

# Contents

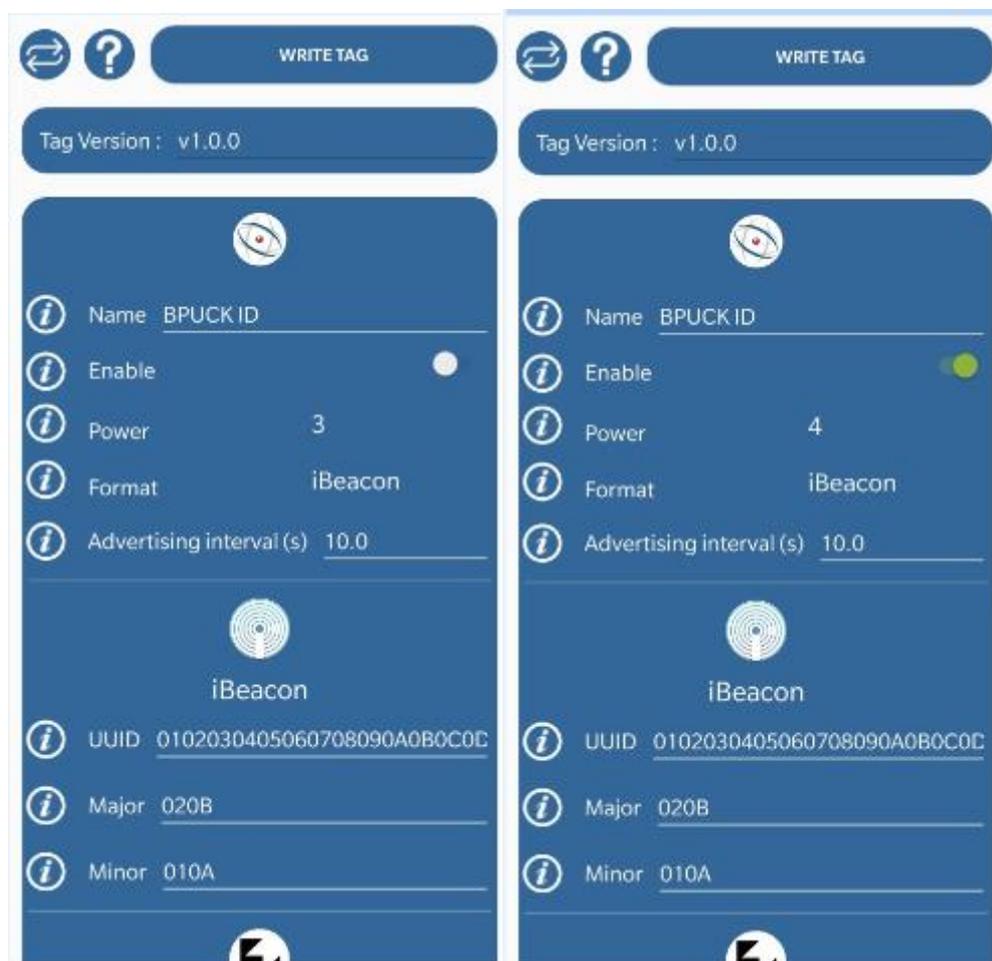
Enable your ELA Beacon or Sensor .....	2
Beacon testing instructions .....	3
Record parsing.....	5
Table 1. Beacon configuration parameters.....	6
Table 2. Beacon Packet structure .....	7
BLE sensors testing instructions .....	7
How to configure a unique sensor if there is no preset.....	9
Table 3. Sensor Elements AVL ID's .....	10
ELA Sensor additional information.....	11

## Enable your ELA Beacon or Sensor

To enable your ELA sensors or ELA Beacons use the ELA application

(Link:[https://play.google.com/store/apps/details?id=com.ela.mobileapp.elamobileapp&hl=en\\_US](https://play.google.com/store/apps/details?id=com.ela.mobileapp.elamobileapp&hl=en_US)). Your phone or device that is used for configuring ELA Sensors or Beacons has to have NFC (configurations are written to a sensor or beacon through NFC).

Below you can find on how to Enable and configure ELA Coin ID or ELA Puck ID to send in the iBeacon format.

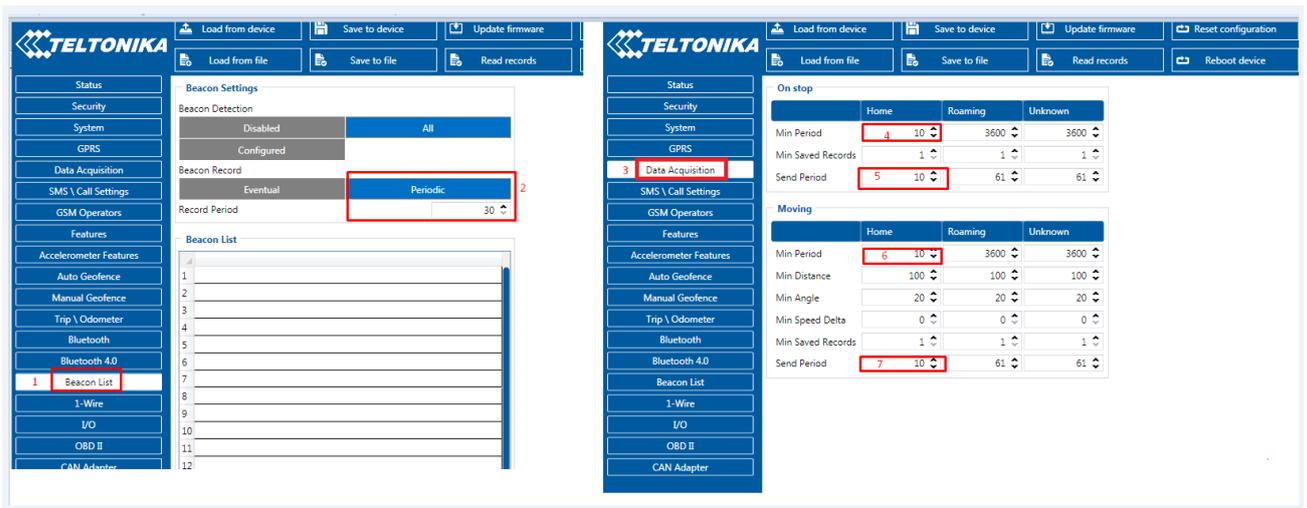


# Beacon testing instructions

Instructions made with firmware version: FMB.Ver.03.25.09.Rev.51

1. Update FMB device firmware to version: FMB.Ver.03.25.09.Rev.51 (or later firmware version which supports beacon functionality)
2. Turn beacons ON (or configure it to send beacon ID)
3. Configure FMB device to send data into your server or computer over Codec 8 extended data sending protocol. Added print screens shows only one of possible ways how to perform test.

Important: To perform test faster, configure Beacon record to → **Periodic**



The left screenshot shows the 'Beacon Settings' configuration page. The 'Beacon Detection' is set to 'All'. Under 'Beacon Record', 'Eventual' is selected, and 'Periodic' is highlighted with a red box and a '2' next to it. The 'Record Period' is set to '30'. The 'Beacon List' is empty.

The right screenshot shows the 'Data Acquisition' configuration page. The 'Data Acquisition' checkbox is checked and highlighted with a red box and a '3' next to it. The 'On stop' and 'Moving' sections have the following settings:

	Home	Roaming	Unknown
Min Period	4 10	3600	3600
Min Saved Records	1	1	1
Send Period	5 10	61	61

	Home	Roaming	Unknown
Min Period	6 10	3600	3600
Min Distance	100	100	100
Min Angle	20	20	20
Min Speed Delta	0	0	0
Min Saved Records	1	1	1
Send Period	7 10	61	61

The screenshot shows the Teltonika configuration tool interface. On the left is a navigation menu with 'System' highlighted. The main area is divided into several sections:

- System Settings:** Includes Movement Source (Ignition, Accelerometer), GNSS (GNSS, CAN Speed), Speed source (GNSS, OBD / CAN), Records Saving/Sending Without TS (After Position Fix: Always, After Time Sync: 9), LED Indication (Disable, Enable), GNSS Source (BeiDou, GLONASS, Galileo, GPS), Battery Charge Mode (On Need, After Ignition ON, Always), Analog Input Value Range (Range 10V, Range 30V), Data Protocol (Codec 8, Codec 8 Extended), and Static Navigation Settings (Static Navigation: Disable, Enable; Static Navigation Source).
- Ignition Source:** Includes Ignition Settings (DIN 1, Accelerometer, Power Voltage, Engine RPM) and High/Low Voltage (mV) settings (30000, 13200).
- Accelerometer Delay Settings:** Movement Start Delay (s) and Movement Stop Delay (s).
- Time Synchronization:** Synchronization Settings (Disable (GNSS only), NITZ+NTP, NTP, NITZ), NTP Resync (h), NTP Server 1 (avl1.teltonika.lt), and NTP Server 2 (pool.ntp.org).
- Accelerometer Auto Calibration:** Accelerometer Auto Calibration (Disabled, Once, Continuous) and Gravity Filter (Disable, Enable).

4. Open Teltonika listener tool or other tool which allows to receive AVL data from FMB device. Records will be received over codec 8 extended data sending protocol
5. After record received copy it into Teltonika Data parser. Beacon ID: 385

The screenshot shows the Teltonika Data Parser interface. At the top, it says 'Enter data:' followed by a long hexadecimal string. Below this is a table showing the decoded data packet structure:

Name	Size	Value	Hex Value
Data Packet	var		
le	4	0	00-00-00-00
ta Length	4	90	00-00-00-5A
ec ID	1	142	8E
Data Count	1	1	01
Data	var		
Timestamp	8	2019.06.18 08:25:22	00-00-01-6B-69-B0-C9-51
Priority	1	0	00
GPS Element	15		
Longitude	4	0	00-00-00-00
Latitude	4	0	00-00-00-00
Altitude	2	0	00-00
Angle	2	0	00-00
Satellites	1	0	00
Speed	2	0	00-00
I/O Element	var		
Event ID	2	385	01-81
Element count	2	1	00-01
1b Element count	2	0	00-00
2b Element count	2	0	00-00
4b Element count	2	0	00-00
8b Element count	2	0	00-00
Xb Element count	2	1	00-01
ID	2	385	01-81
Value	45		11-21-6B-81-7F-8A-27-4D-4F-BD-B6-2D-33-E1-84-2F-8D-F8-01-4D-02-2B-BF-21-A5-79-72-36-75-06-4D-C3-91
Data Count	1	1	01
	4	15965	00-00-3E-5D

## Record parsing

Beacon records are sent as separate Records with Event I/O ID 385 and also include I/O element 385 (Codec8 Extended has to be used, because the I/O element 385 uses Variable size IO element).

Example Beacon Record Packet:

```
000000000000005A8E010000016B69B0C951000000000000000000000000000000000000000000000000010001
0000000000000000000000010181002D11216B817F8A274D4FBDB62D33E1842F8DF8014D022BBF21A579
723675064DC396A7C3520129F61900000000BF0100003E5D
```

**8E** – Codec8 Extended Protocol

**0000016B69B0C95100** – GPS data

**0181** – Event I/O ID : 385 (Generated by Beacon Functionality)

**0181** – Element I/O ID : 385 (Beacon Information stored here)

**002D** – Length of Beacon I/O element that was sent: 45 bytes

**11216B817F8A274D4FBDB62D33E1842F8DF8014D022BBF21A579723675064DC396A7C3520129F6190000000BF** – Beacon Information that has been sent by device.

11 – Data part, 1 Record out of 1 Beacon Record. First half byte specifies, current record, second half specifies how many there are in total. For example: 25 – Second record out of 5 records (that means server can expect 3 more records with Event I/O ID 385).

### 1st Beacon data:

21 – BLE Beacon Flags, this value has to be converted to Binary. 00100001 – Bit 0 is 1, which means Signal Strength is available. Bit 5 is 1, which means iBeacon data has been sent. (0 – Eddystone; 1 - iBeacon).

6B817F8A274D4FBDB62D33E1842F8DF8014D022BBF – iBeacon Data:

17F8A274D4FBDB62D33E1842F8DF8 – UUID

014D – Major

022B – Minor

BF – Signal Strength :Signed 2's complement: -65 RSSI

## 2nd Beacon Data:

21 – BLE Beacon Flags, this value has to be converted to Binary. 00100001 – Bit 0 is 1, which means Signal Strength is available. Bit 5 is 1, which means iBeacon data has been sent. (0 – Eddystone; 1 - iBeacon).

A579723675064DC396A7C3520129F619 – UUID

0000 – Major

0000 – Minor

BF – Signal Strength :Signed 2's complement: -65 RSSI

By parsing the BLE Beacon Flag (It is known how many Bytes of data will follow After, For example if it is iBeacon (21), 21 Bytes of information will follow (UUID:16 Bytes+Major:2 Bytes+Minor:2 Bytes+RSSI:1 Bytes) if it is Eddystone(01), 17 Bytes of information will follow (Namespace: 10 Bytes+Instance: 6 Bytes+RSSI: 1 Bytes))

6. Information for data parsing:

**Table 1. Beacon configuration parameters**

Name	ID	Min	Max	Default	Description
Beacon Detection	134	0	2	0	0 (Disabled) – Functionality disabled. 1 (All) – All beacons are sent to server. 2 (Configured) – Only beacons from the list are sent to server
Beacon Record	136	0	1	0	0 (Eventual) – Disabled. 1 (Periodic) – All beacons are sent to server.
Record Period	137	30	360 0	60	Beacon record saving period
Beacon List	1600 – 1624	Empty	42 B	Empty	List of authorized beacons. Eddystone and iBeacon protocols are supported*.

\*Beacon Ids can be entered the following way: Namespace:CompanyID, UUID:Major:Minor. Any of the ID parts can be omitted ex., Namespace:. UUID::, UUID::Minor, etc.

- Authorized Driving parameter (ID 11703) was extended to include new possible values: 2 – Beacon, 3 – Both (OneWire and Beacon).
- When Authorized Driving is configured to work with beacons, bluetooth scan procedure will be started for 5 minutes once ignition is detected.

Beacon are sent to server in a new variable length Beacon (AVL ID 385). Beacon (AVL ID:385) has the following structure

**Table 2. Beacon Packet structure**

Data part	BLE beacon flags #1	Beacon ID #1	Signal Strength #1	Battery Voltage #1	Temperature #1	Beacon data #2
1 Byte	1 Byte	20B/16B	1 Byte	2 bytes	2 bytes	...
First half byte – current data part Second half byte – total number of data parts	Bitwise parameter, specify BLE beacon type and parameters (signifies if specific data is present for certain beacon) Bit 0 – signal strength 0/1 Bit 1 – Battery Voltage 0/1 Bit 2 – Temperature 0/1 Bit 3 – Reserved Bit 4 – Reserved Bit 5 – BLE beacon type 0/1 Bit 6 – Reserved Bit 7 – Reserved (eddystone/ibeacon)	iBeacon – <b>20B</b> (UUID, major, minor) Eddystone – <b>16B</b> (Namespace, Instance ID)				...

**Important: Battery Voltage and Temperature are not supported in 03.25.09.Rev.51.**

## BLE sensors testing instructions

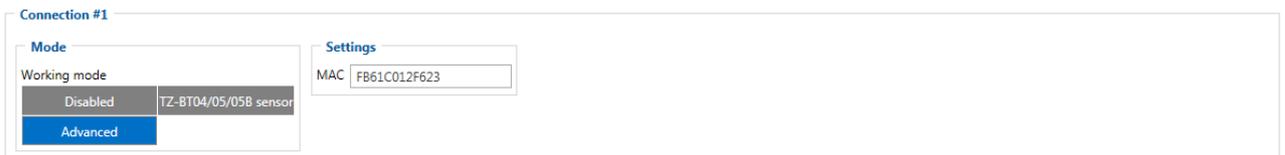
1. Enable BLE sensor via phone application.“Device Manager Mobile“. This application can be downloaded via this link:  
[https://play.google.com/store/apps/details?id=com.ela.mobileapp.elamobileapp&hl=en\\_US](https://play.google.com/store/apps/details?id=com.ela.mobileapp.elamobileapp&hl=en_US). Need to use Firmware 03.25.09.rev.51 (or later firmware version which supports beacon functionality).
2. In this application can be configured ELA sensors:
  - Change ELA sensor Name (Optional)
  - Enable ELA sensor.
  - Set ELA sensor signal power level (-40 is lowest signal power level, 4 is highest signal power level)
  - Choose the format of ELA sensor. For example if it's movement sensor it should be set as MOV
  - Advertising interval (s) with this parameter can be set interval for blue-tooth (optional)
  - Log interval (s) define the data logger period in seconds (optional)
  - Logger enable can enable for log mode.
  - Threshold acc is acceleration threshold for the tag which is working in MOV format only.



3. In configurator select BLE1 Sensor -> Advanced write usable sensor's MAC address. MAC ID can be found by using phone application nRF connect. Link to download:

<https://play.google.com/store/apps/details?id=no.nordicsemi.android.mcp&hl=en>

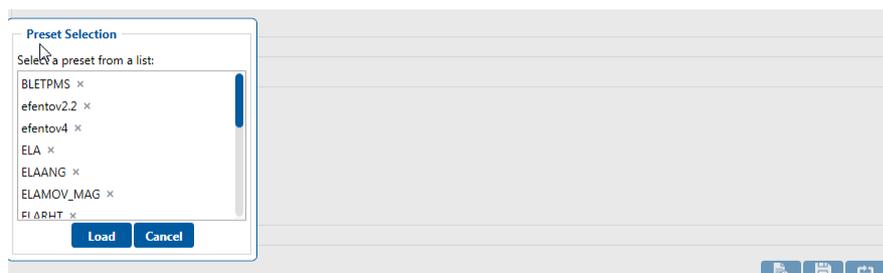
- BLE connectionless functionalities



1st Sensor

Type	Data Offset	Data Size	Action	IO	Match	Endianness	Multiplier	Offset
FE	5	2	Match	None	062A	Little Endian	1	0
FE	7	2	Save	Custom		Little Endian	1	0

4. Select Preset for the sensor which type it is. (If the sensor you are using does not have a made Preset, follow instructions „How to configure a unique sensor if there is no preset”)



5. In the above shown configuration. I/O is set to „Custom“, that means data will be sent under the I/O element BLE Custom 1. However, if the I/O is set to Temperature/Humidity/Fuel etc. – data will be saved and sent under the corresponding I/O elements. Temperature – BLE Temperature #X; Battery – BLE Battery #X; Humidity – BLE Humidity #X etc.

BLE Temperature #1	°C	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Temp #1
BLE Temperature #2	°C	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Temp #2
BLE Temperature #3	°C	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Temp #3
BLE Temperature #4	°C	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Temp #4
BLE Battery #1	%	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Battery #1
BLE Battery #2	%	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Battery #2
BLE Battery #3	%	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Battery #3
BLE Battery #4	%	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Battery #4
BLE Humidity #1	%RH	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Humidity #1
BLE Humidity #2	%RH	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Humidity #2
BLE Humidity #3	%RH	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Humidity #3
BLE Humidity #4	%RH	None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Humidity #4
BLE Custom 1		None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Custom sensor #1
BLE Custom 2		None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Custom sensor #2
BLE Custom 3		None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Custom sensor #3
BLE Custom 4		None	Low	High	Panic	0	0	Crash	Yes	No	Monitoring			BLE Custom sensor #4

### How to configure a unique sensor if there is no preset

This can be done by using the application nRF connect by finding a custom sensor and selecting RAW data. For example there is a screenshot with a RAW data from ELA movement sensor. As a type we are using type 16 to select all the needed data. Data offset is used to specify from which Byte data should be read or taken. We will use match on 062A (see screenshot of ELA movement packet structure). Data 062A is constant, that is why in the configuration we use „Match“ to confirm that this is the correct packet. To take data from the Packet, we enter type 16 and configure „Save“ and Offset is 2 bytes. We select I/O – Custom (Because Movement Sensor data is not measured in Kvants/°C/Voltage/%RH etc. )

Raw data:  
0x0201060516062AB6000709524A204D6F76

Details:

LEN	TYPE	VALUE
2	0x01	0x06
5	0x16	0x062AB600
7	0x09	0x524A204D6F76

LEN - length of 2m packet (Type + Data) in bytes.  
TYPE - the data type as in <https://www.bluetooth.org/en-us/specification/assigned-numbers/generic-access-profile>

Device type: LE only  
Advertising type: Legacy  
Flags: GeneralDiscoverable, BREdrNotSupported  
Service Data: UUID: 0x2A06 Data: 0xB600  
Complete Local Name: RJ Mov

<b>ELA MOV</b>
Length : 2
Type : 0x01
Data : 0x06
Length : 5
Type : 0x16
Alert Charac : 0x06
Alert Charac : 0x2A
MOV (cnt+state) data LSB
MOV (cnt+state) data MSB
Length : 16
Type : 0x09



## AVL ID DETAILS

Property ID in AVL packet	Property Name	Bytes	Type	Min	Max	Multiplier	Units	Description
20	BLE 2 Battery Voltage	1	Unsigned	0	100	-	%	Battery voltage of sensor #2
22	BLE 3 Battery Voltage	1	Unsigned	0	100	-	%	Battery voltage of sensor #3
23	BLE 4 Battery Voltage	1	Unsigned	0	100	-	%	Battery voltage of sensor #4
25	BLE 1 Temperature	2	Signed	-400	1250	0.1	°C	Degrees ( °C ), -40 - +125; Error codes: 4000 - abnormal sensor state 3000 - sensor not found 2000 - failed sensor data parsing
26	BLE 2 Temperature	2	Signed	-400	1250	0.1	°C	Degrees ( °C ), -40 - +125; Error codes: 4000 - abnormal sensor state 3000 - sensor not found 2000 - failed sensor data parsing
27	BLE 3 Temperature	2	Signed	-400	1250	0.1	°C	Degrees ( °C ), -40 - +125; Error codes: 4000 - abnormal sensor state 3000 - sensor not found 2000 - failed sensor data parsing
28	BLE 4 Temperature	2	Signed	-400	1250	0.1	°C	Degrees ( °C ), -40 - +125; Error codes: 4000 - abnormal sensor state 3000 - sensor not found 2000 - failed sensor data parsing
29	BLE 1 Battery Voltage	1	Unsigned	0	100	-	%	Battery voltage of sensor #1
86	BLE 1 Humidity	2	Unsigned	0	1000	0.1	%RH	
104	BLE 2 Humidity	2	Unsigned	0	1000	0.1	%RH	
106	BLE 3 Humidity	2	Unsigned	0	1000	0.1	%RH	
108	BLE 4 Humidity	2	Unsigned	0	1000	0.1	%RH	
270	Eskort LLS Fuel Level #1	2	Unsigned	0	65535		-	Fuel Level
273	Eskort LLS Fuel Level #2	2	Unsigned	0	65535		-	Fuel Level
276	Eskort LLS Fuel Level #3	2	Unsigned	0	65535		-	Fuel Level
279	Eskort LLS Fuel Level #4	2	Unsigned	0	65535		-	Fuel Level

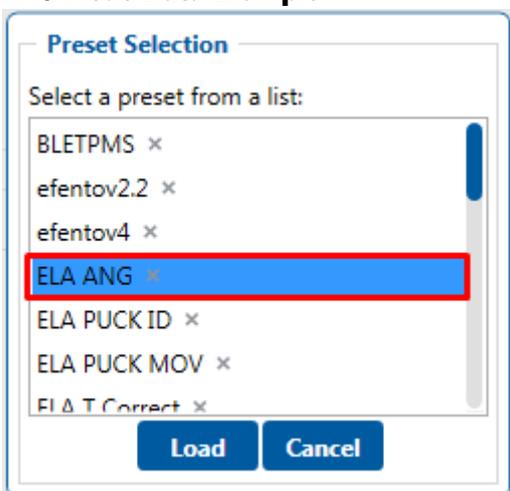
331	BLE Sensor 1	variable	HEX	0	256	-	-	Raw data of BLE sensor
332	BLE Sensor 2	variable	HEX	0	256	-	-	Raw data of BLE sensor
333	BLE Sensor 3	variable	HEX	0	256	-	-	Raw data of BLE sensor
334	BLE Sensor 4	variable	HEX	0	256	-	-	Raw data of BLE sensor
335	BLE Luminosity 1	2	Unsigned	0	0xffff	-	lx	Luminosity value of a BLE sensor
336	BLE Luminosity 2	2	Unsigned	0	0xffff	-	lx	Luminosity value of a BLE sensor
337	BLE Luminosity 3	2	Unsigned	0	0xffff	-	lx	Luminosity value of a BLE sensor
338	BLE Luminosity 4	2	Unsigned	0	0xffff	-	lx	Luminosity value of a BLE sensor

## ELA Sensor additional information

### ELA MOV

ELA MOV sensors can send data either in MOV mode or ANG mode. **ANG Mode:** acceleration (3 axis) on 16 bits / Axis(12 bits + a 16th sign bit); **MOV Mode:** 15-bit counter & instantaneous state\* on 1 bit (LSB)

### ANG Mode Data Example:



IO ID[331] Length[ 6]: 04 9C FF F8 00 03 BLE Custom 1

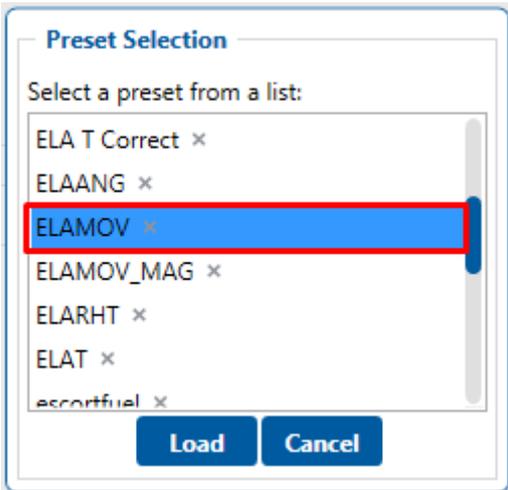
04 9C: 1180 – X axis

FF F8: -8 – Y axis

00 03: 3 – Z axis

Preset to use in the configurator: **ELA ANG**

### MOV Mode Data Example:



IO ID[331] Length[ 2]: 00 02

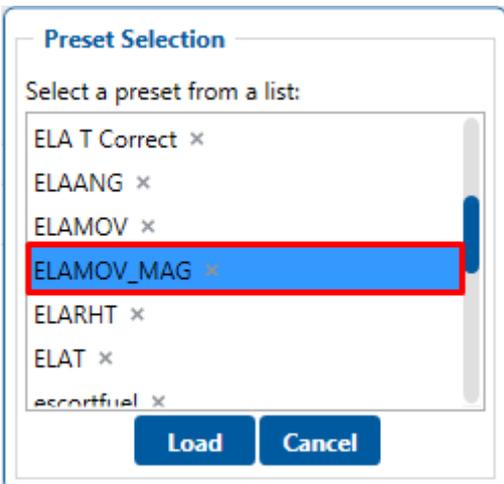
BLE Custom 1

00 02 - 0000 0000 0000 0010 (15 Bits Movement Counter: Movement Detected 1 time; 1 bit State of Sensor: Current State 0, not moving)

Preset to use in the configurator: **ELA MOV**

### ELA MAG

ELA MAG sends data 15-bit counter & instantaneous state\* on 1 bit (LSB).



### MAG Mode data Example:

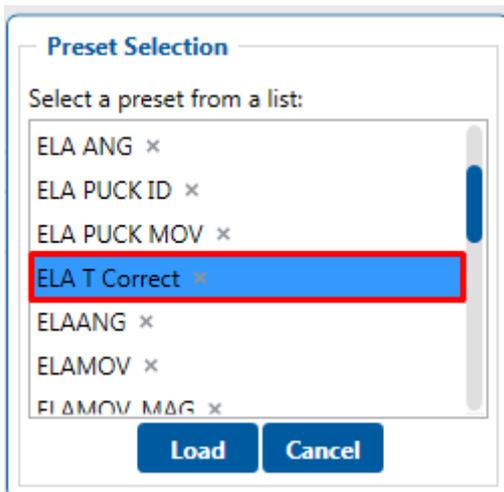
IO ID[331] Length[ 2]: 00 02

BLE Custom 1

00 02 - 0000 0000 0000 0010 (15 Bits Movement Counter: Attached/Detached Counter 1 time; 1 bit State of Sensor: Current State 0, Detached)

Preset to use in the configurator: **ELAMOV\_MAG**

## ELA T



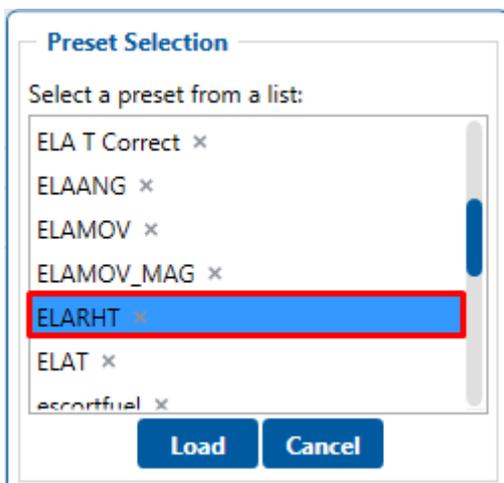
ELA T is a temperature sensor and it sends Temperature data.

### T Mode Data Example:

IO ID[ 26]: 2550 °C BLE Temperature #2

Preset to use in the configurator: **ELA T**

## ELA RHT



ELA RHT is a temperature and humidity sensor and it sends Temperature and Humidity data.

### RHT Mode Data Example:

IO ID[ 27]: 2550 °C BLE Temperature #3

IO ID[106]: 54% BLE Humidity #3

Preset to use in the configurator: **ELA RHT**

**NOTICE:**

**If there are troubles with configuring a specific sensor for usage. Please contact your sales manager and send the RAW data packet from **nRF Connect application** and the **protocol** of the sensors broadcasted frames.**