



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

Enterprise Asset Tracker

Technical Reference Manual

Document Type: Technical Reference Manual

Document Number: T0006940_EAT_TRM

Document Version: 1.0

Product Name: Enterprise Asset Tracker

Product Variants:

Enterprise Asset Trackers (Indoor)	T0007379 – Base T0007128 – Wall-Mount
Enterprise Asset Trackers (C-Cell)	T0007377 – Base T0006909 – Wall-Mount
Enterprise Asset Tracker (ATEX)	T0007367 – Wall-Mount

FW Versions v1.0.15, v1.0.16, and later

Release Date: June 30, 2021

PROPRIETARY:

The information contained in this document is the property of TEKTELIC Communications Inc. Except as specifically authorized in writing by TEKTELIC, the holder of this document shall keep all information contained herein confidential, and shall protect the same in whole or in part from disclosure to all third parties.

© 2021 TEKTELIC Communications Inc., all rights reserved.

All products, names, and services are trademarks and registered trademarks of their respective companies.

DISCLAIMER:

Material contained in this document is subject to change without notice. The material herein is solely for information purposes and does not represent a commitment by TEKTELIC or its representatives. TEKTELIC has prepared the information contained in this document solely for use by its employees, agents, and customers. Dissemination of this information and/or concepts to other parties is prohibited without the prior written consent of TEKTELIC. In no event will TEKTELIC be liable for any incidental or consequential damage in connection with the furnishing, performance or use of this material.

TEKTELIC reserves the right to revise this publication in accordance with formal change control procedures defined by TEKTELIC.

Revision History

Version	Date	Editor	Comments
0.1	June 14, 2021	Carter Mudryk	<ul style="list-style-type: none">• Initial release based on BLE Tracker TRM T0005946_TRM_v0.11.• Includes new averaging mode.
1.0	June 30, 2021	Carter Mudryk	<ul style="list-style-type: none">• Added missing register description sections.• Updated description of filtering packet encoding and decoding for clarity.• Added missing product codes• 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.

Table of Contents

Revision History.....	2
Table of Contents	3
List of Tables.....	4
List of Figures.....	5
List of Acronyms	6
1 Overview.....	7
1.1 BLE Operation.....	9
1.2 Function Button Operation	11
1.3 Magnetic Reed Switch Operation.....	11
1.4 LED Behaviour	14
1.4.1 Power-On and Network Join Operation	14
1.4.2 Normal Operation	15
1.4.3 DEEP SLEEP	15
1.5 Other Transducer Operation	16
2 UL Payload Formats.....	17
2.1 Frame Payload to Report Non-BLE Sensing Data	17
2.2 Reporting Discovered BLE Devices	18
2.3 Response to Configuration and Control Commands.....	21
3 DL Payload Formats.....	22
3.1 Putting Tracker to Sleep OTA	22
3.2 Configuration and Control Commands.....	22
3.2.1 LoRaMAC Configuration	23
3.2.2 Periodic Tx Configuration	25
3.2.3 Function Button Configuration.....	27
3.2.4 Accelerometer Configuration.....	29
3.2.5 BLE Configuration	33
3.2.6 Temperature Threshold Configuration	39
3.2.7 Command and Control	41
4 Appendix A: Default Values	45
References.....	46

List of Tables

Table 1-1: Enterprise Asset Tracker HW Models	8
Table 1-3: Enterprise Asset Tracker Information Streams	8
Table 1-4: Enterprise Asset Tracker Default Reporting Behavior	9
Table 2-1: UL Information Streams	17
Table 2-2: UL Frame Payload Values for Present Sensor Data.....	18
Table 3-1: DL Information Stream	22
Table 3-2: LoRaMAC Configuration Registers	24
Table 3-3: Default Maximum Tx Power in Different Regions.....	25
Table 3-4: Default Values of Rx2 Channel Frequency and DR Number in Different Regions	25
Table 3-5: Periodic Transmission Configuration Registers.....	26
Table 3-6: Function Button Configuration Registers	28
Table 3-7: Accelerometer Configuration Registers	30
Table 3-8: Typical Current Draw Deltas at 3.6 V for Different Accelerometer Sample Rates.....	32
Table 3-9: BLE Configuration Registers	34
Table 3-10: Maximum Number of Reported BLE Devices per LoRaWAN Packet in Different Regions	38
Table 3-11: Temperature Threshold Configuration Registers.....	40
Table 3-12: Command & Control Registers.....	41
Table 3-13: LoRaMAC Regions and Region Numbers for BLE Asset Tracker	43
Table 4-1: Default Values of Configuration Registers	45

List of Figures

Figure 1-1: The BLE scan scheme.	10
Figure 1-2: The Tracker magnetic reset/wake-up pattern.....	12
Figure 1-3: The Tracker magnetic UL-triggering pattern.	13
Figure 1-4: The Tracker magnetic reed switch operation state diagram.....	13
Figure 1-5: The LED pattern for GREEN or RED after the POST.	14
Figure 1-6: The GREEN pattern during join.	15
Figure 1-7: The LED pattern with GREEN after the magnetic wake-up/reset pattern is observed.	15
Figure 1-8: The LED pattern with RED before entering DEEP SLEEP.....	16
Figure 2-1: The frame format in the UL payload.....	17
Figure 2-2: The UL payload format to report discovered BLE devices.	20
Figure 3-1: The format of a DL configuration and control message block.....	23
Figure 3-2: Example FW version format.....	43

List of Acronyms

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

1 Overview

This document contains the technical information about the supported functionality of the TEKTELIC *Enterprise Asset Tracker* sensor device variants, referred to as *Trackers* henceforth. 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 Tracker is a BLE-capable LoRaWAN IoT sensor run on a single LTC battery and packed into a compact polycarbonate casing. The Tracker'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 Tracker 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 Tracker orientation is of interest.

Additional sensing functions on the Tracker include on-board temperature and the battery voltage. The battery lifetime of the Tracker has been estimated to be up to 6 years for AA-cell variants and 10 years for C-cell variants with the default configuration settings.

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 reed switch which is used to wake the device from the DEEP SLEEP (used for shipping) and to force ULs when the device is activated.

Table 1-1 presents the currently available Tracker HW variants.

Table 1-1: Enterprise Asset Tracker HW Models

Product Code, Module-Level T-Code	Product Code, PCBA-Level T-Code	Description	Battery/Enclosure Size
T0007379	T0006983	Enterprise Asset Tracker (Indoor), Base	AA-cell
T0007128		Enterprise Asset Tracker (Indoor), Wall-Mount	AA-cell
T0007377	T0006984	Enterprise Asset Tracker (Outdoor), Base	C-cell
T0006909		Enterprise Asset Tracker (Outdoor), Wall-Mount	C-cell
T0007367	T0007465	Enterprise Asset Tracker (ATEX), Wall-Mount	C-cell

IMPORTANT: The information in this manual is applicable to devices programmed with FW version 1.0.15 or 1.0.16. For newer FW versions, ensure that there is not a more recent TRM available. For older FW versions, please refer to the appropriate TRM version.

The information streams supported by the SW have been shown in Table 1-2.

Table 1-2: 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

The default configuration on the Tracker for reporting transducer readings has been shown in Table 1-3.

Table 1-3: Enterprise Asset Tracker Default Reporting Behavior

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

In Sections 2 and 3, the UL and DL payload formats are explained, respectively. There is a webapp encoding/decoding tool available at sensor-config.tektelic-dev.com that allows for easy DL command formation and UL reading [1]. This tool can be used to change Tracker configuration settings and decode ULs as desired.

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

1.1 BLE Operation

The BLE function of the Tracker 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 an UL.

The BLE scan can be disabled entirely or enabled at any time. 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.

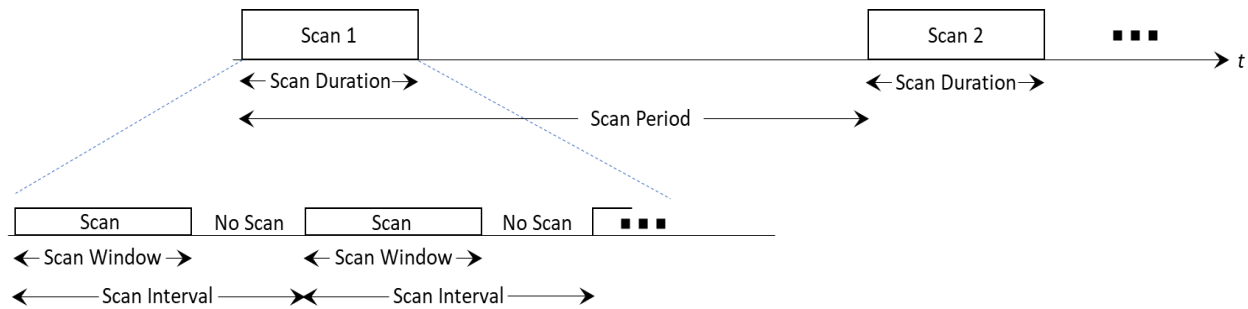


Figure 1-1: The BLE scan scheme.

The scan period, duration, interval, and window are all configurable (see Sections 3.2.2 and 3.2.5).

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 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 a user-configurable value; see Section 3.2.5).
- 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 UL 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 list 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 the battery life. In any 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.

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

1.2 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 Tracker. This also lets the Tracker as a LoRaWAN Class-A end device 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 Tracker sends an uplink. The uplink can be empty, or it can contain the battery voltage, acceleration vector, temperature, or BLE scan report, or any combination of these. Section 3.2.3 describes the configuration for the function button.

1.3 Magnetic Reed Switch Operation

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

The reed 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 Tracker from DEEP SLEEP and having it try to join the network. When the Tracker comes out of 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 Tracker during normal operation, causing it to try to rejoin the network.

The magnetic pattern in this application is illustrated in Figure 1-2. 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 1-2 shows that the pattern involves sustaining a “magnet presence” continuously for at least 3 sec but less than 10 sec.

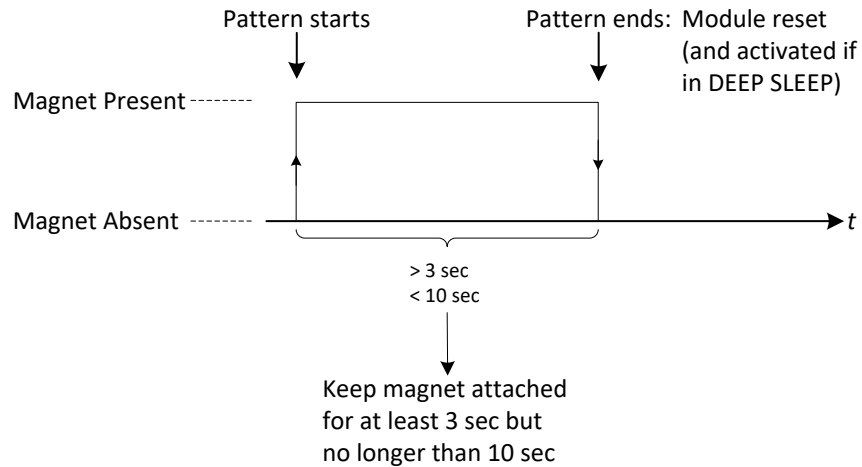


Figure 1-2: The Tracker magnetic reset/wake-up pattern.

When the specified magnetic pattern is applied to the Tracker, the Tracker displays an LED indication, described in Section 1.4.3, to show that it has accepted the magnetic pattern. In all cases, the magnet pattern causes the Tracker to reset. If the Tracker was in DEEP SLEEP when the pattern was applied, after resetting it will wake up and begin trying to join the network. If the Tracker was in normal operation when the magnet pattern was applied, after resetting it will try to rejoin the network.

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

See

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 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 1-3. *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.*

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

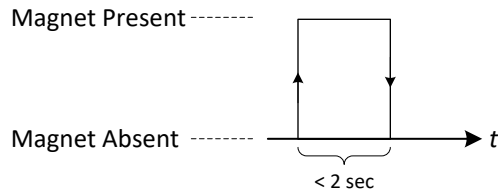


Figure 1-3: The Tracker magnetic UL-triggering pattern.

The state diagram for the magnetic reed switch operation is shown in Figure 1-4, wherein the following definitions apply.

- *Mothball Flag*: a bit saved to flash that is either TRUE or FALSE. When equal to TRUE, the Tracker will enter DEEP SLEEP upon resetting. When equal to FALSE, the Tracker will begin the NS join procedure.
- *Indoor T-Code*: The Tracker’s PCBA-level T-Code as listed in Table 1-1. Only Trackers with indoor device T-Codes are able to be put in DEEP SLEEP.
- *Mothball Sleep*: DEEP SLEEP.

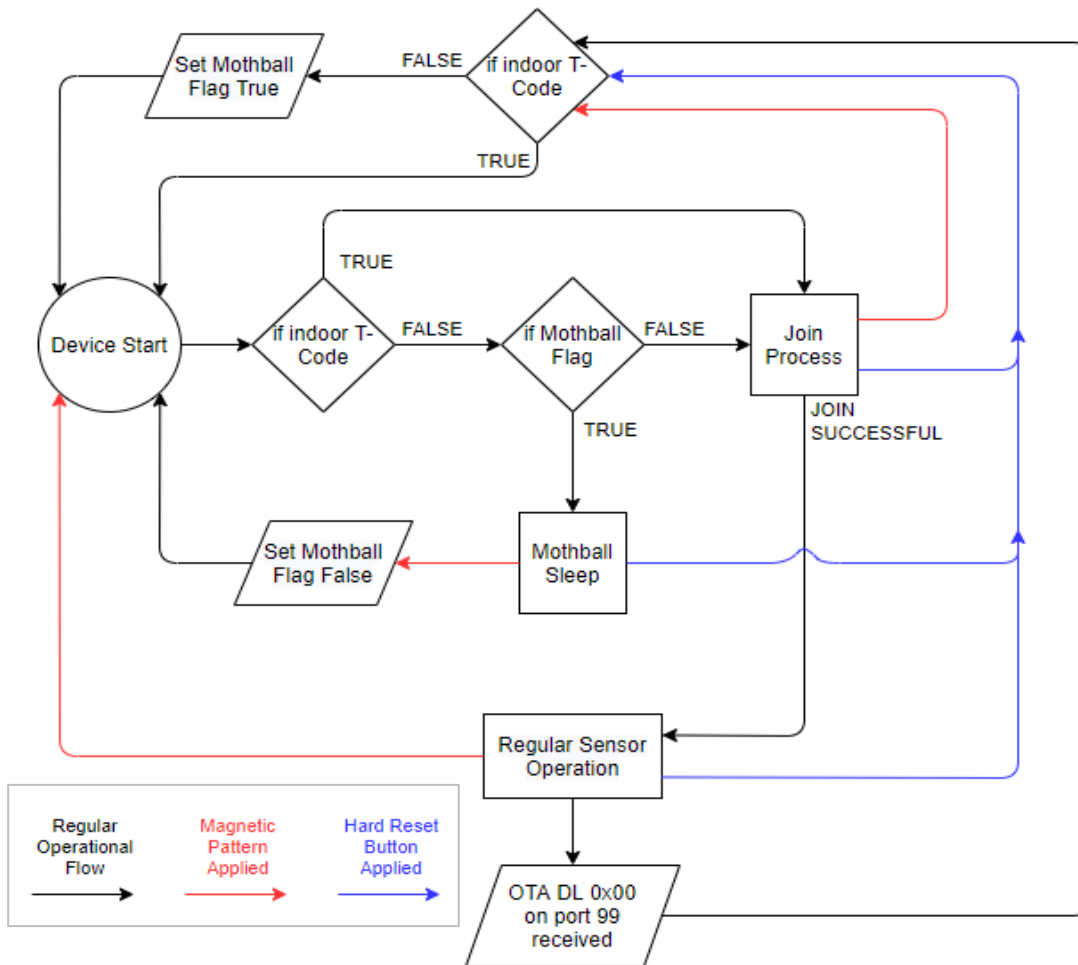


Figure 1-4: The Tracker magnetic reed switch operation state diagram.

1.4 LED Behaviour

The Tracker 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.

1.4.1 Power-On and Network Join Operation

The LED behavior in the Tracker is not user configurable, and is specified as follows:

1. Both **GREEN** and **RED** are turned off when the Tracker 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 1-5.
 - b. If the POST fails, **RED** is toggled ON and OFF every 50 ms for 0.5 sec, as shown in Figure 1-5.

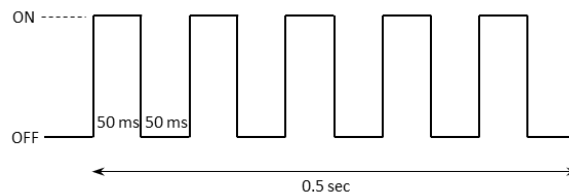


Figure 1-5: The LED pattern for **GREEN** or **RED** after the POST.

4. Both **GREEN** and **RED** are turned off when the POST and the subsequent LED flashing specified in item 3 end.
5. While the Tracker is attempting to join:
 - a. **GREEN** is toggled ON and OFF every 50 ms for the first hour. But after that, it just flashes twice (ON time: 50 ms, OFF time: 50 ms) every 5 sec. This scheme has been shown in Figure 1-6.
 - b. **RED** flashes just once:
 - i. with a pulse duration of 25 ms right after transmitting a JOIN REQUEST.
 - ii. with a pulse duration of 100 ms right after receiving a JOIN ACCEPT.

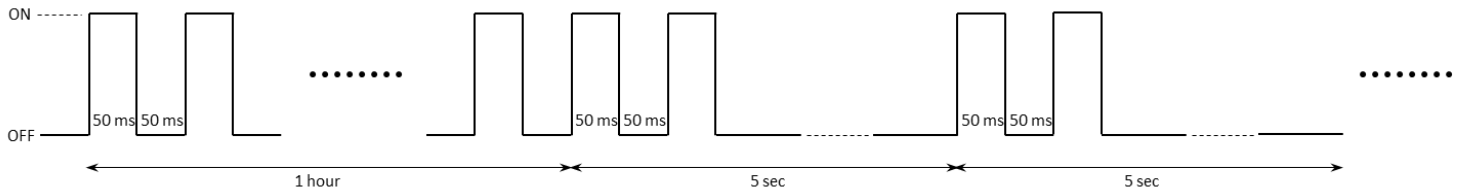


Figure 1-6: The GREEN pattern during join.

1.4.2 Normal Operation

After the Tracker 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.

1.4.3 DEEP SLEEP

The Tracker 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 5 sec after the pattern is applied:

1. The POST LED pattern described in steps 1-3 in Section 1.4.1 above occurs after the device resets.
2. **GREEN** is toggled ON and OFF every 0.5 sec for 3 sec as shown in Figure 1-7.

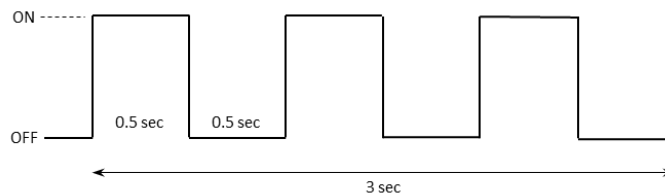


Figure 1-7: The LED pattern with GREEN after the magnetic wake-up/reset pattern is observed.

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

1. The POST LED pattern described in steps 1-3 in Section 1.4.1 above occurs after the device resets.
2. **RED** is toggled ON and OFF every 0.1 sec for 0.6 sec as shown in Figure 1-8.

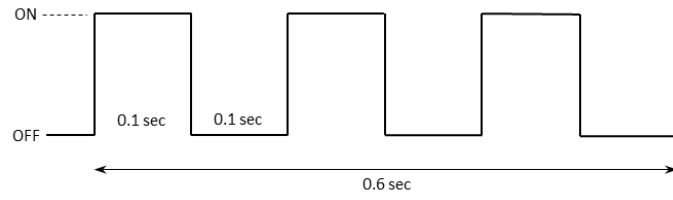


Figure 1-8: The LED pattern with RED before entering DEEP SLEEP.

1.5 Other Transducer Operation

The battery gauge, accelerometer, and MCU thermometer functional operations are described in the corresponding subsections in Section 3.

2 UL Payload Formats

The UL information streams (from the Tracker to the NS) supported by the SW are shown in Table 2-1, and are explained in Sections 2.1 to 2.3. Refer to [1] for a comprehensive tool to decode Tracker UL frame payloads.

Table 2-1: UL Information Streams

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

2.1 Frame Payload to Report Non-BLE Sensing Data

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

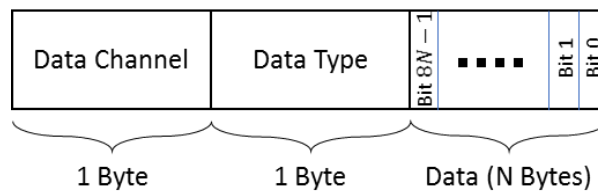


Figure 2-1: The frame format in the UL payload.

A Tracker message payload can include multiple data frames from different sensing entities in the Tracker. Frames can be arranged in any order. The Tracker frame payload values for present sensor 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 Status	0x00	0xBA	1 B	Unsigned	<ul style="list-style-type: none"> Bits 0-6: Battery voltage minus 2.5 V (10 mV/LSb) Bit 7: Not used 	<pre>battery_status { life: <value>, (unsigned/V) eos_alert: <value> (unsigned/no unit) }</pre>
Acceleration Alarm Status	0x00	0x00	1 B	Digital Input	<ul style="list-style-type: none"> 0x00 = Alarm inactive (motion no longer detected) 0xFF = Alarm active (motion detected) 	<pre>acceleration_alarm (unsigned/no unit)</pre>
Acceleration Vector	0x00	0x71	6 B	Acceleration	<ul style="list-style-type: none"> 1 milli-g/LSb (signed) Bits 32-47: X-axis acceleration Bits 16-31: Y-axis acceleration Bits 0-15: Z-axis acceleration 	<pre>acceleration_vector { xaxis: <value>, (signed/g) yaxis: <value>, (signed/g) zaxis: <value> (signed/g) }</pre>
MCU Temperature	0x00	0x67	2 B	Temperature	<ul style="list-style-type: none"> 0.1°C/LSb (signed) 	<pre>temperature (signed/°C)</pre>

Examples:

- Uplink 0x 00 BA 70 is reported to indicate the battery voltage is 3.62 V.
- 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, see Section 1.1.

Discovered BLE devices are reported on **LoRWAN 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 0x0A. 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.

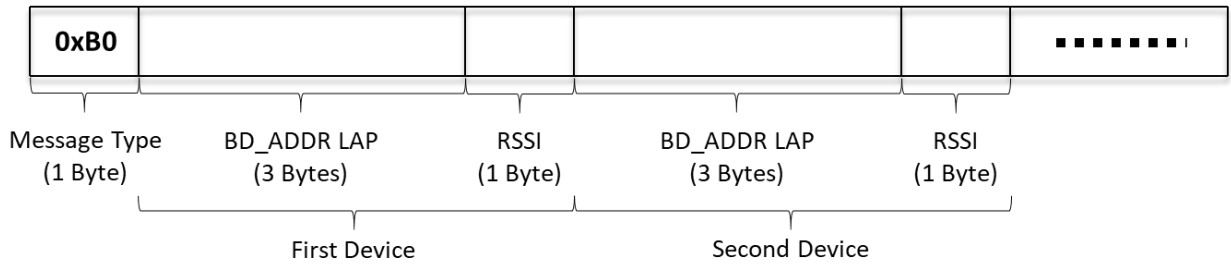
With *filtered* reporting (Figure 2-2-(b)–(e)), up to 4 ranges of BD_ADDR can be defined for filtering discovered devices (see Section 3.2.5). The message type headers 0xB0, 0xB1, 0xB2, 0xB3 correspond to Range 0 to 3. A BD_ADDR consists of an *Organizationally-Unique Identifier* (OUI) comprising the 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. In fact, 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 AC233F (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 AC233F: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 AC233F000000 to AC233FFFFFFF, inclusive.

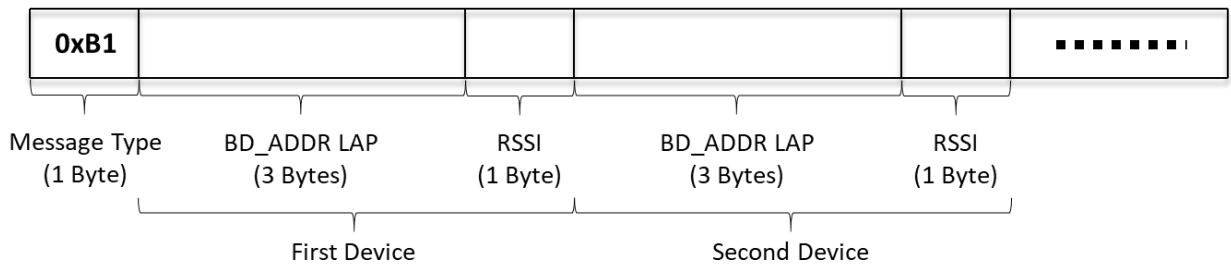
Zero, one, or more devices can be reported in a single message, depending on the number of devices available to report and payload size limitation as determined by LoRaWAN regional parameters [2]. See Section 1.1 for how many and what devices are chosen to get reported OTA. The case of zero devices (an empty BLE device list) happens when no devices are discovered, or when the BLE is disabled (see Section 3.2.5) but a BLE report is due. 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.



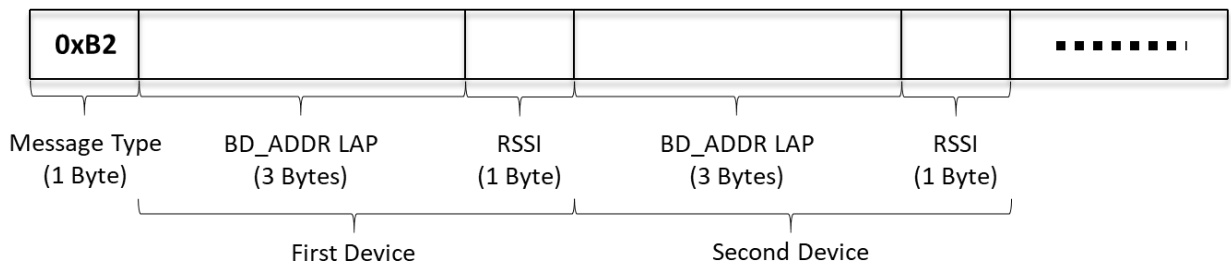
(a) Basic mode.



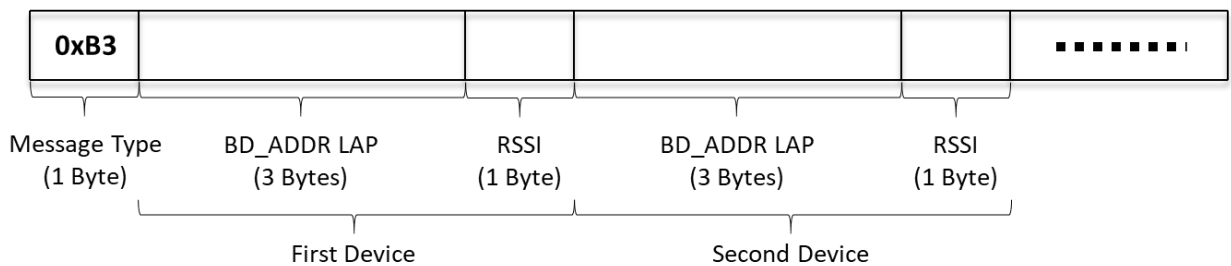
(b) Whitelisting mode, Range 0.



(c) Whitelisting mode, Range 1.



(d) Whitelisting mode, Range 2.



(e) Whitelisting mode, Range 3.

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

2.3 Response to Configuration and Control Commands

Tracker responses to DL configuration and control commands (which are sent on LoRaWAN port 100; see Section 3.1) are sent in the UL on **LoRaWAN port 100**. These responses include,

- returning the value of a configuration register in response to an inquiry from the NS; and
- writing to a configuration register.

In the former case, the Tracker 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 Tracker responds with a 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 Tracker response in the former case).

3 DL Payload Formats

The DL stream (from the NS to the Tracker) supported by the SW is shown in Table 3-1, and is explained in Section 3.1. Refer to [1] for a comprehensive tool to encode DL messages into DL frame payloads.

Table 3-1: DL Information Stream

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

3.1 Putting Tracker to Sleep OTA

For information about the alternative method for putting the Tracker into DEEP SLEEP (i.e. with a magnet), refer to Section 1.3..

The Trackers ship in a state of DEEP SLEEP to conserve power. Once activated by the magnetic wakeup pattern (see Section 1.3), a Tracker 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 be **0x 00**.

Upon receiving this DL, the *Mothball Flag* will be set to TRUE and the Tracker will reset then enter DEEP SLEEP mode as shown in Figure 1-4.

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

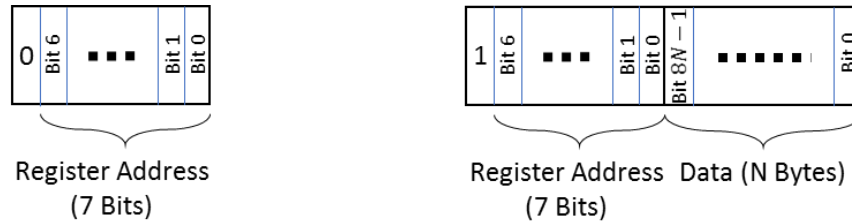
3.2 Configuration and Control Commands

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.

The Register Address is used to access various configuration parameters. These addresses are bound between 0x00 and 0x7F.

Bit 7 of the first byte 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 format of a DL configuration and control message block.

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

Examples:

- Read Registers 0x2A, 0x2B, and 0x2C:
 - DL command: { 0x 2A 2B 2C }
- Read Register 0x20 and Write value 0x8000 to Register 0x10:
 - DL command: { 0x 20 90 80 00 }

When a write command is sent to the Tracker, the Tracker 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 2.2).

DL configuration and control commands fall into one of the following categories and are discussed in Sections 3.2.1–3.2.7:

- LoRaMAC Configuration
- Periodic Tx Configuration
- Function Button Configuration
- Accelerometer Configuration
- BLE Configuration
- Temperature Threshold Configuration
- Command and Control

3.2.1 LoRaMAC Configuration

LoRaMAC options can be configured using the LoRaWAN DL. These configuration options change the default MAC configuration that the Tracker loads on start-up. They can also change certain run-time parameters. Table 3-2 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 3-2: LoRaMAC Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x10	Join Mode	2 B	<ul style="list-style-type: none"> Bit 15: 0/1 = ABP/OTAA mode Bits 0-14: Ignored 	OTAA mode 0x 80 00	<i>loramac_join_mode:</i> <value> (unsigned/no unit)
0x11	Options	2 B	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Unconfirmed UL Public Sync Word Duty cycle enabled¹ ADR enabled 0x 00 0E	<i>loramac_opts</i> { <i>confirm_mode:</i> <value>, (unsigned/no unit) <i>sync_word:</i> <value>, (unsigned/no unit) <i>duty_cycle:</i> <value>, (unsigned/no unit) <i>adr:</i> <value> (unsigned/no unit) }
0x12	DR and Tx Power ²	2 B	<ul style="list-style-type: none"> Bits 8-11: Default DR number Bits 0-3: Default Tx power number Bits 4-7, 12-15: Ignored 	<ul style="list-style-type: none"> DRO Tx Power 0 (max power; see Table 3-3) 0x 00 00	<i>loramac_dr_tx</i> { <i>dr_number:</i> <value>, (unsigned/no unit) <i>tx_power_number:</i> <value> (unsigned/no unit) }
0x13	Rx2 Window	5 B	<ul style="list-style-type: none"> Bits 8-39: Channel frequency in Hz for Rx2 Bits 0-7: DR for Rx2 	As per Table 3-4	<i>loramac_rx2</i> { <i>frequency:</i> <value>, (unsigned/Hz) <i>dr_number:</i> <value> (unsigned/no unit) }

Note: Modifying these values only changes them in the Tracker. Options for the Tracker in the NS also need to be changed in order to not strand a Tracker. Modifying configuration parameters in the NS is outside the scope of this document.

¹ 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.

² 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].

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

RF Region	Max Tx EIRP [dBm]
EU868	16
US915	30
AS923	16
AU915	30
IN865	30
KR920	14
dRU864	16

Table 3-4: 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

3.2.1.1 LoRaMAC Configuration DL Examples

- Switch Device 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

3.2.2 Periodic Tx Configuration

All periodic reporting is synchronized around ticks. The *core tick* is simply a user-configurable time base unit that is used to schedule Tracker measurements. For each sensor in the Tracker, the number of elapsed ticks before transmitting can be defined. Table 3-5 shows a list of registers used to configure the Tracker periodic transmissions. All the registers have R/W access.

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

$$\langle \text{Transducer} \rangle \text{ Reporting Period} = \text{Seconds per Core Tick} \times \text{Ticks per } \langle \text{Transducer} \rangle$$

where $\langle \text{Transducer} \rangle$ can be “Battery”, “Accelerometer”, “BLE”, or “MCU Temperature”, as shown in Table 3-5. If $\langle \text{Transducer} \rangle$ Reporting Period equals 0, it means that the $\langle \text{Transducer} \rangle$ 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 3-5: Periodic Transmission Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x20	Seconds per Core Tick	4 B	<ul style="list-style-type: none"> • 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	<i>seconds_per_core_tick:</i> <value> (unsigned/sec)
0x21	Ticks per Battery	2 B	<ul style="list-style-type: none"> • Ticks between battery reports • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic battery reports 	24 ticks = 1-day period 0x 00 18	<i>ticks_per_battery:</i> <value> (unsigned/no unit)
0x24	Ticks per Accelerometer	2 B	<ul style="list-style-type: none"> • Ticks between accelerometer reports • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic accelerometer reports 	Periodic reporting disabled 0x 00 00	<i>ticks_per_accelerometer:</i> <value> (unsigned/no unit)
0x25	Ticks per BLE	2 B	<ul style="list-style-type: none"> • Ticks between BLE reports • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic BLE reports 	1 tick = 1-hour period 0x 00 01	<i>ticks_per_ble:</i> <value> (unsigned/no unit)
0x28	Ticks per MCU Temperature	2 B	<ul style="list-style-type: none"> • Ticks between temperature reports • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic temperature reports 	Periodic reporting disabled 0x 00 00	<i>ticks_per_mcu_temperature:</i> <value> (unsigned/no unit)

3.2.2.1 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 Temperature set to 1

NOTE: As the function button operation (see Sections 1.2 and 3.2.3) and the magnetic reed switch operation (see Section 1.3) cannot be disabled it is impossible to brick the Tracker with a bad configuration; i.e. it is always possible to trigger the Tracker to uplink something so it can receive DL commands for a desired configuration change. However, there are use cases in which using the function button to trigger the Tracker may not be a convenient option, e.g. due to special mounting location or due to many Trackers to trigger). In these use cases, due care should be taken not to disable all periodic reporting (e.g. by setting register 0x20 to 0) or disable some periodic reporting and make the other periods too long.

3.2.3 Function Button Configuration

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

The operation of the function button has been described in Section 1.2. It is configurable how a function button event can be registered such that the Tracker 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 Tracker is triggered to send an uplink when the enabled event type is registered.

Table 3-6 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 3-6: Function Button Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x2A	Mode	2 B	<ul style="list-style-type: none"> • Bit 15: 0/1 = Event Type I/II enabled • Bit 0: 0/1 = Battery voltage 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 • Bits 4-14: Ignored 	<ul style="list-style-type: none"> • Event Type I enabled • BLE report enabled 0x 80 08	<pre>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) }</pre>
0x2B	Event Type I Configuration	1 B	<ul style="list-style-type: none"> • 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 	<ul style="list-style-type: none"> • $m = 1$ • $n = 1$ 0x 11	<pre>fb_event_type1 { m_value: <value>, (unsigned/no unit) n_value: <value> (unsigned/no unit) }</pre>
0x2C	Event Type II Configuration	1 B	<ul style="list-style-type: none"> • Bits 0-3: Value of T $1 \leq T \leq 15$ $T = 0$: Invalid and ignored • Bits 4-7: Ignored 	<ul style="list-style-type: none"> $T = 1$ 0x 01	<pre>fb_event_type2 { t_value: <value> (unsigned/no unit) }</pre>

3.2.3.1 Mode

Bits 0 and 1 of byte B₀ of the *Mode* register determines which event type is desired. Bits 0-3 of byte B₁ of the *Mode* register determines what the Tracker transmits when an event is registered.³

³ The battery voltage, acceleration vector, temperature, and discovered BLE devices are all reported using their usual uplink formats explained in Section 2.

If all bits 0-3 are 0, an empty uplink is transmitted on port 10 upon registration of a function button event.

3.2.3.2 Event Type I/II Configuration

Registers 0x2B and 0x2C 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.

3.2.3.3 Function Button Configuration DL Examples

- Enable BLE report and battery voltage report (with event type I):
 - DL payload: 0x AA 80 09
- Read current value of *Event Type II Configuration*:
 - DL payload: 0x 2C

3.2.4 Accelerometer Configuration

The accelerometer in the Tracker can be disabled or enabled and supports both periodic-based and event-based reporting.

In the case of the periodic-based reporting, only the acceleration vector (X-axis, Y-axis, Z-axis) is reported, with the period configured via registers 0x20 and 0x24 (see Section 3.2.2). Registers 0x20 and 0x24 can also be used to disable the periodic reporting. Also, in general, the accelerometer can provide raw (unfiltered) X-axis, Y-axis, and Z-axis values, or filtered values by removing the DC components. In the periodic reporting, the raw acceleration vector is always reported.

In the case of event-based reporting, an acceleration event is registered whenever the absolute of filtered (DC-removed) value of any axis exceeds a configurable threshold (register 0x44) for a configurable number of times (register 0x42) within a configurable period (register 0x43). For example, with the accelerometer enabled and otherwise default configuration settings, as soon as the acceleration magnitude on any axis is measured to be greater than 2 g (register 0x44, *Acceleration Event Threshold*) one time (register 0x42, *Acceleration Event Threshold Count*) in less than 10 seconds (register 0x43, *Acceleration Event Threshold Period*), an alarm UL report is sent.

An acceleration alarm is considered “cleared” as soon as no acceleration event is registered for at least a configurable *grace period* (register 0x45). No additional acceleration alarms are sent before an acceleration clear is registered. For example, for a grace period of 5 minutes, after an acceleration event is registered, as soon as 5 minutes without another acceleration event elapses

(i.e. no above-threshold motion for 5 minutes), an acceleration clear is registered. Until that time, no additional acceleration alarm ULs are sent.

Whenever an acceleration alarm 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 will be reported as soon as an “acceleration alarm clear” is registered, as explained in the previous paragraph. No additional acceleration alarms are reported before the first acceleration alarm clear is registered.

Another option of what to report after an acceleration alarm is registered is a BLE scan report. In this case, when an acceleration alarm is registered, a BLE scan with duration as determined by the MSB of register 0x51 is performed and the result is reported just like a regular BLE report.

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).

Table 3-7 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 3-7: Accelerometer Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x40	Mode	1 B	<ul style="list-style-type: none"> • 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 	<ul style="list-style-type: none"> • X-axis enabled • Y-axis enabled • Z-axis enabled • Accelerometer off <p>0x 07</p>	<pre> 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) } </pre>

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x41	Sensitivity	1 B	<ul style="list-style-type: none"> 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 Range⁴): 0/1/2/3 = $\pm 2/\pm 4/\pm 8/\pm 16$ g Bits 3, 6, 7: Ignored 	<ul style="list-style-type: none"> Sample Rate 10 Hz Measurement Range ± 8 g 0x 22	<pre>accelerometer_sensitivity { sample_rate: <value>, (unsigned/Hz) measurement_range: <value> (unsigned/g) }</pre>
0x42	Acceleration Event Threshold Count	2 B	<ul style="list-style-type: none"> Number of acceleration events before an acceleration alarm is registered Acceptable values: ≥ 1 0: Invalid and ignored 	1 event 0x 00 01	<pre>acceleration_event_thres hold_count: <value> (unsigned/no unit)</pre>
0x43	Acceleration Event Threshold Period	2 B	<ul style="list-style-type: none"> 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	<pre>acceleration_event_thres hold_period: <value> (unsigned/sec)</pre>
0x44	Acceleration Event Threshold	2 B	<ul style="list-style-type: none"> Unsigned, 1 milli-g/LSb 	2 g 0x 07 D0	<pre>acceleration_event_thres hold: <value> (unsigned/g)</pre>
0x45	Acceleration Event Grace Period	2 B	<ul style="list-style-type: none"> 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	<pre>acceleration_event_grace _period: <value> (unsigned/sec)</pre>
0x46	Acceleration Event Value to Tx	1 B	<ul style="list-style-type: none"> Bit 0: 0/1 = Acceleration alarm UL report disabled/enabled Bit 1: 0/1 = BLE scan report disabled/enabled Bits 2-7: Ignored 	<ul style="list-style-type: none"> Acceleration alarm report UL enabled BLE scan report enabled 0x 03	<pre>acceleration_event_tx { acceleration_alarm: <value>, (unsigned/no unit) ble: <value> (unsigned/no unit) }</pre>

⁴ Measurement ranges ± 2 g, ± 4 g, ± 8 g, ± 16 g correspond to typical transducer output precisions of 16 mg, 32 mg, 64 mg, 192 mg, 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.

3.2.4.1 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.

3.2.4.2 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 3-8 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 3-8: 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 [μ A]	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 $\pm 2 g$ implies a precision of $4/256 g/LSb$.

3.2.4.3 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.

3.2.4.4 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.

3.2.4.5 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.

3.2.4.6 Acceleration Event Grace Period

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

3.2.4.7 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 0x51; see Section 3.2.5) is performed and the result is reported.

3.2.4.8 Accelerometer Configuration DL Examples

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

3.2.5 BLE 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 how BLE operates in the BLE Asset Tracker. Table 3-9 shows the list of BLE 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: BLE Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x50	Mode	1 B	<ul style="list-style-type: none"> Bits 0-6: N Number of reported devices (1–127) 0: Disables BLE Bit 7: A 0/1 = Averaging mode off/on 	<ul style="list-style-type: none"> Up to 8 reported devices. Averaging mode on. <p>0x 88</p>	<pre>ble_mode: { num_reported_devices : <value>, (unsigned/no unit) averaging_mode: <value> (unsigned/no unit) }</pre>
0x51	Scan Duration	2 B	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> 3 seconds for periodic scans 1 second for event-based scans <p>0x 01 03</p>	<pre>ble_scan_duration: { periodic: <value>, (unsigned/sec) event_based: <value> (unsigned/sec) }</pre>
0x52	Scan Interval	2 B	<ul style="list-style-type: none"> Scan interval (1 ms/LSb) Acceptable values: 3, ..., 10000 Other values: Invalid and ignored 	30 ms 0x 00 1E	<pre>ble_scan_interval: <value> (unsigned/sec)</pre>
0x53	Scan Window	2 B	<ul style="list-style-type: none"> Scan window (1 ms/LSb) Acceptable values: 3, ..., “Scan Interval” Other values: Invalid and ignored 	30 ms 0x 00 1E	<pre>ble_scan_window: <value> (unsigned/sec)</pre>
0x54	BD_ADDR Filter Range 0	9 B	<ul style="list-style-type: none"> Range 0 for filtered BD_ADDRs B₀:B₁:B₂:B₃:B₄:B₅ to B₀:B₁:B₂:B₆:B₇:B₈ OUI = B₀:B₁:B₂ LAP_{start} = B₃:B₄:B₅ LAP_{end} = B₆:B₇:B₈ 	Range inactive 0x 00 00 00 00 00 00 00 00 00	<pre>bd_addr_range_0: { oui: <value>, (unsigned/no unit) lap_start: <value>, (unsigned/no unit) lap_end: <value> (unsigned/no unit) }</pre>

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x55	BD_ADDR Filter Range 1	9 B	<ul style="list-style-type: none"> • Range 1 for filtered BD_ADDRs • B₀:B₁:B₂:B₃:B₄:B₅ to B₀:B₁:B₂:B₆:B₇:B₈ • OUI = B₀:B₁:B₂ • LAP_{start} = B₃:B₄:B₅ • LAP_{end} = B₆:B₇:B₈ 	Range inactive 0x 00 00 00 00 00 00 00 00 00	<i>bd_addr_range_1: { oui: <value>, (unsigned/no unit) lap_start: <value>, (unsigned/no unit) lap_end: <value> (unsigned/no unit) }</i>
0x56	BD_ADDR Filter Range 2	9 B	<ul style="list-style-type: none"> • Range 2 for filtered BD_ADDRs • B₀:B₁:B₂:B₃:B₄:B₅ to B₀:B₁:B₂:B₆:B₇:B₈ • OUI = B₀:B₁:B₂ • LAP_{start} = B₃:B₄:B₅ • LAP_{end} = B₆:B₇:B₈ 	Range inactive 0x 00 00 00 00 00 00 00 00 00	<i>bd_addr_range_2: { oui: <value>, (unsigned/no unit) lap_start: <value>, (unsigned/no unit) lap_end: <value> (unsigned/no unit) }</i>
0x57	BD_ADDR Filter Range 3	9 B	<ul style="list-style-type: none"> • Range 3 for filtered BD_ADDRs • B₀:B₁:B₂:B₃:B₄:B₅ to B₀:B₁:B₂:B₆:B₇:B₈ • OUI = B₀:B₁:B₂ • LAP_{start} = B₃:B₄:B₅ • LAP_{end} = B₆:B₇:B₈ 	Range inactive 0x 00 00 00 00 00 00 00 00 00	<i>bd_addr_range_3: { oui: <value>, (unsigned/no unit) lap_start: <value>, (unsigned/no unit) lap_end: <value> (unsigned/no unit) }</i>

3.2.5.1 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.

3.2.5.2 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 3-9, the scan duration can be independently set for periodic reports and event-based reports through register 0x51. 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 1.2 and 3.2.3). 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.

3.2.5.3 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 0x54 to 0x57. Each BD_ADDR range register has 9 bytes: B₀ (MSB) to B₈ (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 AC233F (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 AC233F: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 AC233F000000 to AC233FFFFFFF, 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 0x54 to 0x57. 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: 0xB0 for Range 0, 0xB1 for Range 1, 0xB2 for Range 2, and 0xB3 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 performed with message type header 0x0A 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. 0xB0, 0xB1, 0xB2, or 0xB3). For example, if only register 0x54 is non-zero, only Range 0 is active, thus, discovered devices are reported in messages with header 0xB0, and if no device is found under Range 0, a one-byte message of 0xB0 is reported. If only registers 0x54 and 0x55 are non-zero, then only Range 0 and Range 1 are active, and discovered devices are reported in messages with header 0xB0 for Range 0 and header 0xB1 for Range 1. In this case, in every BLE scan, we will ALWAYS have reports (of even an empty list) with both headers 0xB0 and 0xB1.

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—0xB0, 0xB1, 0xB2, or 0xB3).
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.

3.2.5.4 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 3-10 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 the Basic mode, where each device is reported using 7 bytes. The second number is for the case of the Whitelisting mode where 4 bytes is needed per device. For example, from Table 3-10, 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 3-10: 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.

3.2.5.5 BLE 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

3.2.6 Temperature Threshold Configuration

The Tracker supports threshold transmission on the temperature. The temperature is measured from the MCU.

When the temperature thresholds are enabled, the Tracker reports the temperature when it leaves the configured threshold window, and once again when the temperature re-enters the threshold window⁵.

The Threshold mode is compatible with periodic reporting of the temperature; both can be disabled or enabled independently.

Table 3-11 shows a list of configuration registers for the temperature threshold setting. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

⁵ The threshold window is defined as the open interval “(Low Threshold, High Threshold)”; i.e., even if the temperature is equal to Low Threshold or High Threshold, the Tracker is considered to have left the threshold window.

Table 3-11: Temperature Threshold Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x60	Sample Period in Idle State	4 B	<ul style="list-style-type: none"> Sample period of temperature in sec in Idle state Acceptable values: 10, 11, ..., 86400 Other values: Invalid and ignored 	300 seconds 0x 00 00 01 2C	<i>temperature_sample_period_idle: <value></i> <i>(unsigned/sec)</i>
0x61	Sample Period in Active State	4 B	<ul style="list-style-type: none"> Sample period of temperature in sec in Active state Acceptable values: 10, 11, ..., 86400 Other values: Invalid and ignored 	60 seconds 0x 00 00 00 3C	<i>temperature_sample_period_active: <value></i> <i>(unsigned/sec)</i>
0x62	High/Low Thresholds	2 B	<ul style="list-style-type: none"> Bits 8-15: High temperature threshold (signed, 1°C/LSb) Bits 0-7: Low temperature threshold (signed, 1°C/LSb) 	<ul style="list-style-type: none"> High threshold = 30°C Low threshold = 15°C 0x 1E 0F	<i>temperature_threshold {</i> <i> high: <value>,</i> <i> (signed/°C)</i> <i> low: <value></i> <i> (signed/°C)</i> <i>}</i>
0x63	Thresholds Enabled	1 B	<ul style="list-style-type: none"> Bit 0: 0/1 = Thresholds disabled/enabled Bits 1-7: Ignored 	Disabled 0x 00	<i>temperature_thresholds_enabled: <value></i> <i>(unsigned/no unit)</i>

3.2.6.1 Sample Period in Idle/Active State

The Idle/Active sample period determines how often the temperature is checked when the reported value is inside/outside the threshold window. When first enabled, the temperature transducer starts in the Idle state.

3.2.6.2 High/Low Thresholds

Temperature thresholds are stored in a single 2-byte register, with the MSB storing the high temperature threshold, and the LSB storing the low temperature threshold. The high threshold must be greater than the low threshold.

3.2.6.3 Thresholds Enabled

The Thresholds Enabled register enables and disables the threshold reporting on the temperature. The thresholds and sample periods can be configured but are not activated unless the Thresholds Enabled bit is set.

3.2.6.4 Temperature Threshold Configuration DL Examples

- Set Temperature Thresholds:
 - DL payload: 0x E2 23 F6
 - Register 62 with write bit set to true
 - High threshold set to 35°C
 - Low threshold set to -10°C
- Read Sample Periods:
 - DL payload: 0x 60 61
 - Register 60 and Register 61 with their write bits set to false

3.2.7 Command and Control

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

Table 3-12 shows the structure of the Command and Control Registers. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-12: Command & Control Registers

Address	Access	Name	Size	Description	JSON Variable (Type/Unit)
0x70	W	Flash Write Command	2 B	<ul style="list-style-type: none"> • 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 Tracker • Bits 1-12, 15: Ignored 	<pre>write_to_flash { app_config: <value>, (unsigned/no unit) lora_config: <value>, (unsigned/no unit) restart_sensor: <value> (unsigned/no unit) }</pre>

0x71	R	Metadata	7 B	<ul style="list-style-type: none"> • 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 3.2.7.1) 	<pre> 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) } </pre>
0x72	W	Reset Configuration to Factory Defaults	1 B	<ul style="list-style-type: none"> • 0x0A = Reset App Configuration • 0xB0 = Reset LoRaMAC Configuration • 0xBA = Reset both App and LoRaMAC Configurations • Any other value: Invalid and ignored 	<pre> config_factory_reset { app_config: <value>, (unsigned/no unit) loramac_config: <value> (unsigned/no unit) } </pre>

Note: The Command and Control Registers 0x70 and 0x72 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.

3.2.7.1 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 Tracker will reset. This means that if the command was sent in a confirmed DL, the confirmation reply UL will not be sent. The Tracker 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.

3.2.7.2 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 3-2, which is shown using the example FW v1.0.15.

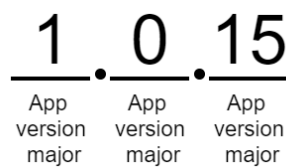


Figure 3-2: Example FW version format

Bits 8-31 in the *Metadata* register contain the LoRaMAC version numbers. The format is the same as shown in Figure 3-2. 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 Tracker are listed in Table 3-13.

Table 3-13: 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

3.2.7.3 *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).

3.2.7.4 *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

4 Appendix A: Default Values

Table 4-1: Default Values of Configuration Registers

Name	Address (Hex)	Default Value (Hex)	Section
LoRaMAC Join Mode	10	80	LoRaMAC Configuration
LoRaMAC Options	11	00 0E	
LoRaMAC DR and Tx Power	12	00 00	
LoRaMAC Rx2 Window (EU868)	13	33 D3 E6 08 00	
LoRaMAC Rx2 Window (US915)	13	37 08 70 A0 08	
LoRaMAC Rx2 Window (AS923)	13	37 06 EA 00 02	
LoRaMAC Rx2 Window (AU915)	13	37 08 70 A0 08	
LoRaMAC Rx2 Window (IN865)	13	33 A6 80 F0 02	
LoRaMAC Rx2 Window (KR920)	13	36 F3 13 E0 00	
LoRaMAC Rx2 Window (RU864)	13	33 CD 69 E0 00	
Seconds per Core Tick	20	00 00 0E 10	
Ticks per Battery	21	00 18	
Ticks per Accelerometer	24	00 00	
Ticks per BLE	25	00 01	
Ticks per Temperature	28	00 00	
Function button Mode	2A	80 08	Function Button Configuration
Function button Event Type I Configuration	2B	11	
Function button Event Type II Configuration	2C	01	
Accelerometer Mode	40	07	Accelerometer Configuration
Accelerometer Sensitivity	41	22	
Acceleration Event Threshold Count	42	00 01	
Acceleration Event Threshold Period	43	00 0A	
Acceleration Event Threshold	44	07 D0	
Acceleration Event Grace Period	45	01 2C	
Acceleration Event Value to Tx	46	03	
BLE Mode	50	88	BLE Configuration
BLE Scan Duration	51	01 03	
BLE Scan Interval	52	00 1E	
BLE Scan Window	53	00 1E	
BD_ADDR Range 0/1/2/3	54/55/56/57	00 00 00 00 00 00 00 00 00 00	
Temperature Sample Period in Idle State	60	00 00 01 2C	Temperature Threshold Configuration
Temperature Sample Period in Active State	61	00 00 00 3C	
Temperature High/Low Thresholds	62	1E 0F	
Temperature Thresholds Enabled	63	00	

References

- [1] TEKTELIC Communications, Inc., "Sensor Configuration Tool," TEKTELIC Communications, Inc., June 2021. [Online]. Available: <http://sensor-config.tektelic-dev.com/>. [Accessed 28 June 2021].
- [2] LoRa Alliance, "LoRaWAN Regional Parameters," ver. 1.0.2, rev. B, Feb 2017.
- [3] LoRa Alliance, "LoRaWAN Specification," ver. 1.0.2, rev. B, Jul 2016.