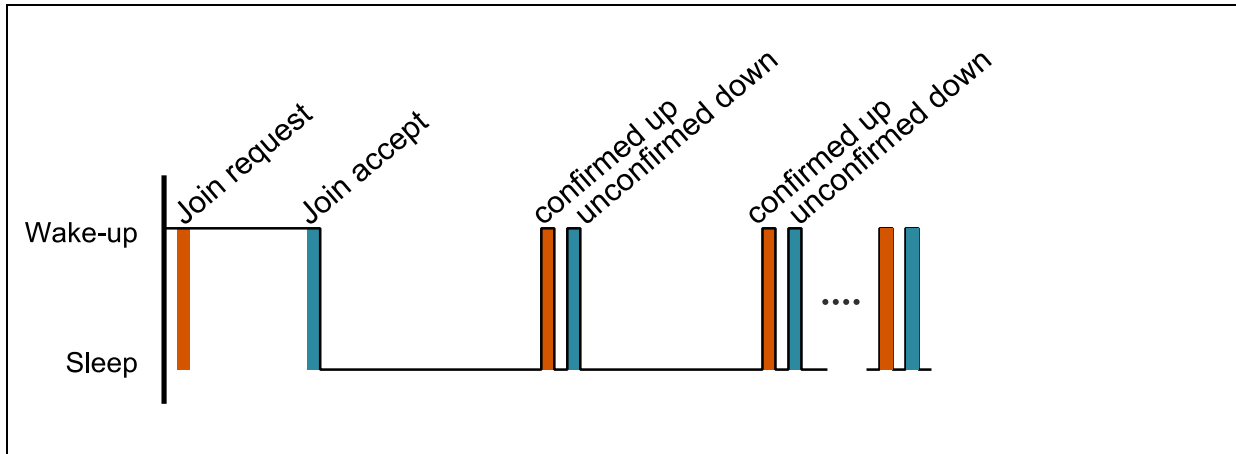
Hardware 구성

LOM102A 모듈과 UART 인터페이스를 통해 CLI command read/write 를 하기 위해서는 아래의 그림과 같이 회로를 구성한다.



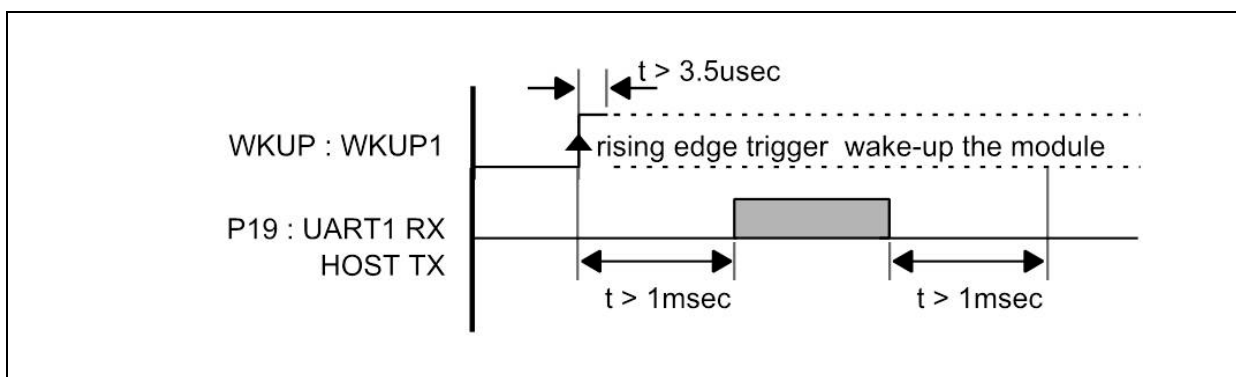
[그림. UART 인터페이스 예시 회로]

LOM102A 모듈은 Class A의 주기전송 ON모드에서 LoRaWAN 규격에 따른 Sleep/Wake-up 주기를 갖는다.



[그림. LOM102A Sleep/Wake-up 주기 예]

Sleep/Wake-up 주기에서 모듈이 Sleep 상태에 있는 경우에는 HOST CPU 로부터의 UART 인터페이스를 통한 CLI command 를 받아들일 수 없게 된다. LOM102A 가 sleep mode 상태에 있을 때 UART TX 를 하기 위해서는 강제로 wake-up 을 해야 한다. UART TX 에 대한 timing sequence는 아래와 같다.



[그림. UART TX timing sequence]

WKUP1(Pin 39)은 rising edge에 의해 트리거 되며, rising 이후 최소 3.5usec 이상 high 상태를 유지하도록 한다. UART 입력은 rising edge 시작 기준 최소 1msec이상의 시간 이후에 진행 한다.

주기전송 ON모드의 주기전송 데이터

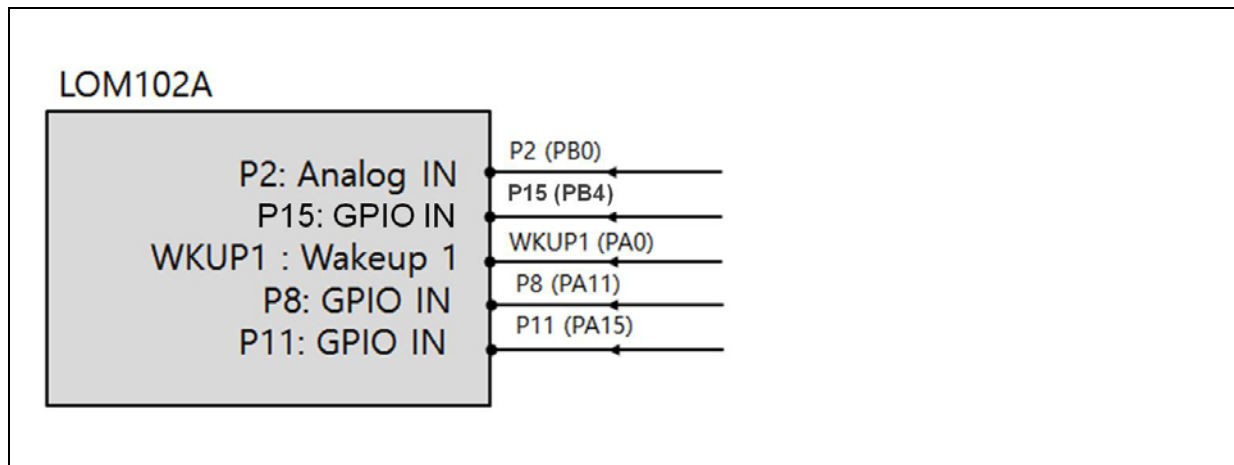
주기전송 ON모드의 주기전송 데이터관련 Pin number

Pin No.	Pin name	MCU pin name		Description
19	P8	PA11	GPIO IN	Payload data bit 1
24	P11	PA15	GPIO IN	Payload data bit 0
29	P15	PB4	GPIO IN	Payload data bit 3
39	WKUP1	PA0_WKUP1	INT IN	Wake Up: Rising Edge, Payload data bit 2
43	P24	PA3	ADC IN	Payload data bit 4 ~ 15

주기전송 ON모드의 주기전송 데이터 구성

1. 주기전송 ON모드에서 설정된 주기시간마다 자동으로 전송하는 주기전송 데이터는 총 2byte이며, GPIO INPUT pin 3개의 3bit와 wake-up pin 1개의 1bit 그리고 ADC pin 1개의 12bit로 구성되어 있다.
2. 주기전송 시 "DataReport : 0x0000" 형태로 UART에 출력된다.
3. 일반적으로 ADC pin은 플로팅 상태이므로 값이 변동되어 보인다.

LOM102A 의 주기 보고 데이터에는 아래 네 핀의 상태 정보가 포함된다.



[그림. 주기보고 데이터 핀]

주기보고 데이터는 아래와 같이 2byte로 구성되어 있다.

bit	15 : 4	3	2	1	0
할당	P2	P15	WKUP1	P8	P11

[표. 주기보고 데이터]

주기보고 데이터에 주기적으로 Logic level 값 및 Analog Data 값을 체크하고자 할 경우에는 P8, P11, P2 핀에 해당 회로를 연결할 수 있으며, LOM102A 모듈을 Sleep mode에서 Wakeup 하기 위해 WKUP1 Pin에 Low Level에서 High Level로 전환함으로써 LOM102A를 Wakeup 할 수 있다.

Tx power와 Atten(antenna gain)

Tx power관련 유의사항

Tx power는 온도/전압에 따른 출력 변동 등 모듈의 환경에 따라 $\pm 1.5\text{dBm}$ 의 변동폭이 발생 할 수 있으므로 전파법 규정에 준수하기 위해 atten 값 조정이 필요하다.

Atten 값은 1dBm단위로 설정할 수 있다.

Tx power 설정값은 다음 식을 이용한다.

$$Tx \text{ 설정 Power} = \text{원하는 Tx power} - \text{변동폭} - \text{안테나 Gain}$$

예) 14dBm의 TX power를 출력하기 위해 2dBi의 안테나를 사용할 경우 다음 식을 이용한다.

원하는 Tx power : 14dBm

변동폭 : 1.5dBm

안테나 Gain : 2dBm

$$14\text{dBm} - 1.5\text{dBm} - 2\text{dBm} = 10.5\text{dBm}$$

Tx 설정 Power는 10dBm으로 설정한다. (1dBm 단위로 설정 가능 하므로)

(atten 값 조정관련 커맨드는 CLI Command list 문서 참조 요망)

각 채널 인덱스 별 default Tx power value

Channel Index	Frequency (MHz)	Power Index	Tx power value (dBm)	Description	
1	922.1	0	14	Default channel	RX1 channel
2	922.3	0	14		
3	922.5	0	14		
4	921.9	4	10	RX2 channel	
5	922.7	0	14		
6	922.9	0	14		
7	923.1	0	14		
8	923.3	0	14		

1. Join 진행 시 Join Request 전송 채널은 Channel Index 1~3(Default channel)에 해당하는 3개의 채널이 hopping 하면서 전송한다.
2. Join 완료 후에는 Channel Index 1~8에 해당하는 총 8개의 채널이 hopping 하면서 전송한다.

Application 프로세서 진행 시 CLI Command 운용

Applicaion 프로세서 진행 시 CLI Command 동작 방식

모듈이 프로세서 진행을 완료하지 않은 상태에서 Application에서 커맨드가 오면 커맨드를 실행하지 않고 Application에게 'Busy', 'ERROR' 실패 코드를 전달한다.

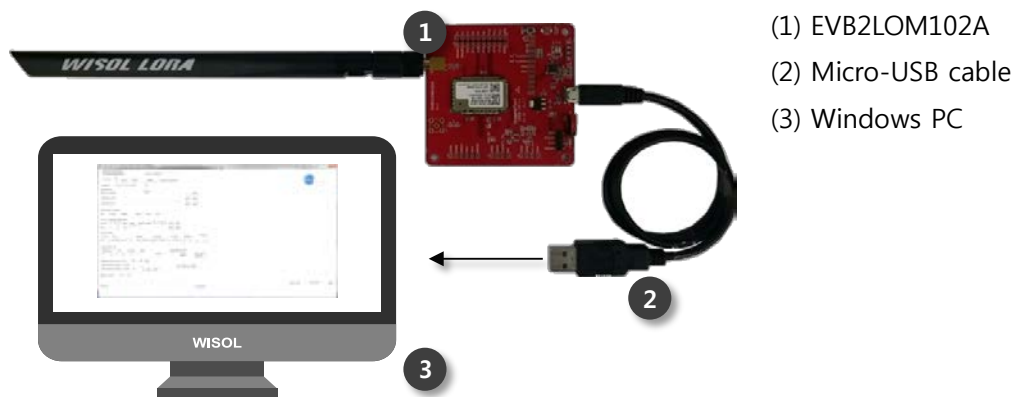
수신 완료 후 CLI Command 입력 시간

서버로부터 Unconfirmed-down 등의 데이터 수신 완료 후 CLI Command 입력 시 200ms 이상의 시간 지연 후 CLI Command 실행이 필요하다.

테스트 프로그램

Evaluation board 연결

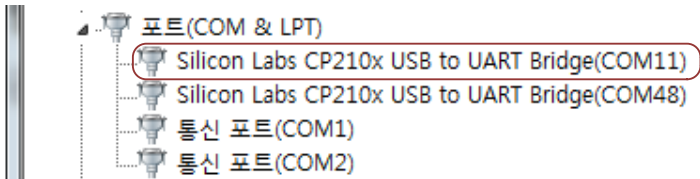
1. EVB2LOM102A 와 Windows PC 를 Micro-USB 케이블로 연결



[Fig. EVB2LOM102A 연결]

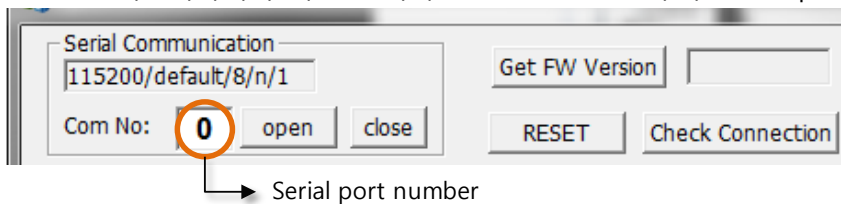
테스트 프로그램 실행

1. EVB2LOM102A 을 Windows PC 에 연결한 상태에서 장치관리자에서 시리얼 포트 연결 및 포트 번호 확인
 - A. Silicon Labs CP210X USB to UART Bridge (Com□□)



[Fig. 장치관리자 시리얼 포트 연결 확인]

2. 제공된 테스트 프로그램 LoRaWAN_GUI 실행
3. 장치관리자에서 확인된 시리얼 COM 번호 입력 후 Port open



Serial port number

[Fig. COM 번호 입력 및 Port open]

가상 시리얼 장치 드라이버 설치

가상 시리얼 장치가 인식되지 않아 드라이버를 수동으로 설치 하기 위해서는

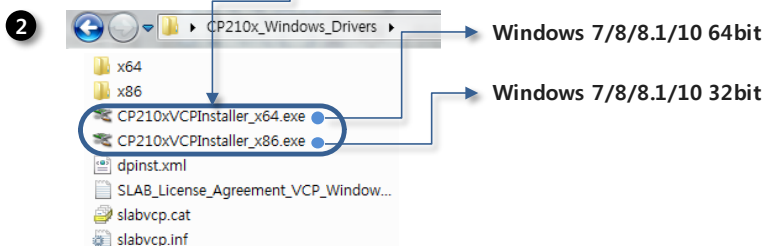
<https://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx> 에서 OS 버전에 맞는 드라이버 파일을 다운로드 한 뒤 장치드라이버를 수동으로 설치한다.

1 Download for Windows 7/8/8.1/10 (v6.7.3)

(1) 드라이버 다운로드

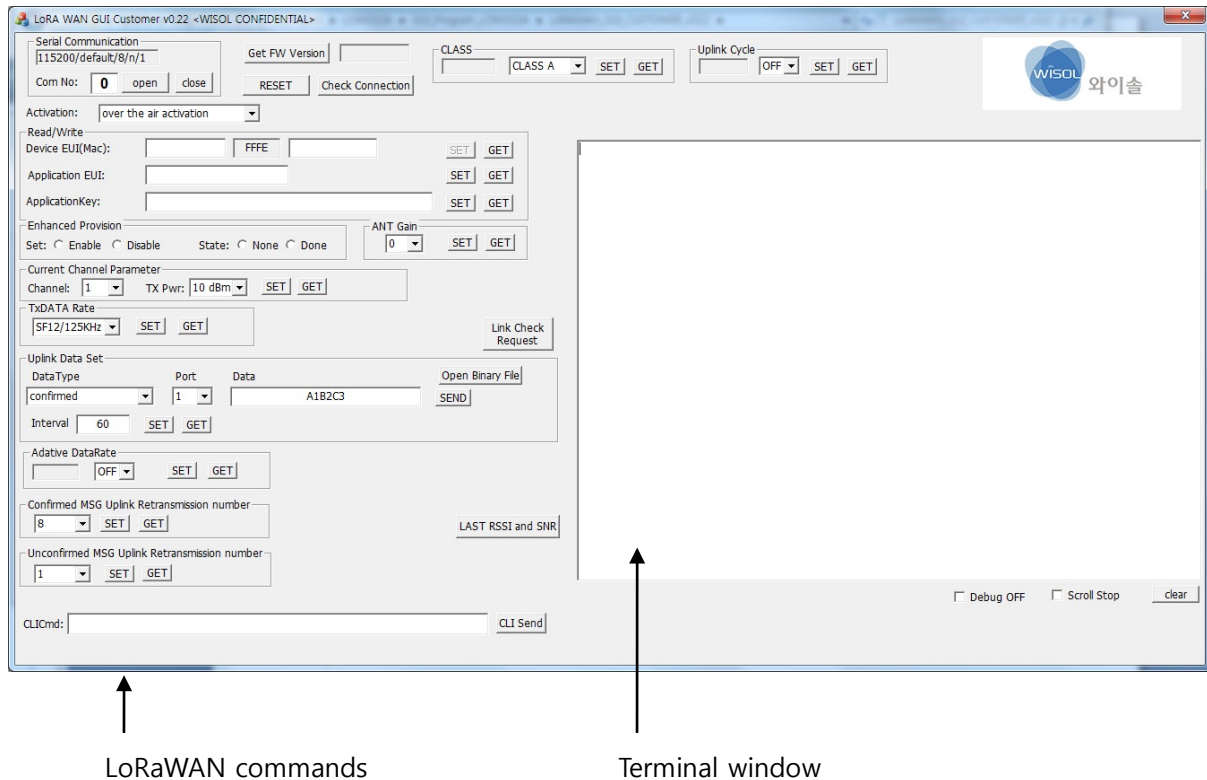
Platform	Software	Release Notes
Windows 7/8/8.1/10	Download VCP (5.3 MB) (Default)	Download VCP Revision History
Windows 7/8/8.1/10	Download VCP with Serial Enumeration (5.3 MB) Learn More »	Download VCP Revision History

(2) 드라이버 설치



[Fig. 가상 시리얼 장치 드라이버 수동 설치]

테스트 프로그램 메뉴 설명



[Fig. 테스트 프로그램 실행 화면]

- Serial Communication
 - ① Com No : 연결된 EVB 의 시리얼 포트 의 포트번호 입력란 (장치관리자에서 확인 후 입력)
 - ② open : 지정된 시리얼 포트 열기
 - ③ close : 지정/연결된 시리얼 포트 닫기
- Get FW Version : LOM102A 펌웨어 버전 확인
- RESET : LOM102A system software reset.
- Check Connection : Serial connection 상태를 체크. 연결에 문제가 없으면 "OK" 응답
- CLASS : class 모드 설정
 - ① CLASS A : class A 모드
 - ② CLASS C : class C 모드
 - ③ SET : 선택한 CLASS 모드 설정 및 저장
 - ④ GET : CLASS 모드 설정 값 확인
- Uplink Cycle : 주기전송 기능 비활성화/활성화 설정
 - ① OFF : 주기전송 기능 비활성화
 - ② ON : 주기전송 기능 활성화
 - ③ SET : Uplink Cycle 주기전송 ON/OFF 설정

- ④ GET : Uplink Cycle 주기전송 설정 값 확인
- Activation : LoRaWAN activation 조건 선택
 - ① over the air activation
 - ② activation by personalization
- Read/Write
 - ① Device EUI
 - (1) GET : Device EUI 설정 값 확인
 - ② Application EUI
 - (1) GET : Application EUI 설정 값 확인
 - (2) SET : Application EUI 설정
 - ② ApplicationKey : Application key 설정
 - (1) GET : Application key 설정 값 확인
 - (2) SET : Application key 설정
- Enhanced Provision
 - ① Set : Enhanced provisioning 활성화 선택
 - (1) Enable : Enhanced provisioning 활성화
 - (2) Disable : Enhanced provisioning 비활성화
 - ② State : Enhanced provisioning 의 Join request 시작 방식 설정
(설정 값 정상 적용을 위해 설정 후 반드시 system을 reset 할 것)
 - ① State None : Pseudo Key로 Join request 시작
 - ② State Done : Real Key 로 Join request 시작
- Current Channel Parameter (채널 별 Tx power 설정)
 - (1) Channel : Channel index 선택
 - ① 설정 값 : 1 ~ 8
 - ② Channel index 별 default Frequency
 - 1 : 922.1 MHz
 - 2 : 922.3 MHz
 - 3 : 922.5 MHz
 - 4 : 921.9 MHz
 - 5 : 922.7 MHz
 - 6 : 922.9 MHz
 - 7 : 923.1 MHz
 - 8 : 923.3 MHz
 - (2) TX Pwr : Transmit output power index 선택
 - ① 설정값 : 0 ~ 14
 - ② power index 별 power 값
 - 0 : 14 dBm
 - 1 : 13 dBm
 - 2 : 12 dBm

3 : 11 dBm
 4 : 10 dBm
 5 : 9 dBm
 6 : 8 dBm
 7 : 7 dBm
 8 : 6 dBm
 9 : 5 dBm
 10 : 4 dBm
 11: 3 dBm
 12 : 2 dBm
 13 : 1 dBm
 14 : 0 dBm

(3) SET : 선택된 Channel index와 power index에 대한 설정 값 변경 및 저장

(4) GET : 전체 각 채널 별 Frequency 및 Tx power 설정 값 확인

- TxDATA Rate : Transmit output data rate 설정

① Data rate 설정 값

- SF12/125KHz
- SF11/125KHz
- SF10/125KHz
- SF9/125KHz
- SF8/125KHz
- SF7/125KHz

② SET : Transmit output data rate 설정

③ GET : Transmit output data rate 설정 확인

- Uplink Data Set : Confirmed / unconfirmed uplink data 설정 및 전송

(1) Type

- ① confirmed
- ② unconfirmed

(2) Port : Port filed

① 범위 : 1~221

(3) Interval : 다음 전송까지의 지연시간(sec)

① 범위 : 0 ~ 31622400 (최대 366일 설정 가능)

(4) Data : Message

(5) SEND : Uplink Data 설정 및 전송

- Link Check Request : Link check request 시작

(Link Check Request 버튼 클릭 후 Uplink Data Set의 SEND버튼을 클릭하여 Uplink Data를 전송하거나 또는 Link Check Request 버튼 클릭 후 자동 주기보고에 의한 Uplink data가 전송될 때 Link check request가 적용 됨)

- Adaptive data rate on/off : Adaptive data rate 기능 활성화/비활성화 설정

- (1) ON : ADR on
- (2) OFF : ADR off
- (3) SET : ADR 설정
- (4) GET : ADR 설정 값 확인
- Confirmed MSG Uplink Retransmission number
 - : Confirmed MSG Uplink transmission retry 수 설정
 - ① 범위 : 1~8
 - ② SET : Confirmed MSG Uplink transmission retry 수 설정
 - ③ GET : Confirmed MSG Uplink transmission retry 수 설정 확인
- Unconfirmed MSG Uplink Retransmission number
 - : Unconfirmed MSG Uplink transmission retry 수 설정
 - ① 범위 : 1~8
 - ② SET : Unconfirmed MSG Uplink transmission retry 수 설정
 - ③ GET : Unconfirmed MSG Uplink transmission retry 수 설정 확인
- LAST RSSI and SNR : 마지막 수신된 RSSI값과 SNR 값 확인

Join Retransmission(재전송) back-off Time 기능

Join Request 송신에 대해 서버로부터 Join Accept를 수신 받지 못하면 Join Request 송신에 대한 재전송을 진행한다. 이 때 LoRaWAN 규격에 의해 back-off Time 기능이 적용되어 아래와 같은 시간 간격으로 재전송을 한다.

모듈 Reset 또는 전원 인가 후 1시간 동안

처음 1분 동안 10초 간격으로 최초 전송 포함 7회 재전송하며
그 이후 나머지 59분 동안 약 2분 52초 간격으로 20회 재전송 한다.

그 이후 10시간 동안

약 24분 11초 간격으로 총 25회 재전송 한다.

그 이후 24시간 동안

24시간 동안 총 5회 재전송하며 그 시간 간격은 일정하지 않다.
대략적인 시간간격은 아래와 같다.

- 1회와 2회 사이 시간간격 : 2시간 23분 9초
- 2회와 3회 사이 시간간격 : 2시간 23분 10초
- 3회와 4회 사이 시간간격 : 11시간 55분 49초

4회와 5회 사이 시간간격 : 3시간 34분 46초

그 이후 24시간 마다

24시간 동안 총 5회 재전송하며 그 시간 간격은 일정하지 않다.

Join-Request 송신관련 디버그메시지

10초 간격으로 Join-Request 송신관련 디버그메시지를 출력하지만, 실제 송신이 되는 것이 아니며, 아래의 디버그 메시지가 출력되면서 실제 Join-Request가 송신 된다.

SEND : Freq: <frequency> DR: <dataRate> PWR: 1

Debug message 가이드

GUI전용 프로그램 Log창에서의 Time stamp Debug message

GUI전용 프로그램 Log창에는 debug message 중간에 '[2016-01-01 12:00:00 000]:'과 같은 형태의 Time stamp가 출력되는데 이는 GUI 프로그램상에서만 출력되며 실제 UART에는 Time stamp가 출력되지 않는다.

Reset 또는 전원 인가 시 Debug message (Jon Request / Accept)

모듈에 대해 reset 또는 전원 인가 시 pseudo join 또는 real join을 시도한다.

처음에는 항상 pseudo join 부터 시작하며 pseudo join 절차에서 서버로부터 real key를 받아 자동 저장되면, real join으로 변경된다. 따라서 그 다음부터는 시스템 reset을 해도 항상 real join 부터 시작한다.

또한 pseudo join 및 real join 절차를 거쳐 최종적으로 join이 완료되어도 그 다음부터는 시스템 reset을 해도 항상 real join부터 시작한다.

1. CLASS A 모드의 pseudo Join request과 real join request의 debug message

1) pseudo join debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 0
 SEND : Freq: 922100000 DR: 0 PWR: 1
 1st Join
 RX1CH Open: freq 922100000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 0

2) real join debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 0
 SEND : Freq: 922100000 DR: 0 PWR: 1
 2nd Join
 RX1CH Open: freq 922100000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 0

2. CLASS C 모드의 pseudo Join request과 real join request의 debug message

1) pseudo join debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 0
 SEND : Freq: 922300000 DR: 0 PWR: 1
 1st Join
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

2) real join debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 0
 SEND : Freq: 922300000 DR: 0 PWR: 1
 2nd Join
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

3. CLASS A 모드 / 주기보고 OFF 모드의 join 완료에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 2 P 0
 SEND : Freq: 922300000 DR: 0 PWR: 1
 2nd Join
 RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_JOIN_ACCEPT / Ver: 0
 OnRadioRxDone: ADR OFF, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 0 P 32
 Join is completed

4. CLASS A 모드 / 주기보고 ON 모드의 join 완료에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 0
 SEND : Freq: 922300000 DR: 0 PWR: 1
 2nd Join
 RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_JOIN_ACCEPT / Ver: 0
 OnRadioRxDone: ADR OFF, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 0 P 32
 PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 1
 SEND : Freq: 922500000 DR: 0 PWR: 1
 DataReport : 0x3360
 RX1CH Open: freq 922500000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_DATA_UNCONFIRMED_DOWN / Ver: 0
 OnRadioRxDone: ADR ON, ADDRACKReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 1 P 192

Join is completed

5. CLASS C 모드 / 주기보고 OFF 모드의 join 완료에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 0
 SEND : Freq: 922300000 DR: 0 PWR: 1
 2nd Join

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_JOIN_ACCEPT / Ver: 0
 OnRadioRxDone: ADR OFF, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 0 P 32
 Join is completed

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

6. CLASS C 모드 / 주기보고 ON 모드의 join 완료에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_JOIN_REQ / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 0
 SEND : Freq: 922500000 DR: 0 PWR: 1
 2nd Join
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 922500000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_JOIN_ACCEPT / Ver: 0
 OnRadioRxDone: ADR OFF, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 0 P 32
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 1
 SEND : Freq: 923100000 DR: 0 PWR: 1
 DataReport : 0x3140
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 923100000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_DATA_UNCONFIRMED_DOWN / Ver: 0
 OnRadioRxDone: ADR ON, ADDRACKReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 1 P 111

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 Join is completed

7. debug message에 대한 설명

- FRAME_TYPE_JOIN_REQ : 송신 메시지 형태로 Join request임을 나타낸다
- FRAME_TYPE_JOIN_ACCEPT : Join request에 대한 수신 ack로 Join accept임을 나타낸다.
- FCnt : 프레임 카운트를 의미하며, 송신할 때마다 1씩 증가한다.
 단, 재전송(retransmission) 시에는 프레임 카운트가 증가하지 않는다.
- P : Fport 번호를 나타낸다. Join request의 Fport 번호는 0이다.
 할당된 Fport 번호는 아래와 같다.
 - ① Fport 0 : Join request, mac command(서버트리거)에 대한 Request-Answer 송수신에 사용 된다.
 - ② Fport 1 ~ 221 : 유저가 송신(Tx)에 사용하는 Fport번호로 범위내에서 랜덤하게 사용 가능하다. 송신에 대해서 서버로부터 수신 받는 ack 형태의 데이터는 Fport번호가 랜덤하게 부여된다.
 - ③ Fport 222 : ThingPlug 서버에 할당되어 있다. 따라서 ThingPlug 제어명령 수신 데이터의 Fport번호는 222번이다.
 - ④ Fport 223 : 네트워크 서버에 할당되어 있다. 따라서 서버 제어명령 수신 데이터의 Fport번호는 223번이다.
- join request '*1st Join*'은 첫 번째 Join을 의미하며 이는 pseudo join을 시도하는 debug message이다.
- SEND : 데이터 송신을 의미 한다.

- Freq : 송신 Frequency(주파수)를 의미한다. 단위는 Hz이다.
- DR : Data rate을 의미한다.
- PWR : Tx power index를 의미한다.
0 : 14dBm, 1 : 13dBm, 2 : 12dBm, 3 : 11dBm, 4 : 10dBm, 5 : 9dBm
6 : 8dBm, 7 : 7dBm, 8 : 6dBm, 9 : 5dBm, 10 : 4dBm, 11 : 3dBm
12 : 2dBm, 13 : 1dBm, 14 : 0dBm
- *1st Join* : First Join(첫 번째 Join)을 의미하며 pseudo Join request에 해당한다.
- *2nd Join* : Second Join(두 번째 Join)을 의미하며 real Join request에 해당한다.
- RX1CH Open : RX1 Channel로 수신 대기 상태를 의미 한다.
- RX2CH Open : RX2 Channel로 수신 대기 상태를 의미 한다.
- freq : 수신 Frequency(주파수)를 의미한다. 단위는 Hz이다.
- SF : SpreadingFactor를 의미한다. (data rate과 매칭된다)
- Join is completed : 최종적으로 join이 완료되었음을 의미 한다.

송수신(TX/RX) Debug message

송수신 데이터는 Confirmed와 Unconfirmed 두 가지 형태가 있다.

Confirmed는 ack를 요청하는 데이터이며 Unconfirmed는 ack를 요청하지 않는 데이터이다.

송신은 Confirmed-up과 Unconfirmed-up의 Uplink이다.

수신은 Confirmed-down과 Unconfirmed-down의 Downlink이다.

Confirmed-up 형태로 송신을 하면 서버로부터 반드시 ack를 받는다.

ack는 Confirmed-down이나 Unconfirmed-down으로 받을 수 있는데 일반적으로 Confirmed-up에 대해서 Unconfirmed-down으로 ack를 받는다. 이는 ack에 다시 ack를 요청하는 데이터를 요청하게 되면 또 다시 서버로 송신을 해야 하고 이는 무한 루프에 빠지게 된다.

이와 같은 현상을 방지하기 위해 아래와 같은 방식으로 운용된다.

Confirmed-up -> Unconfirmed-down

Confirmed-up -> Confirmed-down -> Unconfirmed-up

Unconfirmed-up -> ack 없음

단, Unconfirmed-up일지라도 송신 후 RX 수신모드로 전환되므로 서버로부터 데이터를 수신 할 수도 있다.

1. CLASS A 모드의 Confirmed-up과 Unconfirmed-down에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 1
 SEND : Freq: 922300000 DR: 0 PWR: 1
 DataReport : a1b2c3

OK

RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_DATA_UNCONFIRMED_DOWN / Ver: 0
 OnRadioRxDone: ADR ON, ADDRACKReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 1 P 134

2. CLASS A 모드의 Unconfirmed-up에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_UNCONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 2 P 1
 SEND : Freq: 922100000 DR: 0 PWR: 1
 DataReport : a1b2c3

OK

RX1CH Open: freq 922100000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 0

3. CLASS C 모드의 Confirmed-up과 Unconfirmed-down에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 1 P 1
 SEND : Freq: 922100000 DR: 0 PWR: 1
 DataReport : a1b2c3

OK

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 922100000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_DATA_UNCONFIRMED_DOWN / Ver: 0
 OnRadioRxDone: ADR ON, ADDRACKReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 1 P 210

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

4. CLASS C 모드의 Unconfirmed-up에 대한 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_UNCONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 2 P 1
 SEND : Freq: 923300000 DR: 0 PWR: 1
 DataReport : a1b2c3

OK

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 923300000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

5. debug message에 대한 설명

- FRAME_TYPE_DATA_CONFIRMED_UP : 송신 메시지 형태로 Confirmed-up임을 나타낸다
- FRAME_TYPE_DATA_UNCONFIRMED_UP: 송신 메시지 형태로 Unconfirmed-up임을 나타낸다
- FRAME_TYPE_DATA_UNCONFIRMED_DOWN: 수신 메시지 형태로 Unconfirmed-down임을 나타낸다
- DataReport : DataReport 콜론(:) 다음에 출력되는 데이터는 송신 메시지이다.

ThingPlug 제어 명령 수신 Debug message

ThingPlug 제어 명령은 DevReset, RepPerChange, RepImmediate, extDevMgmt 총 4개가 있다.

모듈은 ThingPlug 제어 명령을 수신하면 제어 명령에 대해 Application에 출력하여 UART를 통해 디바이스에 전달만 할 뿐, 그에 따른 동작을 수행하지 않는다. 따라서 디바이스에서는 이를 전달 받아 CLI Command를 통해 모듈을 제어해야 한다.

ThingPlug 제어 명령을 수신하면 모듈은 즉시 ack를 송신하는데, 이때의 ack는 Unconfirmed-up의 형태이며 payload 값은 Null이다.

DevReset : 디바이스 또는 모듈을 리셋한다.

RepPerChange <interval time(sec)> : 주기전송 interval time을 설정한다.

RepImmediate : 데이터를 즉시 보고(송신) 한다.

extDevMgmt : 사용자 정의 제어 명령을 내린다.

1. ThingPlug 제어명령 DevReset 수신 시의 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0

PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 2 P 1

SEND : Freq: 922300000 DR: 0 PWR: 1

DataReport : a1b2c3

OK

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0

OnRadioRxDone: MType: FRAME_TYPE_DATA_CONFIRMED_DOWN / Ver: 0

OnRadioRxDone: ADR ON, ADDRACKReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 2 P 222

Rx_MSG 222 : 008000

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

AppDataProcessWithThingPlug: RCV 128

DevReset

PrepareFrame: MType: FRAME_TYPE_DATA_UNCONFIRMED_UP / Ver: 0

PrepareFrame: ADR ON, ADDRACKReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 3 P 4

SEND : Freq: 923300000 DR: 0 PWR: 1

DataReport :

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

RX1CH Open: freq 923300000, SF 12, BW 0, TO 25, RXC 0

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

2. ThingPlug 제어명령 RepPerChange 수신 시의 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0

PrepareFrame: ADR ON, ADDRACKReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 4 P 1

SEND : Freq: 922900000 DR: 0 PWR: 1

DataReport : a1b2c3

OK

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

RX1CH Open: freq 922900000, SF 12, BW 0, TO 25, RXC 0

OnRadioRxDone: MType: FRAME_TYPE_DATA_CONFIRMED_DOWN / Ver: 0

OnRadioRxDone: ADR ON, ADDRACKReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 3 P 222

Rx_MSG 222 : 0081010f

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

AppDataProcessWithThingPlug: RCV 129

RepPerChange : 15
 PrepareFrame: MType: FRAME_TYPE_DATA_UNCONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADRAckReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 5 P 4
 SEND : Freq: 921900000 DR: 0 PWR: 2

DataReport :
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

3. ThingPlug 제어명령 ReplImmediate 수신 시의 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADRAckReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 6 P 1
 SEND : Freq: 922100000 DR: 0 PWR: 1
 DataReport : a1b2c3

OK
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 922100000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_DATA_CONFIRMED_DOWN / Ver: 0
 OnRadioRxDone: ADR ON, ADRAckReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 4 P 222
 Rx_MSG 222 : 008200

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 AppDataProcessWithThingPlug: RCV 130
 ReplImmediate
 PrepareFrame: MType: FRAME_TYPE_DATA_UNCONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADRAckReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 7 P 4
 SEND : Freq: 923300000 DR: 0 PWR: 1
 DataReport :
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 923300000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

4. ThingPlug 제어명령 extDevMgmt 수신 시의 debug message

PrepareFrame: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADRAckReq 0, ACK 0 Fpending 0 FOptsLen 0 FCnt 8 P 1
 SEND : Freq: 922700000 DR: 0 PWR: 1
 DataReport : a1b2c3

OK
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 922700000, SF 12, BW 0, TO 25, RXC 0
 OnRadioRxDone: MType: FRAME_TYPE_DATA_CONFIRMED_DOWN / Ver: 0
 OnRadioRxDone: ADR ON, ADRAckReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 5 P 222
 Rx_MSG 222 : 0000170123456789012345678901234567890123456789012345

RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 AppDataProcessWithThingPlug: RCV 0
 extDevMgmt
 PrepareFrame: MType: FRAME_TYPE_DATA_UNCONFIRMED_UP / Ver: 0
 PrepareFrame: ADR ON, ADRAckReq 0, ACK 1 Fpending 0 FOptsLen 0 FCnt 9 P 4
 SEND : Freq: 921900000 DR: 0 PWR: 2
 DataReport :
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1
 RX1CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 0
 RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 1

5. debug message에 대한 설명

- FRAME_TYPE_DATA_CONFIRMED_DOWN : 수신 메시지 형태로 Confirmed-down임을 나타낸다
- Rx_MSG : 수신 데이터의 payload 메시지를 의미하며 메시지 형태는 아래와 같다.
Rx_MSG <Fport> : <LoRaMAC version> <message type> <Payload length> <Payload>
1byte 1byte 1byte 1byte이상
- DevReset : ThingPlug 제어명령 중 하나로 디바이스 또는 모듈에 대 리셋 명령어를 나타낸다.
- RepPerChange : ThingPlug 제어명령 중 하나로 주기전송 interval time 설정 명령어를 나타낸다.
RepPerChange <interval time(sec)>의 형태로 출력된다.
- RepImmediate : ThingPlug 제어명령 중 하나로 데이터 즉시 보고(송신) 요청 명령어를 나타낸다.
- extDevMgmt : ThingPlug 제어명령 중 하나로 사용자 정의 제어 명령어를 나타낸다.

CLI Command Debug message

CLI Command debug message는 CLI Command list 문서에 상세히 기술되어 있으므로 이를 참고하면 된다. (CLI Command list 문서에서 실행 Message 부분을 참고 할 것)

Time Sync 기능

Time Sync Request / Answer

Time Sync 기능은 LoRaWAN 규격의 Mac command에 해당하며, LoRa 단말에서 Time Sync를 요청하면 서버로부터 초(sec)단위의 현재시간 정보를 ack로 받을 수 있다.

Time Sync Request 실행 순서

step 1 : Time Sync Request CLI command를 실행한다.

LRW 39

step 2 : Confirmed-up 형태의 전송(Tx) CLI command를 실행한다.

LRW 31로 시작하는 전송(Tx) CLI command 또는 LRW 4D로 시작하는 전송(Tx) CLI command

Debug message 형태

1. Time Sync Request CLI command에 대한 리턴 debug message

AddMacCommand : MOTE_MAC_TIME_SYNC_REQ

OK

2. 서버로부터 수신 받은 Time Sync Answer debug message

펌웨어 버전 1.05와 1.05_PLUS의 Time Sync Answer debug message는 각각 다르게 출력된다.

1) 펌웨어 버전 1.05

ProcessMacCommands: SRV_MAC_TIME_SYNC_ANS: S <초 단위 데이터> FS <초 단위 소수점 이하 데이터>

예시) ProcessMacCommands: SRV_MAC_TIME_SYNC_ANS: S 1173661165 FS 61056

```
LRW 39
[2017-03-16 10:05:06 904]:AddMacCommand: MOTE_MAC_TIME_SYNC_REQ
OK
LRW 31 a1b2c3 cnf 1
RSP Timeout
LRW 31 a1b2c3 cnf 1
[2017-03-16 10:05:19 509]:PrepareFrame[2017-03-16 10:05:19 525]: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ver: 0
PrepareFrame: ADR ON, ADRACKReq 0, ACK 0 Fpending 0 FOptLen 1 FCnt 4 P 1
[2017-03-16 10:05:19 540]: SEND : Freq: 922300000 DR: 0 PWR: 0
DataReport : a1b2c3
OK
[2017-03-16 10:05:21 771]:RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
[2017-03-16 10:05:23 222]:OnRadioRxDone: MType: FRAME_TYPE_DATA_UNCONFIRMED_DOWN / Ver: 0
OnRadioRxDone: ADR OFF, ADRACKReq 0, ACK 1 Fpending 0 FOptLen 0 FCnt 4 P 0
[2017-03-16 10:05:23 238]:ProcessMacCommands: SRV_MAC_TIME_SYNC_ANS: S 1173661538 FS 56682
```

2) 펌웨어 버전 1.05_PLUS

ProcessMacCommands: SRV_MAC_TIME_SYNC_ANS: S <초 단위 데이터> FS <초 단위 소수점 이하 데이터>

time : YYYY-MM-DD HH:MM:SS <초 단위 소수점 이하 데이터>

예시) ProcessMacCommands: SRV_MAC_TIME_SYNC_ANS: S 1173661165 FS 61056

time : 2017-03-16 09:59:08 931654839

```

LRW 39
[2017-03-16 09:58:48 526]:AddMacCommand: MOTE_MAC_TIME_SYNC_REQ
OK
LRW 31 a1b2c3 cnf 1
[2017-03-16 09:58:49 742]:PrepareFrame[2017-03-16 09:58:49 742]: MType: FRAME_TYPE_DATA_CONFIRMED_UP / Ve
PrepareFrame: ADR ON, ADRACKReq 0, ACK 0 Fpending 0 FOptsLen 1[2017-03-16 09:58:49 742]: FCnt 1 P 1
[2017-03-16 09:58:49 774]: SEND : Freq: 923300000 DR: 0 PWR: 0
DataReport : a1b2c3
OK
[2017-03-16 09:58:52 004]:RX1CH Open: freq 923300000, SF 12, BW 0, TO 25, RXC 0
[2017-03-16 09:58:53 018]:RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 0
[2017-03-16 09:58:56 840]: SEND : Freq: 922900000 DR: 0 PWR: 0
DataReport : a1b2c3
[2017-03-16 09:58:59 071]:RX1CH Open: freq 922900000, SF 12, BW 0, TO 25, RXC 0
[2017-03-16 09:59:00 538]: SEND : Freq: 922500000 DR: 0 PWR: 0
DataReport : a1b2c3
[2017-03-16 09:59:02 769]:RX1CH Open: freq 922500000, SF 12, BW 0, TO 25, RXC 0
[2017-03-16 09:59:03 783]:RX2CH Open: freq 921900000, SF 12, BW 0, TO 25, RXC 0
[2017-03-16 09:59:06 606]: SEND : Freq: 922300000 DR: 0 PWR: 0
DataReport : a1b2c3
[2017-03-16 09:59:08 837]:RX1CH Open: freq 922300000, SF 12, BW 0, TO 25, RXC 0
[2017-03-16 09:59:10 272]:OnRadioRxDone[2017-03-16 09:59:10 272]: MType: FRAME_TYPE_DATA_UNCONFIRMED_DC
OnRadioRxDone: ADR OFF, ADRACKReq 0[2017-03-16 09:59:10 272]: ACK 1 Fpending 0 FOptsLen 0 FCnt 4 P 0
[2017-03-16 09:59:10 288]:ProcessMacCommands: SRV_MAC_TIME_SYNC_ANS: S 1173661165 FS 61056
[2017-03-16 09:59:10 288]:time : 2017-03-16 09:59:08 931654839

```

Time Sync Answer Payload 데이터의 형태

	초 단위 데이터	초 단위 소수점 이하 데이터
Size (bytes)	4	2
DeviceTimeAns Payload	32-bit integer : Seconds since epoch*	fractional-second in $\frac{1}{2}^{16}$ second steps

(*) The GPS epoch (i.e Sunday January the 6th 1980 at midnight) is used as origin. The "seconds" field is the number of seconds elapsed since the origin. This field is monotonically increasing by 1 every second. To convert this field to UTC time, the leap seconds must be taken into account.
 Example : Friday 12th of february 2016 at 14:24:31 UTC corresponds to 483888 sec since GPS epoch. At this date the GPS time is 17seconds ahead of UTC time.

1. Second(4byte) 예시

- 현재 시간(A)=2016년12월27일 13시54분30초
- GPS 기준시간(B)=1980년1월6일 00시00분00초
- UTC 기준시간(C)=1970년1월1일 00시00분00초
- 윤초(D)=2016년까지는 17초, 2018년까지는 18초
- Second=(A-C)-(B-C)+D = 1166849687

2. Fractional-second(2byte) 예시

- 1초 이하의 값들을 표기하기 위한 값
- 0.000000000초~0.999999999초를 0~65535 분할하여 0~65535 사이의 값으로 표기
- ex) 0.839795528초 → (0.839795528 / 0.999999999)*65535 = 55036