Sigfox Workshop
Bogotá - June 2017

Using Thinxtra Xkit

Contribute back

Don’t forget to publish your experiments

Code Samples, HW design, fails … will be useful to other people

We all start by copy/pasting ;)

Your own website, github, hackster.io, instructables … your call!

Online resources

Technical information :http://makers.sigfox.com

Videos : http://youtube.com/sigfox

Q&A : http://ask.sigfox.com

Github: http://github.com/sigfox

Contact info

Alexandre Araujo

alexandre.araujo@sigfox.com

sigfox

Register

http://backend.sigfox.com/activate

Provider: Thinxtra

Country : Colombia

ID/PAC : Check sticker

sigfox

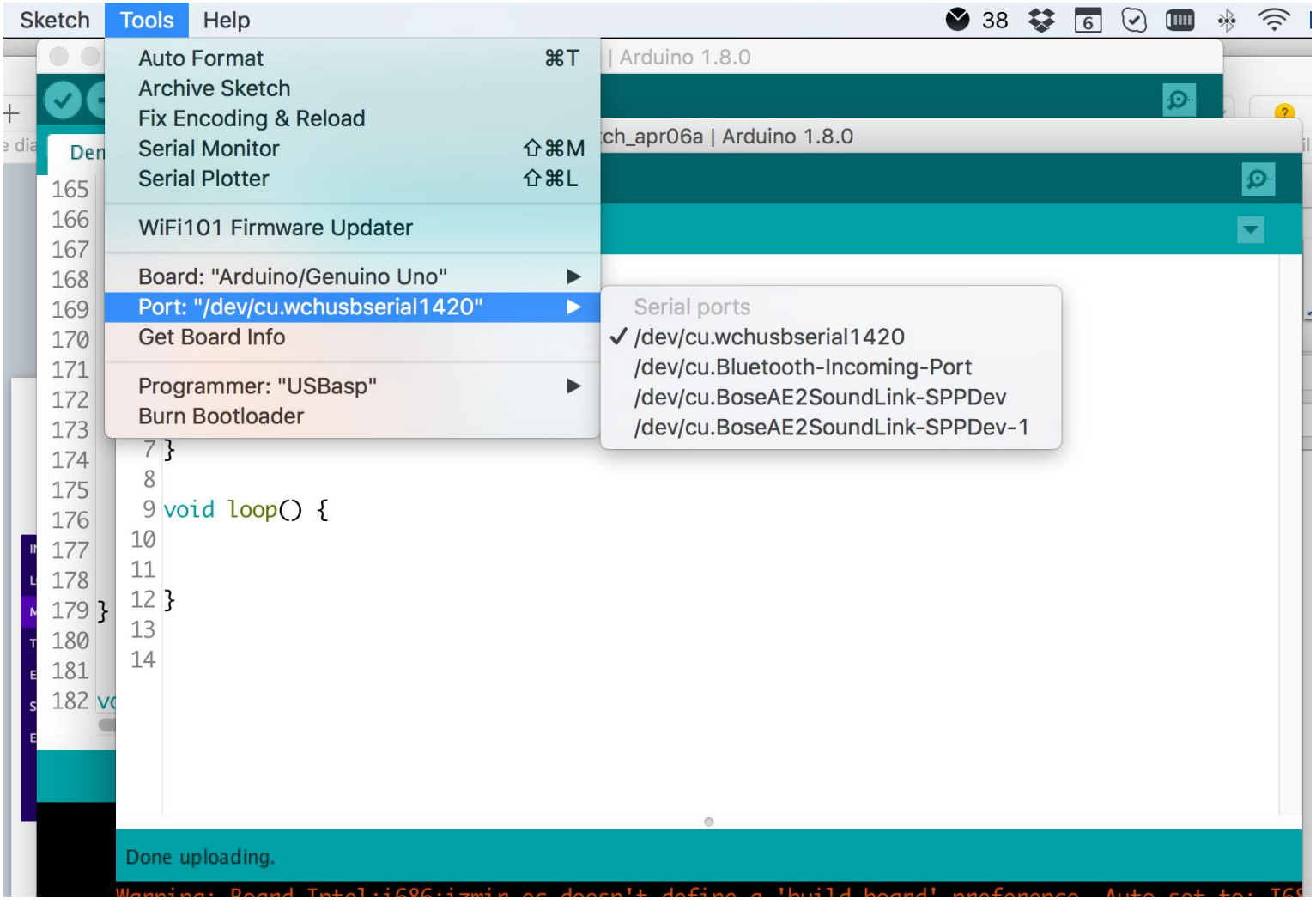
Hello World

Arduino Setup

Open the Arduino IDE

Select the board (COM port)

Board type : Arduino Uno


Sketch
Tools
Help
38
Auto Format ⌘T
Archive Sketch
Fix Encoding & Reload
Serial Monitor ⇧⌘M
Serial Plotter ⇧⌘L
WiFi101 Firmware Updater
Board: "Arduino/Genuino Uno"
Port: "/dev/cu.wchusbserial1420"
Get Board Info
Programmer: "USBasp"
Burn Bootloader
Serial ports
/dev/cu.wchusbserial1420
/dev/cu.Bluetooth-Incoming-Port
/dev/cu.BoseAE2SoundLink-SPPDev
/dev/cu.BoseAE2SoundLink-SPPDev-1
| Arduino 1.8.0
ch_apr06a | Arduino 1.8.0
void loop() {
}
Done uploading.

sigfox

Hello world sketch

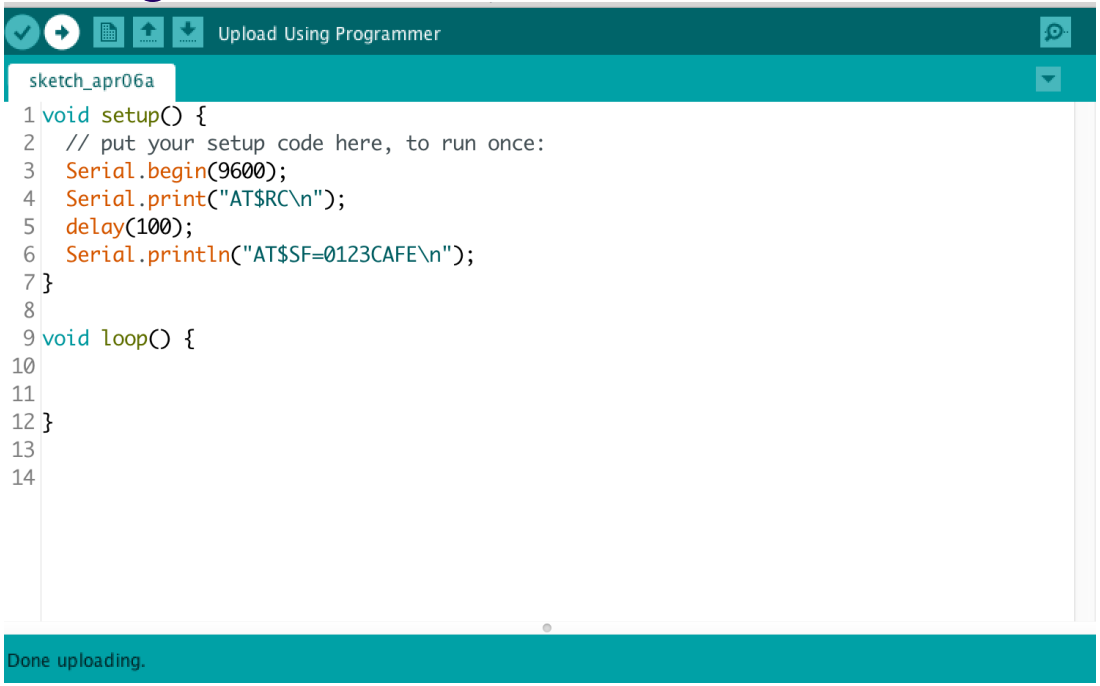
```
void setup() {
  Serial.begin(9600);
  Serial.print("AT$RC\n");
  delay(100);
  Serial.print("AT$SF=0123CAFE\n");
}
void loop() {}
```

sigfox

Upload to the board

Remove the shield first

Then click Upload / Descargar



sigfox

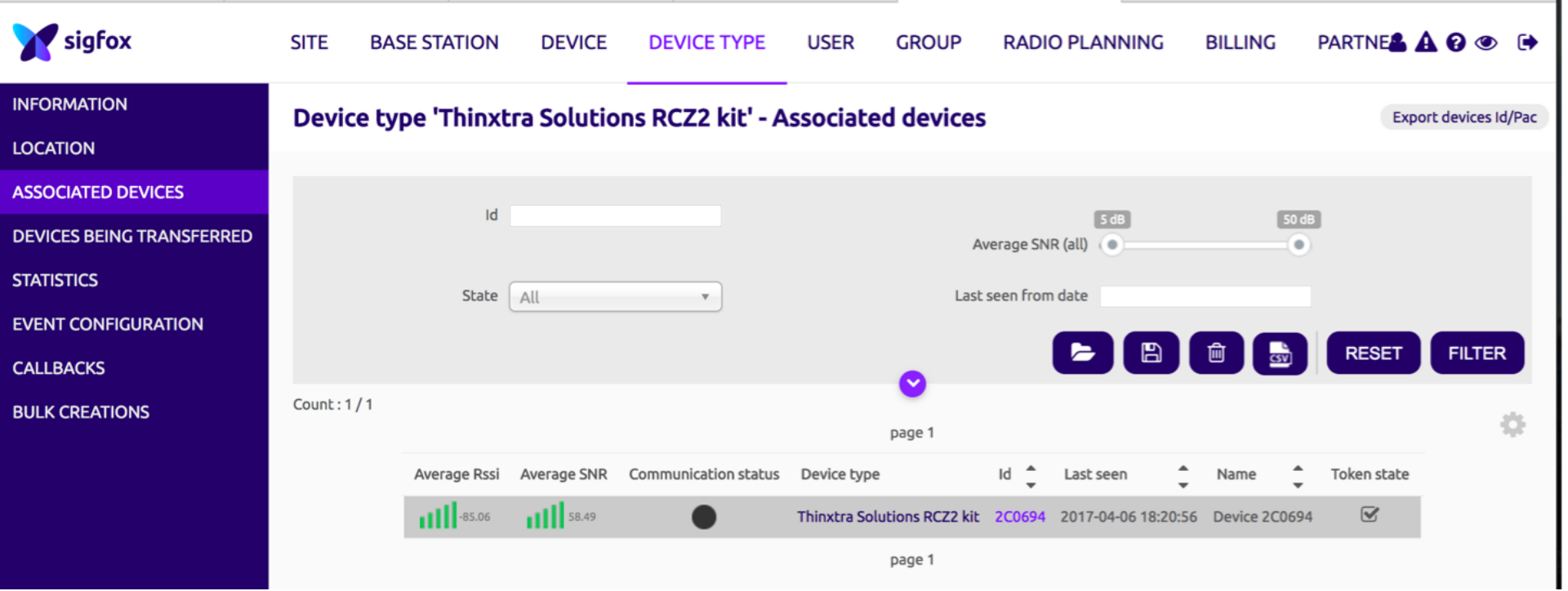
Message received ?

http://backend.sigfox.com

Navigate to the *devices* menu in the top bar

Click on the ID of your device

Enter the *messages menu* from the left navigation column


sigfox
SITE
BASE STATION
DEVICE
DEVICE TYPE
USER
GROUP
RADIO PLANNING
BILLING
PARTNE
INFORMATION
LOCATION
ASSOCIATED DEVICES
DEVICES BEING TRANSFERRED
STATISTICS
EVENT CONFIGURATION
CALLBACKS
BULK CREATIONS
Device type 'Thinxtra Solutions RCZ2 kit' - Associated devices
Export devices Id/Pac
Id
5 dB
50 dB
Average SNR (all)
State
All
Last seen from date
RESET
FILTER
Count : 1 / 1
page 1
Average Rssi
Average SNR
Communication status
Device type
Id
Last seen
Name
Token state
-85.06
58.49
Thinxtra Solutions RCZ2 kit
2C0694
2017-04-06 18:20:56
Device 2C0694
page 1

sigfox

sigfox

First callback

Callback setup

Device Type menu

Click on your *device type* name

Enter the *Callbacks* menu

Select *new default callback*

sigfox

- INFORMATION
- LOCATION
- ASSOCIATED DEVICES
- DEVICES BEING TRANSFERRED
- STATISTICS
- EVENT CONFIGURATION
- CALLBACKS
- BULK CREATIONS

Device type 'Thinxtra Solutions RCZ2 kit' - Information

Id: 58e4135d3c8789274562f9e5

Name: Thinxtra Solutions RCZ2 kit

Description: Auto created device type for EVK user : Nicolas Lesconnec

Keep alive: N/A

Group: Nicolas Lesconnec EVK

Payload display: None

Contract: Free eval board contract

Alert Email:

Downlink data hexa: {tapId}0000{rssi}

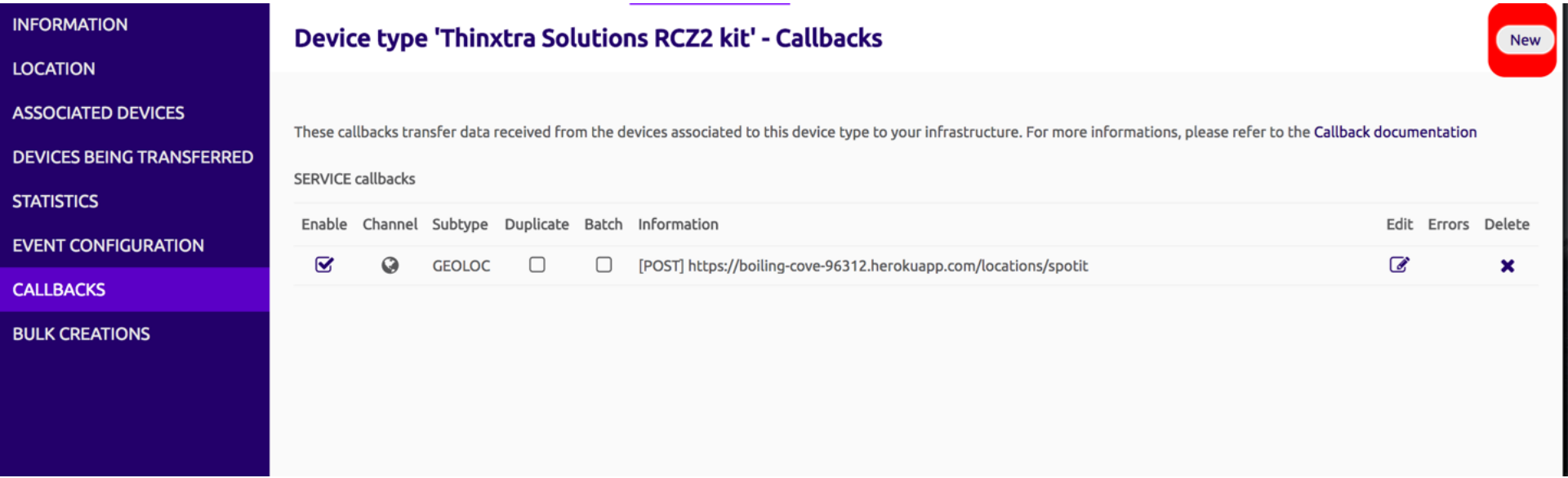
Creation date: 2017-04-04 23:42:53

Created by: Nicolas Lesconnec

Last edition date: 2017-04-04 23:43:11

Last edited by: Nicolas Lesconnec

sigfox


INFORMATION
LOCATION
ASSOCIATED DEVICES
DEVICES BEING TRANSFERRED
STATISTICS
EVENT CONFIGURATION
CALLBACKS
BULK CREATIONS
Device type 'Thinxtra Solutions RCZ2 kit' - Callbacks
New
These callbacks transfer data received from the devices associated to this device type to your infrastructure. For more informations, please refer to the Callback documentation
SERVICE callbacks
Enable
Channel
Subtype
Duplicate
Batch
Information
Edit
Errors
Delete
GEOLOC
[POST] https://boiling-cove-96312.herokuapp.com/locations/spotit

sigfox

Device type 'Thinxtra Solutions RCZ2 kit' - New Callback

Create callbacks to connect Sigfox cloud to your server/platform.
A callback is a custom http request containing your device(s) data, along with other variables, sent to a given server/platform when the aforesaid device(s) message is received by Sigfox cloud.

Custom callback

Creates a new callback from **Sigfox cloud** to **your own server**. This is the "default" callback type.
You can create a full custom request (http method, content type, headers, etc).

AWS IoT

AWS IoT is a managed cloud platform that lets connected devices easily and securely interact with cloud applications and other devices. AWS IoT can support billions of devices and trillions of messages, and can process and route those messages to AWS endpoints and to other devices reliably and securely.

AWS Kinesis

Amazon Kinesis is a platform for streaming data on AWS, offering powerful services to make it easy to load and analyze streaming data, and also providing the ability for you to build custom streaming data applications for specialized needs.

Microsoft Azure™ Event hub

Event Hubs is an event processing service that provides event and telemetry ingress to the cloud at massive scale, with

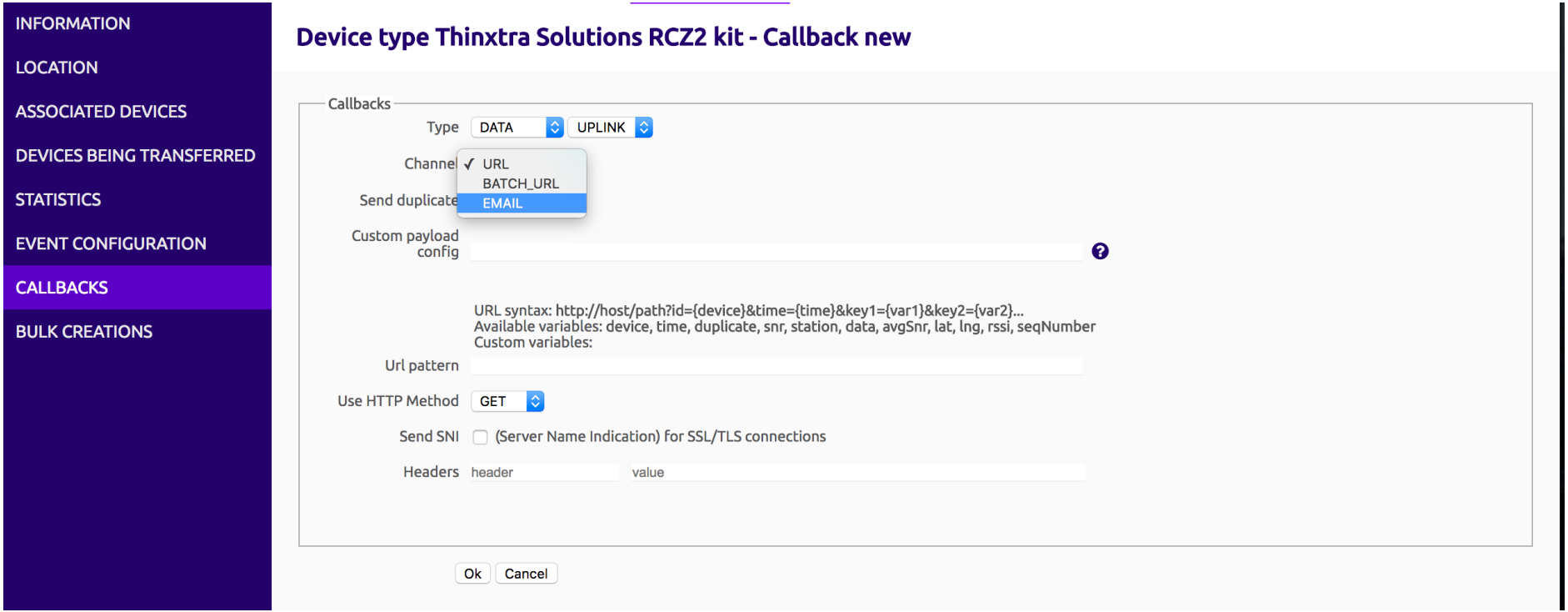
Callback setup

TYPE : DATA UPLINK

Choose a *CHANNEL* : URL (EMAIL for a quick test)

Url pattern: URL of your own server

Use HTTP method: GET/POST/PUT


INFORMATION
LOCATION
ASSOCIATED DEVICES
DEVICES BEING TRANSFERRED
STATISTICS
EVENT CONFIGURATION
CALLBACKS
BULK CREATIONS
Device type Thinxtra Solutions RCZ2 kit - Callback new
Callbacks
Type
DATA
UPLINK
Channel
URL
BATCH_URL
EMAIL
Send duplicate
Custom payload config
URL syntax: http://host/path?id={device}&time={time}&key1={var1}&key2={var2}...
Available variables: device, time, duplicate, snr, station, data, avgSnr, lat, lng, rssi, seqNumber
Custom variables:
Url pattern
Use HTTP Method
GET
Send SNI
(Server Name Indication) for SSL/TLS connections
Headers
header
value
Ok
Cancel

sigfox

Callback status

In the *Devices* > *Messages* panel, you have a indicator of the callback status (an arrow)

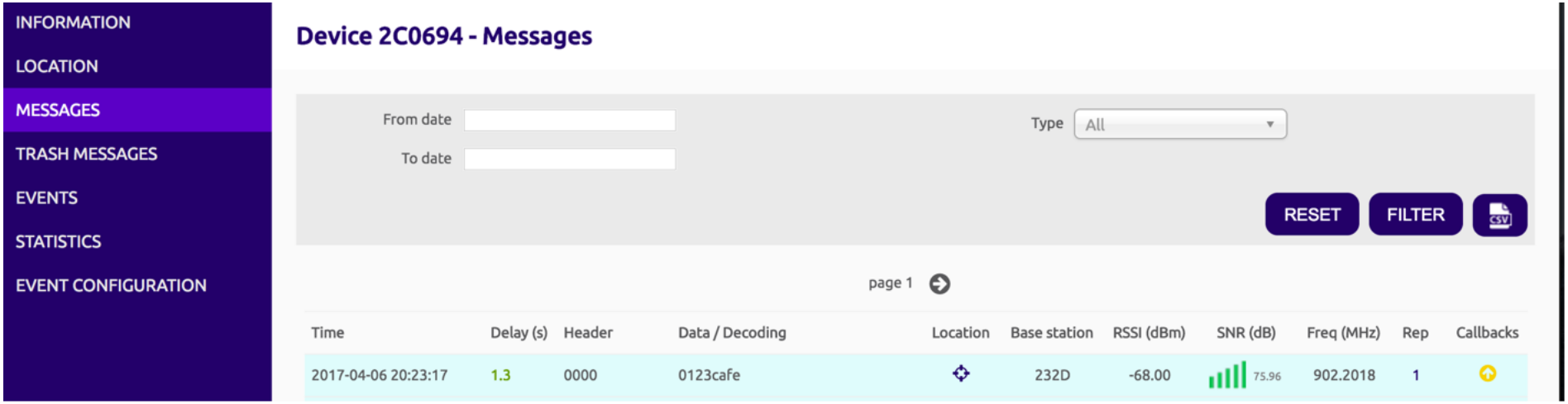
Black : in progress

Green : Callback OK

Red : Callback KO (at least one of the callbacks failed)

Click the arrow to display details.

sigfox


INFORMATION
LOCATION
MESSAGES
TRASH MESSAGES
EVENTS
STATISTICS
EVENT CONFIGURATION
Device 2C0694 - Messages
From date
To date
Type
All
RESET
FILTER
CSV
page 1
Time
Delay (s)
Header
Data / Decoding
Location
Base station
RSSI (dBm)
SNR (dB)
Freq (MHz)
Rep
Callbacks
2017-04-06 20:23:17
1.3
0000
0123cafe
232D
-68.00
75.96
902.2018
1

sigfox

Downlink

How does it work ?

Send a message, with a *downlink* flag

Once message is sent, the module gets back to sleep

After 20s, it will wake up automatically, in Rx mode

It will wait 20s for a *downlink* message

Afterwards it will get back to sleep

Downlink setup

To setup an automatic callback :

Device Type > *Info* > *Edit*

In the *Downlink data* settings, set the following :

Downlink Mode : DIRECT

Set the following value : 123400000BADCAFE

sigfox

INFORMATION
LOCATION
ASSOCIATED DEVICES
DEVICES BEING TRANSFERRED
STATISTICS
EVENT CONFIGURATION
CALLBACKS
BULK CREATIONS

Device type 'Thinxtra Solutions RCZ2 kit' - Information

Disengage sequence number | Edit | Delete

Id: 58e4135d3c8789274562f9e5

Name: Thinxtra Solutions RCZ2 kit

Description: Auto created device type for EVK user : Nicolas Lesconnec

Keep alive: N/A

Group: Nicolas Lesconnec EVK

Payload display: None

Contract: Free eval board contract

Alert Email:

Downlink data hexa: {tapId}0000{rssi}

Creation date: 2017-04-04 23:42:53

Created by: Nicolas Lesconnec

Last edition date: 2017-04-04 23:43:11

Last edited by: Nicolas Lesconnec

Device type Thinxtra Solutions RCZ2 kit - Edition

Device type information

Name: Thinxtra Solutions RCZ2 I

Description: Auto created device type for EVK user : Nicolas

Keep-alive (in minutes): 0

If we fail to call one of your callbacks, an email will be sent to the address below so that you can take action to fix the problem.

Alert email:

Downlink data

Downlink mode: DIRECT / CALLBACK

Expression must either include hexadecimal encoded bytes (ex: deadbeefcafebabe) or the following variables: - {time} 4 bytes - {tapId} 4 bytes - {rssi} 2 bytes

Downlink data in hexa: {tapId}0000{rssi}

Payload display

Select below the most suitable parsing mode for the display of your payloads in the backend (mostly appropriate for debugging and development)

Payload parsing: Regular (raw payload)

Ok Cancel

How to request a downlink

Same AT command, with additional parameters

```
AT$SF=[hex byte]*, 1
```

sigfox

Handle the response

When entering Rx mode, the module will display

```
+RX BEGIN
```

Received frame (if any) will be displayed as:

End of Rx mode

```
+RX END
```

sigfox

Downlink callback

In *Device Type* > *Info* > *Edit*

change *Downlink mode* to CALLBACK

Create a new default callback, with TYPE : DATA | BIDIR

Then set up your URL

Sample input output

```
AT$SF=55 50 4C 49 4E 4B,1

OK

+RX BEGIN

+RX=44 4F 57 4E 4C 49 4E 4B

+RX END
```

sigfox

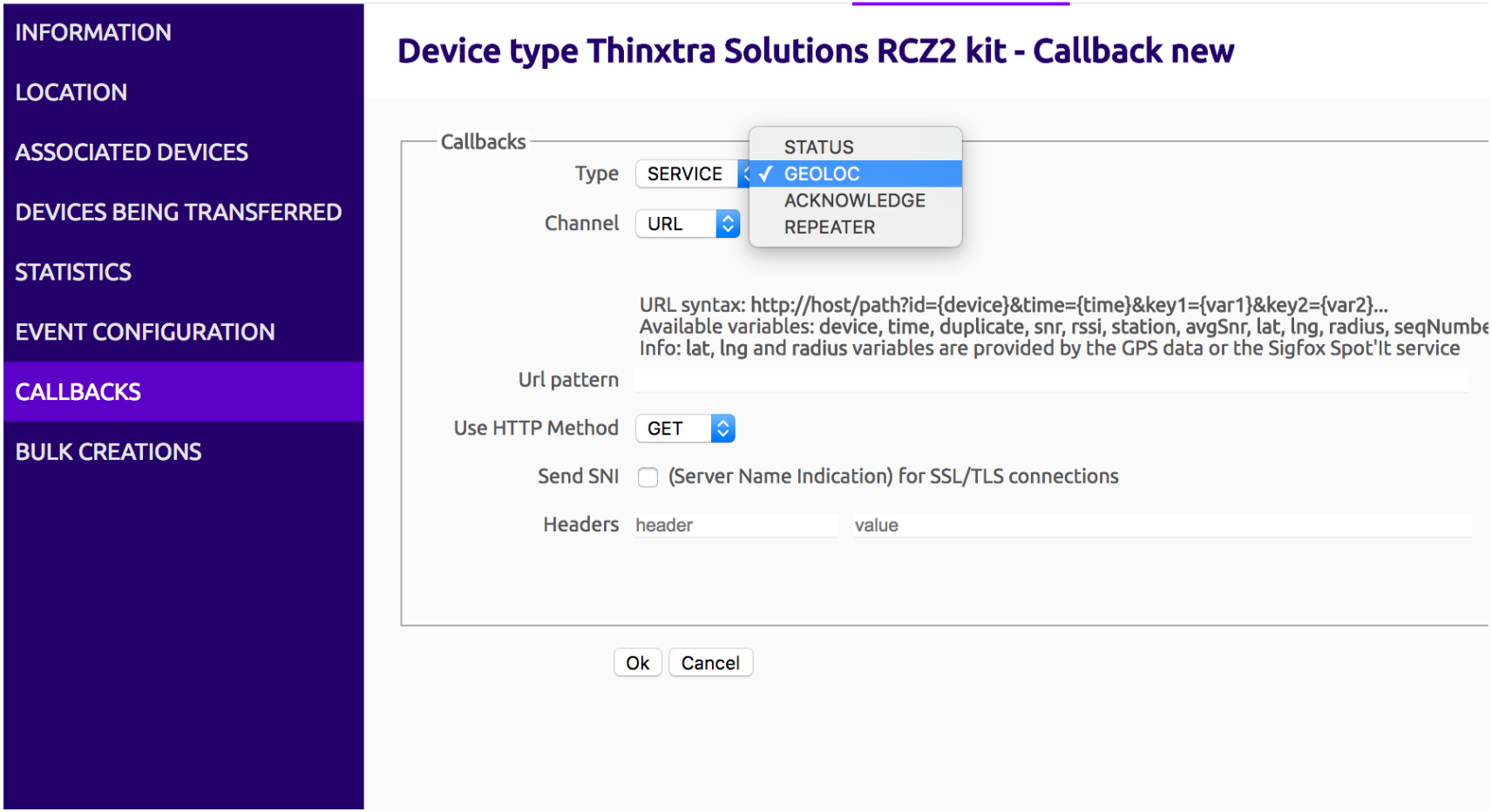
XKit - demo app

sigfox

Geolocation

Geoloc callback

Simply create a SERVICE > GEOLOC callback, and receive latitude + longitude + accuracy


INFORMATION
LOCATION
ASSOCIATED DEVICES
DEVICES BEING TRANSFERRED
STATISTICS
EVENT CONFIGURATION
CALLBACKS
BULK CREATIONS
Device type Thinxtra Solutions RCZ2 kit - Callback new
Callbacks
Type
SERVICE
STATUS
GEOLOC
ACKNOWLEDGE
REPEATER
Channel
URL
URL syntax: http://host/path?id={device}&time={time}&key1={var1}&key2={var2}...
Available variables: device, time, duplicate, snr, rssi, station, avgSnr, lat, lng, radius, seqNumbe
Info: lat, lng and radius variables are provided by the GPS data or the Sigfox Spot'It service
Url pattern
Use HTTP Method
GET
Send SNI
(Server Name Indication) for SSL/TLS connections
Headers
header
value
Ok
Cancel

sigfox