SAMR34 Getting Started LoRaWAN Range Test [Button] Quick Start Guide

Introduction

This document is intended as a guide to help you understand the available features, and to describe how to use the example project: **SAMR34 Getting Started LoRaWAN Range Test [Button]**

Table of Contents

Contents

1. Overview

The "**SAMR34 Getting Started LoRaWAN Range Test [Button]"** example application is an ASF 3 based project that is intended to be used with the SAMR34 XPRO evaluation kit.
The goal of this application is to allow a user to create a LoRaWAN end node application that supports the following capabilities:

- Configuration of parameters required to join a LoRawan Network (thru #defines at build time)
- Selection of the operational region thru a console menu at power up or reset.
- Report operational configurations thru the console serial connection. (EDBG connector)
- Periodically sends temperature sensor data to the networ,k using unconfirmed transmission frames
- Allows a manual transmission of temperature sensor data using confirmed transmission frames by way of a short press of the USER SW0 pushbutton, available on the SAMR34 XPRO eval board.
- Allows modifying the selected Data Rate, used for transmission, by a Long Press of the SW0 button.
- Allows modifying of the selected TX Power Index, used for transmission, by an Extended Long press of the SW0 button.
- Green and Amber leds to signify various operational states, and parameter settings.
- Provides serial console information, thru the EDBG usb serial connection, available on the SAMR34 XPRO evaluation board.
- Implements Standby sleep mode after a network join, and in-between periodic or manual sensor transmissions.

1.1 Supported hardware platforms and IDEs

Microcontroller	Supported Evaluation Kit	Supported IDE's
SAMR34J18B(SIP)	SAMR34 Xplained PRO [DM320111]	Atmel Studio 7

2. Development environment

This section provides information on the required tools needed to setup and build the example project , and the platform to run it on.

2.1. Atmel Studio 7

Atmel Studio 7 can be used to develop and debug applications for MIcrochip ARM-based platforms. Atmel Studio 7 is equipped with the GCC compiler, and does not require any additional external software tools to compile and debug SAMR34 LoRaWAN applications.

Below are a few reference links to documents that will help you get started with Studio 7.
https://www.microchip.com/mplab/avr-support/atmel-studio-7

http://ww1.microchip.com/downloads/en/DeviceDoc/Getting-Started-with-Atmel-Studio7.pdf

http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-Studio-7-User-Guide.pdf

Here are additional links to detailed documents to help you understand the SAMR34/35 LoRaWAN stack

http://ww1.microchip.com/downloads/en/DeviceDoc/SAM-R34-R35-Microchip-LoRaWAN-Stack-Software-API-Reference-Manual-DS70005382A.pdf

http://ww1.microchip.com/downloads/en/DeviceDoc/SAM-R34-MLS-Getting-Started-Guide-User-Guide-DS50002812A.pdf

2.2. SAMR34 Xplained Pro [DM320111]

Full information regarding the SAMR34 Xplained Pro evaluation kit can be found at the links provided:

https://www.microchip.com/DevelopmentTools/ProductDetails/DM320111

https://www.microchip.com/wwwproducts/en/ATSAMR34J18

http://ww1.microchip.com/downloads/en/DeviceDoc/ATSAMR34-Xplained-Pro-User-Guide-DS50002803A.pdf

3. Stack Configurations

This project takes advantage of configuration .h files to define and configure the behavior of the stack.
The configuration file that is important to the initial operation of this example project is:

- \src\config\conf_app.h

Within the file conf_app.h you must initially configure the defines listed below:
The data for each of these defines must be modified to reflect the EUI's and Keys required by your selected Network Server.

```
/*Define the Sub band (where required) of Channels to be enabled by default for the application*/
#define SUBBAND 2

/* OTAA Join Parameters */
#define DEMO_DEVICE_EUI           { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xFA, 0xFC}
#define DEMO_APPLICATION_EUI      { 0x70, 0xB3, 0x00, 0x00, 0xD0, 0x00, 0x00, 0x00}
#define DEMO_APPLICATION_KEY      { 0x00, 0x00, 0x00, 0x00, 0xE7, 0xB0, 0x83, 0xBD, 0x79,0x5E, 0xF4, 0xB4,
                                  0x00, 0x00, 0x00, 0x00}


/* ABP Join Parameters */
#define DEMO_DEVICE_ADDRESS               0xdeafface
#define DEMO_APPLICATION_SESSION_KEY      {0x00, 0x00, 0x74, 0x69, 0x6C, 0x69, 0x74, 0x79, 0x00, 0x04,
                                          0xA3, 0x0B, 0x00, 0x04, 0xA3, 0x0B}
#define DEMO_NETWORK_SESSION_KEY          {0x00, 0x00, 0x00, 0x69, 0x6C, 0x69, 0x74, 0x79, 0x00, 0x04,
                                          0xA3, 0x0B, 0x00, 0x04, 0x00, 0x00}
```

By default, the example uses OTAA Join and *Unconfirmed* periodic transmissions, at an interval of 30 seconds.

Each SAMR34 XPRO board comes pre programmed with a unique device EUI. If during your development cycle you would like to use your own Device EUI, you can accomplish this by modifying a project setting as described below.

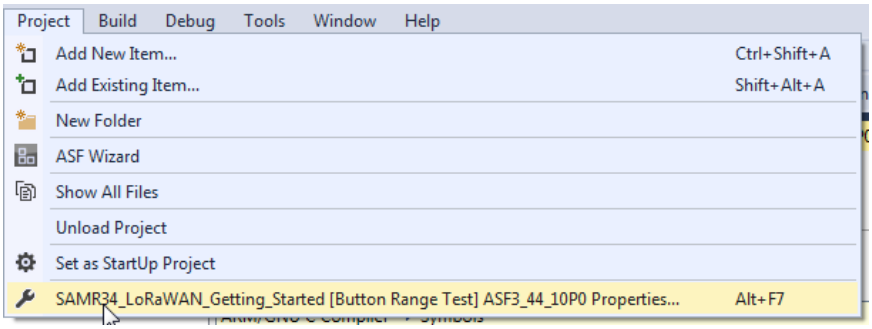
The DEMO APPLICATION EUI and DEMO APPLICATION KEY are provided by the Network Service you are attempting to work with . For Example, For TTN (The Things Network) these parameters are provided thru the TTN console.

4. Building applications in Atmel Studio

Atmel Studio 7 is used to develop and build SAMR34 applications.
To configure or review the project settings before building your project, follow the instructions below.

4.1. Setting Project / Properties

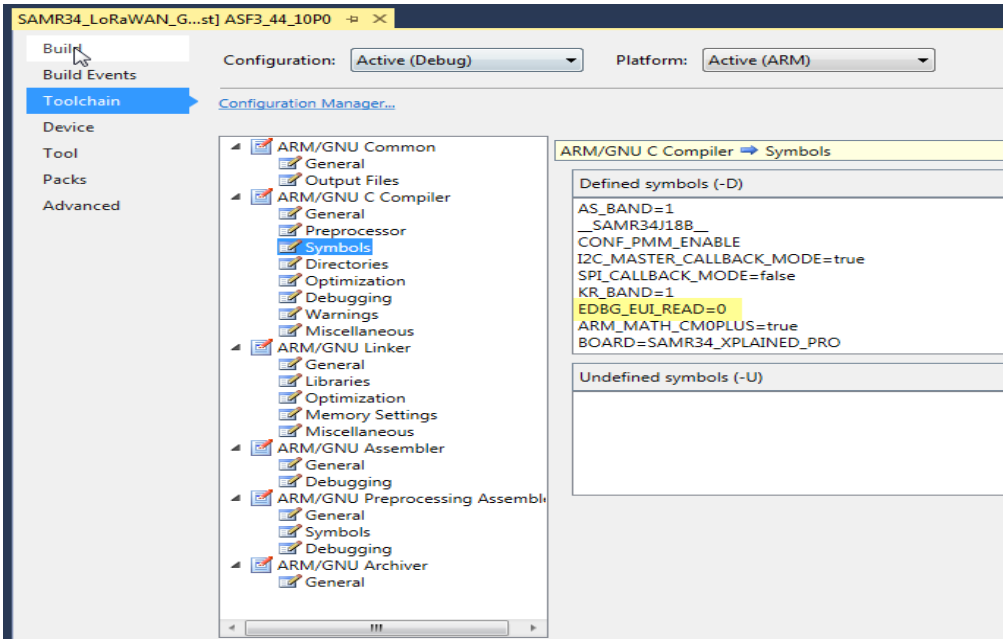
Project settings and properties can be configured thru the Project/Properties menu Tabs



To select which device EUI you will be using, select the Toolchain item under project properties.
Then select Symbols in the list of ARM/GNU C compiler options as shown below.

In the list of Defined symbols shown , edit the symbol EDBG_EUI_READ to either
EDBG_EUI_READ=0 to use your own device EUI entered in file conf_app.h or
EDBG_EUI_READ=1 to use the pre-programmed Microchip assigned device EUI that is pre-programmed into the SAMR34 Xplained PRO evaluation board.

You will see a Properties page similar to what is shown below:

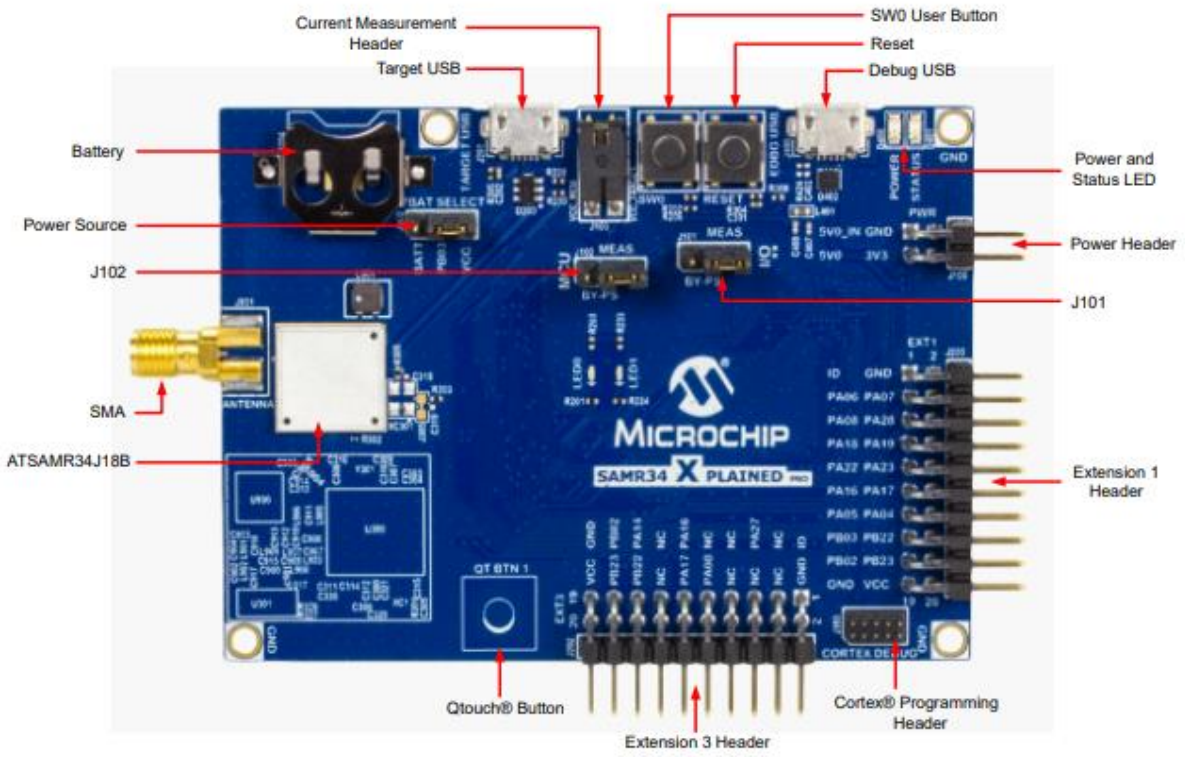


5. Hardware environment setup

5.1. Supported platform and eval board

The following board is used/supported in this release.

SAMR34 Xplained Pro Part# DM320111



This board can be powered from the Debug USB port located at the top right of the image shown above, or by way of the 5v Power Header shown at the top right-hand edge of the above image.

Other methods are available, but not used for this example application.

Note! The small battery located at the top left is only for battery backup of the battery backup memory section, and is NOT used to actually power the board or SAMR34 device. This battery is not required for this example application.

6. LoRaWAN Range Test [Button] example project

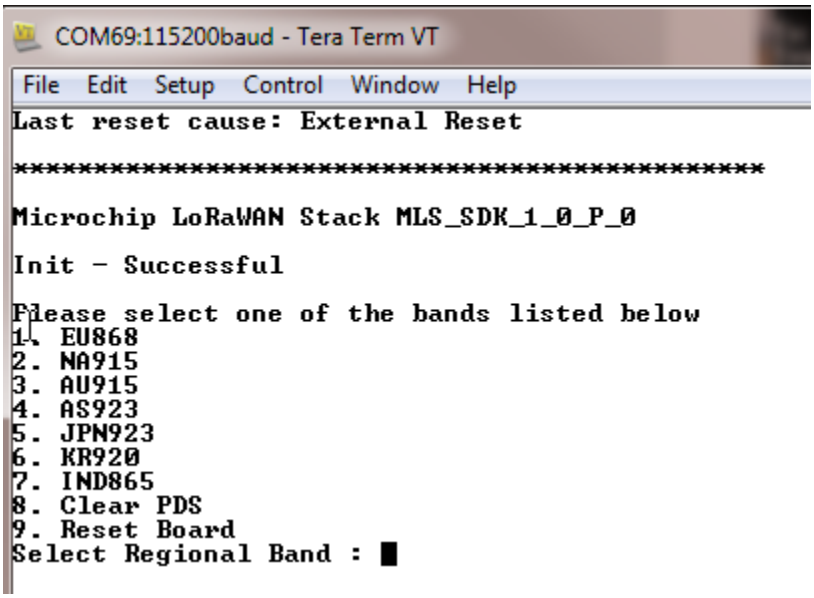
The Range Test example application code focuses on the elements required to take a battery operated Lorawan end node into the field to collect range data between the end node, and the LoRaWAN gateways that it is able to communicate thru.
The following features of the SAMR34 LoRaWAN solution are demonstrated:

- Establish a LoRaWAN connection automatically between the SAMR34 end node and available Gateways that are within radio range.
- Periodically transmit an unconfirmed data frame (every 30 seconds)
- Manually transmit a confirmed data frame upon the press of a pushbutton
- Allow adjustment, in the field, of Data Rate (spreading factor) and Tx Power Levels without the need of a terminal (laptop etc). This is accomplished by using various duration button presses of the SW0 user button on the XPRO board.

To run the example application, follow the instructions given below

1. Connect the XPRO board to your computer using a USB cable connected to the EDBG connector on the XPRO board.
2. Later we can power the board from a 5v battery pack connected to the the 5v pwr header of the SAMR34 XPRO board, for mobile field measurements. Or use a USB rechargeable 5v battery pack connected to the EDBG connector.
3. Program the SAMR34 XPRO board with the Range Test [Button] application firmware using Atmel Studio 7.
4. For the initial setup, connect a terminal application like Teraterm to the EDBG port of the XPRO board. This will be used for serial console input and output during the setup phase.
5. Setup the terminal Application for 115200 baud N 8 1
   Wait a few seconds, until the green led begins to blink.

   If it is the *first time execution* of the application after initial programming , then with a terminal program connected you will see a startup screen similar to the one shown below:



Enter a number between 1 and 7 to select the RF band or region that you intend to operate within.
Menu Selection 8 will allow you to clear the persistent data storage memory, to start again.
Menu Selection 9 performs a reset to the application and stack.

Generally, to start over (and remove all previous settings) you would select 8, and then 9

Once you have selected the region of operation, you will then be presented with the following screen

```
**********Join Parameters**********

DevEUI  : 0x000425191801d47b

AppEUI  : 0x0000000000000005

AppKey  : 0x00000000000000000000000000000005

TxPower Index      = 05
Current Data rate  = DR0
ADR                = OFF

Confirmed Retries  : 04
SUBBAND            = 02

Join Request Sent for NA915
```

Here you will find the parameters that will be used to join the LoRaWAN network
The DevEUI is read from the preprogrammed NVM (non volatile memory) of the XPRO board.
This is a unique EUI supplied by Microchip. Or from the one provided in conf_app.h as discussed earlier.

The AppEUI and AppKey were the parameter's that you configured in the conf_app.h file, before you built the Program, as described in section 3.0 of this document.

Following the display of the above screen, a join request is sent using the displayed parameters/attributes

If the Join operation is **not** successful, you will be shown a screen similar to the one shown below

```
Join Request Sent for NA915

Joining Denied
Join Attempt was on CH  13
Awaiting next Join attempt

**********************************************
```

In this case, a join attempt on Channel 13 (North America band) has failed.
A delay is performed before the next Join attempt.

If the Join attempt is successful, you will be presented with a screen as shown below:

```
Joining Successful
Joined on Channel 15

DevAddr: 0x26022944

**************Application Configuration**************

DevType             : CLASS A

ActivationType      : OTAA

Transmission Type   : UNCONFIRMED

FPort               : 1

TxPower Index       : 05
SUBBAND             = 02

Confirmed Retries   : 04

*****************************************************
```

The successful Join, shown above, occurred on channel 15.
The DevAddr assigned by the Network Server (during an OTAA join) is displayed.
The remainder of the display reports the current application configuration.

This information is useful when trying to confirm that the settings of the end node are proper, and match the configuration used when registering a node with your LoRaWAN network service provider.

Once you have joined the network successfully, the end node application will periodically send a data frame containing temperature sensor data.
The screen below illustrates two periodic transmissions that occur 30 seconds apart.
The first transmission occurred on channel 8 and the second on channel 10

The TX power index of 05 represents the highest ***SAMR34 TX power level*** available for North America.
DR0 represents SF10BW125 for North America, and is the lowest data rate available in North America. This is also the configuration that will provide the highest link budget (highest TX power and best RX sensitivity) to achieve the longest range .

```
*****************************************************
Temperature: 18.0° C  / 64.4° F

Tx Data Sent
Transmission Success
Active Channel 8
Active TxPower Index = 05
Active Data rate = DR0
*****************************************************

wakeup from sleep_ok 27469 ms
*****************************************************
Temperature: 17.0° C  / 62.6° F

Tx Data Sent
Transmission Success
Active Channel 10
Active TxPower Index = 05
Active Data rate = DR0
*****************************************************
```

6.1 Changing Data Rate / Spreading factor in the field

When not connected to a serial console, it is still possible to adjust the Data Rate/spreading factor by way of the SW0 pushbutton, on the SAMR34 XPRO evaluation board. Changing of the Data Rate can be performed by a long press of the SW0 pushbutton. Press and hold the pushbutton until you see the green led turn on. Then release the button.
This action will increment the Data Rate by one step, moving from a lower data rate to a higher data rate.

The new data rate will be indicated by the blinking of the green led after you stop pressing the pushbutton.
The chart at the end of this section lists the number of blinks to expect for the various Data Rates available.

If the serial console was connected, then a message as shown below is displayed on the console.
In this case the Data Rate was changed from DR0 to the new setting of DR1

```
wakeup from sleep_ok 5286 ms
Set DateRate = DR1 Success
```

As you can see from the console listing below, the next transmission was made using DR1

```
**************************************************
Temperature: 18.0° C  / 64.4° F

Tx Data Sent
Transmission Success
Active Channel 9
Active TxPower Index = 05
Active Data rate = DR1
**************************************************
```

Region	Data Rate	Spreading factor	Bandwidth	bps	LED blink count
North America	DR0	SF10	125Khz	980	1
	DR1	SF9	125Khz	1760	2
	DR2	SF8	125Khz	3125	3
	DR3	SF7	125Khz	5470	4

Region	DataRate	Spreading factor	Bandwidth	bit/s	LED blink count
Europe 868	DR0	SF12	125Khz	250	1
	DR1	SF11	125Khz	440	2
	DR2	SF10	125Khz	980	3
	DR3	SF9	125Khz	1760	4
	DR4	SF8	125Khz	3125	5
	DR5	SF7	125Khz	5470	6

NOTE: Only 125Khz channels are used in this application

6.2 Changing TX power index

The initial Tx power index setting is set to the maximum TX output power allowed for the region selected.
For North America this would be a Tx power Index of 5. The SW0 user button on the SAMR34 XPRO can be used to modify this parameter while making field range measurements.
An Extended Long press of the SW0 pushbutton will increment the TX power index and thus reduce the TX output power used for future transmissions.
Press the SW0 button until the amber LED turns on. (***Note!*** First the green LED will turn on, and you must continue to hold/press the button until the amber led also turns on.
Release the button when you see both the green and amber LED's on.

When you release the SW0 button, the amber led will blink a specific number of times to indicate which Tx power index setting you adjusted to. See table below for relationship between the number of blinks and the Tx power index that you have set.

If you happen to have the console connected at this point in program operation, you will also see screens like the following.

The first section indicates that in this example, the Tx power index was changed to 6

And the second section illustrates that on the next sensor transmission, a Tx power index of 6 was actually used.

```
wakeup from sleep_ok 7897 ms
Set Tx Power Index = 6 Success
```

```
wakeup from sleep_ok 29989 ms
**************************************************
Temperature: 19.0° C  / 66.2° F

Tx Data Sent
Transmission Success
Active Channel 8
Active TxPower Index = 06
Active Data rate = DR1
**************************************************
```

Region	Tx Power Index	Tx Power dBm *[30-(2*PwrIndex)]*	LED blink count
North America	**5**	**+20**	**6**
	6	**+18**	**5**
	7	**+16**	**4**
	8	**+14**	**3**
	9	**+12**	**2**
	10	**+10**	**1**

Note! Tx power dBm is related to output pins of SAMR34 and does not include any loss in output matching or interface circuitry.

Region	Tx Power Index	Tx Power dBm	LED blink count
Europe 868	**0**	**+16 max**	**8**
	1	**+14 max -2**	**7**
	2	**+12 max -4**	**6**
	3	**+10 max -6**	**5**
	4	**+8 max -8**	**4**
	5	**+6 max -10**	**3**
	6	**+4 max -12**	**2**
	7	**+2 max -14**	**1**

Note! Tx power dBm is related to output pins of SAMR34 and does not include any loss in output matching or interface circuitry.

6.3 Restoring to default (initial) Data Rate and TX power levels

After making changes to either the Data rate, Tx power index or both, while making range measurements, there may be a need to quickly reset both parameters to their original default settings (lowest data rate and higher Tx power level).
This can be achieved by performing an extra long SW0 pushbutton press allowing a reset to default operation. Press and hold the SW0 pushbutton as you watch the leds. The green led comes on first, (continue holding) while the amber led also turns on, and again continue holding until both leds turn off. You may then release the SW0 pushbutton and both Data Rate and Tx power level will be set back to the initial default values for the region you had selected initially.

```
wakeup from sleep_ok 5473 ms
Set DateRate to default  DR0 Success
Set default Tx Power Index  5 Success
```

6.4 Manual PushButton sensor transmission

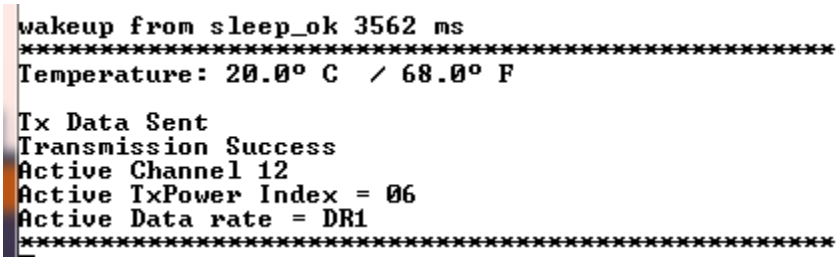
Although the application will periodically transmit a frame containing temperature sensor data, there are times when one would want to manually initiate a sensor transmission by pressing the SW0 pushbutton.
A short press of the pushbutton will initiate a sensor transmission using the confirmed message format.

Upon short pressing the button, the green led will start blinking, and it will remain blinking until the transaction has completed successfully or fails.
If all went well the green led will stop blinking and the amber led will be off.

If there is a failure or no acknowledgement is obtained, then the amber led will remain on to indicate a failure. The amber led will be cleared on the next successful transmission attempt.

NOTE! The SW0 pushbutton is right next to the RESET pushbutton .. Be very careful that you do not press the RESET pushbutton by mistake! This will easily happen if you are not paying attention to which pushbutton you are pressing.

If the console is connected when initiating a manual message transmission, the following will be shown on the screen.



This screen illustrates that the device had been asleep for 3562 ms at the time the pushbutton was pressed. The pressing of the pushbutton woke up the device from standby sleep, and initiated a manual sensor transmission.

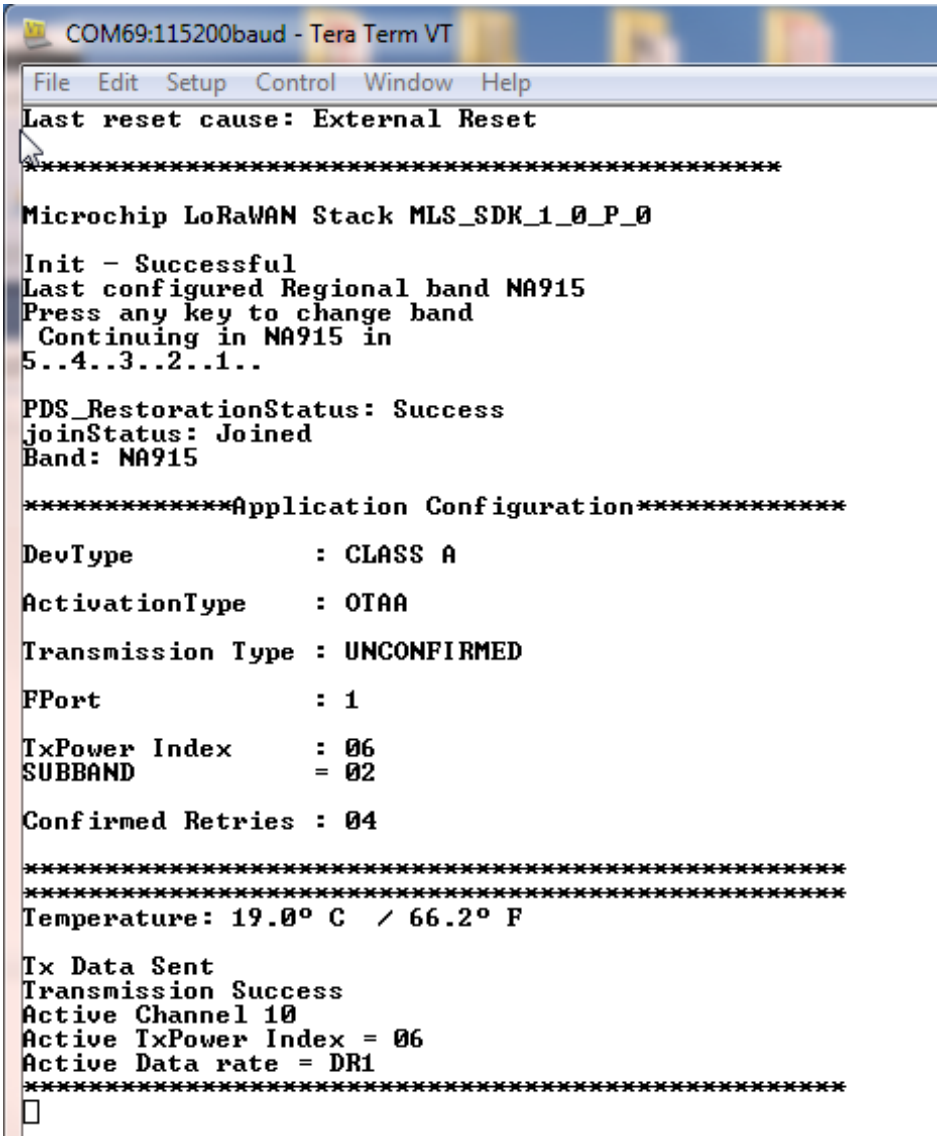
NOTE! When operating in the European 868mhz band or other bands that incorporate duty cycle restrictions, you may see a NO CHANNEL FOUND message indicating that previous transmissions have used the available time, and you will have to wait and try again later when duty cycle allows.

6.5 Restarting / Restoring configuration upon power cycle/reset

The initial selecting of Region is only necessary during the initial setup.
Once the application has been run once, the required parameters and attributes are stored in non volatile memory (persistent storage)

Upon a power cycle or reset of the device, a check is made to determine if there is previously stored information that can be used to restore operation to settings that were in use before the reset or power cycle.

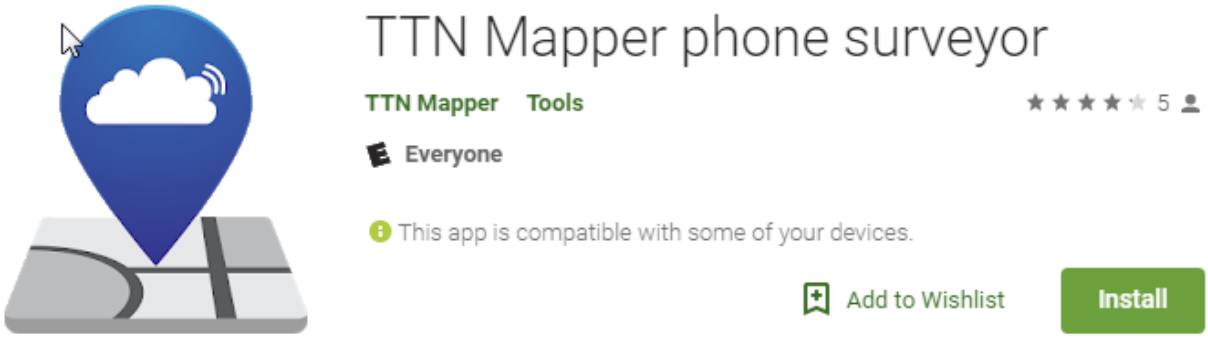
The console screen below illustrates what occurs upon a restoration from a power cycle or reset.

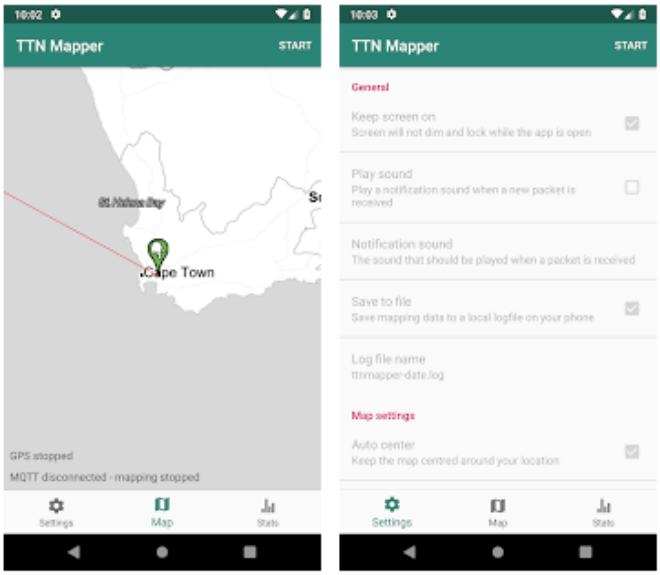


Operation will occur on the same band that was used the last time the application executed.
There will be a count down of approximately 5 seconds to allow the user to press a console key to abort the restoration process, and go back to the main menu , where a new region can be selected if needed.
For field testing, all one needs to do Is initially configure the application, and from that time on, the settings last used will be applied.

7.0 TTNmapper [mobile app for LoRaWAN range testing]

If possible, it is suggested that you use ***The Things Network*** as the LoRaWAN network during your range measurement activity's. This is a free and open network with some very good tools for making range measurements.
One of those tools is TTN Mapper. This free SmartPhone app is available for Android and IOS thru the apps stores

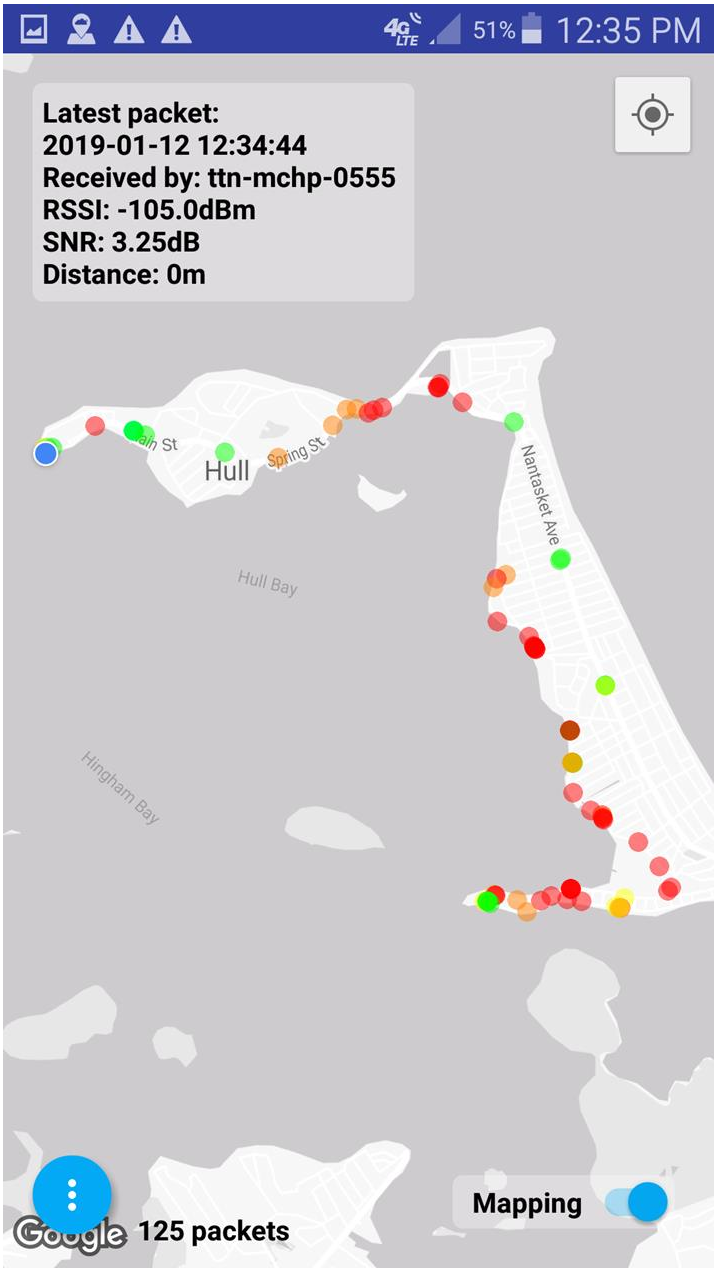




This is the third version of the TTN Mapper Android app. Use this app to map coverage of The Things Network using the GPS receiver in your phone. A LoRa node that is being kept in close proximity of your phone transmits LoRaWAN messages to The Things Network. This app subscribes to TTN to receive these messages. The metadata that is associated with received messages contains the signal strength and quality of the message as received by one or many gateways. This metadata is paired with your phone's location and sent to TTN Mapper. There we use this signal strength and quality data, and the location where the observation is made to draw a heatmap of the coverage of The Things Network.

Typical TTN_mapper screen capture

Below is an example of data captured during range measurement activities.
The color coded dots are locations where the sensor transmissions were captured and logged by the ttn mapper app.
The GPS position was provided by the GPS available on the smartphone, that was used while running the ttn_mapper application
The color code table lists the signal level values associated with the colors used in the map.

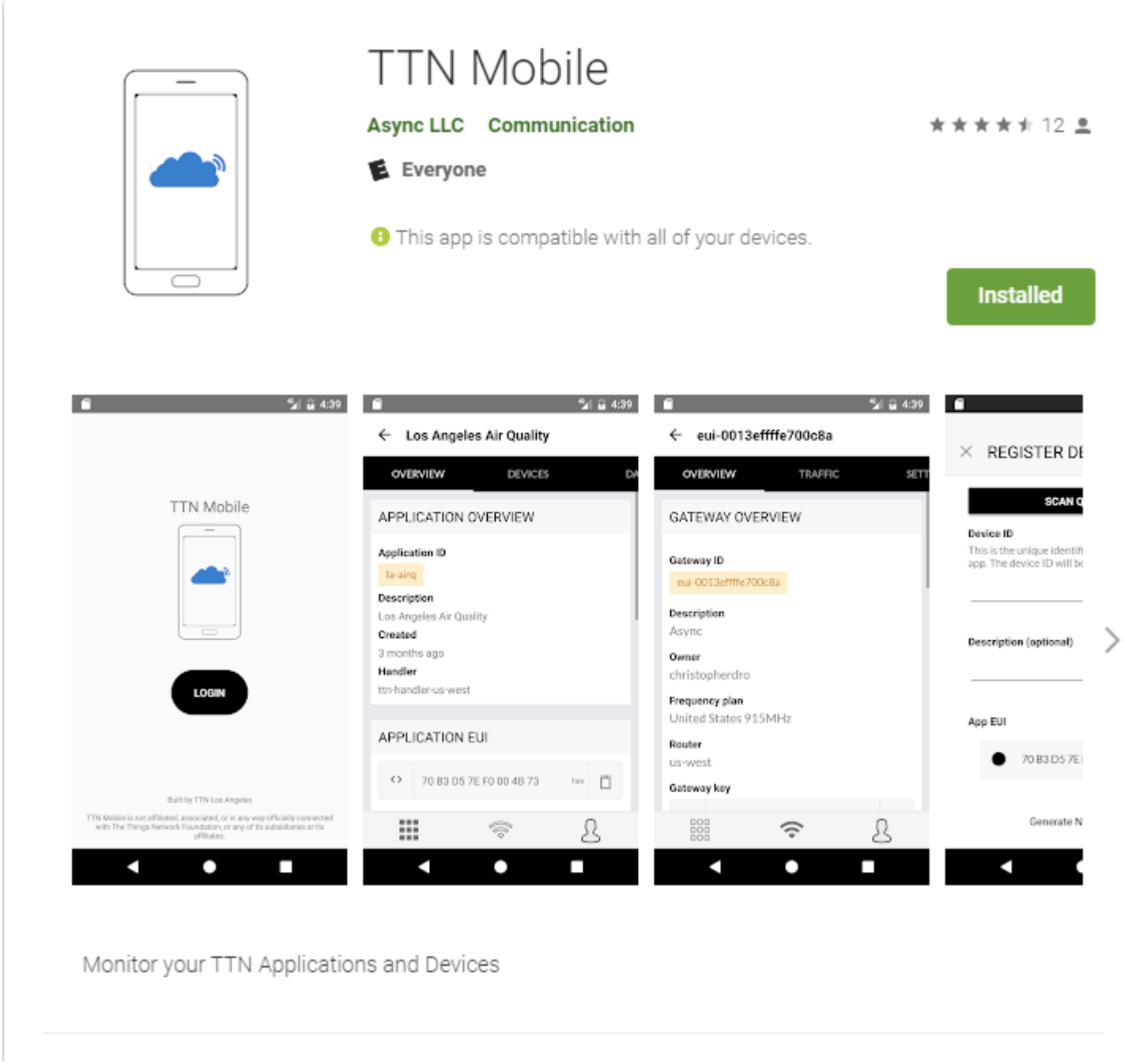


Colour	RSSI
	> -100dBm
	-100 - -105
	-105 - -110
	-110 - -115
	-115 - -120
	< -120dBm

When you return from field testing, you can visit https://ttnmapper.org/
You will then be able to download csv spreadsheet data collected during your mapping and also download an image like the one displayed on your smartphone screen. You can also screen capture your smartphone screen as was done to capture the image shown above.

8.0 TTN mobile app [mobile app for viewing the TTN console]

Another useful smartphone application is TTN mobile.
This application allows you to view the same information that is available using the TTN web based console from a PC.
When using this application during range testing, you can monitor the information being sent and received thru The Things Network (TTN).



Typical screen on Android smartphone
Additional information can be displayed by tapping on a particular line of the display to see additional meta data collected during the sensor transmission.