TEKTELIC communications

TEKTELIC Communications Inc.
7657 10th Street NE Calgary, Alberta
Canada, T2E 8X2

SPARROW | PELICAN | PELICAN EX

Enterprise Asset Tracker & BLE Beacon

Technical Reference Manual

Document Type:	Technical Reference Manual	
Document Number:	T0006940_SPARROW-PELICAN-PELICAN-EX_TRM	
Document Version:	2.9	
Document Status:	Release	
Product Name:	Enterprise Asset Tracker	
Product Variants & T-Codes (Module-Level):	SPARROW Enterprise Asset Trackers (Indoor)	T0007379 – Base T0007128 – Wall-Mount
	PELICAN Enterprise Asset Trackers (C-Cell)	T0007377 – Base T0006906 – Wall-Mount
	PELICAN EX Enterprise Asset Tracker (ATEX)	T0007367 – Wall-Mount
Release Date:	April 19, 2023	

Revision History

TRM Version	Date	Editor	HW Generation: FW Version	Comments
0.1	June 14, 2021	Carter Mudryk	Gen1: 1.0.15 Gen2: 1.0.15	• Initial release based on BLE Tracker TRM T0005946_TRM_v0.11. • Includes new averaging mode.
1.0	June 30, 2021	Carter Mudryk	Gen1: 1.0.15 Gen2: 1.0.15	• Added missing register description sections. • Updated description of filtering packet encoding and decoding for clarity. • Updated to more recently battery life estimates. • Updated List of Acronyms. • Added PCBA T-codes to product code table. • Added DL information stream on port 99 information and added section on how to put the device to sleep OTA. • Minor grammatical changes.
1.1	July 5, 2021	Carter Mudryk	Gen1: 1.0.16 Gen2: 1.0.16	• Added order codes/LoRaWAN regions table. • Removed magnetic pattern operational state description. • Minor error and formatting corrections.
2.0	November 17, 2021	Carter Mudryk	Gen2: 2.0.10	• Added information on new Beacon capabilities. • Added information on new coulomb counting function.
2.1	February 15, 2022	Carter Mudryk	Gen2: 2.0.12	• Corrected product T-Codes. • Changed BLE advertising frequency to 1 register. • Corrected maximum advertising interval to 40 000 ms. • Added list of all registers to Appendix. • Minor corrections based on feedback. • Changed default value of register
2.2	March 31, 2022	Carter Mudryk	Gen2: 2.0.13	Changed default value of register 0x 4A for Beacon mode.
2.3	April 12, 2022	Carter Mudryk	Gen2: 2.0.13	Corrected default values in Table 3-2.
2.4	April 21, 2022	Carter Mudryk	Gen2: 2.0.14	• Added “B” in document title to reflect Beacon mode functionality. • Added anti-brick functional description in Section 3.2.3.1.

TRM Version	Date	Editor	HW Generation: FW Version	Comments

2.5	April 22, 2022	Carter Mudryk	Gen2: 2.0.14
			Gen2.5: 3.0.0
				• Changed and clarified terminology:
				  ○ "Tracker" = specifically when referring to the device operating in Tracker mode.
				  ○ "Beacon" = specifically when referring to the device operating in Beacon mode.
				  ○ "Sensor" = when referring to the general functions not specific to either Tracker or Beacon mode.
				• Removed references to the sensor config web app.
				• Added description of periodic and event-based reporting in Section 2.
				• Minor formatting, grammatical, and clarifying changes throughout.

2.6	July 20, 2022	Carter Mudryk	Gen2: 2.0.16 Gen2.5: 3.0.5	• Minor grammatical corrections. • Moved FW versioning notes to Revision History table. • Updated product name terminology to "PELICAN" and "SPARROW." • Changed default value of register 0x 40 (in Tracker mode only) to 0x 87 to reflect that the accelerometer should be enabled by default. • Moved operational description of the accelerometer to Section 1.2 and added detail for clarity. • Added function for generating an event-based BLE report upon accelerometer clear.
2.7	August 3, 2022	Carter Mudryk	Gen2: 2.0.16 Gen2.5: 3.0.6	Changed default value of register 0x 11 to enable ADR by default.

TRM Version	Date	Editor	HW Generation: FW Version	Comments
2.8	December 12, 2022	Carter Mudryk	Gen2: 2.0.20 Gen2.5: 3.0.16	• Added information about differences between generations. • Changed document name to reflect marketing names. • Updated Gen2.5 T-codes. • Added description of what happens in the case of a delayed BLE scan report in Sections 1.1.1 and 2.2. • Corrected and clarified LED pattern descriptions. • Indicated deprecation of battery voltage reporting for Gen2.5 devices. • Updated descriptions for battery management and gauging. • Changed default value of register 0x 2A in Beacon mode to allow a battery report upon button press. • Changed default value of register 0x 4A to allow battery reporting format to be both remaining capacity [%] and remaining lifetime [days] by default for both Tracker and Beacon Mode. • Added register 0x 4B to reflect new remaining battery lifetime system design. • Changed limit on acceptable values for register 0x 59. • Corrected and updated options for Beacon Tx power in register 0x 5B for Gen2 and Gen2.5 devices. • Updated description for Eddystone TLM battery data to be clear on Gen2 vs Gen2.5 behaviour. • Minor grammatical fixes.

TRM Version	Date	Editor	HW Generation: FW Version	Comments
2.9	April 19, 2023	Carter Mudryk	Gen2: 2.0.22 Gen2.5: 3.2.4	• Added row to Table 1-3 to reflect that accelerometer events and clears also trigger event-based BLE scans by default. • Added register 0x 46 to Table 3-2 as one of the settings that changes depending on whether the Sensor is in Tracker or Beacon mode. • Changed default value of register 0x 12. • Removed MCU temperature threshold-based reporting support. • Minor grammatical and formatting fixes.

Table of Contents

List of Tables

List of Figures

List of Acronyms

ABP .... Activation By Personalization
ADR .... Adaptive Data Rate
AS .... Asia
ATEX ... . Atmosphere Explosible
AU .... Australia
B .... Byte
BD_ADDR .... Bluetooth Device Address
BLE .... Bluetooth Low Energy
CRC .... Cyclic Redundancy Check
DC .... Direct Current
DL .... DownLink
DR .... Data Rate
EAT .... Enterprise Asset Tracker
EIRP .... Equivalent Isotropically Radiated Power
EoS .... End of Service
EU .... European Union
FW .... FirmWare
g .... gravity (unit of acceleration)
HW .... HardWare
ID .... Identity/Identifier
IN .... India
IoT .... Internet of Things
IP .... Ingress Protection
JSON ... . JavaScript Object Notation
KR .... KoRea
LAP .... Lower Address Part
LED .... Light-Emitting Diode
LoRa .... Long Range
LoRAMAC .... LoRaWAN MAC
LoRaWAN .... LoRa Wide Area Network
LoS .... Line-of-Sight
LSb .... Least Significant bit
LSB .... Least Significant Byte
LTC .... Lithium Thionyl Chloride
MAC .... Medium Access Control
MCU .... MicroController Unit
MSb .... Most Significant Bit
MSB .... Most Significant Byte
NA .... North America
NS .... Network Server
OTA .... Over-The-Air
OTAA .. .. OTA Activation
PCBA ... . Printed Circuit Board Assembly
OUI .... Organizationally-Unique Identifier
POST ... . Power-On Self-Test
RF .... Radio Frequency
RO .... Read-Only
RTLS .... Real-Time Location System
R/W .... Read/Write
RSSI .... Received Signal Strength Indicator
Rx .... Receiver / Receive
sec .... second(s)
SW .... SoftWare
TLM .... TeLeMetry
TRM .... Technical Reference Manual
Tx .... Transmitter / Transmit
UL .... UpLink
v. .... version
ver. .... Version

1 Overview

IMPORTANT: Not all features described in this manual may be applicable to devices programmed with older FW versions. Refer to the Revision History table to verify which FW versions included the addition of new features. To check which version of FW your device has, send a command to query your device as described in Section 3.2.9.2.

This document contains the technical information about the supported functionality of the TEKTELIC *Enterprise Asset Tracker* device variants, which include the SPARROW, PELICAN, and PELICAN EX. In particular, the LoRa IoT uplink and downlink payload structures and user-accessible configuration settings are described in detail. This document assumes an understanding of the NS and its command interfaces.

The Enterprise Asset Tracker is a BLE-capable LoRaWAN IoT sensor run on a single LTC battery and packed into a compact polycarbonate casing. The device's primary purpose is to scan for and report nearby BLE peripherals (devices) up to the LoS range of 70 m. Some common applications include, but are not limited to:

- Asset tracking within a network of BLE Beacon devices,
- Proximity detection for equipment in places of interest, and
- Asset movement detection and security.

The device is also equipped with an accelerometer, which, as a motion detector, can be used to detect and report acceleration events, or trigger a BLE scan upon detecting motion. The accelerometer output vector can also be reported periodically if knowledge of the orientation is of interest.

The device can also be configured to act as a BLE Beacon. When configured this way, the device no longer scans but is discoverable by other nearby Trackers or BLE-capable devices.

Although Beacon mode is just a function of the device known as the Enterprise Asset *Tracker*, the terminology throughout this document is chosen to be clear on whether the operation being described relates to Tracker mode, Beacon mode, or both. When referring to parameters specifically pertaining to Tracker mode operation only, the device will be referred to as a ***Tracker***. Similarly, for referring to Beacon mode-specific parameters and operation, the device will be referred to as a ***Beacon.*** When referring to parameters and operations that are available in both modes, the blanket term ***Sensor*** is used.

By default, the Sensor operates in Tracker mode, therefore all following references to default configuration without specifically referring to Beacon mode should be assumed to be for Tracker mode. That is, "default settings" should be taken to be equivalent to both "default Sensor settings" and "default Tracker settings." "Default Beacon settings" is not equivalent to "default Sensor settings."

Additional sensing functions on the Sensor include on-board temperature and the battery voltage. The battery lifetime of the Sensor has been estimated to be up to:

- 5.5 years for AA-cell variants operating with default Tracker mode settings[1].
- 15+ years for C-cell variants operating with default Tracker mode settings[1].
- 5 Years for PELICAN EX operating with default Tracker mode settings[1].
- 16 months for AA-cell variants operating with default Beacon mode settings[2].
- 4.5 years for C-cell variants operating with default Beacon mode settings[2].
- 1.5 years for PELICAN EX operating with default Beacon mode settings[2].

The AA-cell variants have an externally-accessible push-button that can be configured to have various functions. The C-cell variants have a magnetic switch which is used to wake the device from the DEEP SLEEP state (used for shipping) and to force ULs when the device is active.

There have been 3 *generations* of the BLE Sensors. Gen2 and Gen2.5 have all the same functions as each other but have different HW. The main differences between Gen2 and Gen2.5 are summarized in Table 1-1. Note that Gen1 device functional description is outside the scope of this TRM version; for Gen1 devices, please refer to an older document version in accordance with your device as described in the Revision History Table.

Table 1-1: Generational Differences

Generation	MCU	FW Versions	Supported Models
Gen2	Silicon Labs EFR32 Blue Gecko	• 1.x.x (does not support Beacon mode) • 2.x.x (supports Beacon mode)	• SPARROW • PELICAN • PELICAN EX
Gen2.5	STM 32	3.x.x	• SPARROW • PELICAN

Table 1-2 presents the currently available Sensor HW variants. The information streams supported by the SW have been shown in Table 1-3, and the default configuration for reporting data has been shown in Table 1-4.

[1] Default settings plus 24 event-based BLE scans and reports per day, operating at DR2. Applicable to NA region only.

[2] Default settings with no event-based reports, operating at DR2. Applicable to NA region only.

Table 1-2: Enterprise Asset Tracker HW Models

Product Code, Module-Level T-Code	Product Code, PCBA-Level T-Code	Model Name	Description	Battery/Enclosure Size
T0007379	T0006983 (Gen2) T0008265 (Gen2.5)	SPARROW	Enterprise Asset Tracker (Indoor), Base	AA-cell
T0007128			Enterprise Asset Tracker (Indoor), Wall-Mount	AA-cell
T0007377	T0006984 (Gen2) T0008176 (Gen2.5)	PELICAN	Enterprise Asset Tracker (Outdoor), Base	C-cell
T0006906			Enterprise Asset Tracker (Outdoor), Wall-Mount	C-cell
T0007367	T0007465 (Gen2)	PELICAN EX	Enterprise Asset Tracker (ATEX), Wall-Mount	C-cell

Table 1-3: Enterprise Asset Tracker Information Streams

Stream Direction	Data Type	Sent on LoRaWAN Port [decimal]
UL (Tracker to NS)	Sensor data from the MCU, battery gauge, and accelerometer	**10**
	Report of discovered BLE devices	**25**
	Responses to Configuration and Control Commands	**100**
DL (NS to Tracker)	Putting Tracker into DEEP SLEEP	**99**
	Configuration and Control Commands	**100**

Table 1-4: Enterprise Asset Tracker Default Reporting Behavior

Report	Report Type	Default Periodicity
Battery status	Periodic	24 hours
	Event-based	Each time the magnetic switch is triggered (C-cell variants only)
Discovered BLE devices (up to 8) with averaged RSSIs after scanning for 3 sec	Periodic	1 hour

Report	Report Type	Default Periodicity
Discovered BLE devices (up to 8) with averaged RSSIs after scanning for 1 sec	Event-based	Each time the function button is pressed (AA-cell variants only)
		Each time above-threshold motion is detected and each time motion stops
Acceleration vector	Periodic	Disabled
Accelerometer motion alarm	Event-based	Each time above-threshold motion is detected and each time motion stops
MCU temperature	Periodic	Disabled

In Sections 2 and 3 of this document, the UL and DL payload formats are explained, respectively.

The following subsections provide a more detailed description of the functionality of each of the subsystems and user interfaces available on the Sensor.

1.1 BLE Operation

1.1.1 Tracker Mode (BLE Rx)

The BLE function of the Sensor when configured in Tracker (default) mode is Rx only; the Tracker only scans and does not advertise, which means it is not discoverable by other BLE-capable devices. During each scan, other advertising BLE devices can be discovered. Each discovered device has its data (MAC address and the RSSI of the advertisement packet) saved by the Tracker to then be reported in a LoRa UL. This UL is normally reported immediately after the scan concludes, but may be delayed due to duty cycle limitations (LoRa Alliance, Feb 2017). If a new BLE scan occurs before the results of the previous scan have been sent, the old scan results will be discarded.

The BLE scan can be disabled entirely or enabled at any time in Tracker mode. Figure 1-1 shows the BLE scan scheme in the Tracker when the BLE scan is enabled. As shown in the figure, BLE scans are performed periodically with a configurable *scan period*, which is 1 hour by default. Each scan lasts for a configurable *scan duration* and is divided into *scan intervals*. The active BLE scan is performed only in the *scan window* portion of the scan interval.

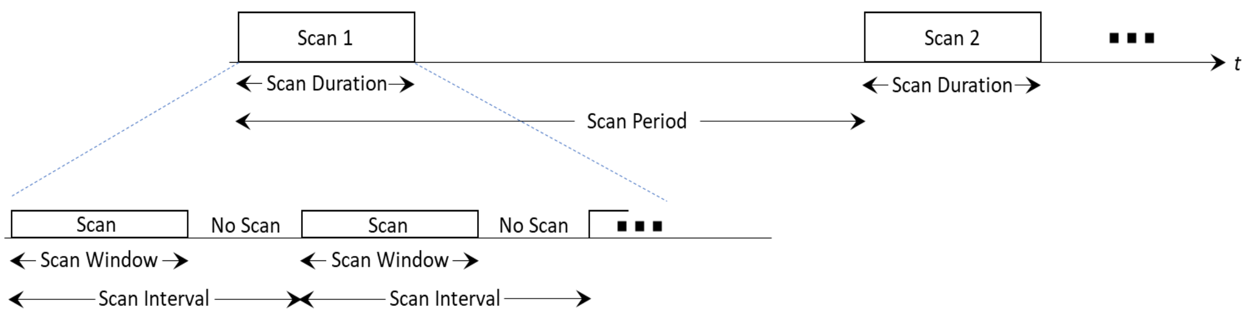


Figure 1-1: The BLE Scan Scheme

The ratio of the scan window to the scan interval is the *scan duty cycle*. A scan window equal to the scan interval represents a scan duty cycle of 100% (a continuous scan) over each scan duration. This is the default behavior as it maximizes the chance of “discovering” nearby BLE advertisement packets for a given scan duration. Reducing the duty cycle below 100% has the advantage of decreasing power consumption at the expense of possibly missing some beacon signals.

The scan period, duration, interval, and window are all configurable (see Sections 3.2.3 and 3.2.7).

Following each scan, there are two different modes for reporting the discovered BLE device data:

A. **Averaging Mode (Default)**: It is often the case that more than one distinct packet from the same BLE device is received during a single scan. When averaging mode is enabled, the RSSIs for each device are averaged over the packets received, then the devices are sorted and uplinked in order from strongest average RSSI to weakest, up to *N* distinct devices (*N* is a user-configurable value; see Section 3.2.7).

B. **Raw Mode:** The packets received during each scan are neither averaged nor sorted before uplinking; the *N* last discovered devices are sent. This means it is possible to have repeated devices in the same UL.

Furthermore, there is the option to send device data with *basic* reporting or *filtered* reporting. In the basic (default) reporting, at the end of each scan duration, up to a maximum *N* discovered BLE devices are reported over LoRaWAN. In filtered reporting, up to maximum *N* discovered BLE from a user-specified range of favorable MAC addresses are reported. This allows for keeping out undesired devices from the OTA report, and thus reducing the OTA time and saving battery life. An example application of this is for an indoor Beacon network; only the Beacon signals are of Interest and not other devices like smartphones. In either reporting type, if no devices are found, an empty list is uplinked.

Averaging/raw modes and basic/filtered reporting are mutually compatible; that is, any combination of the four options is acceptable.

The Tracker supports BLE of Bluetooth 5.0. The BLE scan is performed in the passive mode only; i.e., the Tracker listens to surrounding beacons, but does not transmit to them to request additional information.

Switching from Tracker mode to Beacon mode or vice versa can be done by sending a simple DL. See Section 3.2.1 for how this is done.

NOTE: The BLE scan and LoRa radio activity are mutually exclusive; i.e., they do not overlap. If any LoRa reporting becomes due at the same time of a BLE scan, the reporting will be done after the BLE scan is complete.

1.1.2 Beacon Mode (BLE Tx)

The Sensor can be configured to act as a Beacon, which is equivalent to saying that it operates in BLE Tx mode. When in Beacon mode, the Sensor sends out periodic BLE *advertisements* which are small packets of data. These packets are discoverable by other devices operating in Tracker mode as well as any other BLE-capable device.

When in Beacon mode, the Sensor is still LoRaWAN-backhauled; that is, it can still send data such as battery life in a LoRaWAN UL and can still be reconfigured through a LoRaWAN DL. Furthermore, all other transducer functions such as the accelerometer are still accessible in either Beacon or Tracker mode. For this reason, Beacon mode should be considered as one of many functions of the Enterprise Asset Tracker device.

The advertisements are sent periodically as defined by the *advertising interval*, as shown in Figure 1-2. The advertising interval is user-configurable.

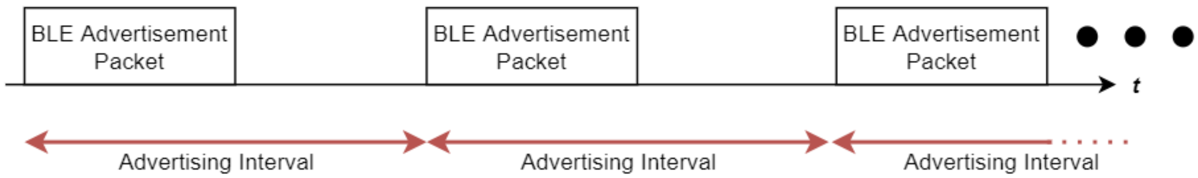


Figure 1-2: The BLE Advertisement Scheme

The advertising packet format supports up to 3 major BLE standards: iBeacon, Eddystone UID, and Eddystone TLM. By default, iBeacon is the only format enabled. Any combination of the 3 formats can be sent. If more than one format is configured, the different formats will alternate in consecutive advertisements. See Section 3.2.8 for more details regarding the different packet formats and configurable settings.

An example use case of the Beacon function is a Real-Time Location System (RTLS), in which a network of Beacons is required to be distributed over an area in which assets need to be tracked. Trackers move within this Beacon network and conduct periodic scans for the Beacon advertisements. The signal strength of these advertisements is then used by an application to resolve the location of the Tracker within the network. For large tracking areas, there could potentially be hundreds or thousands of Beacons required. Having LoRaWAN-managed BLE Beacons allows for the monitoring and management of large quantities of devices, which is in contrast to other 3rd-party Beacons which require individual management through smartphone applications.

Switching from Beacon mode to Tracker mode or vice versa can be done by sending a simple DL. See Section 3.2.1 for how this is done.

NOTE: The BLE advertisement and LoRa radio transmission are mutually exclusive; i.e., they do not overlap. If any reporting becomes due, the BLE advertisements are paused while the LoRa activity is occurring.

1.2 Accelerometer Operation

The accelerometer in the Sensor can be disabled or enabled and supports both periodic-based and event-based reporting. The accelerometer is enabled by default.

In the case of the periodic-based reporting, only the acceleration vector (X-axis, Y-axis, Z-axis) is reported.

In the case of event-based reporting, the accelerometer is used to generate a report when:

- Motion is detected that matches some configurable criteria. This is called an *acceleration event*.
- The motion has ended. This is called an *acceleration clear*.

The criteria that determine whether an acceleration event is registered are whenever the absolute value of any axis exceeds a certain threshold, for a certain number of times, within a certain period. These criteria are all configurable through the registers as described in Section 3.2.5. For example, with default configuration settings, as soon as the acceleration magnitude on any axis is measured to be greater than 2 *g* (register 0x 44, *Acceleration Event Threshold*) one time (register 0x 42, *Acceleration Event Threshold Count*) in less than 10 seconds (register 0x 43, *Acceleration Event Threshold Period*), an acceleration event is registered.

An acceleration clear is when the previous acceleration event is considered "cleared." This occurs as soon as no further above-threshold acceleration is registered for at least a configurable *grace period* (register 0x 45). No additional acceleration-event-based UL reports are sent before an acceleration clear is registered. For example, after an acceleration event has been registered, the Sensor must not sustain any above-threshold movement for the full 5-minute (default) grace period before an acceleration clear is registered. Until that time, no additional acceleration alarm ULs are sent. Every time an above-threshold acceleration is measured after a registered accelerometer event but before that event has been cleared, the grace period timer resets to 0.

Whenever an acceleration event is registered, it is configurable what type of data is reported. As an option, an acceleration alarm UL can be reported. In this case, an acceleration alarm clear UL will be reported as soon as an acceleration clear is registered, as explained in the previous paragraph. No additional acceleration alarms are reported before the first acceleration alarm clear is registered. An example sequence of detected motion and generated acceleration ULs is shown in Figure 1-3.

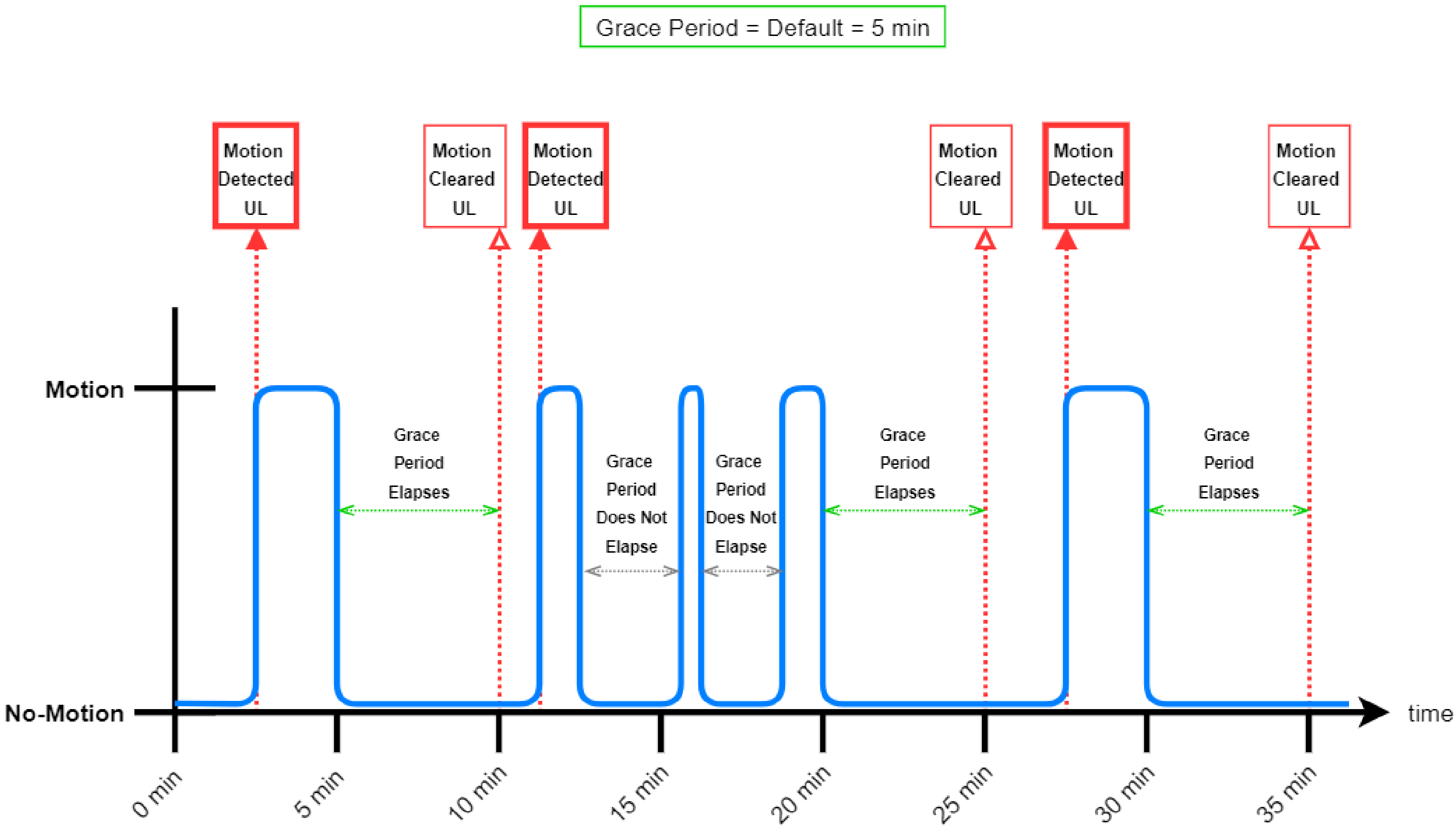


Figure 1-3: Example Sequence of Accelerometer Events and UL Reports

The other option of what to report after an acceleration event is registered is a BLE scan report. In this case, when an acceleration alarm is registered, a BLE scan is performed and the result is reported in the same format as a regular BLE report. When the accelerometer clear is registered, another additional BLE scan is conducted and the results are reported in an UL. If BLE reporting is enabled for acceleration events and clears, then the BLE report ULs will be sent following each motion detected/cleared UL as shown in Figure 1-3 above. This function is particularly useful for use-cases in which the location (as determined by the BLE data) of the Tracker is needed when tracked equipment moves from one place to another; the location would be updated when the device begins to move from location A and then again when it is left motionless to stay in position B.

Either alarm reporting or BLE reporting or both can be enabled.

Event-based reporting in the case of acceleration events can be disabled, in which case neither the acceleration alarm nor BLE scan results will be reported (see register 0x46).

1.3 Function Button Operation

Only the AA-cell variants are equipped with a function button.

The function button is externally-accessible and located on the bottom of the enclosure, actuated by pressing. The purpose of the function button is to elicit an uplink from the Sensor. This also allows the Sensor, as a LoRaWAN Class-A end-device, to open receive windows to receive DL commands.

It is user-configurable how a function button event is registered (e.g., by pushing and releasing the button at least 3 times within 3 sec, or by holding the button pressed for at least 3 sec). Whenever the event is registered, the Sensor sends an uplink. The uplink can be empty, or it can contain the battery voltage, acceleration vector, temperature, BLE scan report, or any combination of these. Section 3.2.4 describes the configuration for the function button.

1.4 Magnetic Switch Operation

Only the C-cell variants are equipped with a magnetic switch.

The magnetic switch can be operated by the provided magnet, and is used for the following purposes:

A. **Device Reset:**

This is mainly used to wake up the Sensor from DEEP SLEEP and have it begin trying to join the network. When the Sensor exits the factory, it is in the low-power DEEP SLEEP mode and can be activated using the specified magnetic pattern. The same magnetic pattern can just be used to reset the Sensor during normal operation, causing it to try to rejoin the network.

The magnetic pattern in this application is illustrated in Figure 1-4. A “magnet presence” is achieved by placing the magnet against the enclosure by the magnet symbol. A “magnet absence” is achieved by taking the magnet away from the enclosure. Figure 1-4 shows that the pattern involves sustaining a “magnet presence” continuously for at least 3 sec but less than 10 sec.

When the specified magnetic pattern is applied to the Sensor it displays an LED indication, described in Section 1.5.3, to show that it has accepted the magnetic pattern. In all cases, the magnet pattern causes the Sensor to reset. If the Sensor was in DEEP SLEEP when the pattern was applied, after resetting it will wake up and begin trying to join the network. If the Sensor was in normal operation when the magnet pattern was applied, after resetting it will try to rejoin the network.

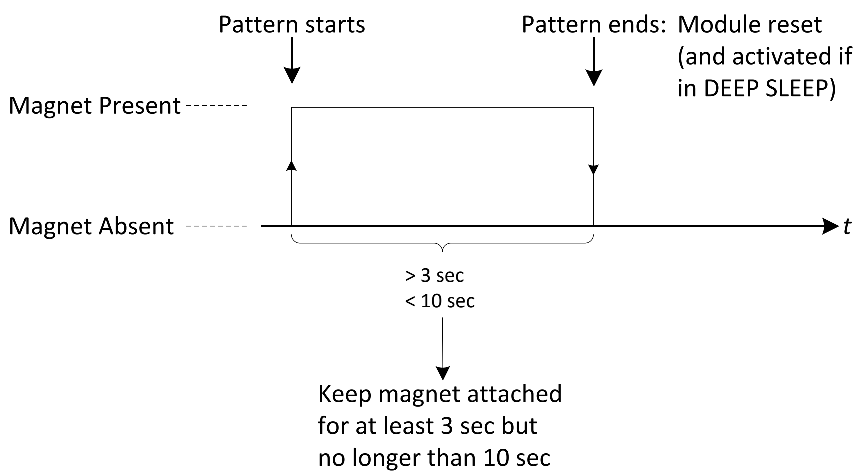


Figure 1-4: The Magnetic Reset/Wake-Up Pattern.

The same magnetic pattern can just be used to put the Sensor back into DEEP SLEEP. If the magnetic pattern is applied while the Sensor is trying to join the network, but before it receives the JOIN ACCEPT DL, the Sensor will go back into DEEP SLEEP mode. Applying the magnetic pattern again will cause it to wake up.

B. **Forced UL:**

This is used to get the LoRaWAN Class-A Sensor to open a receive window so it can receive DL commands from the NS, or simply to trigger the Sensor to uplink the battery voltage data prior to its next scheduled periodic report.

The magnetic pattern in this involves placing and taking away the magnet to and from the magnet sign at the bottom of the enclosure briefly for less than 2 seconds, as shown in Figure 1-5. *It is important to note here that mistakenly holding the magnet attached to the module for more than 3 sec may trigger a module reset, as explained in item A above.*

The magnetic-forced UL contains a regular battery voltage payload on ***LoRaWAN port 10***.

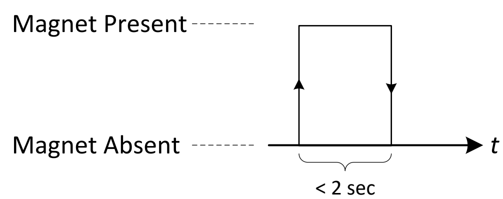


Figure 1-5: The Magnetic UL-Triggering Pattern

1.5 LED Behaviour

The Sensor is equipped with two on-board LEDs: **GREEN** and **RED**. Their behaviour patterns reflect the internal device state and are described in the following subsections. The LED behaviour is not user-configurable.

1.5.1 Power-On and Network Join Operation

When the Sensor is activated or reset:

1. Both **GREEN** and **RED** are turned off when the Sensor is powered on (including after a soft reset).
2. Both **GREEN** and **RED** are turned on when the POST begins.
3. When the POST ends, depending on the POST result:
    a. If the POST passes, **GREEN** is toggled ON and OFF every 50 ms for 0.5 sec, as shown in Figure 1-6. In this case, the LED pattern proceeds to step 4.
    b. If the POST fails, **RED** is toggled ON and OFF every 50 ms for 0.5 sec, as shown in Figure 1-6. In this case, the device restarts and the LED pattern begins again at step 1 after approximately 4 s.

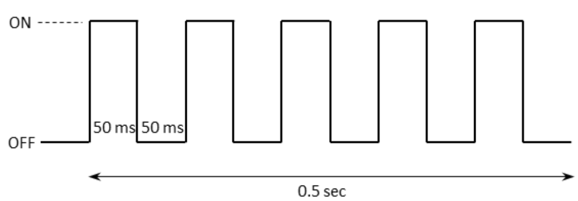


Figure 1-6: The GREEN or RED LED Pattern After the POST

4. Both **GREEN** and **RED** are turned off when the POST and the subsequent LED flashing specified in item 3 end.
5. While the Sensor is attempting to join:
    a. **GREEN** is toggled ON and OFF every 50 ms for the first hour. After that, it only flashes twice (ON time: 50 ms, OFF time: 50 ms) every 5 sec. This scheme has been shown in Figure 1-7.
    b. **RED** flashes just once:
        i. with a pulse duration of 25 ms right after transmitting a JOIN REQUEST. This occurs at approximately 10 s intervals at the beginning of the join process, but at decreasing regularity the longer the join process continues due to battery

saving measures and possible duty-cycle limitations in certain regions (LoRa Alliance, Feb 2017).

ii. with a pulse duration of 100 ms right after receiving a JOIN ACCEPT. This will occur once, after which, the device will have joined the network and normal operation begins.

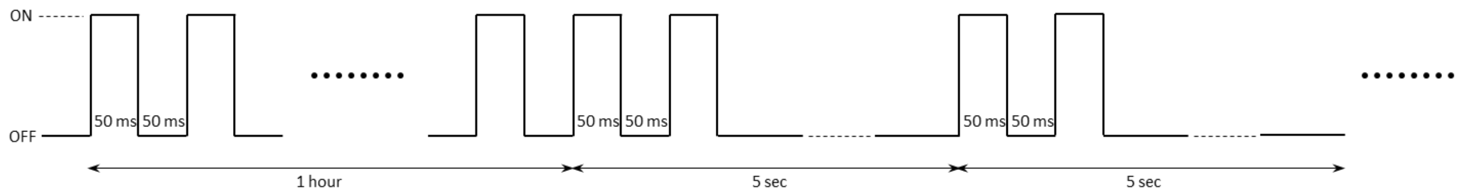


Figure 1-7: The GREEN LED Pattern During Join

1.5.2 Normal Operation

After the Sensor has joined the network:

a. **RED** flashes just once with a pulse duration of 25 ms right after transmitting an uplink.

b. **GREEN** flashes just once with a pulse duration of 25 ms right after receiving a downlink.

1.5.3 DEEP SLEEP

The Sensor displays an LED indication when it is brought out of DEEP SLEEP or reset by applying the magnetic pattern. The following LED pattern is displayed about 3 sec after the pattern is applied:

1. **GREEN** is toggled ON and OFF every 0.5 sec for 3 sec as shown in Figure 1-8.
2. After a 5 s pause, the normal Power-On and Network Join LED patterns described in Section 1.5.1 above occur after the device resets.

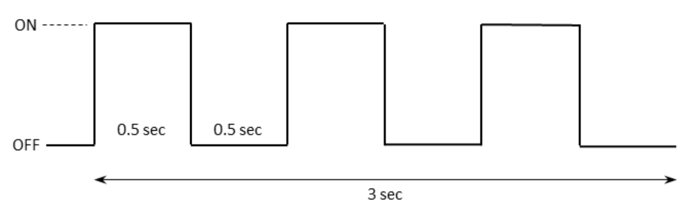


Figure 1-8: The GREEN LED Pattern After the Magnetic Wake-Up/Reset Pattern is Observed

There is another similar LED pattern for when the device is put back into DEEP SLEEP:

1. The POST LED pattern described in steps 1-4 in Section 1.5.1 above occurs after the device resets.
2. Immediately, **GREEN** is toggled ON and OFF every 0.1 sec for 0.6 sec as shown in Figure 1-9.

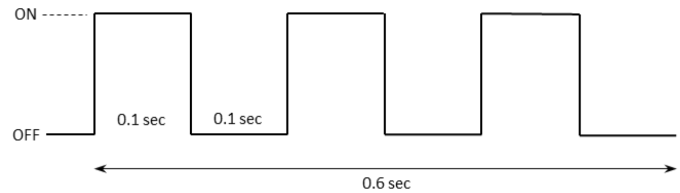


Figure 1-9: The GREEN LED Pattern Before Entering DEEP SLEEP

2 UL Payload Formats

The UL information streams (from the Sensor to the NS) supported by the SW are shown in Table 2-1, and are explained in Sections 2.1 through 2.3.

Table 2-1: UL Information Streams

Data Type	Sent on LoRaWAN Port [decimal]
Sensor data from the MCU, battery gauge, and accelerometer	**10**
Report discovered BLE devices	**25**
Response to Configuration and Control Commands	**100**

All data contained in Sensor telemetry ULs (not responses to configuration and control commands) falls into one of the following reporting categories:

- **Periodic Reporting**: Scheduled reporting of telemetry at regular, configurable intervals. The reporting intervals are configured through the use of the tick registers as described in Section 3.2.3.
- **Event-Based Reporting**: Various external events can trigger unscheduled telemetry ULs outside of the periodic reporting schedule. These external events include function button presses, magnetic switch actuation, motion of the device above the accelerometer threshold, and temperature measurements outside of the threshold window. Each event elicits a different type of response from the Sensor. Not all event-based reporting is enabled by default.

This section describes the *format* of the report payloads. For details on what causes event-based reporting and how to configure the Sensor's event-based behaviour, see the relevant subsections for the particular transducer of interest in Section 3.

2.1 Frame Payload to Report Non-BLE Sensing Data

Each data field from the Sensor is encoded in a frame format shown in Figure 2-1. A big-endian format (MSb/MSB first) is always followed.

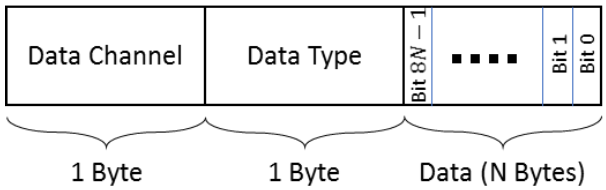


Figure 2-1: The Frame Format of the UL Payload

A Sensor message payload can include multiple data frames from different sensing entities in the Sensor. Frames can be arranged in any order. The Sensor frame payload values for present transducer data are shown in Table 2-2. In this table, the bit indexing scheme is as shown in Figure 2-1. Present sensor data in the UL are sent through ***LoRaWAN port 10***.

Table 2-2: UL Frame Payload Values for Present Sensor Data

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Battery Voltage[3]	0x 00	0x BA	1 B	Analog Voltage	• Bits 0-6: Battery voltage minus 2.5 V (10 mV/LSb, unsigned) • Bit 7: Not used	*battery_status {* *life: <value>,* *(unsigned/V)* }
Remaining Battery Capacity	0x 00	0x D3	1 B	Percentage	• 1% / LSB (unsigned)	*rem_batt_capacity: <value>* *(unsigned/%)*
Remaining Battery Lifetime	0x 00	0x BD	2 B	Days	• 1 day / LSB	*rem_batt_days: <value>* *(unsigned/days)*
Acceleration Alarm Status	0x 00	0x 00	1 B	Digital Input	• 0x 00 = Alarm inactive (motion no longer detected) • 0x FF = Alarm active (motion detected)	*acceleration_alarm* *(unsigned/no unit)*

[3] Battery voltage reporting is not supported on Gen2.5 devices.

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Acceleration Vector	0x 00	0x 71	6 B	Acceleration	• 1 milli-g/LSb (signed) • Bits 32-47: X-axis acceleration • Bits 16-31: Y-axis acceleration • Bits 0-15: Z-axis acceleration	*acceleration_vactor {* *xaxis: <value>, (signed/g)* *yaxis: <value>, (signed/g)* *zaxis: <value> (signed/g)* *}*
MCU Temperature	0x 00	0x 67	2 B	Temperature	• 0.1°C/LSb (signed)	*temperature (signed/°C)*

Examples:

- Uplink 0x 00 D3 5F is reported to indicate the remaining battery capacity is 95%.
- Uplink 0x 00 00 00 is reported to signal an acceleration alarm is cleared.

2.2 Reporting Discovered BLE Devices

For information about how BLE scans are conducted and how discovered device data is handled before reporting over LoRa, see Section 1.1.

When a Sensor is operating in Tracker mode, BLE scans are conducted to search for nearby advertising BLE peripheral devices. During each scan, if a BLE advertisement is received by the Tracker, the BLE device from which that advertisement originated is referred to as a *discovered device*. Discovered devices are reported on ***LoRaWAN port 25***. The payload format to report such devices has been shown in the diagrams in Figure 2-2.

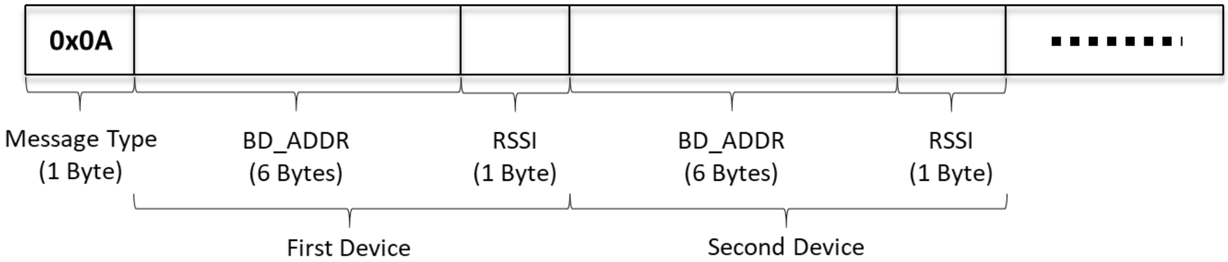
With *basic* reporting enabled (Figure 2-2-(a)), the message type header is 0x 0A. The BD_ADDR for each discovered device is a full 6-byte MAC address, and is followed by the device RSSI, which is a signed one-byte number.

With *filtered* reporting (Figure 2-2-(b)–I), up to 4 ranges of BD_ADDR can be defined for filtering discovered devices (see Section 3.2.7). The message type headers 0x B0, 0x B1, 0x B2, 0x B3 correspond to Ranges 0, 1, 2, and 3 respectively. A BD_ADDR consists of an *Organizationally-Unique Identifier* (OUI) comprising the 3 MSBs followed by a *Lower Address Part* (LAP) comprising the 3 LSBs. Each BD_ADDR range is a 9-byte OUI:LAP_{start}-LAP_{end} that determines the range of BD_ADDRs as OUI:LAP_{start} to OUI:LAP_{end}. Therefore, OUI is the same and known for all devices in each range. The message type header determines the range, and thus the OUI for all devices in the message, such that the devices in each message can be uniquely identified by their LAPs only.

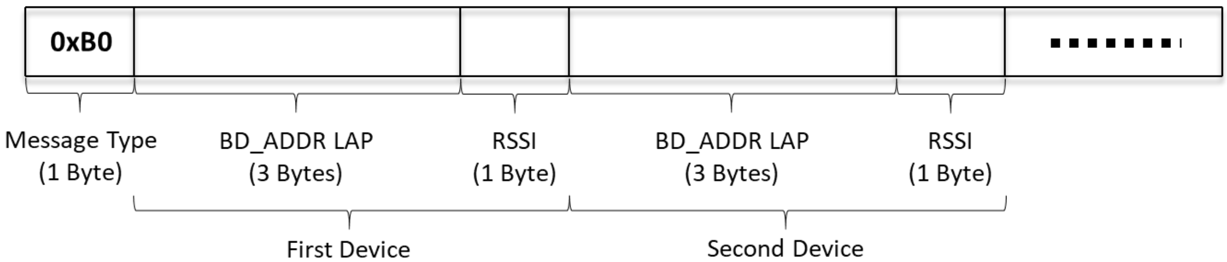
For example, if the only discoverable BLE devices of interest all have MAC addresses that begin with 647FDA (OUI) and only the last 3 bytes are different for each device (LAPs), the BD_ADDR range to filter for only these devices would be 647FDA:000000-FFFFFF. In other words, this BD_ADDR range means that the Tracker will filter the discovered devices to include only those with MAC addresses from 647FDA000000 to 647FDAFFFFFF, inclusive.

Zero, one, or more devices can be reported in a single message, depending on the number of devices available to report and payload size limitation as determined by LoRaWAN regional parameters (LoRa Alliance, Feb 2017). See Section 1.1 for how many and what devices are chosen to get reported OTA. The case of zero devices (an empty BLE device list) happens when no devices are discovered. If there are more devices to be reported than can fit into one message, more than one UL will be subsequently transmitted to report all *N* devices.

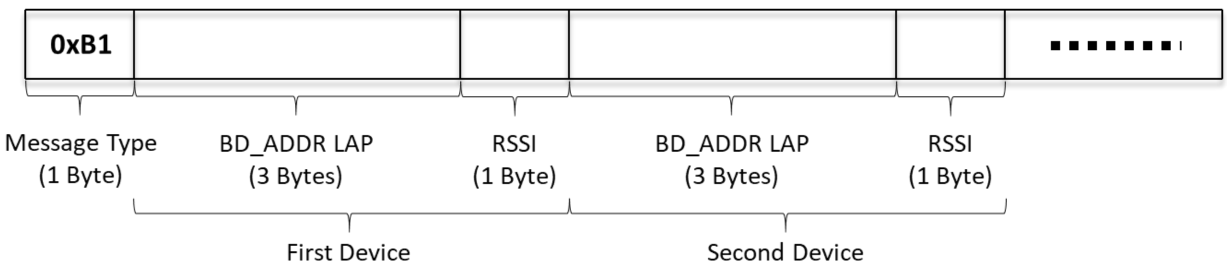
The scan results are normally reported in a UL immediately after the scan concludes, but may be delayed due to duty cycle limitations (LoRa Alliance, Feb 2017). If a new BLE scan occurs before the results of the previous scan have been sent, the old scan results will be discarded.



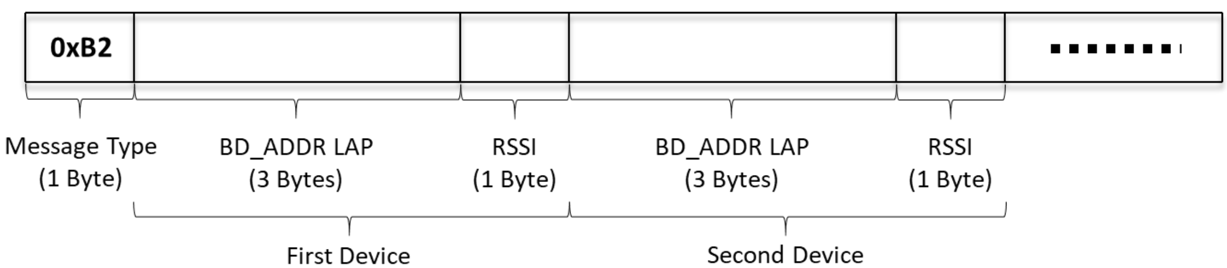
(a) Basic reporting.



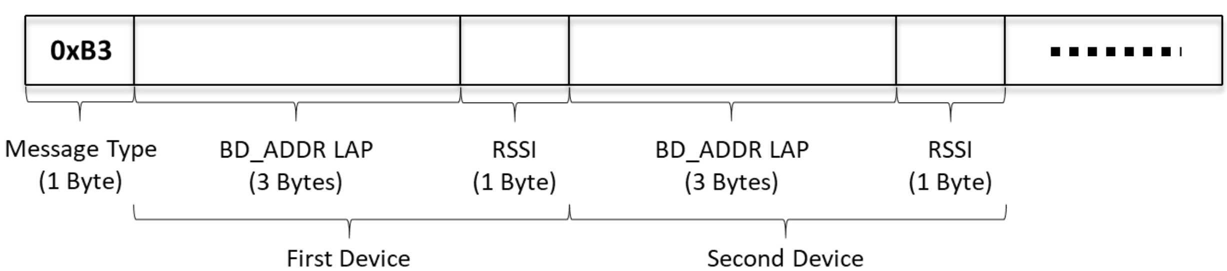
(b) Filtered reporting, Range 0.



(c) Filtered reporting, Range 1.



(d) Filtered reporting, Range 2.



(e) Filtered reporting, Range 3.

Figure 2-2: The UL payload format to report discovered BLE devices.

2.3 Response to Configuration and Control Commands

Sensor responses to DL configuration and control commands (which are sent on LoRaWAN port 100; see Section 3.2) are sent in the UL on ***LoRaWAN port 100***. These responses include:

- Returning the value of a configuration register in response to a query from a DL read command.
- Returning an acknowledgement after a successful reconfiguration of a register(s) following a DL write command.

In the former case, the Sensor responds by the address and value of each of the registers under inquiry (this can be in one or more consecutive UL packets depending on the maximum frame payload size allowed).

In the latter case, the Sensor responds with a 4-Byte CRC32 of the entire DL payload (which may be a combination of read and write commands) as the first 4 bytes of the UL frame. If the DL payload has also had read commands, the 4 CRC32 bytes are followed by the address and value of each of the registers under inquiry (like the Sensor response in the former case).

3 DL Payload Formats and Configuration Settings

The DL streams (from the NS to the Sensor) supported by the SW are shown in Table 3-1, and are explained in Sections 3.1 and 3.2.

Table 3-1: DL Information Streams

Data Type	Sent on LoRaWAN Port
Putting Sensor into DEEP SLEEP	**99**
Configuration and Control Commands	**100**

3.1 Putting Sensor to Sleep OTA

For information about the alternative method for putting the Sensor into DEEP SLEEP (i.e. with a magnet), refer to Section 1.4.

The Sensors ship in a state of DEEP SLEEP to conserve power. Once activated by the magnetic wakeup pattern (see Section 1.4), a Sensor can be remotely put back into DEEP SLEEP by sending an OTA DL. This DL should be sent on ***LoRaWAN port 99*** and the payload should simply contain **0x 00**.

Upon receiving this DL, the Sensor will reset then enter DEEP SLEEP.

The magnetic wakeup pattern will then be required to bring this Sensor out of DEEP SLEEP, as usual.

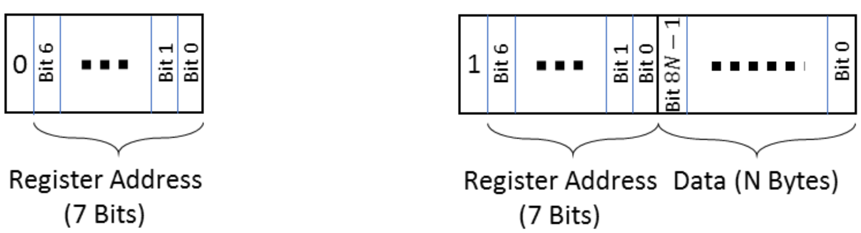
3.2 Configuration and Control Commands

Configuration and control commands are used to query current Sensor settings, reconfigure Sensor settings, or tell the Sensor to perform some action. All configuration and control commands are sent in reference to one or more *configuration register*. Each register has an *address* that is linked to a particular setting or action. These addresses are bound between 0x 00 and 0x 7F, inclusive.

A single DL configuration and control message can contain multiple command blocks, with a possible mix of read and write commands. Each message block is formatted as shown in Figure 3-1. A big-endian format (MSb/MSB first) is always followed.

Bit 7 of the MSB (the left-most bit in the register address) determines whether a read or write action is being performed, as shown in Figure 3-1. All read commands are one-byte long. Data following a read command will be interpreted as a new command block.

Read commands are processed last. For example, in a single DL message, if there is a read command from a register and a write command to the same register, the write command is executed first.



a) The Read Command (b) The Write Command.

Figure 3-1: The Configuration and Control Message Block DL Format

All DL configuration and control commands are sent on ***LoRaWAN port 100***.

Examples:

- Read registers 0x 2A, 0x 2B, and 0x 2C:
  - DL command: {0x 2A 2B 2C}
- Read register 0x 20 and Write value 0x 80 00 to register 0x 25:
  - DL command: {0x 20 A5 80 00}

When a write command is sent to the Sensor, it immediately responds with a CRC32 of the entire DL payload as the first 4 bytes of the UL frame on ***LoRaWAN port 100*** (also see Section 2.3).

DL configuration and control commands fall into one of the following categories and are respectively discussed in Sections 3.2.1 through 3.2.9:

- Toggle Tracker/Beacon Mode Configuration

- LoRaMAC Configuration
- Periodic Tx Configuration
- Function Button Configuration
- Accelerometer Configuration
- Battery Management Configuration
- BLE Scanning (Tracker Mode) Configuration
- BLE Advertising (Beacon Mode) Configuration
- Command and Control

3.2.1 Toggle Tracker/Beacon Mode Configuration

The device is capable of operating in Tracker mode (BLE Rx only) or Beacon mode (BLE Tx only), as explained in Section 1.1. The differences between these modes are due to the different configuration register values listed in Table 3-2. The default values of all other registers not listed in Table 3-2 are not affected by switching between Tracker and Beacon modes; that is, all other registers do not impact specific Tracker or Beacon mode operation and can be configured independently of the current mode.

Table 3-2: Default Register Values for Tracker/Beacon Mode

Name	Register Address [Hex]	Default Value, Tracker Mode (Factory Default) [Hex]	Default Value, Beacon Mode [Hex]
LoRaMAC DR and Tx Power	12	04 00	00 00
Ticks per BLE Scan	25	00 01	00 00 (disabled)
Function Button Mode	2A	00 08	00 01
Accelerometer Mode	40	87	07
Acceleration Event Value to Tx	46	03	01
Advertising Enable / Disable	58	00	01

Instead of changing all 6 registers, the single register 0x0A should be used to easily switch between Beacon and Tracker modes.

DL commands to change this register and UL containing the value of this register are be sent on ***LoRaWAN port 100***, just as for all other sensors.

Table 3-3: Tracker/Beacon Mode Configuration Register

Address	Name	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 0A	Tracker/Beacon Mode	1 B	• 0x 00 = Tracker Mode • 0x 01 = Beacon Mode	Tracker Mode **0x 00**	*tracker_beacon_mode: <value> (unsigned/no unit)*

When a command is sent to the Sensor to write to register 0x 0A (i.e., to switch between Tracker and Beacon Modes), the device always resets and loads the default register values that correspond to the desired mode. Specifically:

- If the DL is sent containing the payload 0x 8A 01, the device would reset, load the Beacon mode default register values, join the network again, and send the MAC address in UL1 on ***LoRaWAN port 100***.
- If the DL is sent containing the payload 0x 8A 00, the device would reset, load the Tracker mode default register values, join the network again, conduct a BLE scan, and send the results of that scan on ***LoRaWAN port 10***.

After the device resets and rejoins the network operating in the desired mode, all registers are reset to the appropriate default values. This means that all other non-default settings will be lost and will need to be reconfigured after the switching of modes. All settings are accessible in either Beacon or Tracker Mode, including the enabling of BLE advertising while in Tracker mode.

3.2.2 LoRaMAC Configuration

LoRaMAC options can be configured to change the default MAC configuration that the Sensor loads on start-up. They can also change certain run-time parameters. Table 3-4 shows the MAC configuration registers. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-4: LoRaMAC Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 10	Join Mode	2 B	• Bit 15: 0/1 = ABP/OTAA mode • Bits 0-14: Ignored	OTAA mode **0x 80 00**	*loramac_join_mode: <value> (unsigned/no unit)*

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 11	Options	2 B	• Bit 0: 0/1 = Unconfirmed/Confirmed UL • Bit 1 = 1 (RO): 0/1 = Private/Public Sync Word • Bit 2: 0/1 = Disable/Enable Duty Cycle • Bit 3: 0/1 = Disable/Enable ADR • Bits 4-15: Ignored	• Unconfirmed UL • Public Sync Word • Duty cycle enabled[4] • ADR enabled **0x 00 0E**	*loramac_opts {* *confirm_mode: <value>, (unsigned/no unit)* *sync_word: <value>, (unsigned/no unit)* *duty_cycle: <value>, (unsigned/no unit)* *adr: <value> (unsigned/no unit)* *}*
0x 12	DR and Tx Power[5]	2 B	• Bits 8-11: Default DR number • Bits 0-3: Default Tx power number • Bits 4-7, 12-15: Ignored	• DR0 • Tx Power 0 (max power; see Table 3-5) **0x 04 00**	*loramac_dr_tx {* *dr_number: <value>, (unsigned/no unit)* *tx_power_number: <value> (unsigned/no unit)* *}*
0x 13	Rx2 Window	5 B	• Bits 8-39: Channel frequency in Hz for Rx2 • Bits 0-7: DR for Rx2	As per Table 3-6	*loramac_rx2 {* *frequency: <value>, (unsigned/Hz)* *dr_number: <value> (unsigned/no unit)* *}*

Table 3-5: Default Maximum Tx Power in Different Regions

RF Region	Max Tx EIRP [dBm]
EU868	16
US915	30
AS923	16
AU915	30
IN865	30

[4] In the LoRaMAC RF regions where there is no duty cycle limitation, such as US915, the "enabled duty cycle" configuration of the Sensor is ignored.

[5] Tx power number m translates to the maximum Tx power, which is a function of the LoRaWAN RF region, minus $2 \times m$ dB. For a list of all values for each region, refer to the corresponding section in the LoRaWAN Regional Parameters specification document [1].

RF Region	Max Tx EIRP [dBm]
KR920	14
RU864	16

Table 3-6: Default Values of Rx2 Channel Frequency and DR Number in Different Regions

RF Region	Channel Frequency [Hz]	DR Number
EU868	869525000	0
US915	923300000	8
AS923	923200000	2
AU915	923300000	8
IN865	866550000	2
KR920	921900000	0
RU864	869100000	0

Note: Modifying these LoRaMAC settings only changes them in the Sensor; LoRaMAC setting in the NS may also need to be changed depending on the desired use case and to ensure a Sensor is not stranded without being able to communicate with the network. Modifying configuration parameters in the NS is outside the scope of this document.

3.2.2.1 LoRaMAC Configuration DL Examples

- Switch Sensor to ABP Mode:
  - DL payload: 0x 90 00 00
- Set ADR On, No Duty Cycle, and Confirmed UL Payloads:
  - DL payload: 0x 91 00 0B
- Set default DR number to 3, default Tx power number to 4:
  - DL payload: 0x 92 03 04

3.2.3 Periodic Tx Configuration

All periodic reporting of telemetry is synchronized around ticks. The *core tick* is simply a user-configurable time base unit that is used to schedule Sensor measurements. For each transducer or subsystem in the Sensor, the number of elapsed ticks before transmitting can be defined. Table 3-7 shows a list of registers used to configure the Sensor periodic transmissions. All the registers have R/W access. Note that these registers only control *periodic reporting* and do not affect *event-based reporting* of the same type of telemetry (see Section 2).

The reporting period for each transducer is obtained as per the following:

<Transducer> *Reporting Period* = *Seconds per Core Tick* × *Ticks per* <Transducer>

where <Transducer> can be "Battery", "Accelerometer", "BLE Scan", or "MCU Temperature", as shown in Table 3-7. If <Transducer> *Reporting Period* equals 0, it means that the <Transducer> periodic reporting is disabled. This happens when either the *Seconds per Core Tick* or *Ticks per* <Transducer> is equal to 0. The above relationship also shows that setting *Seconds per Core Tick* to 0 disables all periodic reporting. However, to disable the periodic reporting of a specific transducer, it is enough to set its *Ticks per* <Transducer> to 0.

Table 3-7: Periodic Transmission Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 20	Seconds per Core Tick	4 B	• Tick value for periodic events • Acceptable values: 0, 3, 4, …, 86400 • 0 disables all periodic transmissions • Other values: Invalid and ignored	3600 seconds **0x 00 00 0E 10**	*seconds_per_core_tick: <value> (unsigned/sec)*
0x 21	Ticks per Battery	2 B	• Ticks between battery reports • Acceptable values: 0, 1, 2, …, 65535 • 0 disables periodic battery reports	24 ticks = 1-day period **0x 00 18**	*ticks_per_battery: <value> (unsigned/no unit)*
0x 24	Ticks per Accelerometer	2 B	• Ticks between accelerometer reports • Acceptable values: 0, 1, 2, …, 65535 • 0 disables periodic accelerometer reports	Periodic reporting disabled **0x 00 00**	*ticks_per_accelerometer: <value> (unsigned/no unit)*
0x 25	Ticks per BLE Scan	2 B	• Ticks between BLE reports • Acceptable values: 0, 1, 2, …, 65535 • 0 disables periodic BLE reports	1 tick = 1-hour period **0x 00 01**	*ticks_per_ble: <value> (unsigned/no unit)*
0x 28	Ticks per MCU Temperature	2 B	• Ticks between temperature reports • Acceptable values: 0, 1, 2, …, 65535 • 0 disables periodic temperature reports	Periodic reporting disabled **0x 00 00**	*ticks_per_mcu_temperature: <value> (unsigned/no unit)*

3.2.3.1 Anti-Bricking Strategy

As a class-A LoRaWAN end-device, the Sensor can only be receptable to a DL in the short period after sending an UL. Therefore, if the Sensor is configured to send ULs very infrequently or not at all, it could become impossible to send a DL command. As the function button operation (see Sections 1.3 and 3.2.4) and the magnetic switch operation (see Section 1.4) cannot be disabled, it is impossible to completely brick the Sensor with a bad configuration; it is always possible to trigger the Sensor to UL something so it can receive DL commands for a desired configuration change.

However, there are use cases in which using the function button or magnetic switch to trigger the Sensor may not be a convenient option, e.g., due to special mounting orientation, remote location, or in the case of reconfiguring a large number of devices. In these use cases, strategies to avoid bricking the Sensor are beneficial and included in the FW as follows.

The undesirable combinations that make the Sensor almost or completely irresponsive are:

> *All periodic reports are disabled OR have a minimum period of larger than a week, AND the BLE receiver AND accelerometer are disabled.*

If, after a configuration update request from the NS, the Sensor SW detects that any of the above situations occurs, the SW automatically sets the *core tick* value to 86400 seconds (i.e. one day) and the *ticks per battery* to 1.

3.2.3.2 Periodic Tx Configuration DL Examples

- Disable all periodic events:
    - DL payload: 0x A0 00 00 00 00
        - Register 20 with the write bit set to true
        - Seconds per Core Tick set to 0—i.e., disable periodic transmissions
- Read current value of Seconds per Core Tick:
    - DL payload: 0x 20
        - Register 20 with the write bit set to false
- Report temperature every tick:
    - DL payload: 0x A8 00 01
        - Register 28 with its write bits set to true
        - Ticks per MCU Temperature set to 1

3.2.4 Function Button Configuration

Only the AA-cell (SPARROW) variants are equipped with a function button and it is used to trigger event-based reporting.

The operation of the function button has been described in Section 1.3. It is configurable how a function button event can be registered such that the Sensor sends an uplink. It is also configurable what uplink is sent when a function button event is registered. Two types of function button events are defined:

- Event Type I: Press the button *for at least* m times within n sec.
- Event Type II: Press and hold the button *for at least* T sec.

Event Type I is registered *as soon as* the button is pressed m times within n sec. Event Type II is registered *as soon as* the button is pressed and held for T sec.

One and only one event type is enabled at any time, and therefore, the function button operation cannot be disabled. The Sensor is triggered to send an uplink when the enabled event type is registered.

Table 3-8 shows the configuration registers for the function button. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-8: Function Button Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 2A	Mode	2 B	• Bit 15: 0/1 = Event Type I/II enabled • Bit 0: 0/1 = Battery report disabled/enabled • Bit 1: 0/1 = Acceleration vector report disabled/enabled • Bit 2: 0/1 = Temperature report disabled/enabled • Bit 3: 0/1 = BLE report disabled/enabled Unavailable in Beacon Mode • Bits 4-14: Ignored	• Event Type I enabled • BLE report enabled **0x 00 08**	*fb_mode {* *event_type: <value>, (unsigned/no unit)* *battery_voltage_report: <value>, (unsigned/no unit)* *acceleration_vector_report: <value>, (unsigned/no unit)* *temperature_report: <value>, (unsigned/no unit)* *ble_report: <value> (unsigned/no unit)* *}*

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 2B	Event Type I Configuration	1 B	• Bits 0-3: Value of m $1 \leq m \leq 15$ $m = 0$: *Invalid and ignored* • Bits 4-7: Value of n $1 \leq n \leq 15$ $n = 0$: *Invalid and ignored* • $m > 2n$: *Invalid and ignored*	• m = 1 • n = 1 **0x 11**	*fb_event_type1 {* *m_value: <value>,* *(unsigned/no unit)* *n_value: <value>* *(unsigned/no unit)* *}*
0x 2C	Event Type II Configuration	1 B	• Bits 0-3: Value of T $1 \leq T \leq 15$ $T = 0$: *Invalid and ignored* • Bits 4-7: Ignored	T = 1 **0x 01**	*fb_event_type2 {* *t_value: <value>* *(unsigned/no unit)* *}*

3.2.4.1 Mode

Bits 0 and 1 of byte B_0 of the *Mode* register determines which event type is desired. Bits 0-3 of byte B_1 of the *Mode* register determines what the Sensor transmits when an event is registered.[6] If all bits 0-3 are 0, an empty uplink is transmitted on port 10 upon registration of a function button event.

In Tracker mode, the default action upon button press is a BLE scan and report. In Beacon mode, the default action upon button press is a battery report, sending the data in the format as determined by register 0x 4A (see Section 3.2.6). Note that a BLE scan and report is unavailable in Beacon mode. If a write command is sent with bit 3 of register 0x 2A set to 1, this command will be ignored.

3.2.4.2 Event Type I/II Configuration

Registers 0x 2B and 0x 2C configure Event Types I and II, respectively. Note that trying to set m, n, or T to 0 is invalid and ignored by the SW. Also, for Event Type I, trying to set m or n in a way that $m > 2n$ is invalid and ignored by the SW. This is to avoid a situation where the user is needed to press the button too fast to register a function button event. For example, it is impractical for a human to press the function button 15 times in 1 sec.

3.2.4.3 Function Button Configuration DL Examples

- Enable BLE report and battery voltage report (with event type I):
  - DL payload: 0x AA 80 09

---

[6] The battery voltage, acceleration vector, temperature, and discovered BLE devices are all reported using their usual uplink formats explained in Section 2.

- Read current value of *Event Type II Configuration:*
    - DL payload: 0x 2C

3.2.5 Accelerometer Configuration

The accelerometer in the Sensor can be disabled or enabled and supports both periodic-based and event-based reporting. The accelerometer is enabled by default.

For a description of how the accelerometer function works, refer to Section 1.3.

Table 3-9 shows a list of accelerometer configuration registers. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-9: Accelerometer Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 40	Mode	1 B	• Bit 0: 0/1 = X-axis disabled/enabled • Bit 1: 0/1 = Y-axis disabled/enabled • Bit 2: 0/1 = Z-axis disabled/enabled • Bits 3-6: Ignored • Bit 7: 0/1 = Accelerometer off/on	• X-axis enabled • Y-axis enabled • Z-axis enabled • Accelerometer on **0x 87**	*accelerometer_mode {* *xaxis_enabled: <value>, (unsigned/no unit)* *yaxis_enabled: <value>, (unsigned/no unit)* *zaxis_enabled: <value>, (unsigned/no unit)* *poweron: <value> (unsigned/no unit)* *}*

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 41	Sensitivity	1 B	• Bits 0-2 (Sample Rate): 0: Invalid and ignored 1/2/3/4/5/6/7 = 1/10/25/50/100/200/400 Hz • Bits 4-5 (Measurement Range[7]): 0/1/2/3 = ±2/±4/±8/±16 *g* • Bits 3, 6, 7: Ignored	• Sample Rate 10 Hz • Measurement Range ±8 *g* **0x 22**	*accelerometer_sensitivity* *{* *sample_rate: <value>, (unsigned/Hz)* *measurement_range: <value> (unsigned/g)* *}*
0x 42	Acceleration Event Threshold Count	2 B	• Number of acceleration events before an acceleration alarm is registered • Acceptable values: ≥ 1 • 0: Invalid and ignored	1 event **0x 00 01**	*acceleration_event_threshold_count: <value> (unsigned/no unit)*
0x 43	Acceleration Event Threshold Period	2 B	• Period in sec over which acceleration events are counted for threshold detection • Acceptable values: ≥ 5 • 0-4: Invalid and ignored	10 seconds **0x 00 0A**	*acceleration_event_threshold_period: <value> (unsigned/sec)*
0x 44	Acceleration Event Threshold	2 B	• Unsigned, 1 milli-*g*/LSb	2 *g* **0x 07 D0**	*acceleration_event_threshold: <value> (unsigned/g)*
0x 45	Acceleration Event Grace Period	2 B	• Time to pass, in sec, after the last acceleration alarm before the alarm can be cleared) • Acceptable values: ≥ 15 • 0-14: Invalid and ignored	5 min **0x 01 2C**	*acceleration_event_grace_period: <value> (unsigned/sec)*
0x 46	Acceleration Event Value to Tx	1 B	• Bit 0: 0/1 = Acceleration alarm UL report disabled/enabled • Bit 1: 0/1 = BLE scan report disabled/enabled • Bits 2-7: Ignored	• Acceleration alarm report UL enabled • BLE scan report enabled **0x 03**	*acceleration_event_tx {* *acceleration_alarm: <value>, (unsigned/no unit)* *ble: <value> (unsigned/no unit)* *}*

[7] Measurement ranges ±2 *g*, ±4 *g*, ±8 *g*, ±16 *g* correspond to typical transducer output precisions of 16 m*g*, 32 m*g*, 64 m*g*, 192 m*g*, respectively. Note that if a threshold configured in register 0x44 is equal to or greater than the configured measurement full scale (2 *g*, 4 *g*, 8 *g*, 16 *g*), then the acceleration alarm will never be triggered.

3.2.5.1 Mode

The accelerometer can be enabled or put in a power-down mode to save battery life. Additionally, it is possible to enable/disable X, Y, Z axes independently. When an axis is disabled, it is not considered in monitoring acceleration events. Also, its corresponding value in the output acceleration vector is 0.

3.2.5.2 Sensitivity

When enabled (powered on), the accelerometer always samples the transducer element at a fixed rate, called the sample rate. To capture an acceleration event, the physical event needs to last longer than the sample period. Larger sample rates have a shorter period and can therefore resolve shorter acceleration events. However, sampling the transducer at a larger rate increases the power usage, and impacts the battery life. Table 3-10 shows typical current draw deltas (with respect to the MCU background current at sleep) for the different sample rates when the accelerometer is enabled.

Table 3-10: Typical Current Draw Deltas at 3.6 V for Different Accelerometer Sample Rates

Sample Rate [Hz]	1	10	25	50	100	200	400
Current Draw [μA]	1.0	1.5	2.3	3.7	6.1	11.4	22.0

Furthermore, the Sensitivity register sets the measurement range or full-scale, which shows the dynamic range of accelerations that can be monitored on any enabled axis. Note that when enabled, the accelerometer is always put in a low-power mode, which means the output acceleration values on any given axis (X, Y, or Z), is an 8-bit signed number. Therefore, a measurement range of ±2 *g* implies a precision of 4/256 *g*/LSb.

3.2.5.3 Acceleration Event Threshold Count

By default, the accelerometer registers an acceleration event each time it detects movement. Depending on the use case, it may be desirable to increase the threshold count to reduce sensitivity. This feature is to allow for filtering out short acceleration events, while still allowing longer acceleration events to be reported.

3.2.5.4 Acceleration Event Threshold Period

The *Acceleration Event Threshold Period* is the amount of time that acceleration events are accumulated for threshold detection. For example, an *Acceleration Event Threshold Period* of 10 sec accumulates acceleration events over a 10-sec period from the time of first detection. An acceleration event is registered only if the number of acceleration events reaches the *Acceleration Event Threshold Count* before the *Acceleration Event Threshold Period* expires.

3.2.5.5 Acceleration Event Threshold

This parameter is the g-threshold for an acceleration event. Acceleration events are registered only if the *Acceleration Event Threshold* is exceeded on at least one of the enabled axes (X, Y, Z) within the *Acceleration Event Threshold Period* for at least the *Acceleration Event Threshold Count* number of times.

3.2.5.6 Acceleration Event Grace Period

The *Acceleration Event Grace Period* determines how long the Sensor waits before the previously registered acceleration event is cleared. For example, an *Acceleration Event Grace Period* of 5 min means that the Sensor clears a previously registered acceleration event only 5 min after the registered event, then starts monitoring to register a new acceleration event.

3.2.5.7 Acceleration Event Value to Tx

This register determines what is reported when an acceleration event is registered. The options are the acceleration alarm and BLE scan. In the case of the BLE scan reporting enabled, a BLE scan with the duration for event-based reports (register 0x 51; see Section 3.2.7) is performed and the result is reported.

3.2.5.8 Accelerometer Configuration DL Examples

- Power on accelerometer and enable all axes:
  - DL payload: 0x C0 87
- Change threshold value to 800 m*g*:
  - DL payload: 0x C4 03 20
- Read *Accelerometer Value to Tx*:
  - DL payload: 0x 46

3.2.6 Battery Management Configuration

The sensor is equipped with a battery-gauging system that allows the ability to report battery data in units of remaining battery capacity [%], or remaining battery lifetime [days], or both. These battery gauging values are based on the nominal capacity of a new battery as well as usage from the time the device is first powered-on.

Table 3-2 shows the battery configuration register options. The value of this register determines what data is sent in a port 10 UL when a battery report is due (periodic or event-based).

Table 3-11: Battery Management Configuration Register

Address	Name	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 4A	Battery Report Options	1 B	• Bit 0: Ignored Deprecated; was used for voltage reporting, which is no longer supported in Gen2.5 devices • Bit 1: 0/1 = Remaining battery capacity [%] not reported/reported • Bit 2: Remaining battery lifetime [days] not reported/reported • Bits 0-2 all set to 0: Invalid and ignored • Bits 3-7: Ignored	Remaining battery capacity [%] and remaining battery lifetime [days] reported **0x 06**	*battery_tx {* *report_voltage_enabled: <value>, (unsigned/no unit)* *report_capacity_enabled: <value> (unsigned/no unit)* *report_lifetime_enabled: <value>, (unsigned/no unit)* *}*
0x 4B	Average Energy Trend Window	1 B	• Bits 0-7: Number of core ticks, ***w*** [1 update/LSB] Acceptable values: 1, 2, …, 255 0: Invalid and ignored	10 core ticks **0x 0A**	*avg_energy_trend_window: <value>* *(unsigned/no unit)*

3.2.6.1 Battery Report Options

This register determines what type of data is reported at the time a battery report is due (according to the periodicity defined by the values of registers 0x 20 and 0x 21). By default, the remaining battery capacity and remaining battery lifetime are reported. Gen2 devices only support battery voltage reporting.

3.2.6.2 Average Energy Trend Window

While the remaining battery capacity gradually drops throughout during normal operation of the Sensor, the remaining battery lifetime may go up or down, depending on the energy consumption of the device. For example, a Sensor configured to send a UL report every 15 minutes will consume more energy than one that is configured to send a UL report every 60 minutes. And since the Sensors can be reconfigured at any time during normal operation and change their energy consumption rate, the remaining lifetime will in-turn change as well.

The Sensor handles this by monitoring the average energy consumption trend over a certain time; the *Average Energy Trend Window*. The Sensor updates how much energy it has consumed at each core tick (register 0x 20). The average energy trend window specifies the number of preceding core ticks over which the energy consumption trend is calculated. This average energy consumption trend is then used to estimate the remaining battery lifetime.

Note that the JOIN procedure consumes energy at a higher rate than normal default operation, so the remaining lifetime value reported will take some time to stabilize after the Sensor joins the network. It will take *w* core ticks over which "steady-state" energy consumption occurs before the remaining battery lifetime value will stabilize.

It is recommended that the core tick settings and average energy trend window be configured in relation to each other. For example, a large core tick setting and large window will result in a long time for the remaining battery lifetime to be calculated accurately and will take a long time to respond to changes in energy consumption. A small core tick setting and small window will result in more fluctuations in consecutive battery reports.

3.2.6.3 Note on End-of-Service Gauging

The algorithm is based on the average nominal battery capacity of a new battery, so when the battery is replaced, the remaining capacity is automatically reset. However, the remaining battery capacity will not be reset when the device is soft-reset for any reason, including an OTA reboot or switching between Beacon and Tracker modes.

3.2.7 BLE Scanning (Tracker Mode) Configuration

The BLE module is embedded in the MCU. It plays the role of a BLE central device that can search to discover nearby BLE peripherals periodically or on-demand. It can be used as a standalone proximity sensor used for positioning.

Figure 1-1 shows how periodic BLE scans are performed; also see Section 1.1 for details about how BLE operates in the Sensor. Table 3-12 shows the list of BLE Rx configuration registers. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Registers 0x 50 to 0x 57 are applicable to Tracker mode only. By default, the device is in Tracker mode. For Beacon Mode configuration, see Section 3.2.8. To switch between modes, see Section 3.2.1.

Table 3-12: BLE Scanning (Tracker Mode) Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 50	Mode	1 B	• Bits 0-6: ***N*** Number of reported devices (1–127) 0: Disables BLE • Bit 7: ***A*** 0/1 = Averaging mode off/on	• Up to 8 reported devices. • Averaging mode on. **0x 88**	*ble_mode: {* *num_reported_devices : <value>,* *(unsigned/no unit)* *averaging_mode: <value>* *(unsigned/no unit)* *}*
0x 51	Scan Duration	2 B	• Bits 0-7: Scan duration for periodic reports (1 sec/LSb) Acceptable values: 1, 2, …, 255 0: Invalid and ignored • Bits 8-15: Scan duration for event-based reports (1 sec/LSb) Acceptable values: 1, 2, …, 255 0: Invalid and ignored	• 3 seconds for periodic scans • 1 second for event-based scans **0x 01 03**	*ble_scan_duration: {* *periodic: <value>,* *(unsigned/sec)* *event_based: <value>* *(unsigned/sec)* *}*
0x 52	Scan Interval	2 B	• Scan interval (1 ms/LSb) • Acceptable values: "Scan Window", …, 10000 • Other values: Invalid and ignored	30 ms **0x 00 1E**	*ble_scan_interval: <value>* *(unsigned/sec)*
0x 53	Scan Window	2 B	• Scan window (1 ms/LSb) • Acceptable values: 3, …, "Scan Interval" • Other values: Invalid and ignored	30 ms **0x 00 1E**	*ble_scan_window: <value>* *(unsigned/sec)*
0x 54	Filter Range 0	9 B	• Range 0 for filtered BD_ADDRs • $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ • OUI = $B_0:B_1:B_2$ • $LAP_{start} = B_3:B_4:B_5$ nhjk • $LAP_{end} = B_6:B_7:B_8$	Range inactive **0x 00 00 00 00 00 00 00 00 00**	*filter_range_0: {* *oui: <value>,* *(unsigned/no unit)* *lap_start: <value>,* *(unsigned/no unit)* *lap_end: <value>* *(unsigned/no unit)* *}*

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 55	Filter Range 1	9 B	• Range 1 for filtered BD_ADDRs • B_0:B_1:B_2:B_3:B_4:B_5 to B_0:B_1:B_2:B_6:B_7:B_8 • OUI = B_0:B_1:B_2 • LAP_{start} = B_3:B_4:B_5 • LAP_{end} = B_6:B_7:B_8	Range inactive **0x 00 00 00 00 00 00 00 00 00**	*filter_range_1: {* *oui: <value>, (unsigned/no unit)* *lap_start: <value>, (unsigned/no unit)* *lap_end: <value> (unsigned/no unit)* *}*
0x 56	Filter Range 2	9 B	• Range 2 for filtered BD_ADDRs • B_0:B_1:B_2:B_3:B_4:B_5 to B_0:B_1:B_2:B_6:B_7:B_8 • OUI = B_0:B_1:B_2 • LAP_{start} = B_3:B_4:B_5 • LAP_{end} = B_6:B_7:B_8	Range inactive **0x 00 00 00 00 00 00 00 00 00**	*filter_range_2: {* *oui: <value>, (unsigned/no unit)* *lap_start: <value>, (unsigned/no unit)* *lap_end: <value> (unsigned/no unit)* *}*
0x 57	Filter Range 3	9 B	• Range 3 for filtered BD_ADDRs • B_0:B_1:B_2:B_3:B_4:B_5 to B_0:B_1:B_2:B_6:B_7:B_8 • OUI = B_0:B_1:B_2 • LAP_{start} = B_3:B_4:B_5 • LAP_{end} = B_6:B_7:B_8	Range inactive **0x 00 00 00 00 00 00 00 00 00**	*filter_range_3: {* *oui: <value>, (unsigned/no unit)* *lap_start: <value>, (unsigned/no unit)* *lap_end: <value> (unsigned/no unit)* *}*

3.2.7.1 Mode

The *Mode* register controls both:

- ***N:*** the number of reported devices (MAC address + RSSI value pairs) to be reported in an UL following a BLE scan, and
- ***A:*** enabling or disabling averaging mode.

Following each scan, there are two different modes for reporting the discovered BLE device data, depending on the value of ***A***:

A. **Averaging Mode (Default), *A* = 1**: It is often the case that more than one packet from the same BLE device is received during a single scan. When averaging mode is enabled, the RSSIs for each device are averaged, then the devices are sorted and uplinked in order from strongest average RSSI to weakest, up to ***N*** distinct devices. ***N*** is user-specified by bits 0-6, and can range from 0 – 127.

    i. If there are less than or equal to ***N*** averaged values, all of the discovered devices with their averaged RSSIs are sent in the UL, in order of decreasing average RSSI.
    ii. If there are more than ***N*** averaged values, the ***N*** strongest discovered devices and their averaged RSSIs are sent in the UL.

B. **Raw Mode, *A* = 0:** The packets received during each scan are neither averaged nor sorted before uplinking; the ***N*** last discovered devices are sent. This means it is possible to have repeated devices in the same UL.

    i. If there are less than or equal to ***N*** devices discovered during the scan, all of the discovered devices shall be sent in the UL.
    ii. If there are more than ***N*** devices discovered during the scan, the ***N*** last discovered devices shall be sent in the UL.

3.2.7.2 Scan Duration, Interval, and Window.

Figure 1-1 shows the BLE scan scheme in the Tracker when the BLE scan is enabled. As shown in the figure, BLE scans are performed periodically with a configurable scan period. Also, each scan duration is divided into scan intervals. The BLE scan is performed only in a scan window portion of the scan interval. The ratio of the scan window to the scan interval is the scan duty cycle. A scan window equal to the scan interval represents a scan duty cycle of 100% (a continuous scan) over each scan duration. This is the default behavior as it maximizes the chance of "discovering" nearby BLE advertisement packets for a given scan duration. Reducing the duty cycle below 100% has the advantage of decreasing power consumption at the expense of possibly missing some beacon signals.

As observed from Table 3-12, the scan duration can be independently set for periodic reports and event-based reports through register 0x 51. A BLE event-based scan and report is made after a function button event if such events are configured to trigger a BLE scan (see Sections 1.3 and 3.2.4). Other BLE scan configuration parameters, i.e. number of reported devices, scan interval, and scan window, remain the same for both periodic and event-based scans.

3.2.7.3 Filtering

As explained in Section 1.1, there are two types of BLE scan reporting: *Basic* and *Filtered*.

With basic reporting, (up to) ***N*** devices (MAC address + RSSI value pairs) are reported.

With filtered reporting, (up to) ***N*** devices (MAC address + RSSI value pairs) with BD_ADDRs (Bluetooth Device MAC Addresses) within one of configured BD_ADDR ranges are reported. Up to 4 BD_ADDR ranges can be defined through registers 0x 54 to 0x 57. Each BD_ADDR range register has 9 bytes: B_0 (MSB) to B_8 (LSB).

A BD_ADDR consists of an *Organizationally-Unique Identifier* (OUI) comprising the 3 MSBs, followed by a *Lower Address Part* (LAP) comprising the 3 LSBs. Each BD_ADDR range is a 9-byte OUI:LAP_{start}-LAP_{end} that determines the range of BD_ADDRs as OUI:LAP_{start} to OUI:LAP_{end}. Therefore, OUI is the same and known for all devices in each range.

For example, if the only discoverable BLE devices of interest all have MAC addresses that begin with 647FDA (OUI) and only the last 3 bytes are different for each device (LAPs), the BD_ADDR range to filter for only these devices would be 647FDA:000000-FFFFFF. In other words, this BD_ADDR range means that the Tracker will filter the discovered devices to include only those with MAC addresses from 647FDA000000 to 647FDAFFFFFF, inclusive.

With filtered reporting, after a scan is complete, the list of the discovered devices is first filtered to include only those lying in one of the ranges defined in registers 0x 54 to 0x 57. Then, (up to) ***N*** devices in the filtered list are reported using the format described in Section 2.2. Devices falling under each range are reported in one or more messages with a range specific header: 0x B0 for Range 0, 0x B1 for Range 1, 0x B2 for Range 2, and 0x B3 for Range 3. Only LAPs of the devices followed by their RSSIs are reported in the filtered message, as the first 3 bytes are implicitly known.

The following are the rules around using the 4 BD_ADDR ranges:

1. A range set to all 0's: An **inactive range**. Otherwise: an **active range**.

2. All 4 ranges inactive: Basic reporting is enabled (the reporting is be performed with message type header 0x 0A as per Section 2.2). Otherwise, filtered reporting is enabled.

3. With filtered reporting, in every round of BLE scanning, there will be reports corresponding to each active range. If no devices are found to be reported for an active range, an empty list with the header corresponding to that active range is reported (i.e. 0x B0, 0x B1, 0x B2, or 0x B3). For example, if only register 0x 54 is non-zero, only Range 0 is active, thus, discovered devices are reported in messages with header 0x B0, and if no device is found under Range 0, a one-byte message of 0x B0 is reported. If only registers 0x 54 and 0x 55 are non-zero, then only Range 0 and Range 1 are active, and discovered devices are reported in messages with header 0x B0 for Range 0 and header 0x B1 for Range 1. In this case, in every BLE scan, we will ALWAYS have reports (of even an empty list) with both headers 0x B0 and 0x B1.

4. A range with $LAP_{start} > LAP_{end}$: The range is active, but empty (i.e., always an empty list is reported with its corresponding header—0x B0, 0x B1, 0x B2, or 0x B3).

5. An active range with $LAP_{start} = LAP_{end}$: The range has only one BD_ADDR in it.

6. It is possible that the ranges overlap. A beacon that is in at least one of the ranges and is to be reported, is always reported under the first range (from Range 0 to 3) that it falls into.

3.2.7.4 Guidelines on BLE Scan Configuration

In the case of periodic BLE scanning, although the BLE scan period, scan duration, and number of devices to report can be freely configured to different values, a bad combination can result in the Tracker not responding as desired. The general rule of thumb is that the scan duration plus the time to report the discovered devices should be smaller than the scan period. The report time is a function of the UL DR and number of devices to report. For example, while a larger DR takes fewer packets to report a number of devices, a smaller DR would require more packets for the same number. Moreover, due to LoRaWAN standard requirements, the packets cannot be sent out faster than about every 3 sec.

Table 3-13 shows the maximum number of BLE devices that can be accommodated in a single packet, as a function of the LoRaMAC region and DR. In the table entries, the first number is for the case of basic reporting, where each device is reported using 7 bytes. The second number is for the case of filtered reporting where 4 bytes is needed per device. For example, from Table 3-13, it respectively takes (at least) 10 and 5 packets to report 10 discovered beacons using DR0 of US915. But the same 10 devices can be reported in 1 packet using DR3 of EU868.

Table 3-13: Maximum Number of Reported BLE Devices per LoRaWAN Packet in Different Regions

Region	DR0	DR1	DR2	DR3	DR4	DR5	DR6
EU868	7/12	7/12	7/12	16/28	34/60	34/60	34/60
US915	1/2	7/13	17/31	34/60	34/60	N/A	N/A
AS923	7/12	7/12	7/12	16/28	34/60	34/60	34/60
AU915	7/12	7/12	7/12	16/28	34/60	34/60	34/60
IN865	7/12	7/12	7/12	16/28	34/60	34/60	N/A
KR920	7/12	7/12	7/12	16/28	34/60	34/60	N/A
RU864	7/12	7/12	7/12	16/28	34/60	34/60	34/60

Whenever the DR is not certain (e.g., due to enabled ADR, which can change the DR used by the Tracker), it is recommended enough margin for the report time be considered between the scan duration and scan period.

3.2.7.5 BLE Scanning (Tracker Mode) Configuration DL Examples

- Configure the Tracker to report the nearest (strongest) beacon only:
    - DL payload: 0x D0 81
- Decrease the scan duty cycle to 50% while keeping scan durations the same to save battery life:
    - DL payload: 0x D3 00 0F
- Set filters to only report devices with MAC addresses ABCDEF000001 and ABCDEF500000 – ABCDEF999999:
    - DL payload: 0x D4 AB CD EF 00 00 01 00 00 01 D5 AB CD EF 50 00 00 99 99 99

3.2.8 BLE Advertising (Beacon Mode) Configuration

The BLE module is embedded in the MCU. It plays the role of a BLE central device that can periodically broadcast advertisements and act as a Beacon to be discoverable to other BLE devices.

Figure 1-2 shows how periodic BLE advertisements are performed; also see Section 1.1 for details about how BLE operates in the Sensor. Table 3-14 shows the list of BLE Tx configuration registers. All the registers have R/W access, except for 0x 5F, which is RO. In this table, the bit indexing scheme is as shown in Figure 3-1.

By default, the Sensor is in Tracker mode. To switch to Beacon Mode and have access to registers 0x 58 to 0x 5F, see Section 3.2.1. For Tracker mode configuration registers, see Section 3.2.7.

Table 3-14: BLE Advertising (Beacon Mode) Configuration

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 58	Advertising Enable / Disable	1 B	• Bits 7-1: ignored • Bit 0: Advertising enable/disable • 0/1 = Advertising off/on	Advertising disabled **0x 00**	*advertising_enabled: <value> (unsigned/no unit)*
0x 59	Advertising Interval	2 B	Advertising interval (1 ms / LSb) • Acceptable values: 30 ms – 10240 ms • Other values: invalid and ignored	100 ms **0x 00 64**	*min_advertising_interval: <value> (unsigned/ms)*

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 5B	Tx Advertising Power	1 B	BLE Tx Power (dBm)[8] For Gen2 devices: • 0x00 = 0 dBm • 0x01 = -1.5 dBm • 0x02 = -3.0 dBm • 0x04 = -6.5 dBm For Gen2.5 devices: • 0x00 = 0 dBm • 0x01 = -0.85 dBm • 0x02 = -1.8 dBm • 0x03 = -3.15 dBm • 0x04 = -4 dBm • 0x05 = -4.95 dBm • 0x06 = -5.9 dBm • 0x07 = -6.9 dBm • 0x08 = -7.8 dBm • 0x09 = -8.85 dBm • 0x0A = -9.9 dBm • 0x0B = -12.05 dBm • 0x0C = -14.1 dBm • 0x0D = -16.5 dBm • 0x0E = -20.85 dBm • 0x0F = -40 dBm • Other values: invalid and ignored	0 dBm **0x 00**	*tx_advertising_power: <value>* *(unsigned/no unit)*

[8] To determine whether a device is Gen2 or Gen2.5, refer to Table 1-1.

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 5C	Advertisement Packet Format	1 B	• Bit 0: 0/1 = iBeacon advertising disabled/enabled • Bit 1: 0/1 = Eddystone UID advertising disabled/enabled • Bit 2: 0/1 = Eddystone TLM advertising disabled/enabled • Bits 3-7: RFU • All set to 0: invalid and ignored	iBeacon enabled Eddystone UID and TLM disabled **0x 01**	*advertising_packet_format: {* *ibeacon: <value>, (unsigned/no unit)* *eddystone_uid: <value>, (unsigned/no unit)* *eddystone_tlm: <value> (unsigned/no unit)* *}*
0x 5F	BLE MAC Address	6 B	• READ ONLY • This contains the 6-Byte MAC address that will be present in BLE advertisement packets	**Different for every device**	*mac_address: <value>, (unsigned/no unit)*

3.2.8.1 Advertising Enable/Disable

By default, BLE advertising is disabled and the FW is in Tracker mode. If Beacon mode is required instead, it is strongly recommended that register 0x 0A be used to switch configurations instead of changing numerous individual registers. See Section 3.2.1 for more information about this.

3.2.8.2 Advertising Interval

The periodicity of the advertising is defined by setting a value for the interval between successive packet transmissions. This is defined by register 0x 59.

The default value is 100 ms in order to match the 100 ms interval requirement for RTLS use-cases. If using a Beacon in a use -case other than RTLS, it is recommended that the interval be increased to save battery life.

NOTE: The advertising interval is independent from the application tick settings (Section 3.2.3). The ticks and BLE advertising intervals are independently configurable.

3.2.8.3 Tx Advertising Power

The BLE Tx advertising power is controlled with register 0x 5B and is set to 0 dBm by default. Other configurable options shall be -5 to 0 dBm in 1 dB increments, according to the supported power levels of the MCU.

3.2.8.4 Advertisement Packet Format

There are several widely-accepted, standard formats for BLE advertisements. The ones supported by the TEKTELIC Beacon include iBeacon (default), Eddystone UID, and Eddystone TLM. Register 0x 5C allows the user to select which type(s) of advertisement packet format is desired. Figure 3-2 shows the contents of each of these packet formats.

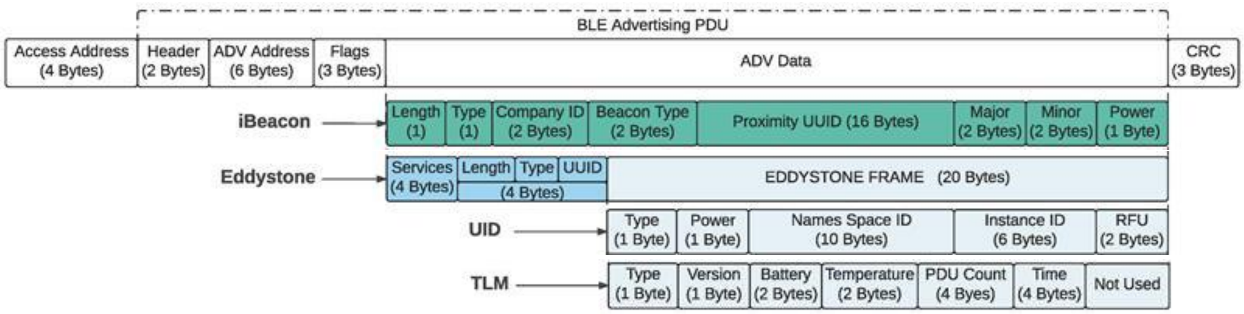


Figure 3-2: Supported Standard BLE Beacon Packet Formats

None of the frame contents are user-configurable. The important frames that relate directly to device operation are described as follows.

- **iBeacon <Proximity UUID>**: This is a 16-byte unique identifier. Each device has this field populated by its LoRaWAN DevEUI twice. For example, if the Beacon's DevEUI is 647FDA0000001234, then the iBeacon Proximity UUID would be 647FDA0000001234-647FDA0000001234.
- **Eddystone UID <Namespace ID**>: This is a 10-byte unique identifier. Each device has this field populated by its LoRaWAN DevEUI, followed by 2-bytes of 0s. For example, if the Beacon's DevEUI is 647FDA0000001234, then the Eddystone UID Namespace ID would be 647FDA00000012340000.
- **Eddystone TLM <Battery>:** This field contains battery report data. The type of data depends on the generation of the device[9]:
  - Gen2 devices report the battery voltage. This is the same number that gets reported OTA in a LoRaWAN battery report UL, except in this case it is 1 mV/LSB. For example, if the battery voltage was measured to be 3.590 V, then <Battery> would be 0x 0E 06.
  - Gen2.5 devices report the battery lifetime. This is the same number that gets reported OTA in a LoRaWAN battery report UL, in units of 1 day/LSB. For example, if the remaining battery lifetime is 1029 days, then <Battery> would be 0x 04 05.

[9] To determine whether a device is Gen2 or Gen2.5, refer to Table 1-1.

- **Eddystone TLM <Temperature>:** This is the temperature as measured by the device's MCU. This is the same number that gets reported OTA in a LoRaWAN UL, except represented in 8.8 fixed-point notation (standard for Eddystone encoding/decoding). For example, if the MCU temperature was measured to be 23.8°C, then <Temperature> would be 0x 17 CD.

- **Eddystone TLM <Advertisement Counter>:** This is a running counter that increments every time the device emits an advertisement frame (of any type). This counter is reset to 0 upon hard or soft device resets.

- **Eddystone TLM <Advertisement Counter>:** This is a counter that represents the time since the device is powered on or rebooted. This is represented in 0.1 s/LSb. This counter is reset to 0 upon hard or soft device resets.

The descriptions of the purposes of the other frames are unrelated to Beacon function and outside the scope of this document. For complete descriptions of the frames, refer to the BLE 5.0 Core Specification (Bluetooth SIG, 2016).

Since register 0x 5C allows for the user to configure more than 1 type of advertising packet format, the system accounts for this by interleaving the packets if more than one format is enabled.

Example: The mean advertising interval (registers 0x 59 and 0x 5A) is set such that advertisements will happen at a frequency at 10 Hz. All 3 advertisement packet formats are enabled on register 0x 5C. In this case, at 0.1 s, an iBeacon packet is broadcasted. At 0.2 s, an Eddystone UID packet is broadcasted. At 0.3 s, an Eddystone TLM packet is broadcasted. At 0.4 s, an iBeacon packet is broadcast. Etc.

3.2.8.5 BLE MAC Address

Any device capable of BLE has a unique 6-Byte MAC address identifier. This is the ID that is contained in the BLE advertisement packets and by which discovered devices are filtered.

Register 0x 5E contains the MAC address of that device and it has read-only access. This is the same MAC address that is printed on the device label.

When the device is placed into Beacon mode (described in Section 3.2.1), upon resetting, UL 0 will be sent immediately following the join process and the payload will contain the MAC address of the device. For example, if the device's MAC address is ABCDEF012345, the UL payload is **0x 5F AB CD EF 01 23 45** and is sent on **LoRaWAN port 100** as if in response to a DL request to read register 0x 5F.

3.2.8.6 BLE Advertisement (Beacon Mode) Configuration DL Examples

- Change the Beacon's BLE advertisement power to -3 dBm:

    - DL payload: 0x DB 03
- Change the median advertising interval to 5 s:
    - DL payload: 0x D9 11 94 DA 15 7C
- Read MAC address and change the advertising packet format to Eddystone TLM:
    - DL payload: 0x 5F DC 04

3.2.9 Command and Control

The *command and control* registers are used are used to save settings, restart the device OTA, read the application and LoRaMAC versions, and reset the configuration settings to default.

Table 3-15 shows the structure of the command and control Registers. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-15: Command & Control Registers

Address	Access	Name	Size	Description	JSON Variable (Type/Unit)
0x 70	W	Flash Write Command	2 B	• Bit 14: • 0/1 = Do not write/Write LoRaMAC Configuration • Bit 13: • 0/1 = Do not write/Write App Configuration • Bit 0: • 0/1 = Do not restart/Restart Sensor • Bits 1-12, 15: Ignored	*write_to_flash {* *app_config: <value>, (unsigned/no unit)* *lora_config: <value>, (unsigned/no unit)* *restart_sensor: <value> (unsigned/no unit)* *}*

Address	Access	Name	Size	Description	JSON Variable (Type/Unit)
0x 71	R	Metadata	7 B	• Bits 48-55: App version major • Bits 40-47: App version minor • Bits 32-39: App version revision • Bits 24-31: LoRaMAC version major • Bits 16-23: LoRaMAC version minor • Bits 8-15: LoRaMAC version revision • Bits 0-7: LoRaMAC region number (see Section 3.2.9.1)	*metadata {* *app_ver_major: <value>, (unsigned/no unit)* *app_ver_minor: <value>, (unsigned/no unit)* *app_ver_revision: <value>, (unsigned/no unit)* *loramac_ver_major: <value>, (unsigned/no unit)* *loramac_ver_minor: <value>, (unsigned/no unit)* *loramac_ver_revision: <value>, (unsigned/no unit)* *loramac_region: <value> (unsigned/no unit)* *}*
0x 72	W	Reset Configuration to Factory Defaults	1 B	• 0x 0A = Reset App Configuration • 0x B0 = Reset LoRaMAC Configuration • 0x BA = Reset both App and LoRaMAC Configurations • Any other value: Invalid and ignored	*config_factory_reset {* *app_config: <value>, (unsigned/no unit)* *loramac_config: <value> (unsigned/no unit)* *}*

Note: The command and control registers 0x 70 and 0x 72 are always executed after the full DL configuration message has been decoded. The reset command should always be sent as an unconfirmed DL message. Failure to do so may cause the device to continually reboot.

3.2.9.1 Flash Write Command

Configuration changes are not retained after a power cycle unless they are saved in the flash memory. The *Flash Write Command* register should be written to in order to save changes that have been written to other registers. This can be done in a separate DL at any time, or be included in the same payload as the other write commands.

Changes made to the LoRaMAC registers (0x 10 to 0x 15) must have bit 14 in the command set to 1 in order to be saved. Changes made to the application registers (0x 20 to 0x 63) must have bit 13 set to 1 in order to be saved. Both bits can be set to any combination of 1s and 0s.

The *Flash Write Command* register can also be used to reset the device and cause it to rejoin the network. This is done by setting bit 0 to 1. Immediately after receiving this command in a DL, the Sensor will reset. This means that if the command was sent in a confirmed DL, the confirmation reply UL will not be sent. The Sensor will rejoin the network but then get the command sent again, causing a loop of continual rebooting. It is important to not send the reset command as a confirmed DL.

3.2.9.2 Metadata

Bits 32 to 55 of the *Metadata* register contain the application revision numbers which define the FW version. The FW version is reported in the format as shown in Figure 3-3, which is shown using the example FW v1.0.15.

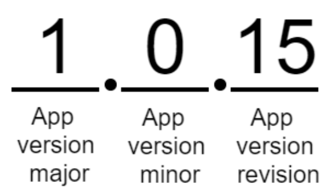


Figure 3-3: Example FW version format

Bits 8-31 in the *Metadata* register contain the LoRaMAC version numbers. The format is the same as shown in Figure 3-3. This number is not to be confused with the LoRaWAN specification version according to the LoRa Alliance standards. The LoRaMAC version number is the version of the LoRaMAC layer of the FW developed by TEKTELIC.

The LoRaMAC region number is the last byte of the *Metadata* register. Current LoRaMAC regions and corresponding region numbers for the Sensor are listed in Table 3-16.

Table 3-16: LoRaMAC Regions and Region Numbers for BLE Asset Tracker

LoRaMAC Region	Region Number
EU868	0
US915	1
AS923	2
AU915	3
IN865	4
KR920	6
RU864	7

3.2.9.3 Reset Configuration to Factory Defaults

The *Reset Configuration to Factory Defaults* register is written to in order to reset all of the other register values (0x 10 to 0x 63).

3.2.9.4 Command and Control Examples

- Write Application Configuration to flash
  - DL payload: 0x F0 20 00
- Write Application and LoRaMAC Configurations to flash
  - DL payload: 0x F0 60 00
- Reboot Device
  - DL payload: 0x F0 00 01
- Get FW version, and reset App Configuration to factory defaults
  - DL payload: 0x 71 F2 0A

Appendix – Default Register Value Summary

Name	Category	Register Address [Hex]	Default Value, Tracker Mode [Hex]	Default Value, Beacon Mode [Hex]
Tracker/Beacon Mode	Toggle Tracker/Beacon Mode	0A	00: Tracker Mode is Default	
Join Mode	LoRaMAC	10	80 00	80 00
LoRaMAC Options		11	00 0E	00 0E
LoRaMAC DR and Tx Power		12	04 00	00 00
LoRaMAC Rx2 Window		13	As per Table 3-6	
Seconds per Core Tick	Periodic Transmission	20	00 00 0E 10	00 00 0E 10
Ticks per Battery		21	00 18	00 18
Ticks per Accelerometer		24	00 00	00 00
Ticks per BLE		25	00 01	00 00 (disabled)
Ticks per MCU Temperature		28	00 00	00 00
Mode	Function Button	2A	00 08	00 01
Event Type I Configuration		2B	11	11
Event Type II Configuration		2C	01	01
Mode	Accelerometer	40	87	07
Sensitivity		41	22	22
Acceleration Event Threshold Count		42	00 01	00 01
Acceleration Event Threshold Period		43	00 0A	00 0A
Acceleration Event Threshold		44	07 D0	07 D0
Acceleration Event Grace Period		45	01 2C	01 2C
Acceleration Event Value to Tx		46	03	03
Battery Report Options	Battery Management	4A	06	06
Average Energy Trend Window		4B	0A	0A
Mode	BLE Scanning (Tracker Mode)	50	88	88 (Not used)
Scan Duration		51	01 03	01 03 (Not used)
Scan Interval		52	00 1E	00 1E (Not used)
Scan Window		53	00 1E	00 1E (Not used)

Name	Category	Register Address [Hex]	Default Value, Tracker Mode [Hex]	Default Value, Beacon Mode [Hex]
Filter Range 0		54	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 (Not used)
Filter Range 1		55	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 (Not used)
Filter Range 2		56	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 (Not used)
Filter Range 3		57	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 (Not used)
Advertising Enable / Disable	BLE Advertising (Beacon Mode)	58	00	01
Advertising Interval		59	00 64 (Not used)	00 64
Tx Advertising Power		5B	00 (Not used)	00
Advertisement Packet Format		5C	01 (Not used)	01
BLE MAC Address		5F	Unique for every device	

References

Bluetooth SIG. (2016, Dec 6). *Core Specification 5.0.* Retrieved November 15, 2021, from https://www.bluetooth.com/specifications/specs/core-specification-5/

LoRa Alliance. (Feb 2017). *LoRaWAN Regional Parameters.* ver. 1.0.2, rev. B.