USI® 環鴻科技股份有限公司
Universal Global Scientific Industrial Co., Ltd.

WM-N-BM-22

Lite Lora-gateway User Guide

Version: 1.0

Amendment Records

Item:	Date:	Revision:	Page:	Change Description:	Changed by:
1	7/1/2017	1.0	All	Initial release	Andy

Contents

1. Overview

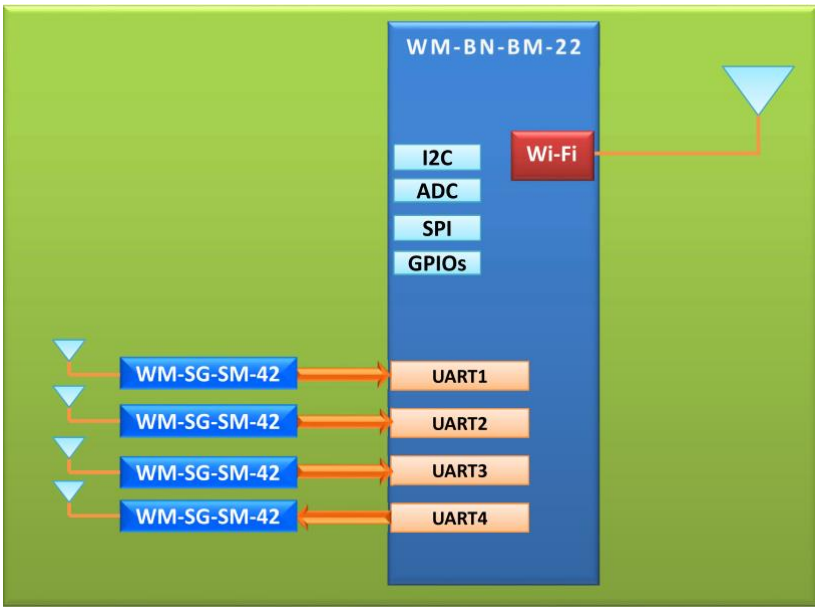
This document is for introduce the lite-gateway and how to set it up.

1.1 The features of the lite-gateway

- Packet forward for LoraWAN Class C compatible devices with ABP activation protocol
- EU868 & US915 Band
- Multiple I/O Function
  - SPI x 1
  - I2C x1
  - ADC x 3
  - PWM x 2
- Integrated with WiCED SDK

1.2 The hardware block diagram

The WM-N-BM-22 integrated IEEE 802.11b/g/n and BT4.1, it can connect 4 WM-SG-SM-42 Lora modules in maximum, and works as a network bridge for Wi-Fi and Lora. The gateway also provides multi IO function like I2C, SPI, GPIO and ADC which could be used for various IOT applications.



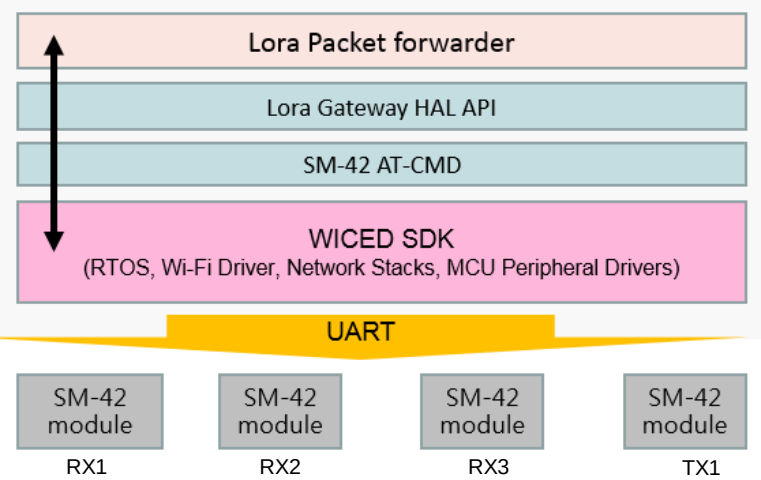
1.3 The software overview

The WM-BN-BM-22 provides rich software libraries based on the WiCED SDK for various IOT applications. There are 3 additional software parts for support the basic function for the lite Lora gateway.

SM-42 AT-CMD: this is used for control the four SM-42 modules through UART interface.

Lora Packet forwarder: this is used for forwarding nodes data to/from cloud server.

Lora Gateway HAL API: this is the Lora concentrator hardware abstraction layer API defined by Semtech, the purpose is for distinguish between the hardware driver (SM-42-AT-CMD) and software application (Lora Packet forward).



1.4 The software versions

The followings shows the required software version for this lora-gateway.

WM-N-BM-22:

- WICED SDK 4.x version above
- WM-N-BM-22 platform patch for lite Lora-gateway
- WM-N-BM-22 lit Lora-gateway Software Package v1.0

WM-SG-SM-42:

- WM-SG-SM-42 firmware version v3.0 above

2 Setup the lite Lora-gateway

2.1 Install the software packages

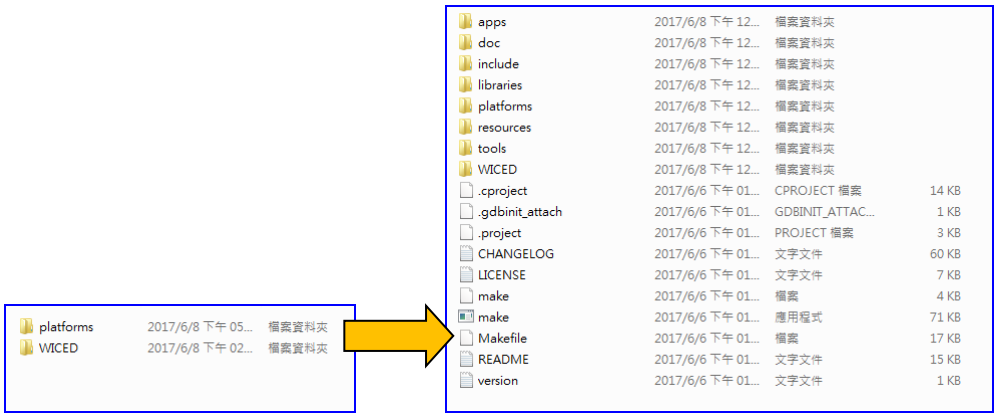
2.1.1 Install WiCED SDK

1. The lite gateway software package is developed with the WiCED SDK 4.x, please download WiCED SDK version 4.x from cypress web site and finish the installation.
https://community.cypress.com/community/wiced-wifi/wiced-wifi-documentation

2.1.2 Install the platform patch for WM-N-BM-22

1. Please get the WiCED platform patch for WM-N-BM-22 from the contact window of the product.
2. Unzip the platform patch package into the root path of WICED SDK as below:

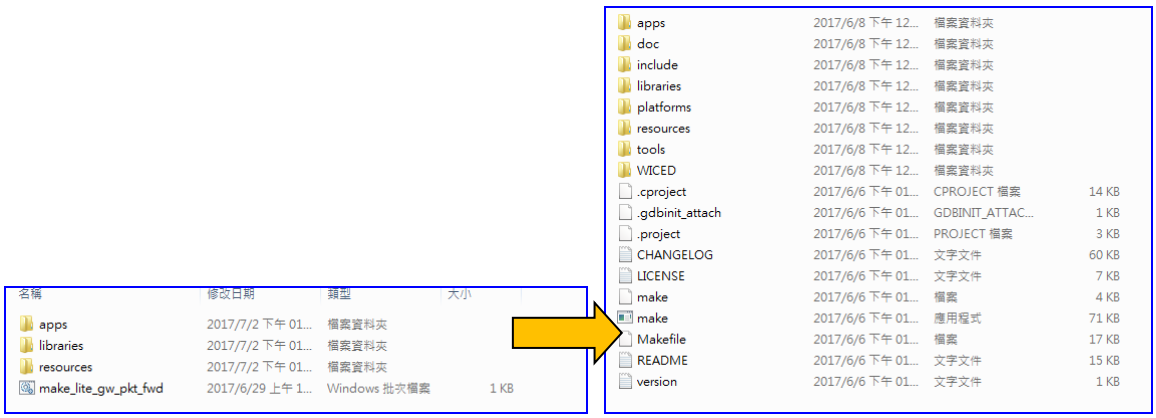
◆ $(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\



2.1.3 Install the lite Lora-gateway software package

1. Please download the latest lite-gateway software package from the from the GitHub link below:
https://github.com/USILoRaModule/USI_BM-22-Lite-LoRa-Gateway/blob/master/WM-N-BM-22_lite_lora-gateway_software_package_v1.0.zip

2. Unzip the platform patch package into the root path of WICED SDK as below:
$(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\



2.2 The configuration of packet forwarder

The related configuration about the packet forwarder was defined in the source files below, they also can be changed depended on the application requirements.

◆ $(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\apps\snip\lora_lite_gw_pkt_fwd\lite_gw__pkt_fwd.c
By the default, the lite Lora-gateway will forward Lora packets to TTN (the things network), the URL and UDP port was defined in this file as below, you can change them depended on the application requirements.

```
18
19 #define PKT_FWD_TARGET_URL  "router.eu.thethings.network"
20 #define PKT_FWD_DLINK_PORT   1700
21 #define PKT_FWD_ULINK_PORT   1700
22
23 #if 0
24 #define PKT_FWD_TARGET_URL  "192.168.11.7"
25 #define PKT_FWD_DLINK_PORT   1680
26 #define PKT_FWD_ULINK_PORT   1680
27 #endif
28
```

◆ $(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\libraries\lora_gateway\lib_legacy_pkt_fwd/legacy_pkt_fwd.h

The gateway ID is simply defined in a constant value as below, you can change it at here.

```
6
7
8  #define THE_GATEWAY_ID "\xFF\xFE\x02\x0A\xF7\x58\x20\x9D"
9
10 int init_legacy_pkt_fwd( wiced_interface_t intf,  char *host_ip
11 wiced_result_t pull_legacy_pkt( struct lgw_pkt_tx_s *lrtx_pkt )
12 int request_pull_legacy_pkt( void );
13 int push_legacy_pkt( struct lgw_pkt_rx_s *p );
14
15
```

2.3 Make & Download the lite Lora-gateway firmware into WM-N-BM-22

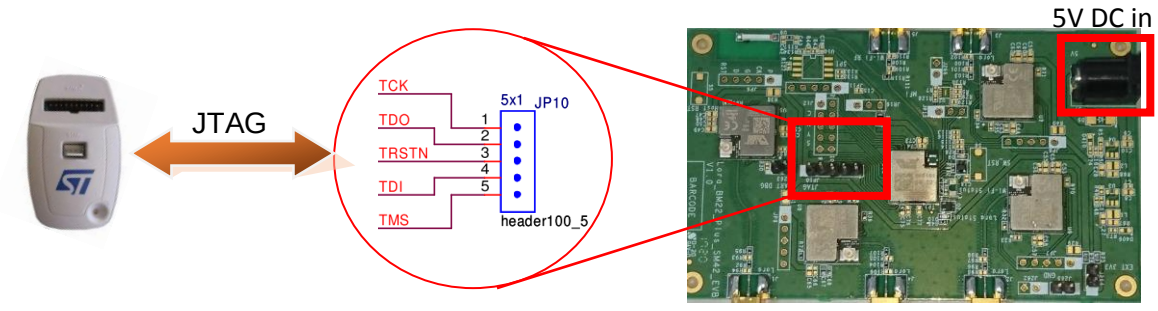
1. The WICED SDK build system uses OpenOCD + FTDI USB JTAG device for download the images on the WM-N-BM-22 EVB. Since the lite Lora-gateway just reserves a 5 pins JTAG interface for download the images, so the following shows how to modify the build system for download the images on the WM-N-BM-22 via ST-Link programmer.

a) Open file $(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\tools\makefiles\wiced_toolchain_common.mk in WICED Eclipse IDE or text editor.

b) Please comment out this line which contains the description 'JTAG ?= BCM9WCD1EVAL1' (near by line #38) and insert a line contains the description below:

```
32
33 TOOLS_ROOT ?= $(SOURCE_ROOT)tools
34
35 OPENOCD_PATH      := $(TOOLS_ROOT)/OpenOCD/
36 PATH :=
37
38 #JTAG          ?= BCM9WCD1EVAL1
39 JTAG          ?= stlink-v2-1
40
41
```

c) Connect ST-LINK programmer to the JP10 on the lite Lora-gateway as below:



2. Open a command window and change the working directory to the path below:
$(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\

3. Build all the code of lite Lora-gateway by using the command below:
make_lite_gw_pkt_fwd.bat

4. Download and run the lite Lora-gateway on WM-N-BM-22 by using the command below:
make_lite_gw_pkt_fwd.bat download run

2.4 Initial the Wi-Fi network and Lora transmitter and receivers

WICED SDK provides a device configuration code to demonstrating how to configure WICED device to join a Wi-Fi network via Soft AP and a web page, this Lora-gateway leverages the device configuration function to demonstrating the initial functions, the gateway will run in (or return to) this configuration mode at the following situations:

- The first time to run the lite Lora-gateway software on the BM-22
- Rerun make & download command without no_dct parameter

At the configuration mode, we can configure the BM-22 to join a Wi-Fi network for internet or intranet connections and configure the SM-42 modules in the best operation mode for the lite gateway.

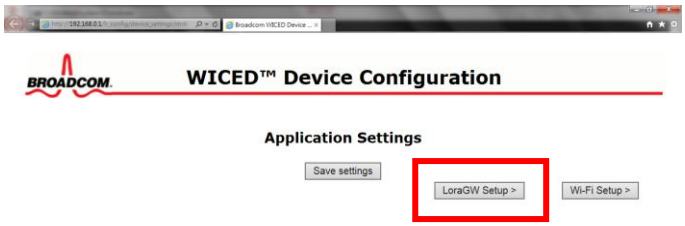
2.4.1 Connect to the soft AP and the setup webpage

1) Connect the device using Wi-Fi to the soft AP 'WICED_Config'
   The soft AP name and password is: 'WICED_Config' and '12345678'

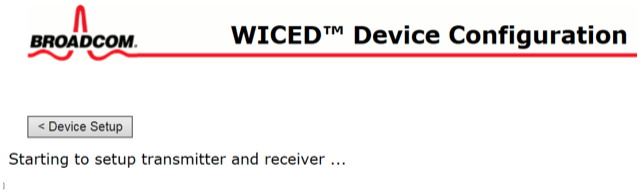
2) Open a web browser and enter '192.168.0.1' in the URL, and then the device configuration webpage appears. (192.168.0.1 is the IP address of the soft AP interface)

2.4.2 Initial Lora transmitter and receivers

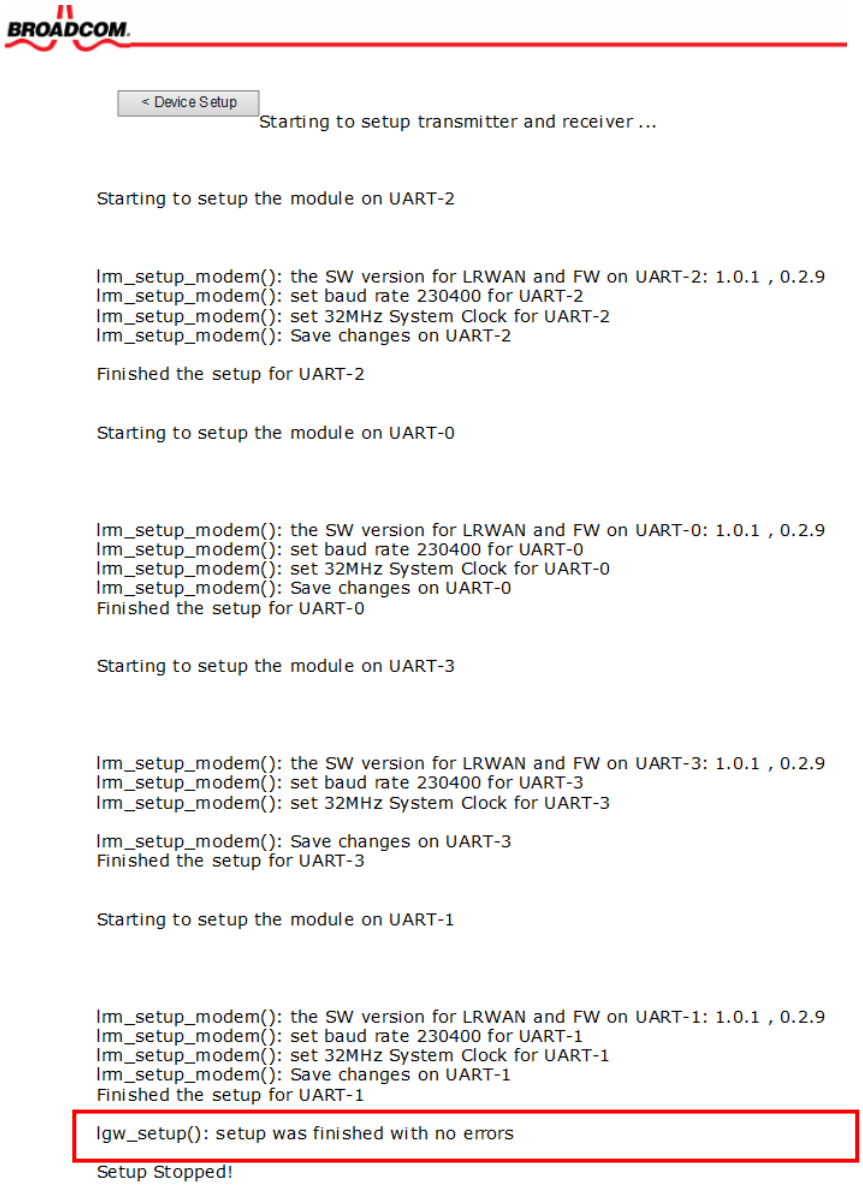
1) Click the LoraGW Setup button on this configuration home page.



2) A Lora Device Setup page appears and transmitter and receiver configuration is proceed as below.



3) Please wait for a while until the configuration completed, and you will get the log as below if setup completed with no any error.

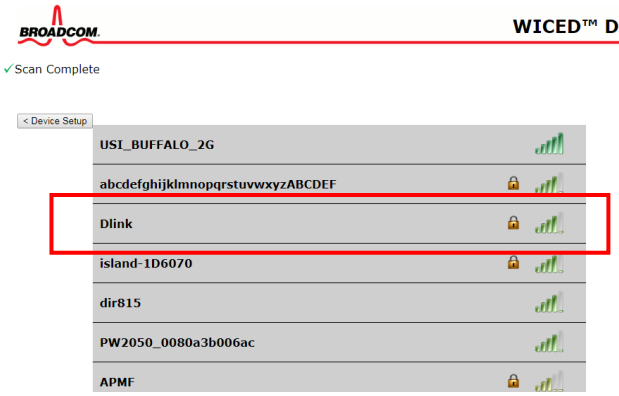


3) Click the device setup button for back to the device setup page (the configuration home page)

2.4.3 Setup Wi-Fi network

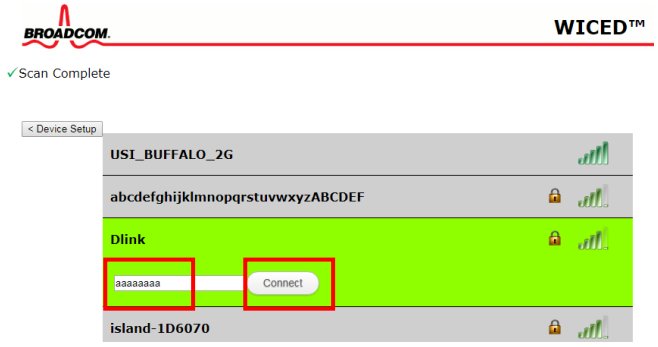
1) Click the Wi-Fi Setup button on the configuration home page.

2) A Wi-Fi Setup webpage appears and scanning the Wi-Fi networks in the range and then will show the scan result in a list.

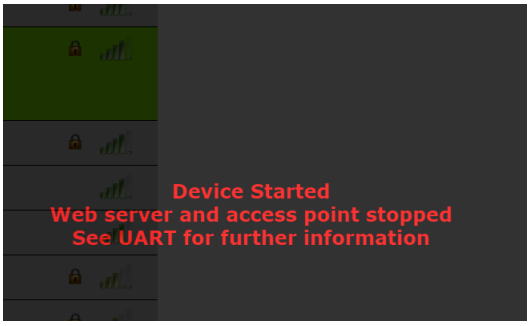


3) Find and click on a Wi-Fi AP which you want to join from the list and a password box will appears as below:

4) Enter the password for the Wi-Fi AP in the password box, and then click button 'Connect'



5) Wi-Fi setup is completed if you saw the screen as below, the Lora-Gateway will being to forwarding the Lora packets to/from the remote gateway (TTN) after a repower.



6) In order to avoid resetting these Wi-Fi setting in the image update later, please add a parameter 'no_dct' at the end of make command.

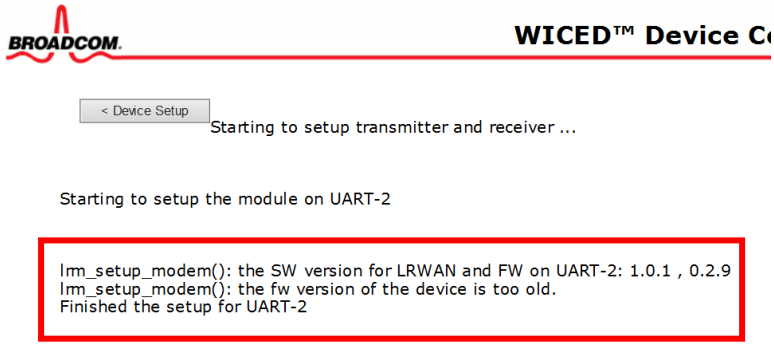
2.4.4 Update WM-SG-SM-42 preloaded software

The Lora-gateway can connect 4 SM-42 modules in maximum, and the preloaded software version must be 3.0 above. When you get an error message about the incorrect SM-42 firmware version when initial the Lora the transmitter/receivers (as below), please refer to the preloaded software update application note to complete the firmware update.

◆ The SM-42 preloaded software v3.0 link:
https://github.com/USILoRaModule/USI_I-NUCLEO-LRWAN1/blob/master/preloaded_firmware/wm-sg-sm-42_firmware_v3.0.hex

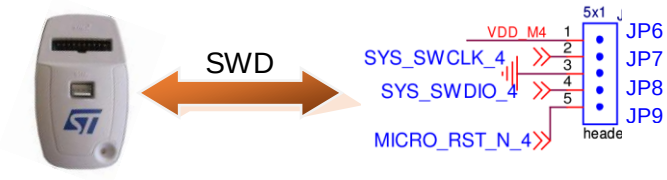
◆ The SM-42 preloaded software update manual link:

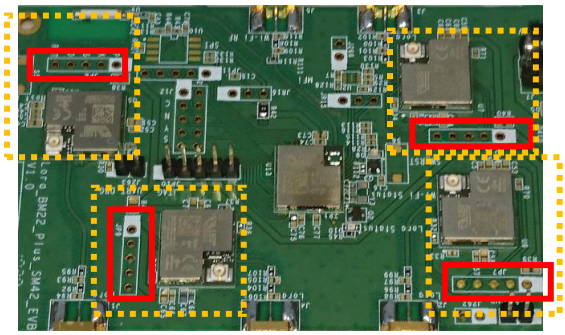
https://github.com/USILoRaModule/USI_I-NUCLEO-LRWAN1/blob/master/preloaded_firmware/WM-SG-SM-42%20Update%20Preloaded%20AT%20Command%20FW%20Application%20Note%20rev.%201.2.pdf



◆ The preloaded software package v3.0 and the update application note can be downloaded from the URL below:

◆ The SWD interfaces mapping to each SM-42 module is as below:





2.5 Test the lite Lora-gateway

2.5.1 Monitoring debug log

By the default, all the debug log from the gateway application will be forwarded to UDP port 50007, a python script can be used for show the debug log. The file is located at the root path of WICED SDK as below:
$(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\udp_receive.py

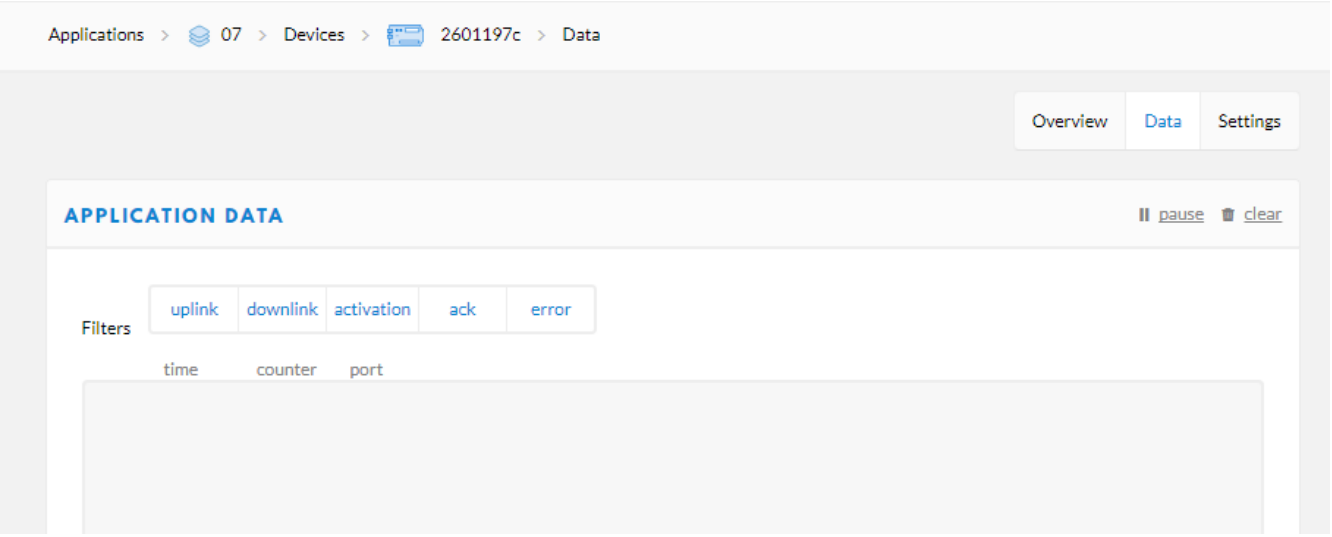
◆ The following is the snapshot of the debug log from the UDP port, the begin of each line is the device IP address and the line number of log

```
<x.x.x.9::00000038> resp_OK(): got 12th OK.
<x.x.x.9::00000039> ---lrm_init()
<x.x.x.9::0000003A> +++lrm_init()
<x.x.x.9::0000003B> +++atcmd_init_uart()
<x.x.x.9::0000003C> atcmd_init_uart(): use default configuration for UART-0.
<x.x.x.9::0000003D> atcmd_init_uart(): allocating memory for UART-1 RX.
<x.x.x.9::0000003E> atcmd_init_uart(): allocating memory for UART-0 TX.
<x.x.x.9::0000003F> ---atcmd_init_uart()
<x.x.x.9::00000040> atcmd_add_console(): allocating CMD buffer for UART-0
<x.x.x.9::00000041> atcmd_add_console(): uses default size of CMD buffer for UART-0
<x.x.x.9::00000042> atcmd_add_console(): uses default size for RESP buffer on UART-0
<x.x.x.9::00000043> lrm_set_op_modem(): set op mode at 9
```

2.5.2 Test Packet forwarding on TTN

The followings demonstrates how to forwarding packets to/from TTN using the lite Lora-gateway and a standalone SM-42 module.

1) Please register a virtual gateway and application on TTN and open the device data webpage for monitor the traffic between the lite gateway and the virtual gateway on the TTN.
   (There is an example of registration in section 4, you can refer this example to create the gateway and application on TTN for this test)



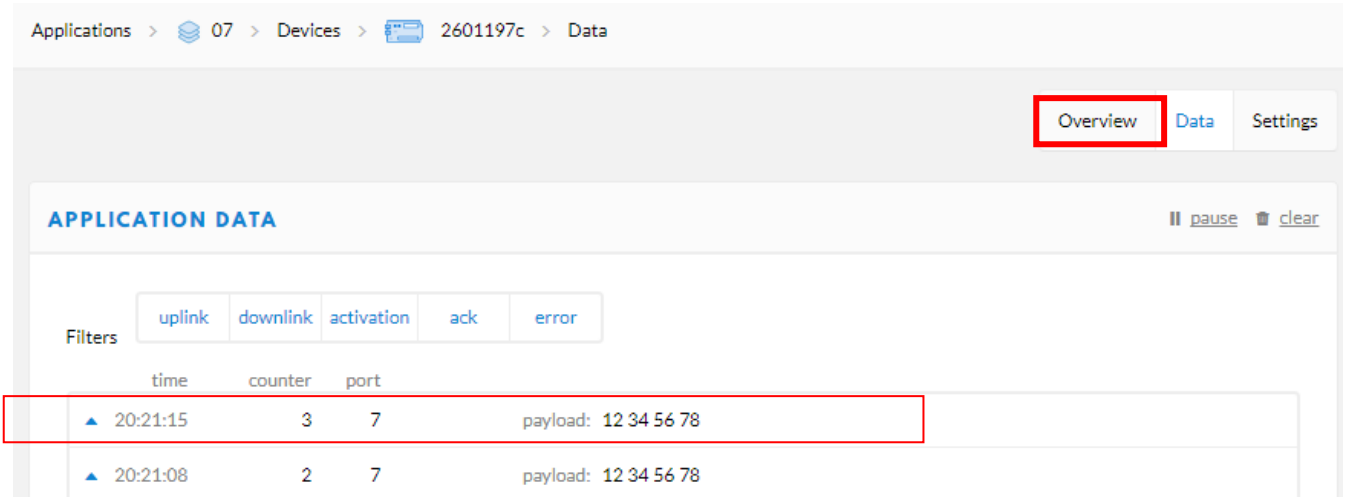
2) Join SM-42 module to the lite Lora-gateway using the command sequence below:

```
# ATZ
# AT+DR=3
# AT+RX2DR=3
# AT+CLASS=2
# AT+DC=0
# AT+JOIN=0
```

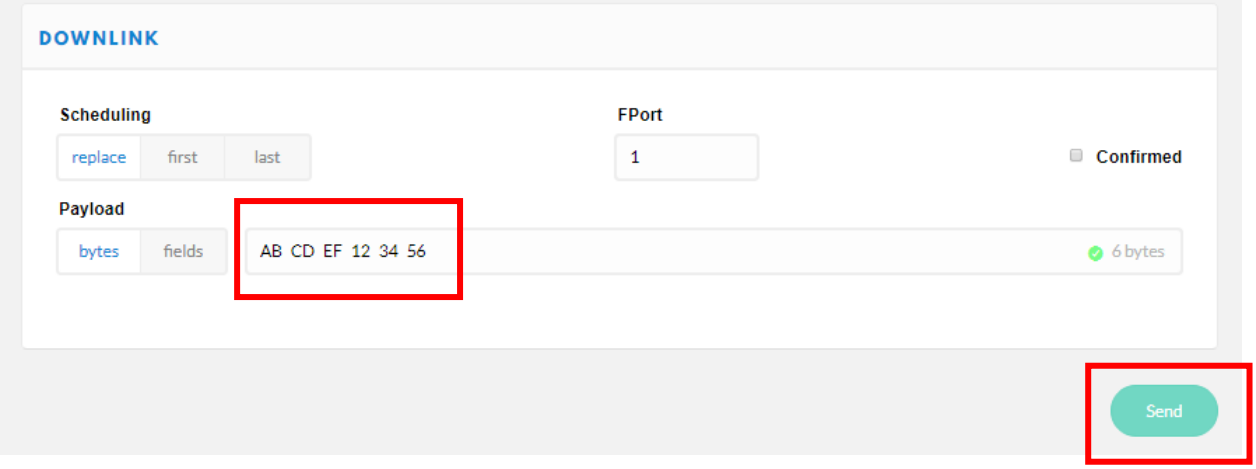
3) Transmit a packet from the SM-42 module to the application on TTN using the command below:

```
# AT+SEND=7 12345678 0
```

4) Then the packet from the SM-42 appears on the TTN device data webpage, it means the uplink is no problem. The next, please click the button 'Overview' for back to the device webpage.



5) Please fill the data by follow the red box below:



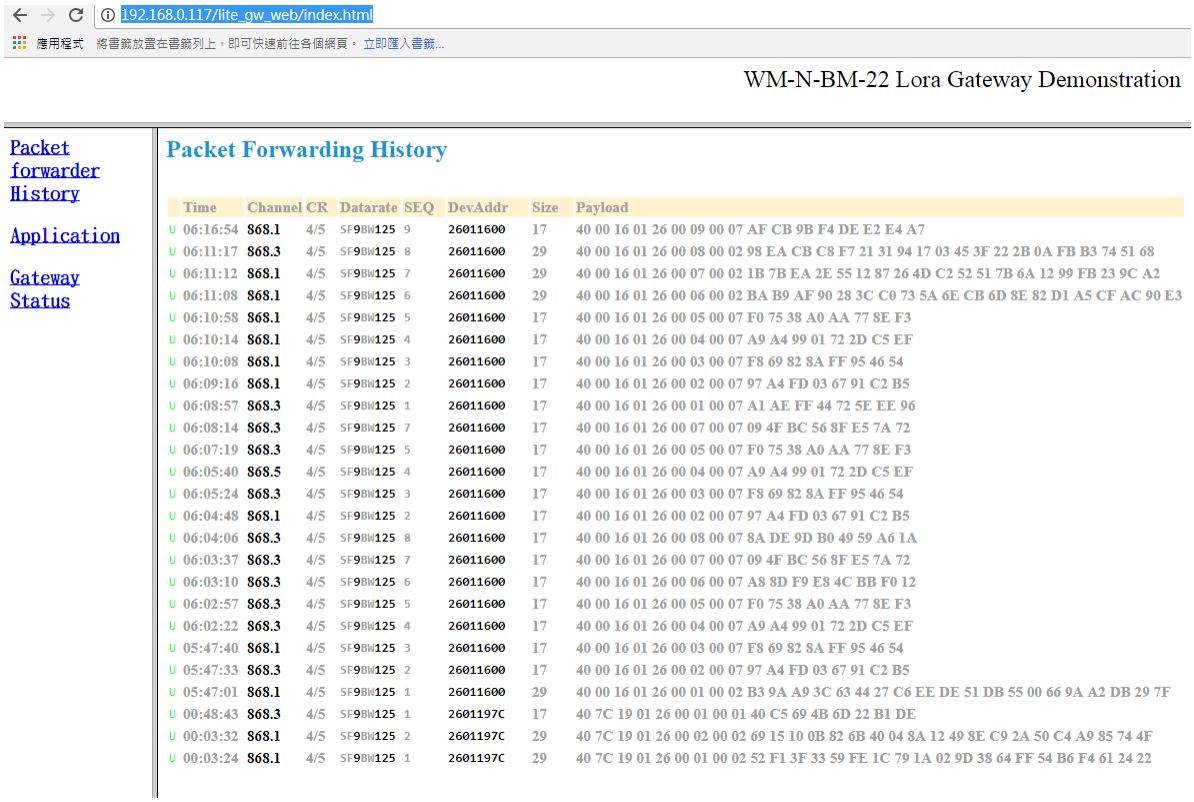
6) Transmit a packet from the SM-42 module to the application on TTN using the command below:

AT+SEND=7 12345678 0

7) And then SM-42 reports a +RCV event as below on the UART console, it means the downlink is no problem.
+RCV=1,6,ABCDEF123456

2.5.3 Packet forwarding history webpage

The lite Lora-gateway built-in a webpage for review the packet forwarding history, just enter the IP address of the Lora-gateway in the URL of web browser, you can see the packet history either downlink and uplink.



3. Configurations in the source code

3.1 Debug Log

By the default, the debug log from the gateway application can be forwarded to UDP port 50007, because all of the UART port was used for the TX and RX, and no extra UART port can be used for show debug log, a python script (named 'udp_receive.py') can be used for show the debug log from UDP port 50007. The file is located at the root path of WICED SDK as below:
$(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\libraries\lora_gateway\lib_log_tracer\udp_receive.py

3.1.1 Change Debug UDP port

You can change the Debug UDP port by modify the source code at the line no. 112 (around) in the file below, or comment out this line for disable to forwarding log on the UDP port.

- $(WICED_SDK_PATH)\43xxx_Wi-Fi\apps\snip\lora_lite_gw_pkt_fwd\lite_gw_pkt_fwd.c

```
103 #endif
104
105     /* Disable roaming to other access points */
106     wiced_wifi_set_roam_trigger( -99 ); /* -99dBm ie. extremely
107
108     /* Bringup the network interface */
109     wiced_network_up( WICED_STA_INTERFACE, WICED_USE_EXTERNAL_D
110
111     /* enable to forward debug log on UDP port */
112     log_trace_enable_udp( WICED_STA_INTERFACE, 50007 );
113
114     /* not supported yet */
115 #if defined( LOCAL_GW_APPLICATIONS )
116     /* starting up the gateway portal on the device */
117     start_lite_gw_web();
```

Once you changed the UDP port, you must also change the UDP port in the python file below:

```
1 import sys
2 import socket
3
4 address = ('', 50007)
5
6
7 s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
8
9 s.bind(address)
10
11 while True:
12     data,addr = s.recvfrom(1024)
13     try:
14         sys.stdout.write(data.decode('utf-8'))
15     except:
16         print (" ")
17
18
19 s.close()
```

3.1.2 Enable Debug UART Port

By the default, the Debug UART port is disabled, if you want to enable it, the following three steps need to be completed.

1) Please open line no. 17 and comment out line no. 18 in the make file below:

◆ $(WICED_SDK_PATH)\43xxx_Wi-Fi\apps\snip\lora_lite_gw_pkt_fwd\lora_lite_gw_pkt_fwd.mk

```
15
16
17 #GLOBAL_DEFINES      := USE_SELF_SIGNED_TLS_CERT
18 GLOBAL_DEFINES      := USE_SELF_SIGNED_TLS_CERT WICED_DISABLE_STDIO
19
20
21 VALID_PLATFORMS     := BCMUSI22
22
```

2) WICED SDK uses UART1 as the debug port, since UART1 is connected to RF chain 2, so we need to disable it by comment out the source code at line no. 92 in the file below:

```
83  void application_start( void )
84  {
85      int nb_pkt;
86      struct lgw_pkt_tx_s txpkt; /* array containing
87
88      wiced_init( );
89
90      /* register RF chain (SM-42 modules) */
91      slgw_add_rf_chain( &receiver1_conf );
92      slgw_add_rf_chain( &receiver2_conf );
93      slgw_add_rf_chain( &receiver3_conf );
94      slgw_add_rf_chain( &transmitter_conf );
95
96      /* Configure the device */
97      wiced_configure_device( NULL );
98
```

3) Rebuild and download the application on WM-N-BM-22 with no_dct parameter.
make_lite_gw_pkt_fwd.bat download run no_dct

3.2 The configuration of transmitter and receivers

By the default, the lite Lora-gateway uses 3 SM-42 modules for RX function and uses 1 SM-42 module for TX function, the default configuration is compatible with EU868 band, the following shows the detail configuration on the TX and RX:

Function	Frequency (MHz)	LoraWAN Data Rate	Spreading Factor	Bandwidth	Code Rate	UART Port
RX1	868.1	DR3	SF9	125K	4/5	UART3
RX2	868.3	DR3	SF9	125K	4/5	UART1
RX3	868.5	DR3	SF9	125K	4/5	UART4
TX1	869.5	DR3	SF9	125K	4/5	UART2

The corresponding configuration for the TX and RX was defined in the source files below, they can be changed depended on the application requirements.

◆ $(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\apps\snip\lora_lite_gw_pkt_fwd/rf_chain_conf.h
The configurations for RF frequency, UART connection and TX/RX assignment was defined in this file as below.

```
/*
 * the RF receiver configuration for 868.3Mhz
 * you can override the configuration by follow the examples in the previous 'lrm_conf_t receiver1_conf'
 */
static lrm_conf_t receiver2_conf = {

        /* console configuration (use for AT-CMD communications)*/
        .cmd_console = {
                .if_conf = {
                        .port_id = (int)WICED_UART_1,  /* the UART port connected to the receiver (SM-42) */
                },
        },

        /* general RF chain configuration */
        .rf_conf = {
                .enable = 1,
                .freq_hz = 868300000,    // the frequency that we want the module listens
                .rssi_offset = 0,
                .tx_enable = 0,          // use the module as a receiver
        },

        .sync_gpio = -1,   // no need to sync the TX/RX timing at this moment. */
};
```

◆ $(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\libraries\lora_gateway\include/lrm_lib.h

The configurations for RF data rate, code rate, bandwidth and output power for TX and RX was defined in this file as below.

```
/* the default module configuration for the gateway receivers */
#if 0

#define DEFAULT_RECEIVER_OP_MODE       LRM_RX_MODE   /* the modem's operation mode  */
#define DEFAULT_RECEIVER_DR            (DR3 << 24) | (LRM_SF_9 << 16) | (LRM_BW_125K << 8) | LRM_CR_4_5  /* sir
#define DEFAULT_RECEIVER_POWER         20            /* it is the TX power if used to transmit data  */
#define DEFAULT_RECEIVER_IQ_INVERTED   0             /* it is the TX power if used to transmit data  */

#else

#define DEFAULT_RECEIVER_OP_MODE       LRM_FGW_MODE  /* the modem's operation mode  */
#define DEFAULT_RECEIVER_DR            (DR3 << 24) | (LRM_SF_9 << 16) | (LRM_BW_125K << 8) | LRM_CR_4_5
#define DEFAULT_RECEIVER_JOIN_TYPE     0             /* APB join mode */
#define DEFAULT_RECEIVER_CLASS         2             /* LoraWAN class-C */
#define DEFAULT_RECEIVER_BAND          0             /* EU868 */
#define DEFAULT_RECEIVER_DC            0             /* disable ducty-cycle */
#define DEFAULT_RECEIVER_RX2DR         3             /* the data rate of RX2: DR3 (SF9, BW125) */
#define DEFAULT_RECEIVER_RX2DT         2000          /* the delay ms of rx2 window: 2000ms */
#define DEFAULT_RECEIVER_RX1DT         1000          /* the delay ms of rx1 window: 1000ms (used for join) */
#define DEFAULT_RECEIVER_POWER         20            /* it is the TX power for the TX RF chain */
#define DEFAULT_RECEIVER_IQ_INVERTED   1             /* LoraWAN sends packet with inverted-iq */

#endif

/* the default module configuration for the gateway transmitters */
#define DEFAULT_TRANSMITTER_OP_MODE       LRM_WLAN_MODE   /* the modem's operation mode  */
#define DEFAULT_TRANSMITTER_DR            (DR3 << 24) | (LRM_SF_9 << 16) | (LRM_BW_125K << 8) | LRM_CR_4_5
#define DEFAULT_TRANSMITTER_JOIN_TYPE     0         /* APB join mode */
#define DEFAULT_TRANSMITTER_CLASS         2         /* LoraWAN class-C */
#define DEFAULT_TRANSMITTER_BAND          0         /* EU868 */
#define DEFAULT_TRANSMITTER_DC            0         /* disable ducty-cycle */
#define DEFAULT_TRANSMITTER_RX2DR         3         /* the data rate of RX2: DR3 (SF9, BW125) */
#define DEFAULT_TRANSMITTER_RX2DT         2000      /* the delay ms of rx2 window: 2000ms */
#define DEFAULT_TRANSMITTER_RX1DT         1000      /* the delay ms of rx1 window: 1000ms (used for join) */
#define DEFAULT_TRANSMITTER_POWER         20        /* it is the TX power for the TX RF chain */
#define DEFAULT_TRANSMITTER_IQ_INVERTED   1         /* LoraWAN sends packet with inverted-iq */
```

4. Register gateway & application on TTN

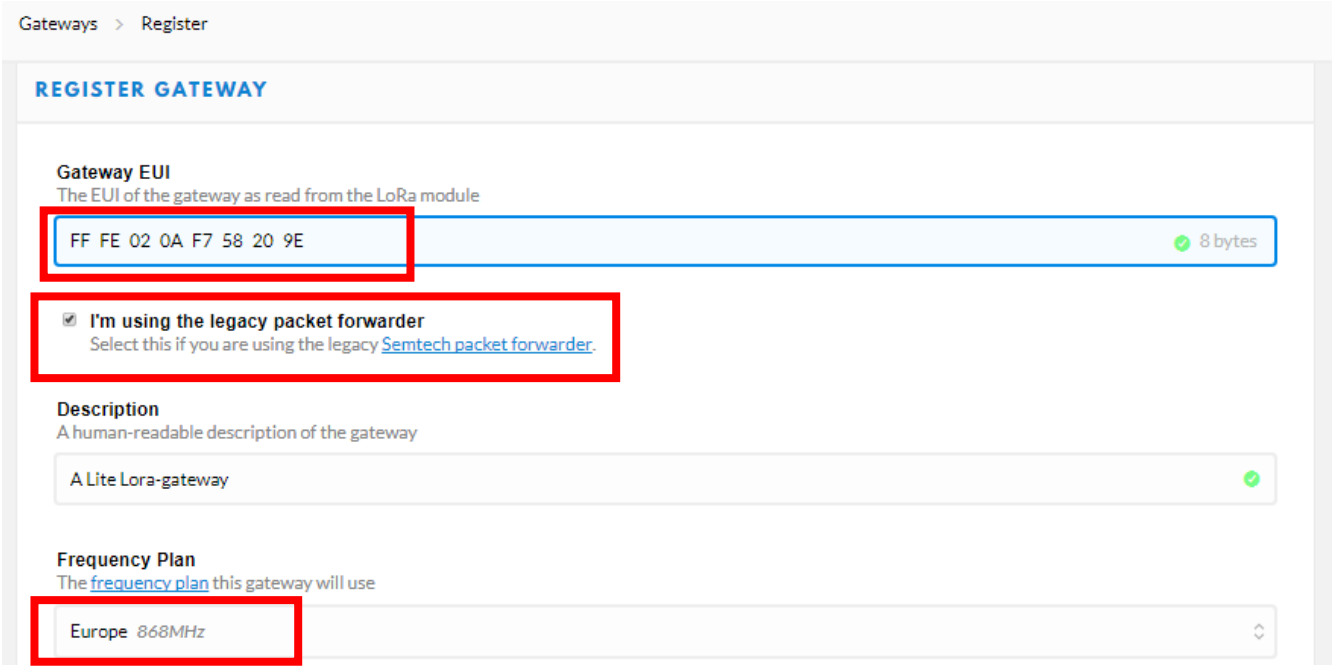
The followings provides an example to register a virtual gateway and application on TTN.

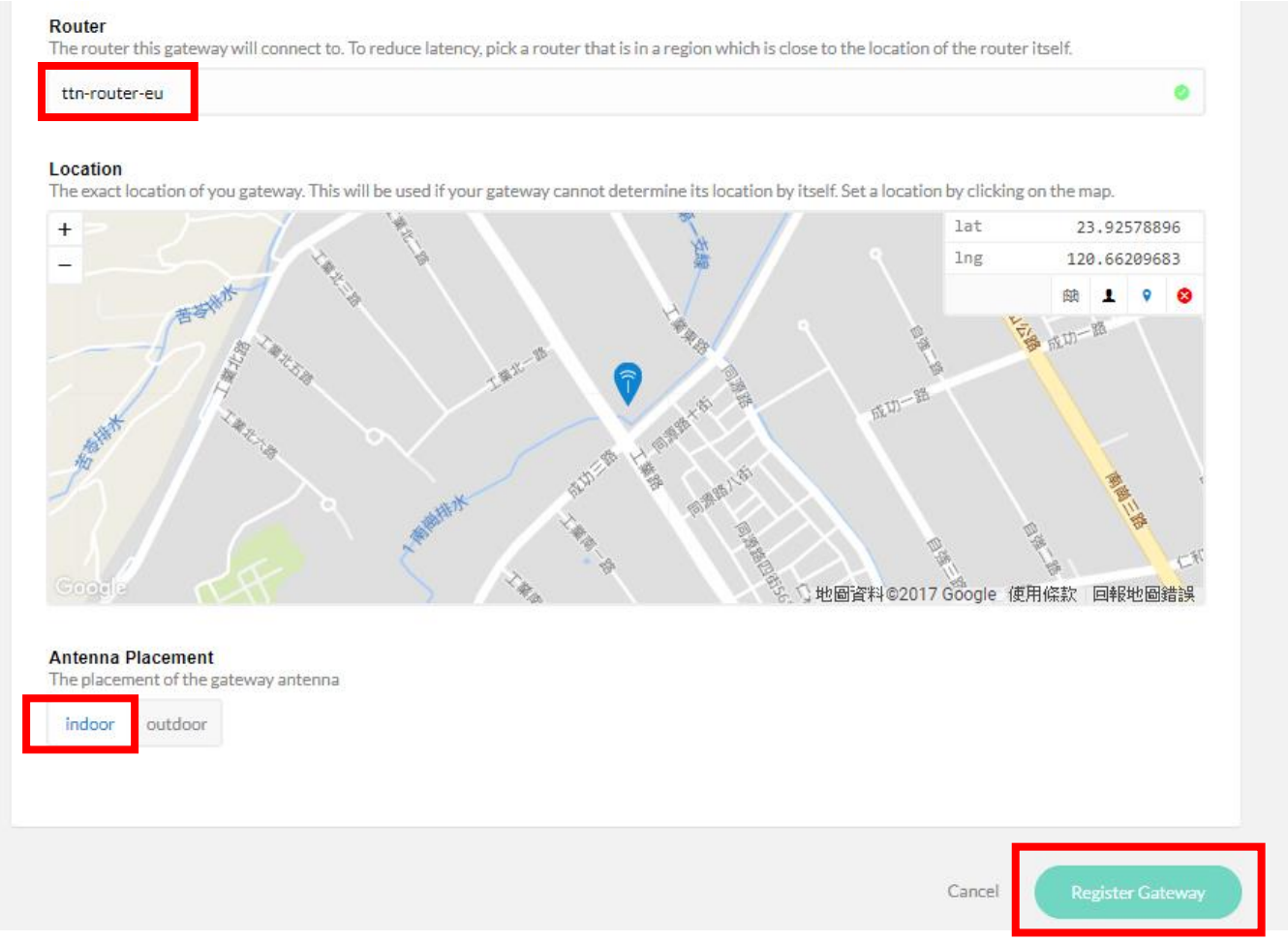
1) In order to avoid collision with other gateway ID, please change the gateway ID defined in the file below, the new gateway ID will be used to registering a gateway on TTN:

- $(WICED_SDK_ROOT_PATH)\43xxx_Wi-Fi\libraries\lora_gateway\lib_legacy_pkt_fwd/legacy_pkt_fwd.h

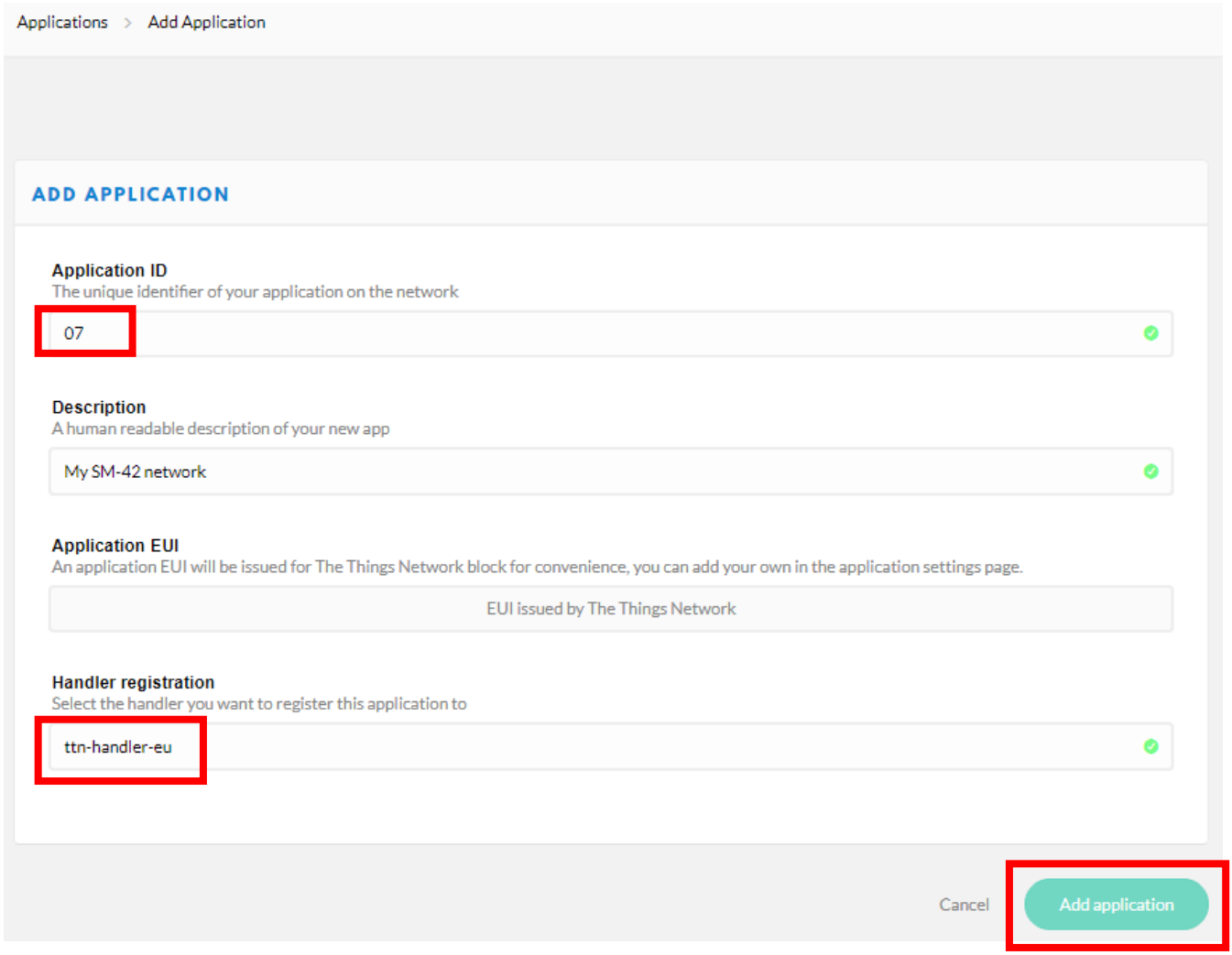
```
#define __LEGACY_PKT_FWD_H__

#include "loragw_hal.h"

#define THE_GATEWAY_ID "\xFF\xFE\x02\x0A\xF7\x58\x20\x9E"

int init_legacy_pkt_fwd( wiced_interface_t intf, char *host_i
wiced_result_t pull_legacy_pkt( struct lgw_pkt_tx_s *lrtx_pkt
int request_pull_legacy_pkt( void );
int push_legacy_pkt( struct lgw_pkt_rx_s *p );
```

2) Sign up a TTN account then go to the gateway register webpage for register a gateway, and please fill the data by follow the red box below:



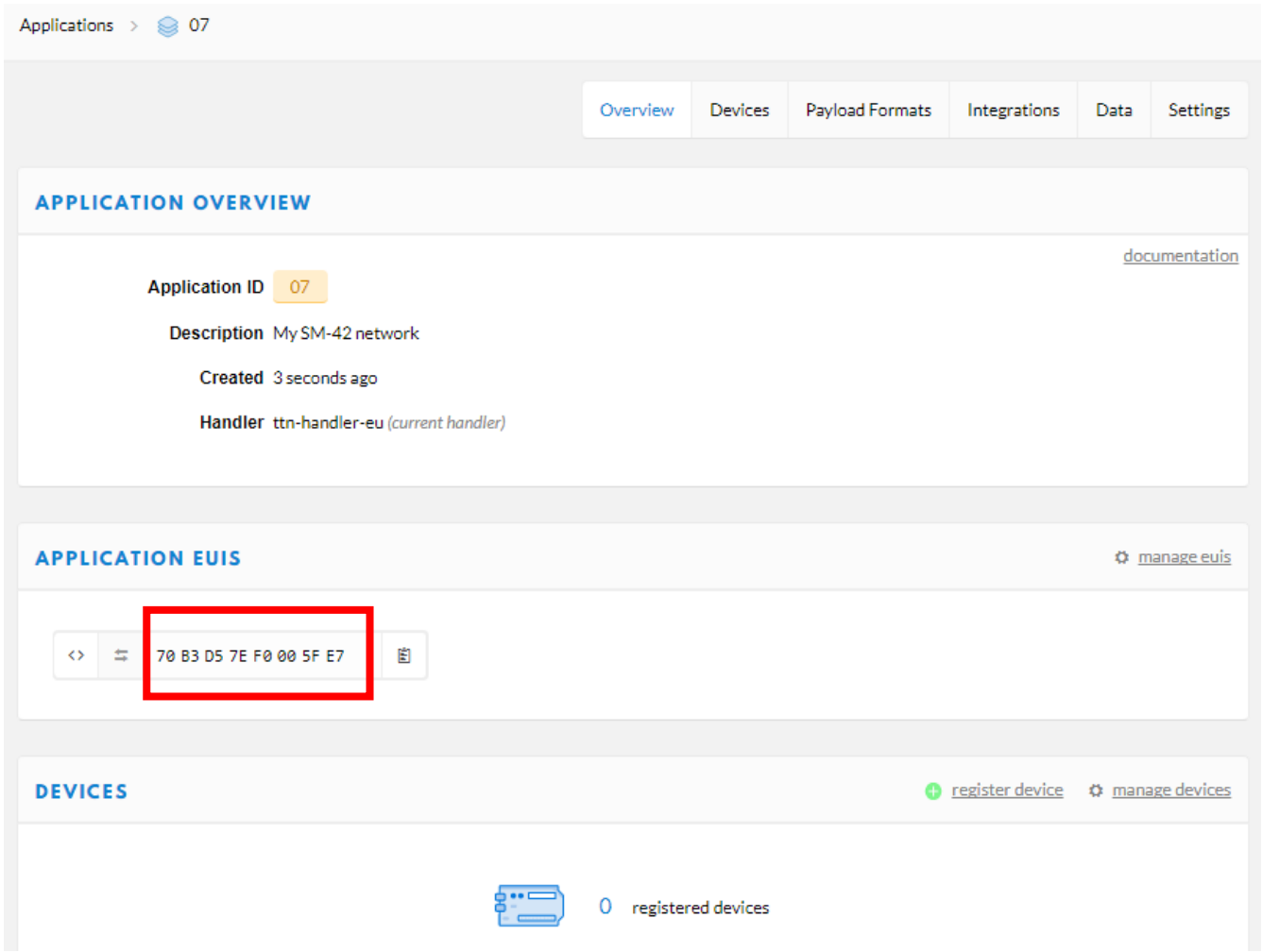


3) Go to the application webpage for register an application, and please fill the data by follow the red box below:

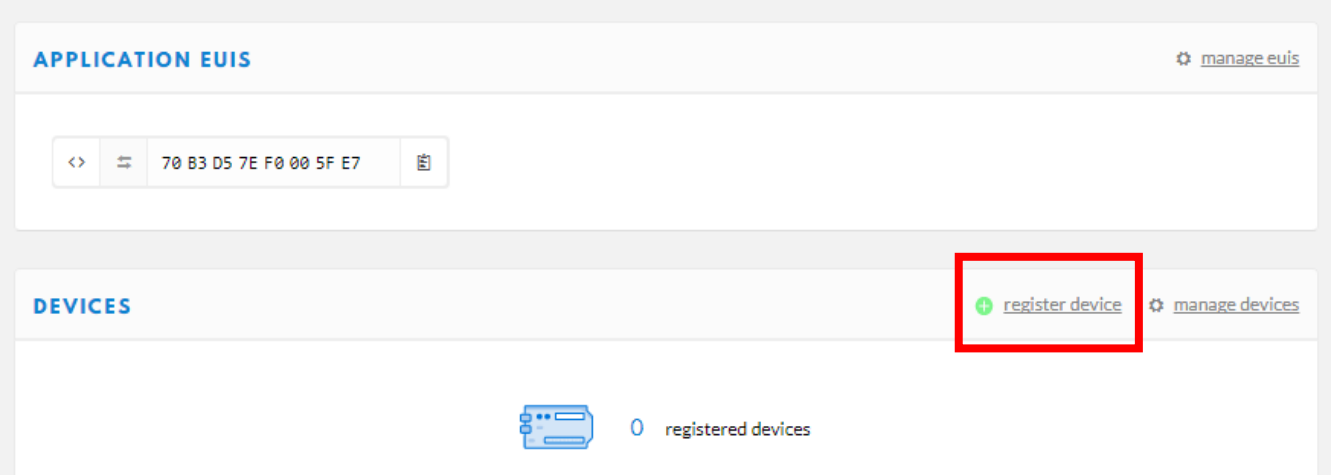


4) Then we will go to the new application webpage as below, we need to set the application EUI (shown in the red box below) in the SM-42 module by the AT command below:

AT+APPEUI=70B3D57EF0005FE7

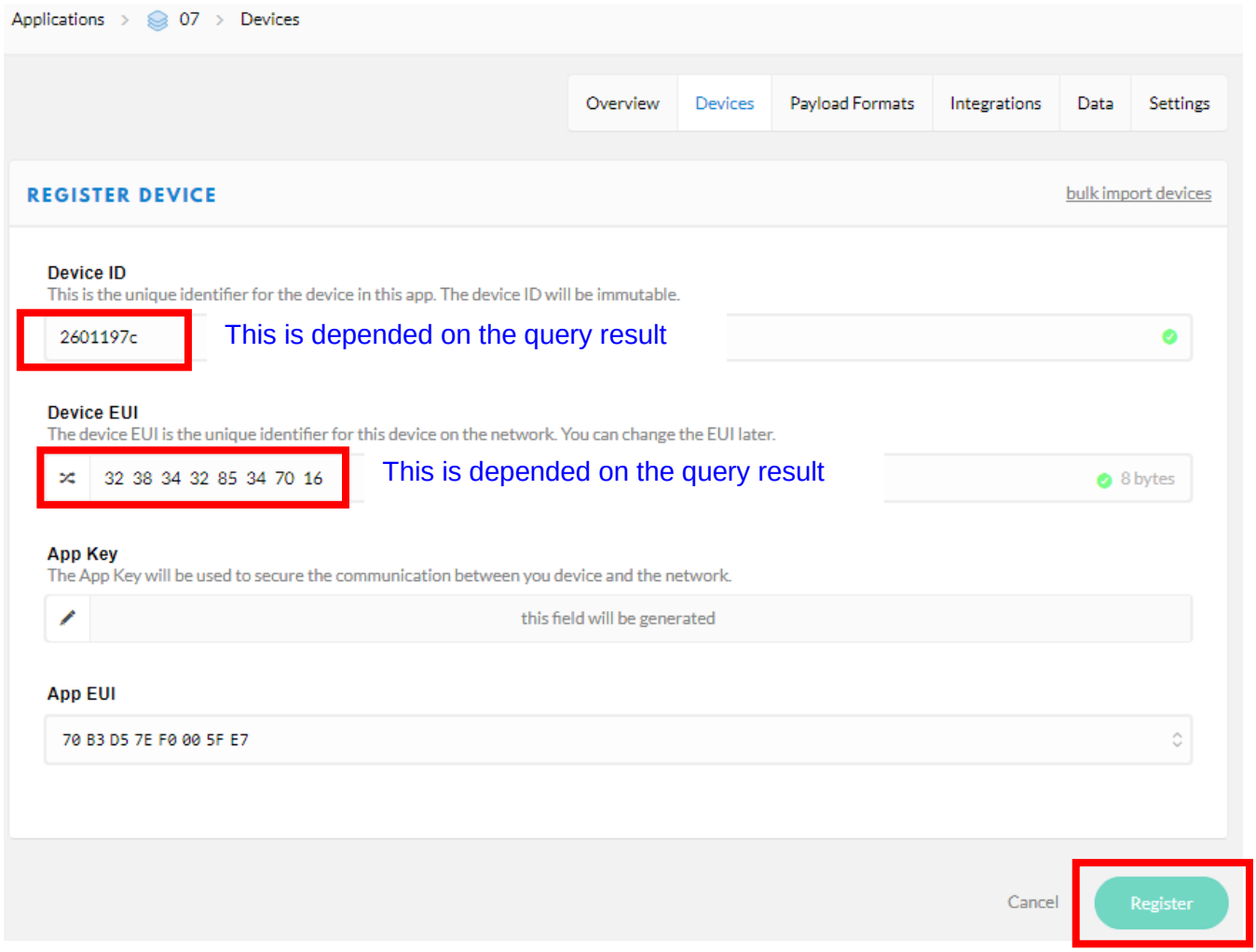


4) After finish the application EUI setting, we need to join the SM-42 module in this application, please click the button 'register device' to starting the registration.

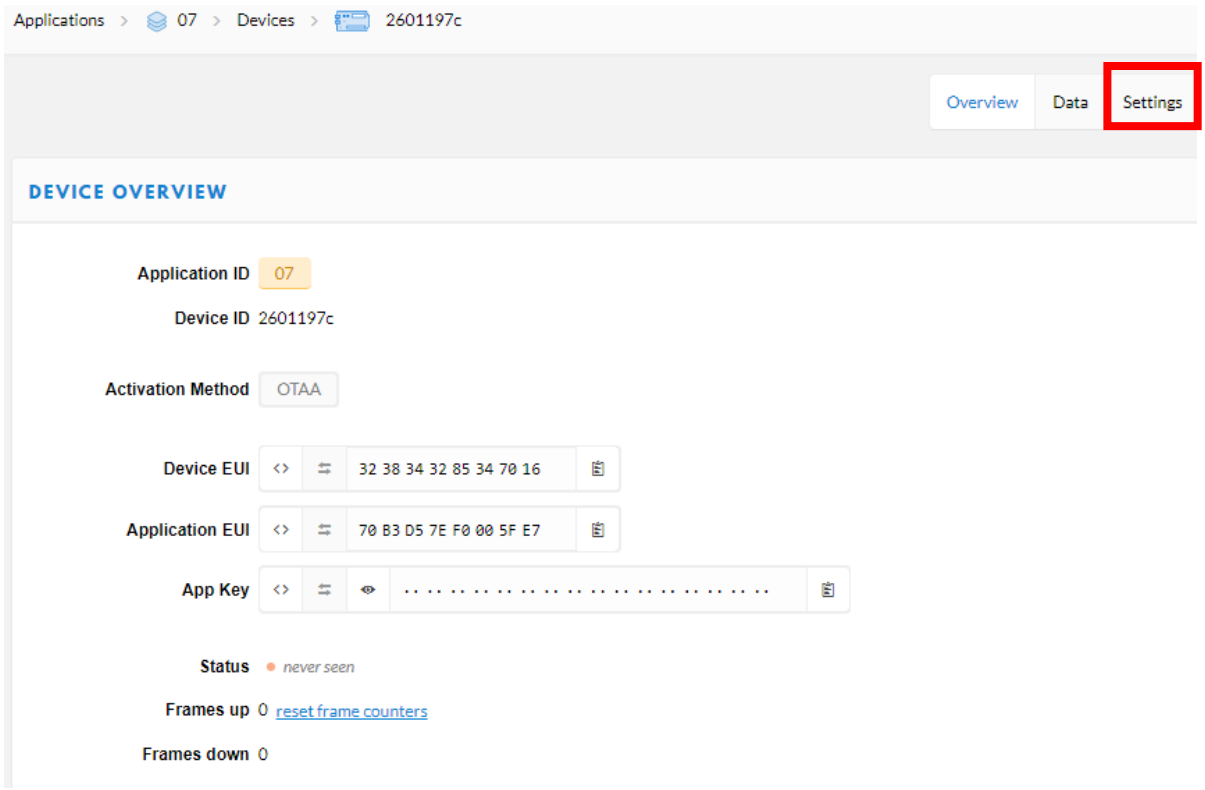


4) Then a register device webpage appears, please also fill the data by follow the red box below, the device ID and device EUI in the SM-42 module can be queried by the command below:

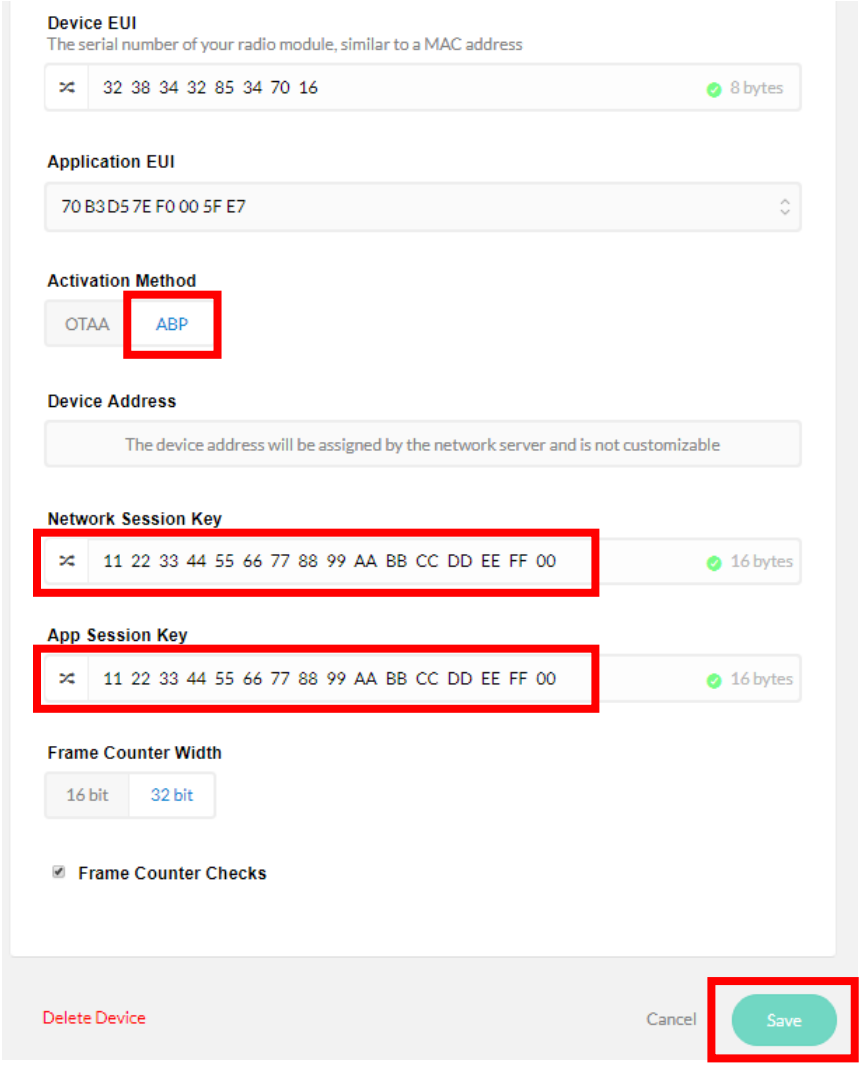
AT+ADDR
AT+EUI



5) After finish the device registration then we will go to the new device webpage, since the lite Lora-gateway only support the ABP activation protocol, please click the button 'Settings' to changing some settings.



6) Please fill the settings by follow the red box below:



6) After finished the settings, we will back to the new device webpage, a new device address was generated on this webpage, and we must set it in the SM-42 module by the command below:

AT+ADDR=26011600