ela INNOVATION

# BLE RANGE USER GUIDE

# TABLE OF CONTENTS

# 1 GENERAL INFORMATION ABOUT BLUETOOTH LOW ENERGY

Bluetooth **Low Energy** technology is also called **LE** or **BLE Bluetooth**. This technology appeared in 2010 with the release of version 4.0 of the Bluetooth Core Specification.

Bluetooth Low Energy is an alternative to "classic Bluetooth". By "classic Bluetooth", we mean all versions of Bluetooth released before Core Specification 4.0.

Low Energy Bluetooth technology operates in the free band **ISM 2.4 GHz**. This technology relies on a **frequency hopping radio**. 40 physical channels are allocated and separated from each other by 2 MHz and used according to the FDMA. Three of them consist in **advertising channels** (they might be considered as signalization) and all the others are data channels. In contrast, conventional Bluetooth uses 80 channels separated from each other by 1 MHz.

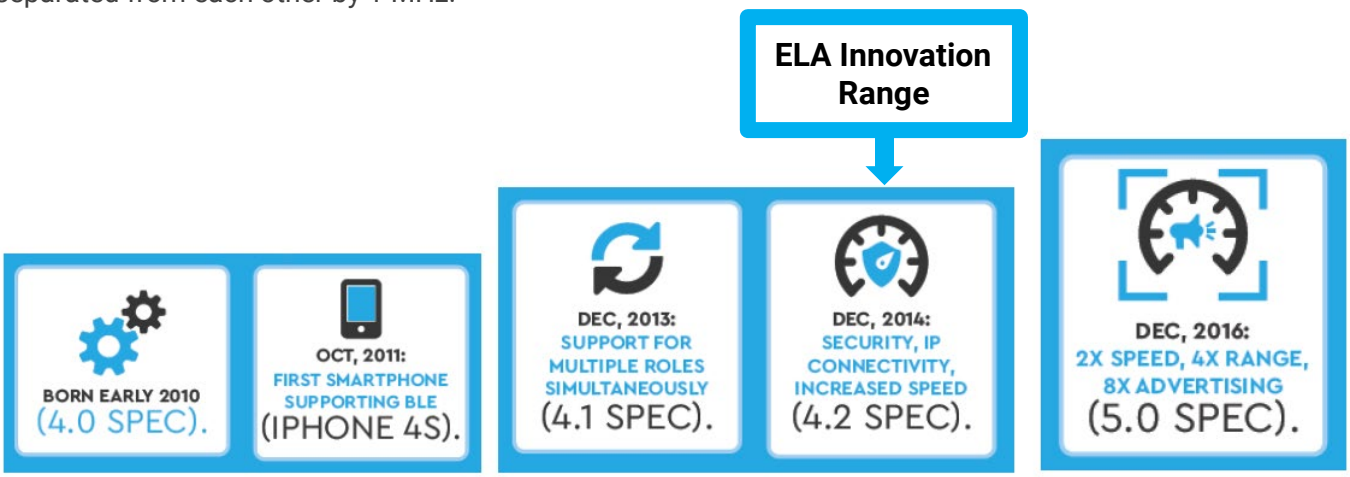



*Figure 1: Evolution of Bluetooth Low Energy versions*

**Bluetooth SIG** is the current standard in terms of information and specifications. The **Bluetooth Special Interest Group**, known as **SIG**, is the body that oversees the development of Bluetooth specifications, manages the various technology qualification processes and grants the needed licenses of the Bluetooth brand and technology to manufacturers.

| | |
|---|---|
| **Bluetooth SIG website** | https://www.bluetooth.com/bluetooth-technology |
| **BLE Specification** | https://www.bluetooth.com/specifications |
| **BLE Services and features** | https://www.bluetooth.com/specifications/gatt |

## 2 BLUETOOTH LOW ENERGY PRODUCTS BY ELA INNOVATION

| *DESIGNATION* | *PRODUCT REFERENCE* | *DESCRIPTION* |
|---|---|---|
| **Blue** PUCK **ID** | IDF25240x | Tag Bluetooth PUCK Format with Identifier Option – iBeacon – Eddystone |
| **Blue** PUCK **BUZZ** | IDF25245x | Tag Bluetooth PUCK format with Identifier Option – Buzzer |
| **Blue** PUCK **T** | IDF25241x | Tag Bluetooth PUCK Format with integrated temperature sensor |
| **Blue** PUCK **T EN12830** | IDF30241x | Tag Bluetooth Format PUCK, integrated temperature sensor, EN12830 (2018) certified |
| **Blue** PUCK **T PROBE** | IDF25250x | Tag Bluetooth Format PUCK external temperature probe, EN12830 (2018) certified |
| **Blue** PUCK **RHT** | IDF25242x | Tag Bluetooth PUCK Format with humidity and temperature sensor option |
| **Blue** PUCK **MAG** | IDF25243x | Tag Bluetooth PUCK Format with magnetic sensor option |
| **Blue** PUCK **MOV** | IDF25244x | Tag Bluetooth PUCK Format with motion sensor option |
| **Blue** PUCK **PIR** | IDF25249x | Tag Bluetooth PUCK format with presence detection sensor |
| **Blue** PUCK **DI** | IDF24246x | Tag Bluetooth PUCK Format with digital input option |
| **Blue** PUCK **DO** | IDF25247x | Tag Bluetooth PUCK Format with digital output option |
| **Blue** PUCK **AI** | IDF25248x | Tag Bluetooth PUCK Format with analog input option |
| **Blue** COIN **ID** | IDF10240x | Tag Bluetooth Format COIN with Identifier option – iBeacon – Eddystone |
| **Blue** COIN **T** | IDF10241x | Tag Bluetooth COIN Format with temperature sensor option |
| **Blue** COIN **MAG** | IDF10243x | Tag Bluetooth COIN Format with magnetic sensor option |
| **Blue** COIN **MOV** | IDF10244x | Tag Bluetooth COIN Format with motion sensor option |
| **Blue** SLIM **ID** | IDF03240x | Tag Bluetooth SLIM Format with Identifier option– iBeacon – Eddystone |
| **Blue** LITE **ID** | IDF28240x | Tag Bluetooth LITE Format with Identifier option– iBeacon – Eddystone |
| **Blue** WATCH **ID** | IDP27240x | Tag Bluetooth WATCH Format with Identifier option– iBeacon – Eddystone |

# 3 BLUE RANGE OPERATIONS BY ELA INNOVATION

## *1.1. REGULAR OPERATING MODE*

- **Advertising mode**

Frames are disseminated through "**Advertising**". Packets are sent periodically at a configurable recurrence comprised within the [0.1s; 10s] interval.

User data size is of 29 bytes. Data content are sensor information or fixed identifier, according to product (Identifier or Sensor).

*See chapter 5 ADVERTISING DATA CONTENT for more information on data sent in "**Advertising**" mode.*

In some cases, a "**Scan Response**" frame may follow the "**Advertising**" frame:

- Battery level below 15%: battery level service available in the Scan Response section.
- A 15-character "Name" added in iBeacon or Eddystone UID format: "*Complete Local Name*" available in the "*Scan Response*" section.

- **Connected Mode**

The BLUE product range by ELA Innovation uses several functions in "Connected Mode". A link is set up between two devices and only these devices can communicate and exchange with each other.

You may establish a connection using a smartphone or a mobile application, or with a PC equipped with the ELA "*Device Manager*" application (provided you activated Bluetooth or connected a BLE dongle to the PC).

- The *Advertising Recurrence* must be less than or equal to 3 seconds to be able to establish a connection.
- Once you enter "*Connected Mode*", "***Advertising***" **is stopped by default**.
- It is possible to send commands to the tag to perform special actions or read data.
- It is possible to get a record of saved data (***Datalogger***) using *Connected Mode*. This datalogger will contain sensor data saved at a defined period with a timestamp for each data.

### *1.2. SPECIFIC OPERATING MODES: MAG, MOV, PIR & DI*

**MAG**, **MOV**, **PIR** and **DI** formats provide the **fast event frame functionality**.

- This frame sends data to a **faster recurrence** (equal to one tenth of the advertising tag recurrence set in NFC). Data contained in this frame is the same as that contained in the simple advertising frame, but its recurrence varies.

- **Fast frames** appear during a period equal to the advertising period, and with a recurrence equal to one tenth of it. Thus, there are **10 frames**.

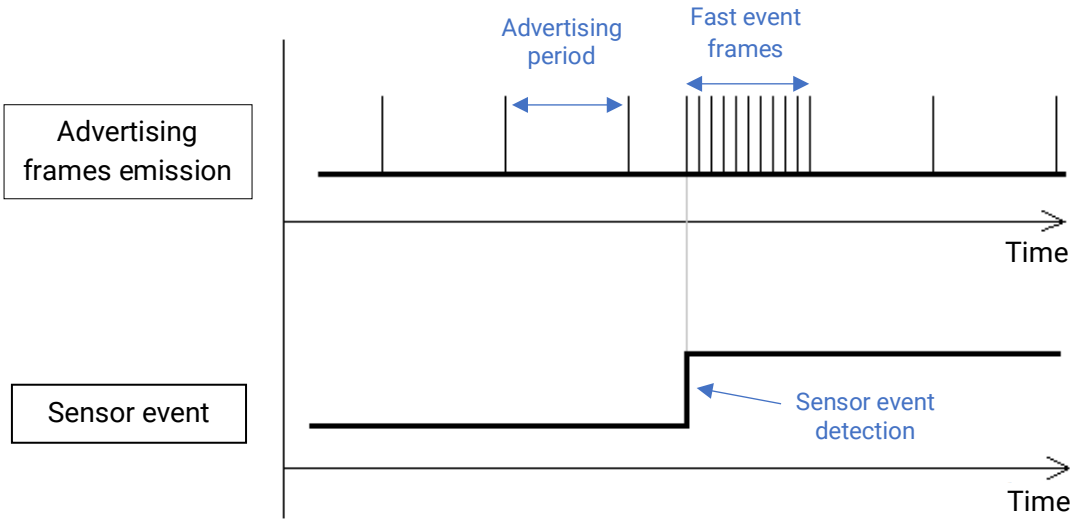



*Figure 2: Demonstration diagram of fast event frames occuring during an event.*

These **fast frames** emission takes place at each sensor event:

- For **MAG format**: With each new magnet detection state (present and absent)
- For **MOV format**: At the beginning and end of each movement (depending on the submitted threshold
- For **DI format**: With each new digital input state (logical state 1 or 0)
- For **PIR format**: with each movement detected (sensor state stay at 1 during movement and falls to 0 few seconds after last movement)

In addition to these fast frames, the MAG, MOV and DI formats data also contains an event counter. This counter is incremented at each “rising edge” event detected by the sensor:

- For **MAG format**: With each new magnet detection (magnet present)
- For **MOV format**: At the beginning of each movement (depending on the submitted threshold)
- For **DI format**: With each new logical state 1 of digital input (input shorted)
- For **PIR format**: With each new infrared movement detected

The counter overflow value is 32767 (maximum counter value before reset to zero). The counter resets when a *Connected mode* command ***“RAZ_COUNT”*** is sent, or when the tag reboot.

## 4 ELA INNOVATION BLUE RANGE CONFIGURATION

ELA Innovation BLUE range products are equipped with a NFC chip used for tag configuration. This chip, used with Device Manager suite, allows to write operating parameters, among the following fields:

| ***PARAMETER*** | ***POSSIBLE VALUES*** | ***ACTION*** | ***AVAILABILITY*** |
|---|---|---|---|
| **Name** | Maximum 15 characters [0-9 ; A-Z ; a-z ; SPACE, _, -] | Definition of the tag *Name*, transmitted by *Advertising* | Complete *Blue* range |
| **Enable** | True / False | *True*: Enable product operation.<br>*False*: Turn OFF the product. | Complete *Blue* range |
| **Power** | [-40, -20, -16, -12, -8, -4, 0, +3, +4] | Definition of the product BLE power emission, unit is dBm | Complete *Blue* range |
| **Format** | [Id, T, RHT, MAG, MOV, ANG, iBeacon, Eddystone, Analog IN, Digi IN, Digi OUT, PIR] | Definition of BLE data sent | According to product |
| **Advertising interval** | [0.1 -> 10] | Definition de la periode d'emission des trames BLE en secondes | Complete *Blue* range |
| **UUID (iBeacon)** | 32 characters [0-9 ; A-F] | Definition of iBeacon *UUID*, transmitted by *Advertising* | *Id* products with iBeacon |
| **Major (iBeacon)** | 4 characters [0-9 ; A-F] | Definition of iBeacon *Major*, transmitted by *Advertising* | *Id* products with iBeacon |
| **Minor (iBeacon)** | 4 characters [0-9 ; A-F] | Definition of iBeacon *Minor*, transmitted by *Advertising* | *Id* products with iBeacon |
| **NID (Eddystone)** | 20 characters [0-9 ; A-F] | Definition of the Eddystone *NID*, transmitted by *Advertising* | *Id* products with Eddystone |
| **BID (Eddystone)** | 12 characters [0-9 ; A-F] | Definition of the Eddystone B*ID*, transmitted by *Advertising* | *Id* products with Eddystone |
| **Log Interval** | [10 -> 86400] | Definition of the sensor data saving period for *datalogger* feature | *Blue* products with sensors |
| **Logger enable** | True / False | *True*: Enable datalogger feature.<br>*False*: Disable datalogger feature. | *Blue* products with sensors |
| **Acc. threshold** | [32 ;8000] | Definition of the acceleration threshold for MOV format, unit is mg | *MOV* products |
| **MOV sensor sensitivity** | High sensitivity / Energy saving | *High sensitivity*: configure accelerometer with high sensitivity mode – ***12,5Hz acquisition frequency***<br>*Energy saving*: configure accelerometer with energy saving mode – ***1,6Hz acquisition frequency*** | *MOV* and *ANG* products with *firmware vers. >2.2.0* |
| **MAG sensor reactivity** | Reactive / Energy saving | *Reactive*: configure MAG sensor with reactive mode – ***acquisition is made in real-time***<br>*Energy saving*: configure MAG sensor with energy saving mode – ***data sampling every 1.5 seconds*** | *MAG* products with *firmware vers. >2.2.0* |
| **PIR sensor sensitivity** | [0,1,2,3] | *Sensitivity level for the PIR sensor: define the maximum detection distance*<br>*0: 50cm*<br>*1: 1m*<br>*2: 2m*<br>*3 :5m* | *PIR* products with *firmware vers. >3.0.1* |

| PARAMETER | POSSIBLE VALUES | ACTION | AVAILABILITY |
|---|---|---|---|
| **Mfr. Data Enable** | True / False | *True*: Enable data transmission in ***Manufacturer Specific Data*** mode.<br>*False*: Enable data transmission in ***Service Data*** mode. | Complete *Blue* range |
| **MFR. Data ID** | 12 characters [0-9 ; A-F] | Definition of an hexadecimal identifier used in Id format when ***Manufacturer Specific Data*** are enabled. | Only used in *Id* products |
| **Battery voltage presence** | True/false | *True* : Enable ***Battery voltage transmission into Scan Response*** frame.<br>See related section of this document. | All products with *firmware vers. >3.0.0* |

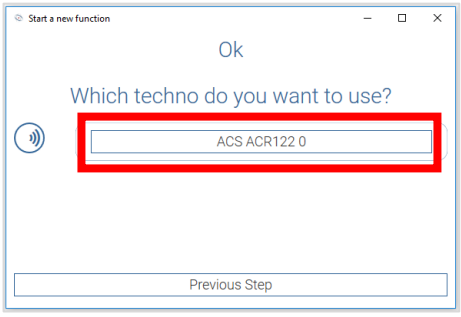
## *2.1. TAG CONFIGURATION USING DEVICE MANAGER PC SOFTWARE*

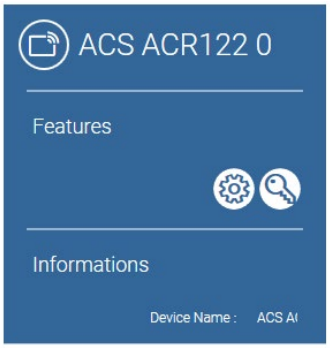

**1. Connect a NFC reader** to your desktop (example: NFC R/W 01 - ref. ACIOM177)

**2.** Start the "**Device Manager**" of your desktop


**3.** Dans le panneau principal de l'application, lancer le **widget « NFC »**


**4.** Choose the available **NFC reader** by **clicking** on the button 



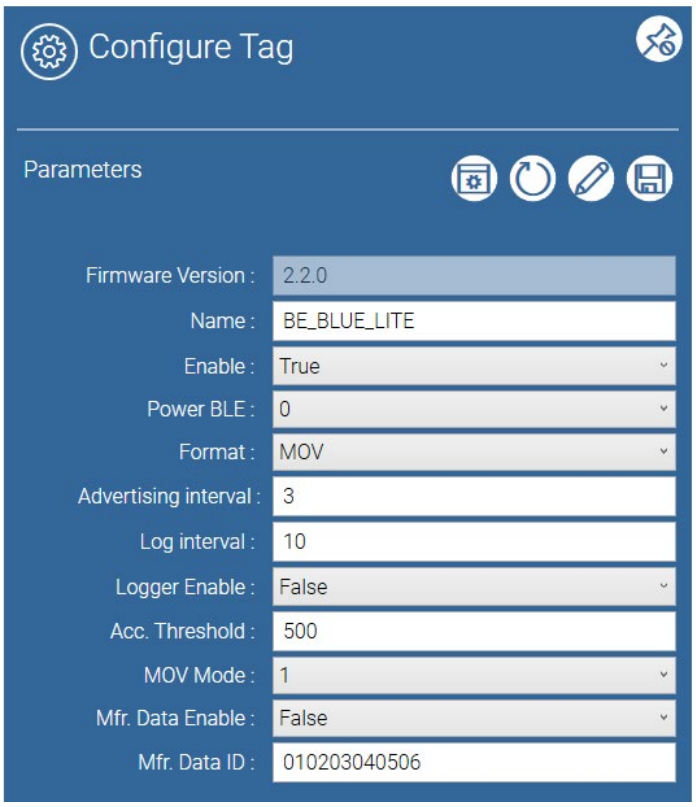

Once reader is selected, this window appears

**5. Place the tag on the NFC reader**

**6. Click** on the "***Configuration***" pictogram  to bring up the tag configuration window:


**7. Click** on « Refresh »  to bring up the current configuration read from the tag.

Configuration example




 : Factory settings parameters

 : Read and refresh tag configuration

 : Write the configuration to the tag

 : Display datalogger options (when applicable to the tag)

 : Display tag security options (when applicable to the tag)

 : Save of the configuration

*Configuration example, the fields are displayed according to the tag configuration.*

## 2.2. TAG CONFIGURATION USING A SMARTPHONE


## 2.3. SETTINGS RESTRICTIONS

### 4.3.1 Restrictions applying to the “Name” field

- Name must include **less than or up to 15 characters**
- Name **should not contain special characters** (but rather only letters, numbers, spaces, dash - and underscore _).

### 4.3.2 Datalogger restrictions

- When the "**Logger Enabled**" **field** of the NFC settings located under the device manager is **disabled**, the **tag reboots** and you will **lose all registered data** contained in the data logger.
- If you proceed to a **complete re-setting** of the tag by NFC, **data** contained in the **data logger is erased** from the tag memory.

### 4.3.3 Connected mode restrictions

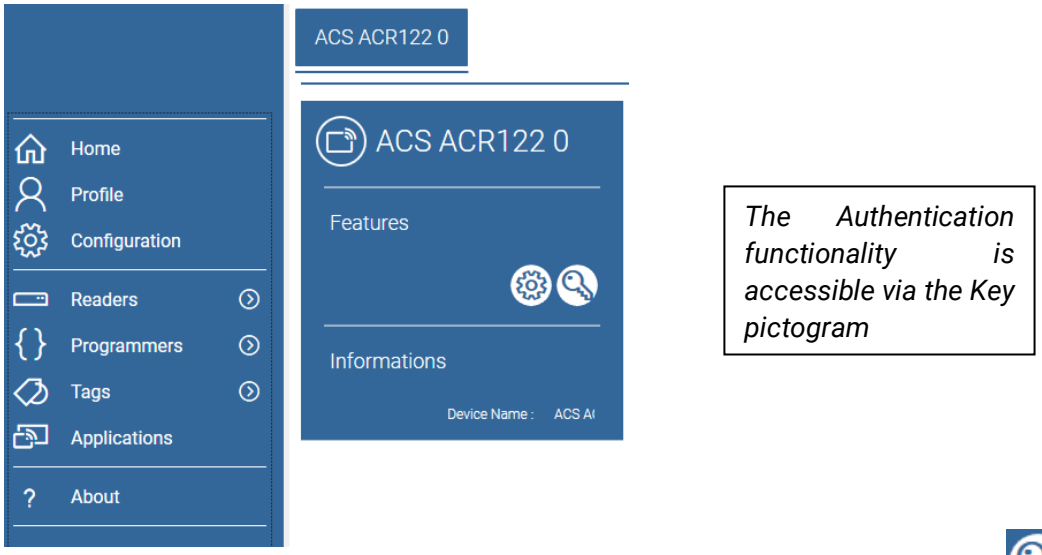
- If the tag is connected to a device and is approached by a NFC field, the tag will disconnect and reboot.

### 4.3.4 Other restrictions

- **iBeacon format**
  - You must fill in the complete UUID field of the iBeacon format: 32 characters ([0-9]; [A-F]).
  - You must fill in the complete Major field of the iBeacon format: 4 characters ([0-9]; [A-F]).
  - You must fill in the complete Minor field of the iBeacon format: 4 characters ([0-9]; [A-F]).
- **Eddystone format**
  - You must fill in the complete NID field of the Eddystone format: 20 characters ([0-9]; [A-F]).
  - You must fill in the complete BID field of the Eddystone format: 12 characters ([0-9]; [A-F]).

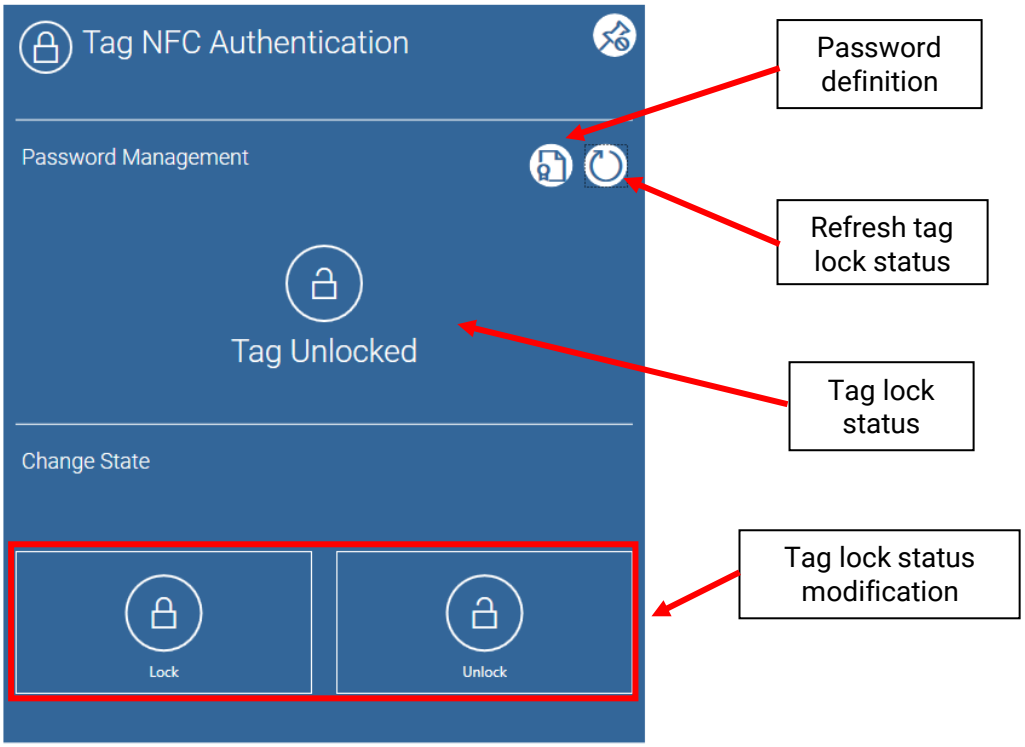
### *2.4. TAG NFC-CHIP PASSWORD PROTECTION*

Starting from firmware version 2.1.0, it is possible to protect the tags NFC-chip writing by a password. The functionality is accessible on ***Device Manager***, on the ***Programmers*** section:

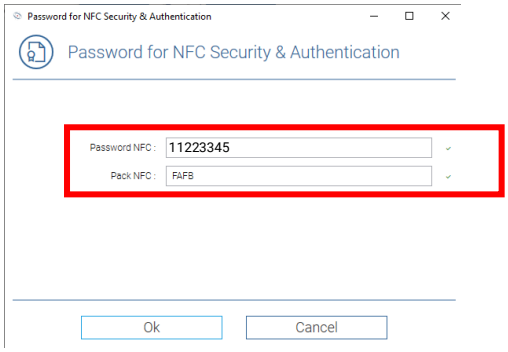



*The Authentication functionality is accessible via the Key pictogram*

**1.** Display the tag authentication window by clicking on the Key pictogram: 



**2.** Click on **Define password** 

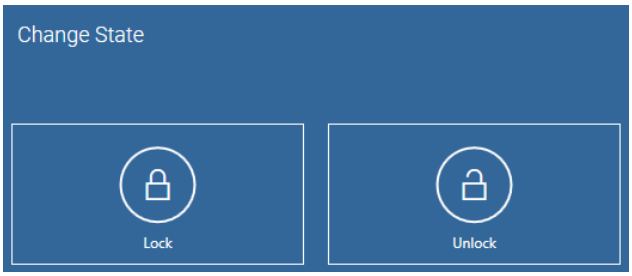



- NFC password must have exactly **8 hexadecimal characters default value is 11223345.**
- Pack NFC validate the authentication of the tag but has no consequence on the password modification. It is recommended to leave it to its defaults value: 0x***FAFB***.

*Note: It is not possible to recover a lost password. If you forgot your password, it will be necessary to return the product to ELA Innovation.*

**3.** Click on **Update state**  to read the current lock status of the tag (locked / unlocked)

Into the ***Change State*** area, the transition from a *Lock/Unlock* or *Unlock/Lock* state is done by clicking on the ***Lock*** or ***Unlock*** icons:




*Note: If the password set in step 2. Is not correct, the Unlock command will have no effect on the tag.*

### *2.5. TEMPERATURE CALIBRATION NFC CONFIGURATION*

Starting from firmware version 3.0.0, it is possible to configure a 2nd-polynomial calibration that can be used to correct temperature sensor value, to adjust measurement precision.

#### General Information

Calibration uses the format $aT^2+bT+c$, where ***a***, ***b*** and ***c*** are configurable coefficients (T being the original temperature value measured by the sensor). These coefficients may only be written with tag configuration via NFC. The coefficients may be read in *connected mode*.

They are transmitted in the format ***X***e***Y***, where ***X*** is an integer between -32768 and 32767, followed by a exponent ***Y*** from -128 to 127. ***X***e***Y*** is equivalent to X.10^Y. Examples:

- 125e-5 = 0.00125
- 1e-2 = 0.01
- 12e-1 = 1.2

Examples of complete calibration procedure:

- Sensor reading before calibration = 25.00°C. Calibration polynomial [c, b, a]: [ 5e-1, 1e0, 0e0 ]. The calculated value is therefore: $T_{cal}$ = 25.5°C
- Sensor reading before calibration = 25.00°C. Calibration polynomial [c, b, a]: [ 0e0, 101e-2, 0e0 ]. The calculated value is therefore: $T_{cal}$ = 25.25°C

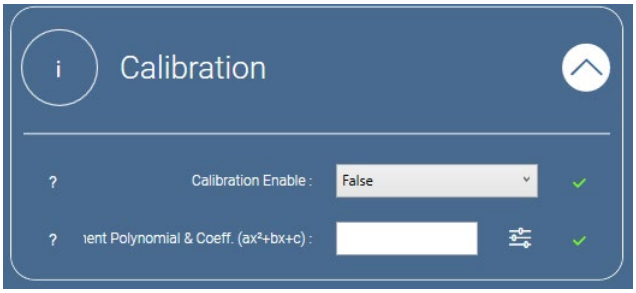
You may not change the calibration state (activated/deactivated) in connected mode when datalogging is running.

You may only define a first-degree polynomial (***b***T + ***c***), or a 0-degree polynomial (only ***c***, which can be used for testing). Coefficients are always sent in the following order: [c, b, a].

#### NFC Configuration

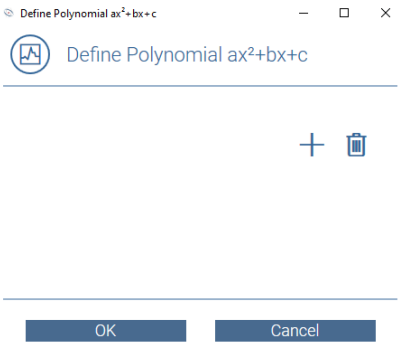
The fields for calibration and the calibration report are configured using the Device Manager application.

The window for configuring the fields is accessible via the "***Calibration***" icon. 



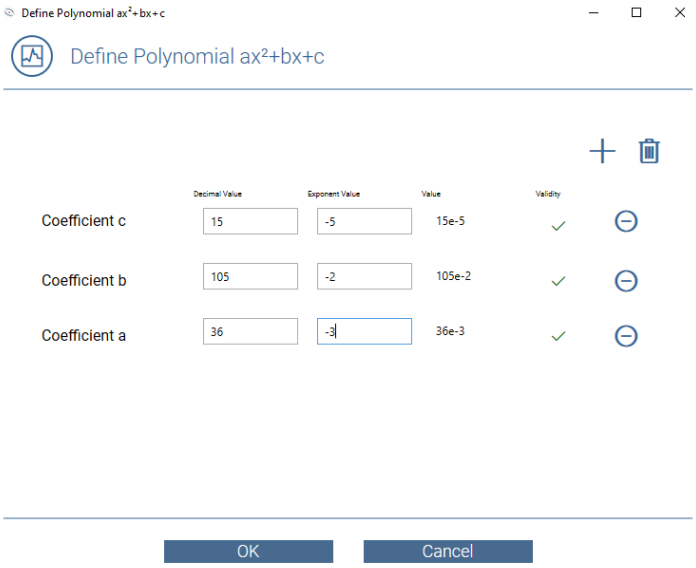

Here you can enable the calibration and also configure polynomial coefficients.

Click on the button to pop the Polynomial value configuration window :

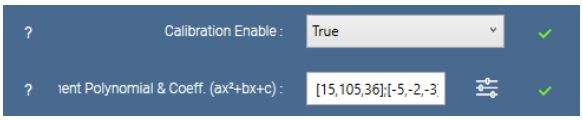



You can add a calibration coefficient by clicking on the button.
You can add up to 3 coefficients.

The window will check the coefficients and exponent values for integrity.



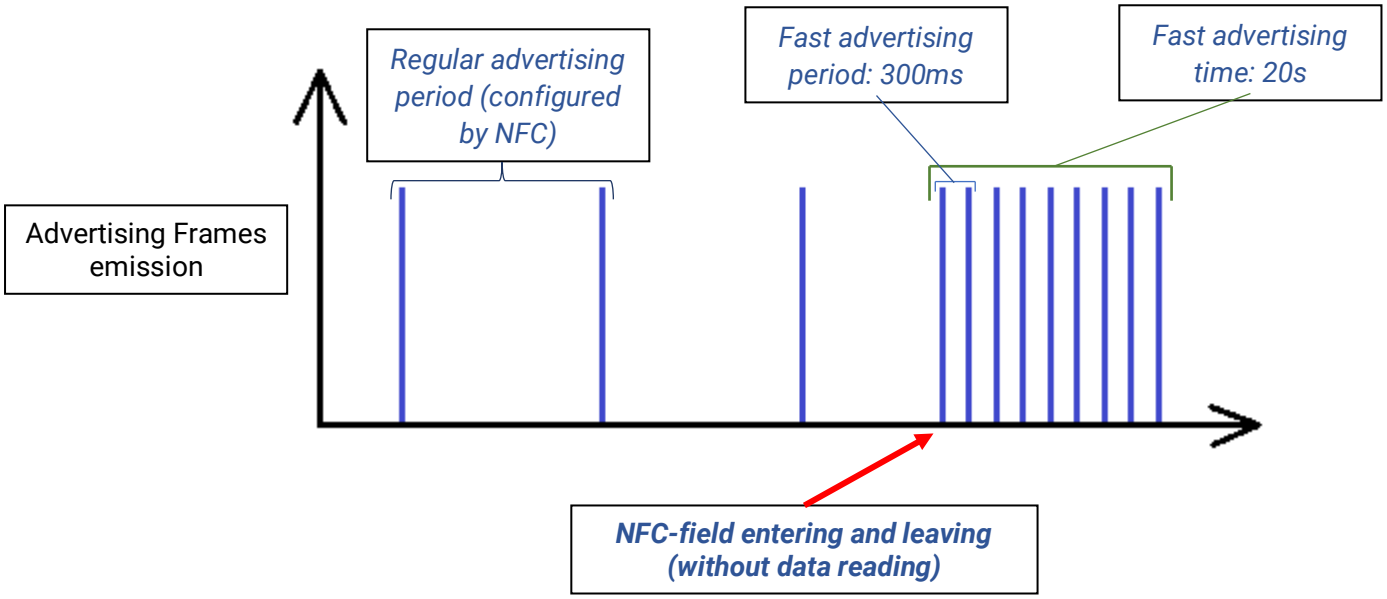

Click on ***OK*** when the values are set. They will appear on the previous window:




Do not forget to write the NFC configuration to the tag.

### 2.6. FAST ADVERTISING AFTER NFC-FIELD DETECTION

Starting from firmware version 3.0.0, the tag advertising period will be modified right after you approach an NFC-field to the tag.




After 20 seconds, the advertising period will come back to its normal value if there is not any connection to the tag. There is no need to read the NFC memory to activate this function, any field leaving will trigger it.

This behaviour also happen when the tag reboot, after a reconfiguration for example.

This behaviour can allow easier connection to tags which advertising period is configured with a value greater than 3 seconds.

Note: The advertising is stopped when the tag is on an NFC-field, thus the advertising will resume right after the tag leave the field.

# 5 CONTENT OF ADVERTISING DATA

Based on the sensor information obtained during configuration, the tag creates the frame that will be transmitted. It is possible to send sensor or identification data through "***Service Data***" or through "***Manufacturer Specific Data***". These « ***Mfr data*** » are company-specific BLE frame fields, which can be used to transmit data in *advertising* packets. If the ***Manufacturer Specific Data*** are not enabled, all sensor data are sent through ***Service Data***.

It is possible to find the detailed *BLE frame specification* in the ELA download area:

- https://elainnovation.com/downloads.html

## 3.1. SENSOR DATA IN « SERVICE DATA » FRAME

- « T », « T EN » and « T Probe » formats example:

Raw data:
0x02010605166E2AAB0A1009425055434B53543830304131324E41

0x6E2A : *Temperature* service

*T°* data:
- 0xAB : LSB
- 0x0A : MSB

T° = 0AAB = 2731 * 0.01 = 27.31°C

Details :

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x6E2AAB0A |
| 11 | 0x09 | 0x425055434B53543830304131324E41 |

Name (ASCII)

Note: For a negative temperature, data is sent in 2-complement: for example, -27.31°C is 6E2A55F5

- « MAG » format example:

Raw data:
0x02010605166062AFB0A1009425055434B53543830304131324E41

0x062A : *Alert Status* service

*MAG* data:
- 0xFB : LSB
- 0x0A : MSB

| Hexa. | 0 | A | F | B |
|---|---|---|---|---|
| Binary | 0000 | 1010 | 1111 | 1011 |

⇨ 1: instantaneous sensor state (magnet present)
⇨ 1010 1111 101: event counter value on 15 bits, 1405 in this example

Details :

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x062AFB0A |
| 11 | 0x09 | 0x425055434B53543830304131324E41 |

Name (ASCII)

- « MOV » format example:

Raw data:

0x0201060516062AFB0A1009425055434B5354
3830304131324E41

Details :

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x062AFB0A |
| 11 | 0x09 | 0x425055434B53543830304131324E41 |

Name (ASCII)

0x062A : *Alert Status* service

*MOV* data:
- 0xFB : LSB
- 0x0A : MSB

| Hexa. | 0 | A | F | B |
|---|---|---|---|---|
| Binary | 0000 | 1010 | 1111 | 1011 |

⇨ 1 : instantaneous sensor state (movement detection)
⇨ 1010 1111 101 : event counter value on 15 bits, 1405 in this example

- « ANG » format example:

Raw data:
0x0201060516A12A05FF0AFBC90755434B53
543830304131324E41

Details :

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0xA12A05FF0AFBC907 |
| 11 | 0x09 | 0x55434B53543830304131324E41 |

Name (ASCII)

0xA12A : *Magnetic 3D* service
*ANG* data:

- 0x05 : X-axis LSB
- 0xFF : X-axis MSB
- 0x0A : Y-axis LSB
- 0xFB : Y-axis MSB
- 0xC9 : Z-axis LSB
- 0x07 : Z-axis MSB

X-axis : 0xFF05 => -251 mg acceleration on X-axis
Y-axis : 0xFB0A => -1270 mg acceleration on Y-axis
Z-axis : 0xFF05 => +1993 mg acceleration on Z-axis

Note: Values are coded on 16-bits with 12 significant bits and 4 sign bits. The values are expressed in mg (+2g/-2g). Negative data are sent in 2-complement.

- « RHT » format example:

Raw data:
0x02010605166E2A5E0A004166F2A3010094
25055434B53543830304131324E41

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x6E2A5E0A004166F2A30 |
| 11 | 0x09 | 0x55434B53543830304131324E41 |

Name (ASCII)

0x6E2A : *Temperature* service
0x6F2A : *Humidity* service

*RH* data:
- 0x30 : RH data i.e. 48% relative humidity

*T°* data:
- 0x5E : LSB
- 0x0A : MSB

T° = 0A5E = 2654 * 0.01 = 26.54°C

Note: For a negative temperature, data is sent in 2-complement

- « DI » format example:

Raw data:

0x0201060516062A0A0004163F2A020E0942
455F544553545F544F52494E

Details :

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x062A0A00 |
| 11 | 0x09 | 0x42455F544553545F544F52494E |

Name (ASCII)

0x062A : *Alert Status* service

*DI* data:
- 0x0A : LSB
- 0x00 : MSB

| Hexa. | 0 | 0 | 0 | A |
|---|---|---|---|---|
| Binary | 0000 | 0000 | 0000 | 1010 |

⇨ 0 : instantaneous input state (input state OFF)
⇨ 0000 0000 0000 101 : event counter value on 15 bits, 1405 in this example

Note: In this example, the *DI* data is 0x000A, i.e. counter is at 4 increments (transition from state 0 to state 1 on digital input), and the instantaneous input state is 0 (input in OFF state).

- « Analog IN » format example:

Raw data:
0x0201060516582AAC0B1009425055434B5354
3830304131324E41

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x582AAC0B |
| 11 | 0x09 | 425055434B53543830304131324E41 |

Name (ASCII)

0x582A : *Analog Output* service

*Analog IN* data:

- 0xAC : LSB
- 0x0B : MSB

Analog voltage measure: 0x0BAC = 2988mV

Note: Analog input voltage measure are in mV unit.

- « PIR » format example:

Raw data:
0x0201060516782A1B00110942505543 4B5354
3830304131324E41

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x782A1B00 |
| 11 | 0x09 | 425055434B53543830304131324E41 |

Name (ASCII)

0x282A : *Rainfall service*

*PIR* data:
- 0x1B : LSB
- 0x00 : MSB

| Hexa. | 0 | 0 | 1 | B |
|---|---|---|---|---|
| Binary | 0000 | 0000 | 0001 | 1011 |

⇨ 1: instantaneous sensor state (infrared movement detected)
⇨ 0000 0001 101: event counter value on 15 bits, 13 in this example

### *3.2. SENSOR DATA IN « MANUFACTURER SPECIFIC DATA » FRAME*

1) ELA Innovation *Company Identifier* (*CIN*) is 0x0757.

2) In *ELA_ID* and *Digi OUT* formats, it is possible to configure a hexadecimal number (max. 0xFFFFFFFFFFFF) which will be sent in advertising frame. This field is named "***Manufacturer Data ID***" in the NFC configuration. This number is called "MFR_ID" in this document frame formats.

In « *Manufacturer Specific Data* », sensor data are encoded the same way as in "Service Data" mode, only the "data type" in hexadecimal is modified:

- 0x16: for « ***Service data*** »
- 0xFF: for « ***Manufacturer Specific Data*** »

*Manufacturer Specific Data* are transmitted as follows:

| ELA CIN | DataInfo | Data |
|---|---|---|

- <u>Here is an example with a « T » format frame:</u>

Raw data:
0x02010606FF5707124D0A0B09425055434B
53543830304131324E41

Details :

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0xFF | 0x5707124D0A |
| 11 | 0x09 | 0x425055434B53543830304131324E41 |

Name (ASCII)

0x5707 => **ELA CIN** : ELA Innovation *Company Identifier number*
- 0x07 : LSB
- 0x57 : MSB

Soit 0x**0757**.

**0x12 : DataInfo :** Indicate the following data : Temperature data here

**Data** : T° data:
- 0x4D : LSB
- 0x0A : MSB

T° = 0A4D = 2637 * 0.01 = 26.37°C

All sensor data are listed in the table below:

| Field | | Length | Description |
|---|---|---|---|
| **Temperature data (T)** | DataInfo | 1 byte | 0x12 (bit7-4=1 et bit3-0=2) |
| | Data | 2 bytes | Temperature on 16 signed bits / 0,01°C step |
| **Humidity data (RH)** | DataInfo | 1 byte | 0x21 (bit7-4=2 and bit3-0=1): Relative Humidity |
| | Data | 1 byte | Humidity on 8 unsigned bits / 1 % step RH de 0 à 100 % |
| **Magnetic data (MAG)** | DataInfo | 1 byte | 0x32 (bit7-4=3 and bit3-0=2) |
| | Data | 2 bytes | Event (state change) counter on the 15 (unsigned) MSB<br>Instantaneous state on LSB |
| **Movement data (MOV)** | DataInfo | 1 byte | 0x42 (bit7-4=4 and bit3-0=2) |
| | Data | 2 bytes | Event counter (threshold overflow) on the 15 (unsigned) MSB<br>Instantaneous state on LSB |
| **Infrared movement data (PIR)** | DataInfo | 1 byte | 0x92 (bit7-4=9 and bit3-0=2) |
| | Data | 2 bytes | Event (infrared movement detected) counter on the 15 (unsigned) MSB<br>Instantaneous state on LSB |
| **Accelerometer data (ANG)** | DataInfo | 1 byte | 0x56 (bit7-4=5 and bit3-0=6) |
| | Data | 6 bytes | X-axis acceleration on 16 signed bits (range +/-2G)<br>Y-axis acceleration on 16 signed bits (range +/-2G)<br>Z-axis acceleration on 16 signed bits (range +/-2G) |
| **Digital Input data (DI)** | DataInfo | 1 byte | 0x62 (bit7-4=6 and bit3-0=2) |
| | Data | 2 bytes | Event (input state change) counter on the 15 (unsigned) MSB<br>Instantaneous input state on LSB |
| **Analog Input data (AI)** | DataInfo | 1 byte | 0x72 (bit7-4=7 and bit3-0=2) |
| | Data | 2 bytes | Voltage measured in mV on 16 unsigned bits |

Identifiers format data (*Id* and *DO*) offer to transmit an identifier configured by NFC:

| | | | |
|---|---|---|---|
| ***Id* format data** | DataInfo | 1 byte | 0x06 (bit7-4=0 and bit3-0=6) |
| | Data | 6 bytes | *MFR_ID* configured by NFC |
| **Digital output data (*DO*)** | DataInfo | 1 byte | 0x86 (bit7-4=8 and bit3-0=6) |
| | Data | 6 bytes | *MFR_ID* configured by NFC |

*BLE frame specification* is available in the ELA download area:

- https://elainnovation.com/downloads.html

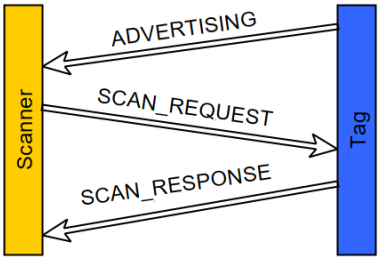
### 3.3. SCAN RESPONSE FRAME

In some formats and versions, the tag can send a frame called « Scan Response frame ».

Once an advertising packet has been received by a scanner, further information can be requested. Then the tag responds with the “scan response” frame.

This frame is located right after the advertising frame, and contains different data depending on the version and format.

The data sent in “Scan response” frame is also formatted either in Service mode or in Manufacturer Specific mode.




### 3.4. BATTERY INFORMATION

#### Battery capacity

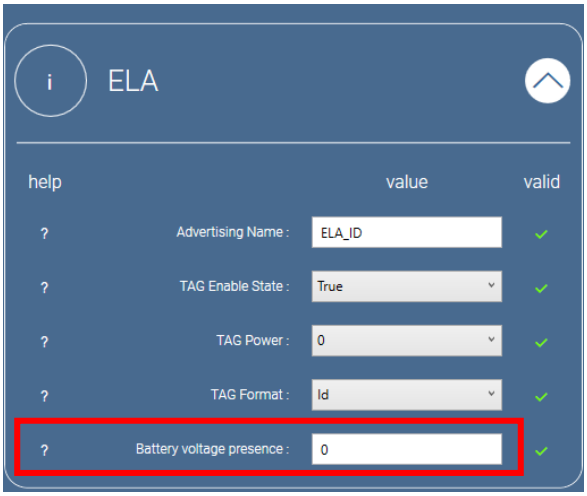
ELA Innovation's tags are based on the transmission of battery information in the Scan Response when the capacity of the battery falls below 15%. The formatting of the information is as follows:

| Frame type | | Service Data | Service Data | Mfr. Spec. Data |
|---|---|---|---|---|
| Version | | 1.0.0, 2.0.0, 2.1.x | ≥2.2.0 | ≥2.0.0 |
| Transmission | | Batt. capacity < 15% | Batt. capacity < 15% | Batt. capacity < 15% |
| **Frame bytes** | 1 | Length : 0x04 | Length: 0x04 | Length : 0x05 |
| | 2 | Type : 0x16 | Type : 0x16 | Type : 0xFF |
| | 3 | Battery Serv. LSB : 0x0F | Battery Serv. LSB : 0x19 | ELA_CIN_LSB : 0x57 |
| | 4 | Battery Serv. MSB : 0x18 | Battery Serv. MSB : 0x2A | ELA_CIN_MSB: 0x07 |
| | 5 | Batt. data (%) | Batt. data (%) | BATT_DATA_ID: 0xF1 |
| | 6 | Not used | Not used | Batt. data (%) |
| | 7 | Not used | Not used | Not used |

#### Battery voltage

From version 3.0.0 onwards, it is possible to transmit battery voltage information for all formats. For this purpose, the "**Battery voltage presence**" option must be configured in the NFC memory.

**When the option is activated, the tag no longer transmits battery capacity information below 15%.**




Once the option is enabled, the battery voltage information is transmitted in the "Scan Response" frame with the following formatting:

| Frame type | | All |
|---|---|---|
| Version | | ≥3.0.0 |
| Transmission | | Battery voltage presence = 1 |
| **Frame Bytes** | 1 | Length : 0x06 |
| | 2 | Type : 0xFF |
| | 3 | ELA_CIN_LSB : 0x57 |
| | 4 | ELA_CIN_MSB: 0x07 |
| | 5 | BATT_DATA_ID: 0xF2 |
| | 6 | Batt. voltage (mV) LSB |
| | 7 | Batt. voltage (mV) MSB |

Frame examples showing battery information:

| Received frame: ELA ID, Service Data, v3.0.0<br>Battery voltage presence = 0 | |
|---|---|
| **Name** | *BE_BATTERY* |
| **Battery cap.** | *13% (0x0D)* |

Raw data:

0x0201060B0942455F424154544552590416192A0D

Details:

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 11 | 0x09 | 0x42455F42415454455259 |
| 4 | 0x16 | 0x192A0D |

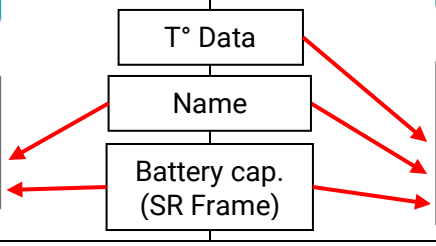
| Received frame: ELA T, MFR Spec. Data, v3.0.0<br>Battery voltage presence = 0 | |
|---|---|
| **Name** | *BE_BATTERY* |
| **Measured temp.** | *27.12°C (0x0A98)* |
| **Battery cap** | *13% (0x0D)* |

Raw data:

0x02010606FF570712980A0B0942455F4241545445525905FF5707F10D

Details:

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 6 | 0xFF | 0x570712980A |
| 11 | 0x09 | 0x42455F42415454455259 |
| 5 | 0xFF | 0x5707F10D |




In Eddystone and iBeacon formats, the battery information is located before the Tag Name :

| Received frame : iBeacon, v2.1.0 | |
|---|---|
| **Name** | *BE_BATTERY* |
| **Battery cap.** | *13% (0x0D)* |

Raw data:

0x0201061AFF4C000215010203040506070809 0A0B0C0D0E0F10020B010AC404160F180D0B0942455F42415454455259

Details:

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 26 | 0xFF | 0x4C000215010203040506070809 0A0B0C0D0E0F10020B010AC4 |
| 4 | 0x16 | 0x0F180D |
| 11 | 0x09 | 0x42455F42415454455259 |

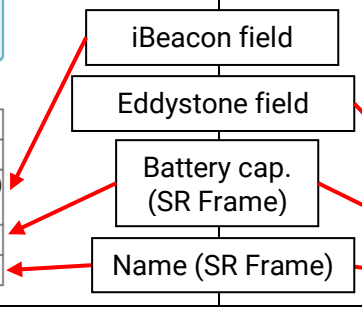
| Received frame: Eddystone, v3.0.0<br>Battery voltage presence = 0 | |
|---|---|
| **Name** | *BE_BATTERY* |
| **Battery cap.** | *13% (0x0D)* |

Raw data:

0x0201060303AAFE1716AAFE00ED0102030405060708090A010203040A0B000004 16192A0D0B0942455F42415454455259

Details:

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 3 | 0x03 | 0xAAFE |
| 23 | 0x16 | 0xAAFE00ED010203040506070809 0A010203040A0B0000 |
| 4 | 0x16 | 0x192A0D |
| 11 | 0x09 | 0x42455F42415454455259 |




| Received frame: iBeacon, v3.0.0<br>Battery voltage presence = 1 | |
|---|---|
| **Name** | *BE_BATTERY* |
| **Batt. voltage** | *2.478V (0x09AE)* |

Raw data:

0x0201061AFF4C000215010203040506070809 0A0B0C0D0E0F10020B010AC406FF5707F2AE090B0942455F42415454455259

Details:

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 26 | 0xFF | 0x4C000215010203040506070809 0A0B0C0D0E0F10020B010AC4 |
| 6 | 0xFF | 0x5707F2AE09 |
| 11 | 0x09 | 0x42455F42415454455259 |

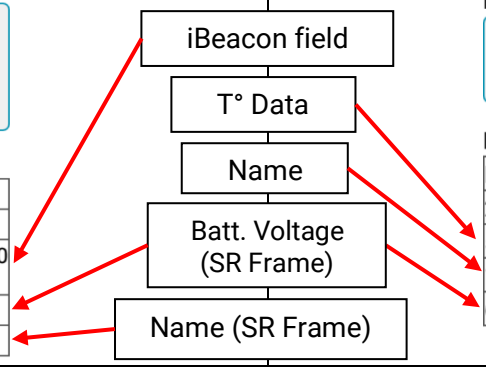
| Received frame: ELA T, Service Data, v3.0.0<br>Battery voltage presence = 1 | |
|---|---|
| **Name** | *BE_BATTERY* |
| **Measured temp** | *21.87°C (0x088B)* |
| **Batt. voltage** | *2.988 V (0x0BAC)* |

Raw data:

0x02010605166E2A8B080B0942455F4241545445525906FF5707F2AC0B

Details:

| LEN. | TYPE | VALUE |
|---|---|---|
| 2 | 0x01 | 0x06 |
| 5 | 0x16 | 0x6E2A8B08 |
| 11 | 0x09 | 0x42455F42415454455259 |
| 6 | 0xFF | 0x5707F2AC0B |



*3.5. INFORMATION ABOUT IBEACON AND EDDYSTONE FORMATS*

- **Description of owner formats such as Apple (iBeacon) and Google (Eddystone)**


  - Tags settings available in iBeacon format
  - Compliance with Apple specific data such as:

    ***Flags – Length – Type - Company ID - Beacon Type - Proximity UUID - Major - Minor***

  - You can add an additional "Name", which is send it in the "Scan Response" BLE frame and configure it in the "Name" field from the Device Manager
  - Specification: https://developer.apple.com/ibeacon/


  - Tags settings available in Eddystone UID format
  - Compliance with specific Google Data Eddystone UID format such as:

    ***A unique, static ID with a 10-byte Namespace component and a 6-byte Instance component***

  - You may add an additional "Name", which is send in the "Scan Response" BLE frame and configure it in the "Name" field
  - Specification: https://developers.google.com/beacons/overview

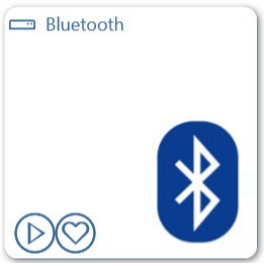
### *3.6. DATA VIZUALISATION USING DEVICE MANAGER*

The ELA Innovation *Device Manager* application can perform BLE scans in order to view advertising data from BLE ELA Innovation products:

**1. Enable internal Bluetooth** or connect a Bluetooth device (typ. Dongle) to your PC

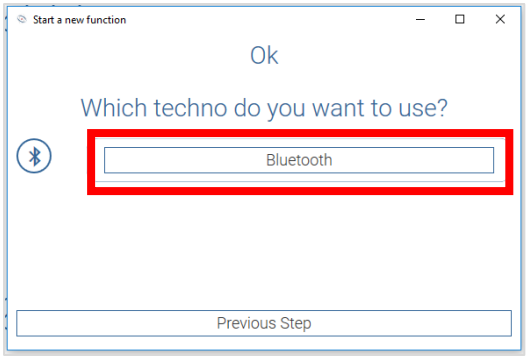
**2.** Launch the "**Device Manager**" desktop application


**3.** Start the **"Bluetooth" widget** by clicking button




**4.** Start the **BLE device search**

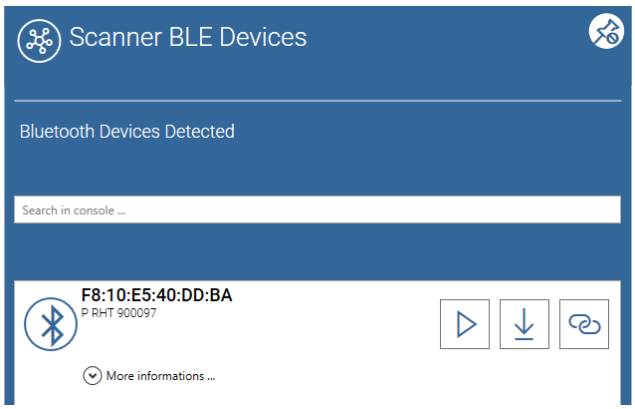
**5. Click** on the found device. The ***Bluetooth*** windows appears



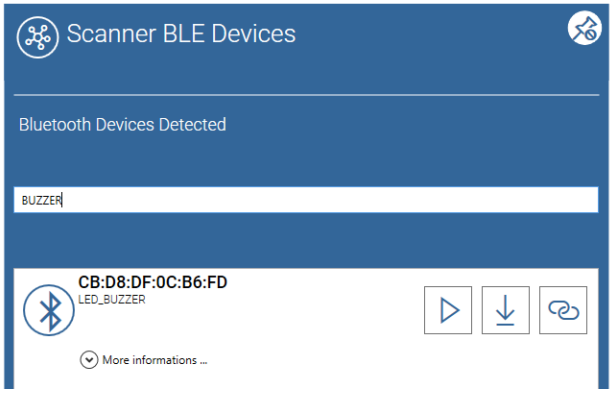


Once reader has been chosen, this window appears

**6.** Start the ***BLE Device scanner***

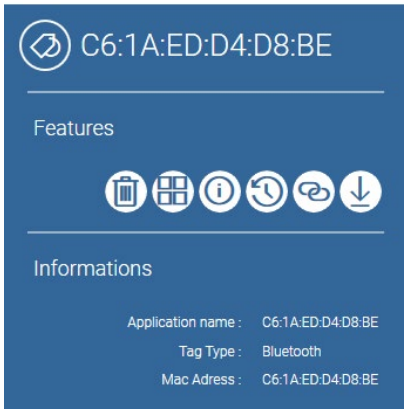



*On this window, it is possible to search for a Name or MAC Address from the entire drop-down list (see next page)*




*Tag name has been filtered*

**7.** Click on the ***Tag info visualization*** button.  This window opens:




**8.** It is possible to view the tag data with the Information button.  This opens a window displaying tag Name, RSSI and MAC address:

# 6 CONNECTED MODE OPERATION

In "***Connected Mode***", a link is established between two devices and only they can communicate and exchange with each other. You may establish a connection using a smartphone or a mobile application, or with a PC equipped with the ELA "*Device Manager*" application (provided you activated Bluetooth or connected a BLE dongle to the PC).

- The *Advertising Recurrence* must be less than or equal to 3 seconds to be able to establish a connection.
- Once you enter "*Connected Mode*", "***Advertising***" **is stopped by default**.
- It is possible to send commands to the tag to perform special actions or read data.
- It is possible to get a record of saved data (***Datalogger***) using *Connected Mode*. This datalogger will contain sensor data saved at a defined period with a timestamp for each data.

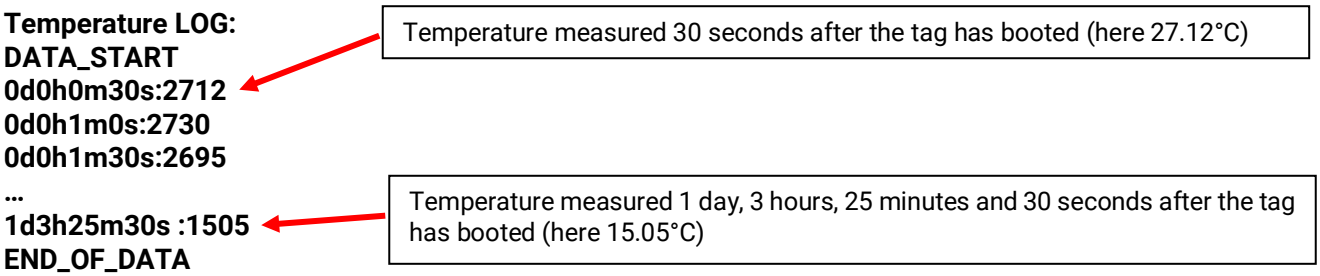
## 4.1. CONNECTED MODE LIST OF COMMANDS

| COMMANDS | ACTIONS | MINIMUM FIRMWARE VERSION |
|---|---|---|
| **LED_ON** | Turn ON the LED (infinite Blink) | **≥1.0.0** |
| **LED_OFF** | Turn OFF the LED | **≥1.0.0** |
| **LED_ON XX** | Turn ON the LED (for XX seconds) | **≥2.0.0** |
| **BUZZ_ON** | Turn ON the buzzer (Repeated beep) | **≥1.0.0** |
| **BUZZ_OFF** | Turn OFF the buzzer | **≥1.0.0** |
| **BUZZ_ON XX** | Turn ON the buzzer (for XX seconds) | **≥2.0.0** |
| **DIGI_ON** | Turn Digital Output to "ON" state | **≥2.1.0** |
| **DIGI_OFF** | Turn Digital Output to "OFF" state | **≥2.1.0** |
| **DIGI_ON XX** | Turn Digital Output to "ON" state for XX seconds | **≥2.1.0** |
| **RAZ_COUNT** | Counter reset | **≥2.0.0** |
| **LOG_DL** | Download datalogger values | **≥2.0.0 (non-EN12830)** |
| **LOG_RST** | Erase datalogger values and timestamp | **≥2.0.0 (non-EN12830)** |
| **GET_BATT_VOLTAGE** | Return battery voltage in mV | **≥2.1.0** |
| **GET_SENSOR_DATA** | Return the last measured sensor value | **≥2.2.0** |

### *4.2. SIMPLE DATALOGGER*

Ela innovation sensor tags can operate the "*Datalogger*" feature. The datalogger is a record of saved data, memorized while advertising, with each value associated with a time stamp, to be able to recover the moment when it was measured.

The datalogger can be retrieved in *Connected mode* using the "***LOG_DL***" command. The simple datalogger (non-EN12830) formatting for a temperature sensor with a log interval of 30 seconds is the following:




The simple datalogger formatting is the same for all ELA Innovation sensor products (xx**d**xx**h**xx**m**xx**s** followed by sensor data). The sensor data is the same as the one transmitted in advertising frames.

The "***LOG_RST***" command is used to delete datalogger data content.

### *4.3. BLUE PUCK T EN12830 (2018) & BLUE PUCK T PROBE DATALOGGER*

The EN12830 firmware has several new features:

- EN12830 Datalogger
- Calibration by 2nd-degree polynomial of temperature values
- Saving tag calibration values (Target values – measured values)

These EN12830 (2018) dedicated functionalities are protected by a BLE password. This password is inserted by the NFC configuration. The EN12830 tag configuration options are only available from *Device manager* version 1.3.0.

The PUCK T EN12830 dedicated documentation can be found in the download area:

- https://elainnovation.com/downloads.html

### *4.4. CONNECTED MODE RESTRICTIONS*

During a ***complete NFC re-configuration***, ***datalogger data is erased*** from the tag memory.

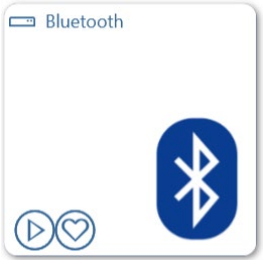
- If the tag is in *Connected Mode* and goes under an ***NFC-field***, then the tag will restart.

### *4.5. CONNECTING TO A BLE ELA INNOVATION TAG*

**1. Enable internal Bluetooth** or connect a Bluetooth device (typ. Dongle) to your PC

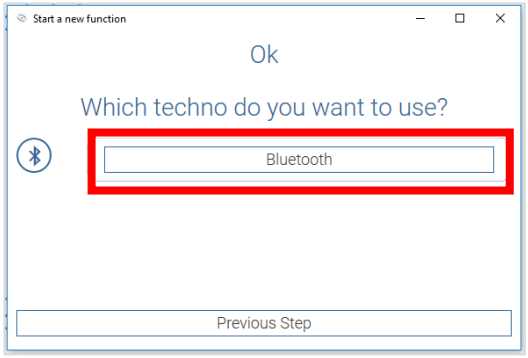
**2.** Launch the "***Device Manager***" desktop application

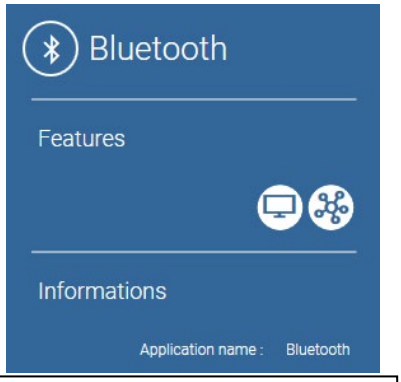

**3.** Start the "***Bluetooth***" ***widget*** by clicking button




**4.** Start the ***BLE device search***

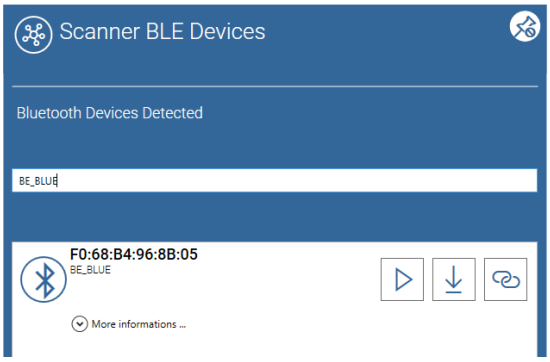
**5. Click** on the found device. The ***Bluetooth*** windows appears:



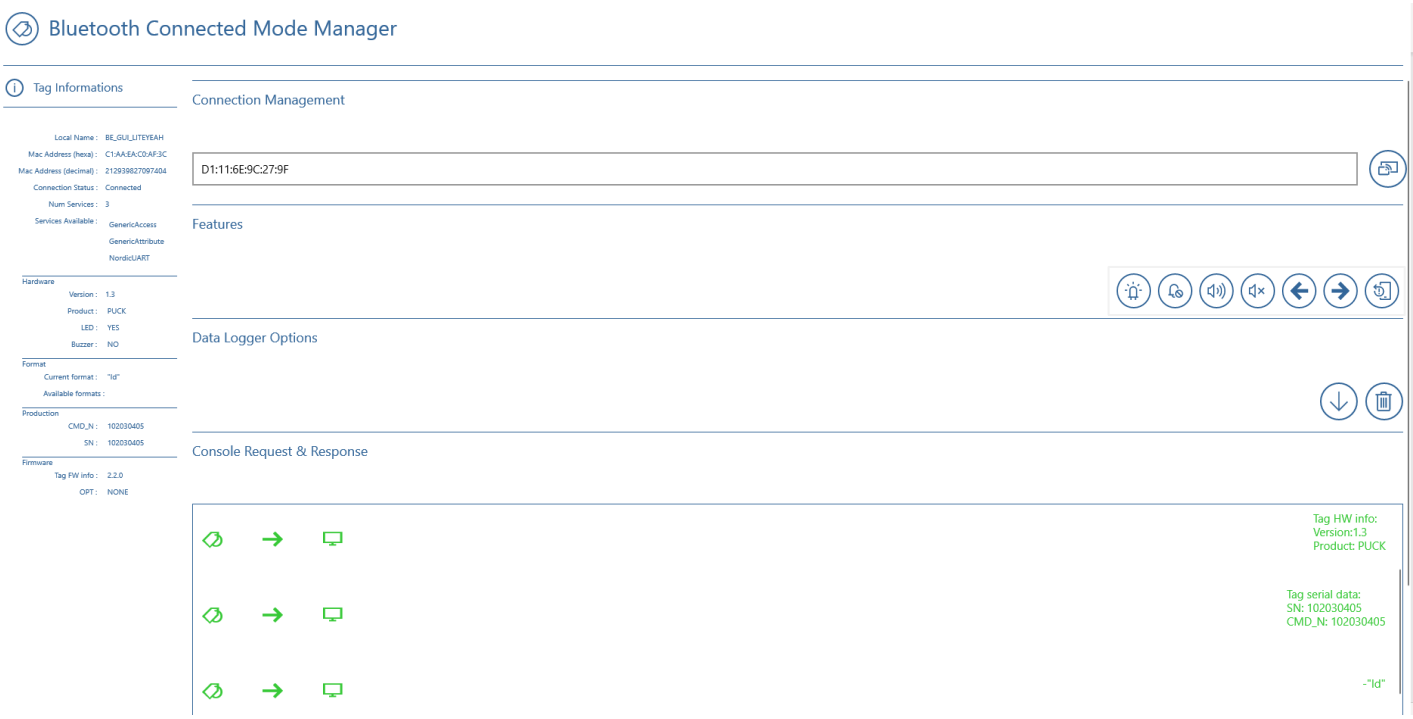

Once reader has been chosen, this window appears

**6.** Start the BLE Device scanner 




*On this window, it is possible to search for a Name or MAC Address from the entire drop-down list (see example)*

**7.** Start the connection by pressing icon  . The *Device Manager Connector* window opens:




- **Features:** Commands to send to the tag (see next page for commands syntax and use)
- **Informations :** Name –Mac Adress – Connection status – Available services
- **Hardware – Format - ...:** Services details and tag options

« Commands » description

| ICONS | COMMANDS | ACTIONS |
|---|---|---|
| | LED_ON | Turn ON the LED (infinite Blink) |
| | LED_OFF | Turn OFF the LED |
| | BUZZ_ON | Turn ON the buzzer (repeated beep) |
| | BUZZ_OFF | Turn OFF the buzzer |
| | DIGI_ON | Turn Digital Output to “ON” state |
| | DIGI_OFF | Turn Digital Output to “OFF” state |
| | RAZ_COUNT | Counter reset (for MAG, MOV and DI formats) |
| | LOG_DL | Download datalogger values |
| | LOG_RST | Erase datalogger values and timestamp |

- **LED & BUZZER commands:**

  For lifetime constraints, LED and BUZZER commands cannot be turned ON at the same time.

- **Datalogger download :**

  The « **LOG_DL** » command is used to download the recorded log data.
  Detailed Data according to sensor can be found on the application note on the ELA website.

# 7 PRODUCT OPERATION

## 4.6. OTAP FUNCTIONNALITY

OTAP (Over-The-Air Programming) is a method used to update a software, data or settings of a product without having to disassemble it and do it in a completely wireless way.

Ela Innovation products programmed with firmware version >3.0.0 can use OTAP Mechanism to update the tag embedded firmware, which can be done without having to return the product to ELA Innovation.

The OTAP procedure is secured by 2 methods:

- The switch into OTAP mode of ELA Innovation products is protected by a password that can be set by the user with NFC configuration
- The firmware update package is signed by a SHA256 private key.

The OTAP procedure and material requirements are described on the OTAP Application Note, available on the ELA Innovation Website.

## 8 NORMS & STANDARDS

### FCC Statement

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference; and
2. This device must accept any interference received, including interference that may cause undesired operation.

### Industry Canada Statement

This device complies with ISED's licence-exempt RSSs. Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and
2. This device must accept any interference received, including interference that may cause undesired operation.

- CE Mark

- FCC Mark

- RoHS Certified

- Bluetooth 4.2