

ROBIN | FINCH

BLE Panic Button

Technical Reference Manual

Document Type: Technical Reference Manual

Document Number: T0006940_ROBIN-FINCH_TRM

Document Version: 1.0

Document Status: Draft

Product Name: BLE Panic Button

Product Variants & T-Codes (Module-Level):

FINCH	T0007127
BLE Panic Button (Indoor)	
ROBIN	T0007125
BLE Panic Button (C-Cell)	

Release Date: October 2, 2025

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.

© 2025 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

TRM Version	Date	Editor	HW Generation: FW Version	Comments
1.0	October 2, 2025	Emma Tholl		<ul style="list-style-type: none">Initial release based on T0006940_SPARROW-PELICAN_TRM_v3.2.

How to Use This Manual

IMPORTANT: Not all features described in this manual may be applicable to sensors 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 sensor has, send a query as described in §6.7.4.

Goal	Suggestions
To turn the sensor on/off.	Go to §2.1 - DEEP SLEEP.
To quickly look up how to decode an uplink payload.	Go to Appendix 1 – List of Sensor Data Report UL Frame Formats. The tables and figures are organized according to the LoRaWAN port that the uplink is sent on.
To quickly look up configuration settings details (register addresses and sizes, default values, allowable values, encoding details, or json variable names).	Go to Appendix 2 – Lists of Configuration Registers and DL Payload Formats. The tables list the configuration settings in order of register address.
To formulate a downlink payload to send a command (change settings, read current settings, or perform an operation).	§4.3 has a description of general downlink details including the general frame formatting schemes. To find the specific encoding details for the frame contents of the desired command, look it up in Table 0-1 in Appendix 2 or go to the related feature subsection. The simplest way to form payloads is to use the web application called <i>KONA ATLAS</i> available as a comprehensive packet codec tool [1].
To familiarize myself more with a particular sensor feature. E.g.: Learn about possible capabilities, configurable settings, uplink formats, or operational details.	<p>The main sections in this document are organized by sensor function:</p> <ul style="list-style-type: none"> • Magnetic-triggered function • External user interfacing • General sensor and LoRaWAN communication information • Basic operation • BLE operation • Accelerometer operation <p>Each main section is broken into subsections covering individual capabilities or features, which will contain any operational descriptions, suggested use cases, instructions, uplink packet information and examples, configuration settings and downlink payload examples.</p>
To use the sensor to track an asset's location and view this on a dashboard.	<p>The FINCHs/ROBINs act as end-devices in an end-to-end asset tracking solution. All components of this system must be set up to view location on a map, as explained in §1.1.</p> <p>This document is a manual for only the FINCH/ROBIN sensor, not for the other components in the end-to-end solution (Gateways, Network Server, Applications, etc.). For information about setting up a complete asset tracking solution, refer to the articles on the support portal Knowledge Base [2].</p>

Table of Contents

Revision History	2
How to Use This Manual	3
Table of Contents	4
List of Tables	8
List of Figures	10
List of Acronyms	11
1 Introduction	12
1.1 Overview	12
1.2 Summary of HW Information, Streams, and Default Behaviour	16
2 Magnetic Function	18
2.1 DEEP SLEEP	19
2.1.1 Sensor Activation	19
2.1.2 Re-entering DEEP SLEEP	19
2.2 Sensor Reset	20
2.3 Forced Uplinks	20
3 External User Interfacing	22
3.1 Function Button	22
3.2 Reset Button	23
3.3 LED Behaviour	23
3.3.1 Power-On and Network Join Patterns	23
3.3.2 Normal Operation Patterns	25
3.3.3 DEEP SLEEP and Magnetic Reset Patterns	25
3.3.4 Entering and Exiting PANIC Mode Patterns	26
3.4 Buzzer and Vibration Motor	27
3.4.1 Entering and Exiting PANIC Mode Patterns	27
4 General Sensor and LoRaWAN Communication Information	29
4.1 Configuration Settings Format	29
4.2 UL Payload Formats	30
4.2.1 Sensor Data Reports (<i>LoRaWAN Port 10</i> and <i>25</i> ULs)	30

4.2.2	Responses to Commands (<i>LoRaWAN Port 100</i> and <i>101</i> ULs)	31
4.3	DL Payload Formats and Configuration Settings	36
4.3.1	Entering DEEP SLEEP Command (<i>LoRaWAN Port 99</i> DLs)	36
4.3.2	Configuration and Control Commands (<i>LoRaWAN Port 100</i> DLs)	36
5	State Machine Implementation – PANIC vs. NORMAL State	39
5.1	Operational Description	39
5.2	NORMAL State Operation	42
5.3	PANIC State Operation	43
5.4	State Change Request DLs (PARs and NARs)	45
5.5	UL Report Frame Format	45
5.6	Configuration Settings	46
6	Basic Operation	50
6.1	LoRaMAC Options	51
6.1.1	Operational Description	51
6.1.2	Configuration Settings	51
6.2	Command Response Options	54
6.2.1	Operational Description	54
6.2.2	Configuration Settings	54
6.3	Periodic Report Scheduling	55
6.3.1	Operational Description	55
6.3.2	Configuration Settings	57
6.4	MCU Data Reporting	59
6.4.1	Operational Description	59
6.4.2	UL Report Frame Format	59
6.4.3	Configuration Settings	59
6.5	Battery Management	62
6.5.1	Operational Description	62
6.5.2	UL Report Frame Formats	64
6.5.3	Configuration Settings	64
6.6	Function Button Operation	68

6.6.1	Operational Description	68
6.6.2	Configuration Settings.....	70
6.7	General Command-and-Control Operations	73
6.7.1	List of Operations and Register Values	73
6.7.2	Save Current Configuration Settings.....	74
6.7.3	Sensor Restart	75
6.7.4	Read FW and BLE Metadata.....	76
6.7.5	Factory Reset.....	77
7	BLE Operation	78
7.1	Tracker Mode (BLE Rx).....	79
7.1.1	Operational Description	79
7.1.2	UL Report Frame Formats	83
7.1.3	Configuration Settings.....	85
7.2	Beacon Mode (BLE Tx)	94
7.2.1	Operational Description	94
7.2.2	Configuration Settings.....	97
7.3	Toggling Tracker Mode ↔ Beacon Mode.....	101
7.3.1	Toggling Procedure and Commands	101
7.3.2	Differences Between Modes.....	102
8	Accelerometer Operation	104
8.1	General Sampling Behaviour	105
8.1.1	Operational Description	105
8.1.2	Configuration Settings.....	105
8.2	Orientation Detection.....	108
8.2.1	Operational Description	108
8.2.2	UL Report Frame Formats	109
8.2.3	Configuration Settings.....	109
8.3	Motion Detection Alarms	112
8.3.1	Operational Description	112
8.3.2	UL Report Frame Formats	113

8.3.3	Configuration Settings.....	114
8.4	Accelerometer Assist for BLE Asset Tracking	117
8.4.1	Operational Description	117
8.4.2	UL Report Frame Formats	117
8.4.3	Configuration Settings.....	117
8.5	Device Status Accelerometer Trigger	120
8.5.1	Operational Description	120
8.5.2	UL Report Frame Formats	120
8.5.3	Configuration Settings.....	120
Appendix 1 – List of Sensor Data Report UL Frame Formats		122
<i>LoRaWAN Port 10 UL Frame Formats</i>		<i>122</i>
LoRaWAN Port 25 UL Frame Formats		124
<i>LoRaWAN Port 100 and 101 UL Frame Formats</i>		<i>125</i>
Appendix 2 – Lists of Configuration Registers and DL Payload Formats.....		126
References.....		137

List of Tables

Table 1-1: Generational Differences	16
Table 1-2: Enterprise Asset Tracker HW Models	16
Table 1-3: Enterprise Asset Tracker Information Streams	16
Table 1-4: Default Reporting Behavior.....	17
Table 3-1: Summary of LED Patterns.....	23
Table 3-2: Summary of Buzzer and Vibration Motor Patterns.....	27
Table 4-1: UL Information Streams	30
Table 4-2: Command Response Option Behaviour	32
Table 4-3: DL Information Streams	36
Table 5-1: FINCH/ROBIN State Machine Transition Summary.....	39
Table 5-2: Request Type DLs	45
Table 5-3: Device Status Data Report UL Frame Format Details	46
Table 6-1: LoRaMAC Options Configuration Registers.....	51
Table 6-2: Default Max Tx Power, Rx2 Channel Frequency, and Rx 2 DR Number by Regions ...	52
Table 6-3: Command Response Option Configuration Register	54
Table 6-4: Periodic Transmission Configuration Registers.....	57
Table 6-5: MCU Data Report UL Frame Format Details	59
Table 6-6: MCU Data Reporting Configuration Settings	60
Table 6-7: Battery Report UL Frame Formats	64
Table 6-8: Battery Management Configuration Registers	65
Table 6-9: Function Button Behavior	68
Table 6-10: Function Button Configuration Registers.....	70
Table 6-11: Command & Control Register Details	73
Table 6-12: LoRaMAC Regions and Region Numbers.....	76
Table 7-1: Tracker Mode Configuration Registers	86
Table 7-2: Maximum Number of Reported BLE Devices per LoRaWAN Packet in Different Regions	92
Table 7-3: BLE Advertising Configuration.....	97
Table 7-4: Tracker/Beacon Mode Configuration Register	101
Table 7-5: Default Register Values for Tracker/Beacon Mode	102
Table 8-1: General Accelerometer Configuration Registers	106
Table 8-2: Accelerometer Vector Report UL Frame Format	109
Table 8-3: Accelerometer Vector Report Configuration Registers	110
Table 8-4: Accelerometer Vector Report UL Frame Format	113
Table 8-5: Accelerometer Motion Detection Alarm Configuration Registers.....	114
Table 8-6: Accelerometer Assist Configuration Registers.....	118
Table 8-7: Device Status Accelerometer Trigger Configuration Registers.....	121

Table 0-1: Complete List of <i>LoRaWAN Port 10</i> Sensor Data Frame Formats	123
Table 0-1: Complete list of Configuration Registers	126
Table 0-2: Comparison of Default Configuration Values for Tracker and Beacon Mode	135

List of Figures

Figure 1-1: BLE Asset Tracking End-to-End Architecture	13
Figure 2-1: Enclosure View Showing Magnetic Activation Site and Symbol.....	18
Figure 2-2: The Magnetic Activation/Reset Pattern.	19
Figure 2-3: The Magnetic Forced UL Pattern	20
Figure 3-1: External User Interfacing on ROBIN (left) and FINCH (right) Enclosures.....	22
Figure 3-2: The GREEN POST Pass (left) and RED POST Failure (right) LED Patterns	24
Figure 3-3: The LED Patterns During JOIN Mode	25
Figure 3-4: The RED LED Pattern Before Entering DEEP SLEEP	26
Figure 3-5: The LED Pattern for Entering PANIC Mode.....	26
Figure 3-6: The LED Pattern for Exiting PANIC Mode.....	27
Figure 3-7: The Buzzer and Vibration Motor Pattern for Entering PANIC Mode.....	28
Figure 3-8: The LED Pattern for Exiting PANIC Mode.....	28
Figure 4-1: Bit Indexing Scheme for Configuration Registers	29
Figure 4-2: The UL Frame Format for a Sensor Data Report.....	31
Figure 4-3: The <i>LoRaWAN port 101</i> Write/Error Response UL Frame Format	34
Figure 4-4: The DL Formats for Configuration and Control Message Blocks	37
Figure 5-1: FINCH/ROBIN State Machine While Connected to Network (Normal Operation)	39
Figure 5-2: NORMAL Process Flow	43
Figure 5-3: PANIC Process Flow.....	44
Figure 6-1: Example Visualization of the Default Average Energy Trend Window.....	62
Figure 6-2: Example FW Version Format.....	76
Figure 7-1: The BLE Scan Scheme	80
Figure 7-2: The BLE Report UL Payload Format for Basic Reporting	83
Figure 7-3: BLE Report UL Payload Formats for Filtered Reporting.....	84
Figure 7-4: Periodic BLE Scanning	90
Figure 7-5: The BLE Advertisement Scheme	94
Figure 7-6: Supported Standard BLE Beacon Advertisement Packet Formats	95
Figure 8-1: Directions of Axes for ROBIN (left) and FINCH (right).	108
Figure 8-2: Example Sequence of Accelerometer Events and UL Reports.....	113
Figure 0-1: The UL Frame Format for a Sensor Data Report.....	122
Figure 0-2: The BLE Report UL Payload Format for Basic Reporting	124
Figure 0-3: BLE Report UL Payload Formats for Filtered Reporting.....	124
Figure 0-4: The <i>LoRaWAN Port 100</i> Read Response Format	125
Figure 0-5: The <i>LoRaWAN port 101</i> Write Response Format	125
Figure 0-1: The <i>LoRaWAN Port 100</i> DL Formats for Configuration and Control Message Blocks	126

List of Acronyms

ABP	Activation by Personalization	MAC	Medium Access Control
Accl	Accelerometer	Max	Maximum
ADDR ..	Address	MCU	MicroController Unit
ADR	Adaptive Data Rate	MSb	Most Significant bit
ADV	ADvertisement	MSB	Most Significant Byte
App	Application	NE	North East
AS	ASia	NS	Network Server
ATEX ...	ATmosphere EXplosible	Opt	Option
AU	AUstralia	OTA	Over-The-Air
BD	Bluetooth Device	OTAA ..	OTA Activation
BLE	Bluetooth Low Energy	OUI	Organizationally Unique Identifier
CRC	Cyclic Redundancy Check	PCBA ...	Printed Circuit Board Assembly
DevEUI	Device Extended Unique ID	PDU	Protocol Data Unit
DL	DownLink	POST ...	Power-On Self-Test
DR	Data Rate	R	Read
EIRP ...	Equivalent Isotropically Radiated Power	Rev	Revision
EU	EUrope	RF	Radio Frequency
FCntDown	Frame Counter Down	RFU	Reserved for Future Use
FW	FirmWare	RO	Read-Only
Gen	Generation	RSSI	Received Signal Strength Indicator
HW	HardWare	RTLS	Real-Time Location System
ID	IDentity/IDentifier	RU	RUssia
IN	India	R/W	Read/Write
Inc	Incorporated	Rx	Receiver / Receive
IoT	Internet of Things	SGS	Società Generale Semiconduttori
IP	Ingress Protection	STM	SGS-TS Microelectronics
JSON ...	JavaScript Object Notation	SW	SoftWare
KR	KoRea	TLM	TeLeMetry
LAP	Lower Address Part	TRM	Technical Reference Manual
LED	Light-Emitting Diode	TS	Thomson Semiconducteurs
LoRa	Long Range	Tx	Transmitter / Transmit
LoRAMAC	LoRaWAN MAC	UID	Unique ID
LoRaWAN	LoRa Wide Area Network	UUID ...	Universally Unique ID
LoS	Line-of-Sight	UL	UpLink
LSb	Least Significant bit	v	version
LSB	Least Significant Byte	W	Write
LTC	Lithium Thionyl Chloride	X	Executable

1 Introduction

This document contains the technical information about the supported functionality of the TEKTELIC *BLE Panic Button* sensors, which include the FINCH and ROBIN. 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.

1.1 Overview

The FINCH and ROBIN Panic Buttons are BLE-capable LoRaWAN IoT sensors run on a single LTC battery and packed into a compact polycarbonate casing. The primary purpose of the sensors is to scan for nearby BLE peripherals (devices) and report information about them over LoRaWAN messages.

LoRaWAN is the LoRa wireless communications standard protocol. This technology provides a low-bandwidth, low-power, and long-range¹ means of transmitting small amounts of data. It has been developed with wireless sensing in mind, and to enable new means of gathering telemetry in numerous environments. The FINCH and ROBIN support LoRa and (G)FSK modulations according to the LoRaWAN 1.0.4 Specification [3]. The 150 MHz-960 MHz ISM bands are utilized to meet different application requirements from the standards and proprietary protocols of the given region.

The sensor can also be configured to act as a BLE beacon. When configured this way, the device no longer conducts scans, but instead sends out advertisements which make it discoverable by other nearby trackers or BLE-capable devices.

¹ Up to 2 km NLoS and more than 22 km LoS.

The primary intended use case is to use multiple FINCHs and ROBINS together as components in an end-to-end asset tracking solution as shown in

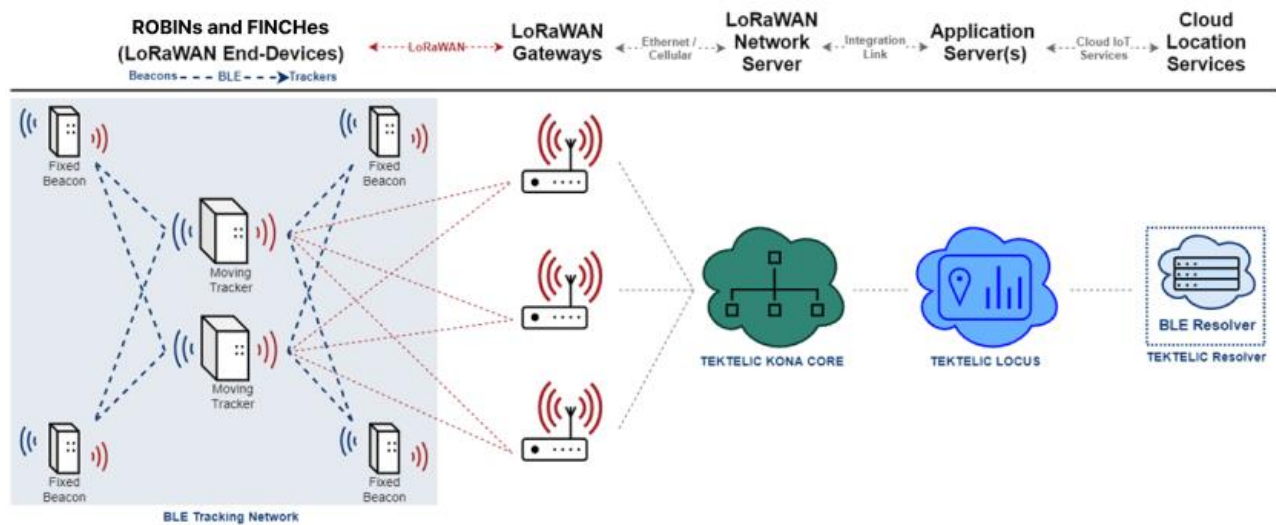


Figure 1-1.

NOTE: Only raw scan data is present in the LoRaWAN payloads, not sensor location information. In order to track and visualize a FINCH or ROBIN's location, an application server must be set up, integrated with the NS, and enabled to use the proper cloud location services. The information in this document is for the sensor only; for information about setting up the rest of the end-to-end solution shown in **Error! Reference source not found.**, refer to the TEKTELIC support portal

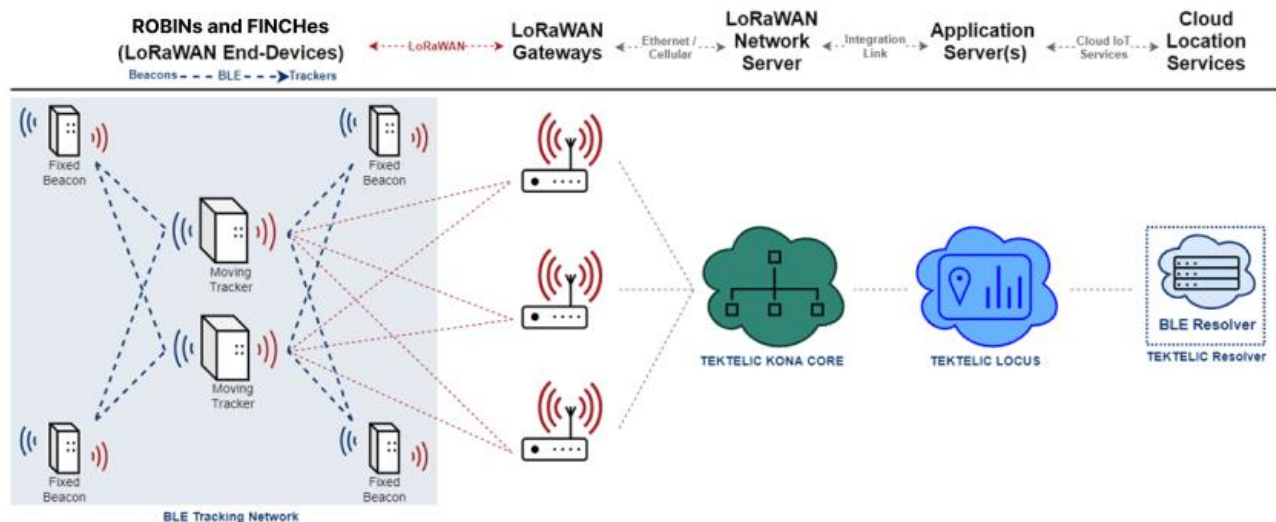


Figure 1-1: BLE Asset Tracking End-to-End Architecture

The asset location information flows in this order:

1. The FINCHs and/or ROBINS acting as beacons in fixed locations broadcast periodic BLE advertisement packets.
2. The FINCHs and/or ROBINS acting as trackers conduct BLE scans to gather raw information from the beacon signals.
3. The raw scan results are conveyed via LoRa transmissions to 1 or more LoRaWAN GateWay (GW).
4. The GWs forward the packets to the LoRaWAN Network Server (NS) either by ethernet or cellular backhaul.
5. The raw scan results are forwarded to the application layer via integration link.
6. The application uses cloud IoT location resolver service(s) to processes the raw scan data to calculate a position fix. TEKTELIC's LOCUS application is designed to work natively with FINCH/ROBIN, but any compatible 3rd-party application can be used.
7. The resolved fixes are returned to the application where they can then be visualized on a virtual dashboard.

In addition to BLE function, the FINCH and ROBIN are multipurpose devices equipped with a variety of technology:

- **Core design:** Based on the low-power, IoT-targeted STM32WB55CWU6 MCU, which runs the system SW and has a built-in BLE module.
- **Function button:** An externally accessible push-button that can be configured to have various functions, including putting the device in PANIC mode.
- **Magnetic detection:** A magnet can be used to wake a ROBIN from the DEEP SLEEP state (used for shipping) and to force ULs during normal operation.
- **Accelerometer:** Detects device motion state so additional BLE scan results can be sent when beginning and ending movement. Motion alarms and the raw acceleration vector can also be reported if knowledge of sensor orientation is of interest.
- **Battery data:** The remaining capacity and lifetime can be reported. The battery lifetime of the sensor has been estimated to be up to:
 - 5.5 years for FINCH operating with default Tracker mode settings².
 - 15+ years for ROBIN operating with default Tracker mode settings².

² Default settings plus 24 event-based BLE scans and reports per day, operating at DR2. Applicable to NA region only.

- 16 months for FINCH operating with default Beacon mode settings³.
 - 4.5 years for ROBIN operating with default Beacon mode settings³.
- **MCU temperature reporting:** The on-board internal temperature can be polled and reported.

Some other common, non-tracking use-cases include, but are not limited to:

- Proximity detection for equipment in places of interest.
- Asset movement detection and security.
- Broadcasting sensor telemetry over BLE for surveying purposes.

³ Default settings with no event-based reports, operating at DR2. Applicable to NA region only.

1.2 Summary of HW Information, Streams, and Default Behaviour

There have been 3 *generations* of the BLE Sensors: Gen1, Gen2, and Gen2.5. Note that Gen1 and Gen2 functional descriptions are outside the scope of this TRM version.

Table 1-1: Generational Differences

Generation	MCU	FW Versions	Supported Models
Gen1	Silicon Labs EFR32 Blue Gecko	<ul style="list-style-type: none"> 1.x.x (does not support <i>beacon mode</i>) 	<ul style="list-style-type: none"> FINCH
Gen2	Silicon Labs EFR32 Blue Gecko	<ul style="list-style-type: none"> 1.x.x (does not support <i>beacon mode</i>) 2.x.x (supports <i>beacon mode</i>) 	<ul style="list-style-type: none"> FINCH ROBIN
Gen2.5	STM 32	<ul style="list-style-type: none"> 3.x.x 	<ul style="list-style-type: none"> FINCH ROBIN

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.

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
T0007127	T0008845	FINCH	MODULE, FINCH, BLE PANIC BUTTON, AA CELL, BELT CLIP	AA-cell
T0007125	T0008594	ROBIN	MODULE, ROBIN, BLE PANIC BUTTON, C-CELL	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	100/101
DL (NS to Tracker)	Putting Tracker into DEEP SLEEP	99
	Configuration and Control Commands	100

Table 1-4: Default Reporting Behavior

Report Type	Periodic Report Schedule	Event-Based Report Trigger
Battery status	1 report / day	Each time the magnetic sensor is triggered ⁴
Discovered BLE devices following a scan	1 report / hour	Each time the function button is pressed
		Each time above-threshold motion is detected, and each time motion stops
Acceleration vector	Disabled by default	Disabled by default
Accelerometer motion alarm	N/A	Each time above-threshold motion is detected, and each time motion stops
MCU temperature	Disabled by default	Disabled by default

⁴ Only the ROBIN is equipped with magnet function.

2 Magnetic Function

Only the ROBIN is equipped with magnetic functionality; the information in this section is not applicable to FINCH.

The ROBIN is equipped with an internal hall-effect sensor, which detects the presence and absence of a magnet. A magnet will be detected if it is sufficiently strong and brought into contact with the enclosure at the *magnetic activation site*, which is located on the enclosure body right above the magnet symbol on the endcap, as shown in Figure 2-1.

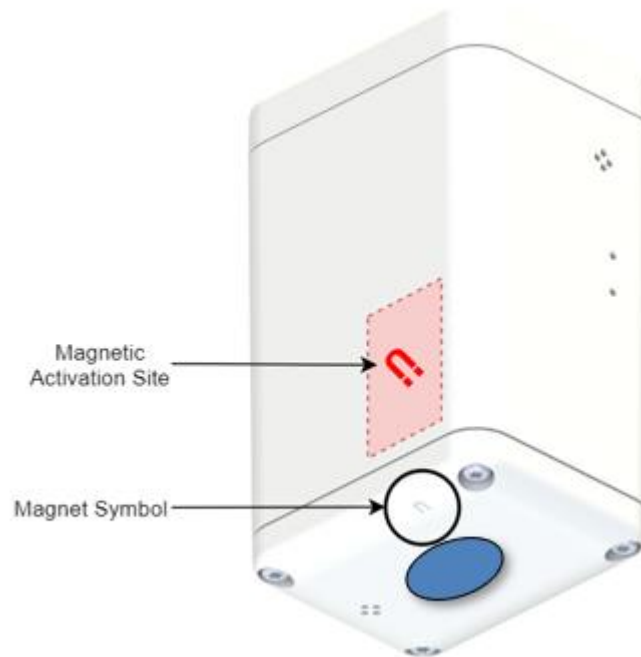


Figure 2-1: Enclosure View Showing Magnetic Activation Site and Symbol

There are 3 main magnetic functions of the ROBIN:

1. Waking from DEEP SLEEP or putting into DEEP SLEEP.
2. Sensor reset.
3. Forced ULs.

2.1 DEEP SLEEP

2.1.1 Sensor Activation

When a ROBIN exits the factory, it is put into low-power DEEP SLEEP mode to conserve battery life during shipping and storage. To activate the sensor and wake it from DEEP SLEEP, the user must apply a *magnetic activation/reset pattern*⁵.

The magnetic activation/reset pattern is illustrated in Figure 2-2. A “magnet presence” is achieved by placing a magnet against the enclosure at the magnetic activation site as shown in Figure 2-1. A “magnet absence” is achieved by taking the magnet away from the enclosure. Figure 2-2 shows that the pattern involves sustaining a “magnet presence” continuously for at least 3 sec but less than 10 sec.

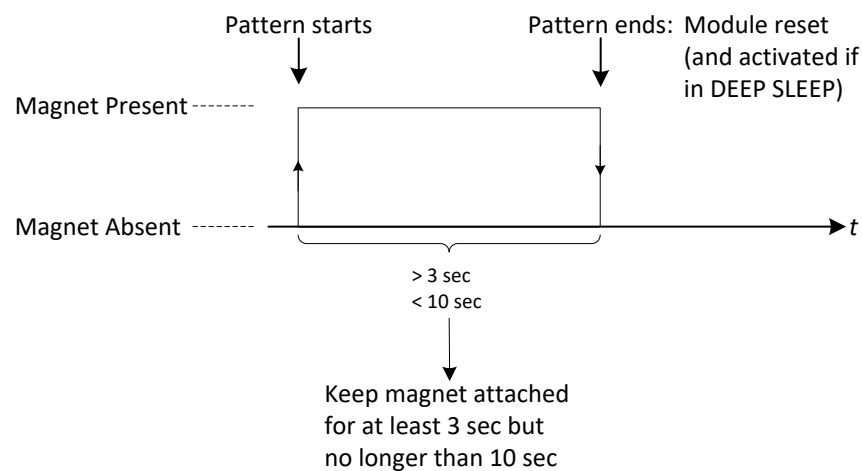


Figure 2-2: The Magnetic Activation/Reset Pattern.

When the magnetic activation/reset pattern is applied to the sensor it displays an LED indication (described in §3.3.3) to show that it has accepted the pattern. In all cases, this 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 network⁶.

2.1.2 Re-entering DEEP SLEEP

There are 2 methods for putting the sensor into DEEP SLEEP mode:

⁵ The FINCH does not support DEEP SLEEP mode nor any magnetic function. A FINCH is activated simply by the removal of the battery tab [7].

⁶ The very first time a sensor is activated out of the box or after a battery replacement, there might be some ramp-up time required due to battery passivation. See §6.5.1.2 for details.

1. **Magnetic Activation/Reset Pattern:** The same pattern to wake the sensor from DEEP SLEEP (Figure 2-2) can be used to put the sensor back into DEEP SLEEP, if applied while the sensor is in JOIN mode. I.e.: If the sensor detects the magnetic pattern before it has received the JOIN ACCEPT DL, then it will re-enter DEEP SLEEP mode.
2. **OTA DL Command:** If the sensor has already joined the network and is in normal operation mode, a DL can be sent to remotely put it into DEEP SLEEP mode. The frame payload must contain the single byte **0x 00** and the DL must be sent on **LoRaWAN port 99** as explained in §4.3.1.

After completing either of these methods, the magnetic activation/reset pattern will then be required to bring this sensor back out of DEEP SLEEP, as usual.

2.2 Sensor Reset

If the sensor is in normal operation - already joined to the network and not still in JOIN mode - applying the magnetic activation/reset pattern as described in §2.1.1 will cause it to reset. After resetting, the sensor will enter JOIN mode and begin attempting to join the network.

2.3 Forced Uplinks

As a Class-A LoRaWAN end-device, the sensor only opens LoRaWAN receive windows immediately following uplink transmissions [3]. It is therefore useful to be able to force the sensor to UL so that it can receive DL commands from the NS ahead of its next scheduled periodic report.

The magnetic pattern for forced ULs involves briefly tapping the magnet against the magnetic activation site then removing it in less than 2 seconds, as illustrated in Figure 2-3. The magnetic activation site is shown in Figure 2-1.

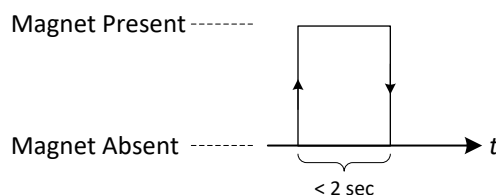


Figure 2-3: The Magnetic Forced UL Pattern

The magnetic forced UL has a payload which contains a regular battery report on **LoRaWAN port 10**, as described in §4.2.1.

NOTE: Mistakenly holding the magnet against the magnetic activation site for more than 3 s may trigger a module reset, as explained in §2.2

3 External User Interfacing

The ROBIN and FINCH enclosures are shown in Figure 3-1 and the locations of the external user interfacing are identified. The function button, reset button, and LED behaviour are described in the following subsections.

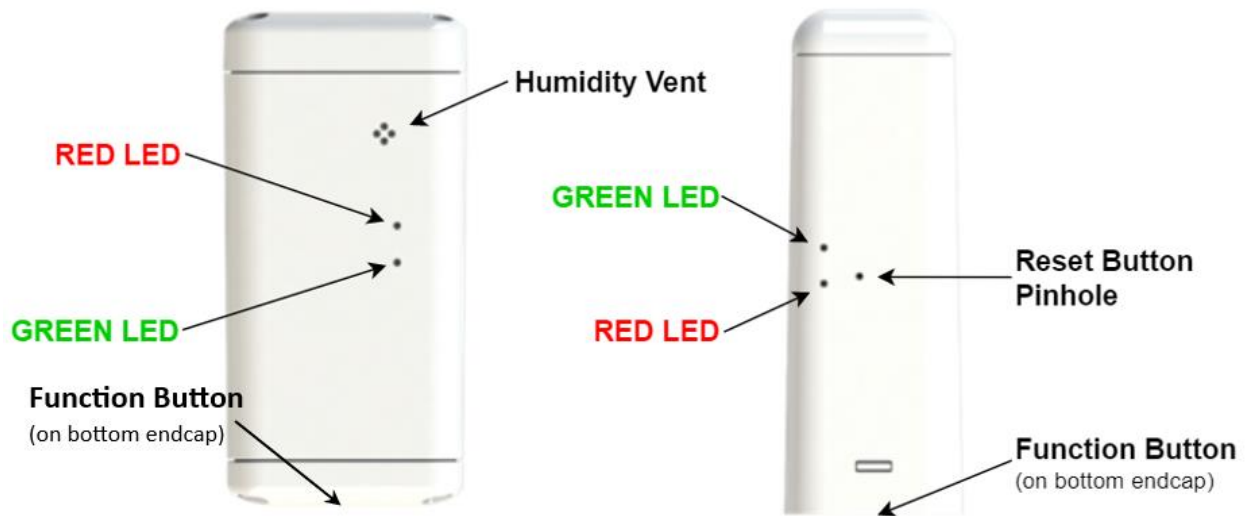


Figure 3-1: External User Interfacing on ROBIN (left) and FINCH (right) Enclosures

3.1 Function Button

The function button is externally accessible and located on the bottom of the FINCH and ROBIN enclosures, as shown in Figure 3-1. It is actuated by pressing until haptic click feedback is felt.

The button supports four distinct press types: **Data Press**, **Panic Press**, **Normal Press**, and **Device Reset Press**. Each type is defined by a unique press pattern and results in specific actions, such as triggering uplink messages, transitioning between NORMAL and PANIC states, or performing a soft reset. These patterns and behaviors are configurable through dedicated registers.

The button cannot be disabled and requires at least one data type to be configured for transmission. Further details are described in the following subsections. For more information on state transitions (N1 and N2), refer to §5. For details on function button behavior and configuration, refer to §6.6.

3.2 Reset Button

Only the FINCH has a physical reset button. To reset a ROBIN, a magnet must be used as described in §2.2.

To reset the FINCH, insert a pin or paperclip into the reset button pinhole at the location shown in Figure 3-1. Press firmly until the button is pressed, then release. As soon as the button is released, the FINCH will reset and begin the network join process. The LEDs will indicate the reset as described in §3.3.1.

3.3 LED Behaviour

The sensor is equipped with two on-board LEDs: **GREEN** and **RED**. They are visible through holes in the sensor enclosure at the locations shown in Figure 3-1. The LED behaviour is not user configurable.

The LEDs are normally off. Their blinking patterns reflect different actions and states of the sensor. At a high level, the main patterns are summarized in Table 3-1. The detailed sequence and timings for each are described in the following subsections.

Table 3-1: Summary of LED Patterns

LED Pattern	Meaning
GREEN blinking rapidly and single RED flash every 10 s	JOIN mode; attempting to join the network
Single RED flash	UL sent
Single GREEN flash	DL received
3 quick RED flashes	Entering DEEP SLEEP
3 RED flashes for 0.5 seconds	Entering PANIC Mode
3 GREEN flashes for 0.2 seconds	Exiting PANIC Mode

3.3.1 Power-On and Network Join Patterns

When the sensor is activated or reset:

1. Both **GREEN** and **RED** are OFF for approximately 0.5 s after any reset occurs.
2. Upon startup, the SW conducts its POST. Both **GREEN** and **RED** are turned on when the POST begins.
3. When the POST ends (about 2 s), both **GREEN** and **RED** are turned off. Immediately following, the sensor will do 1 of 2 things, depending on the POST result:

- a. If the POST passes, **GREEN** is toggled ON and OFF 3 times: every 100 ms for 0.6 s, as shown in Figure 3-2. In this case, the LED pattern proceeds to step 4.
- b. If the POST fails, **RED** is toggled ON and OFF 3 times: every 100 ms for 0.6 s, as shown in Figure 3-2. In this case, the device restarts and the LED pattern begins again at step 1 after approximately 4 s.

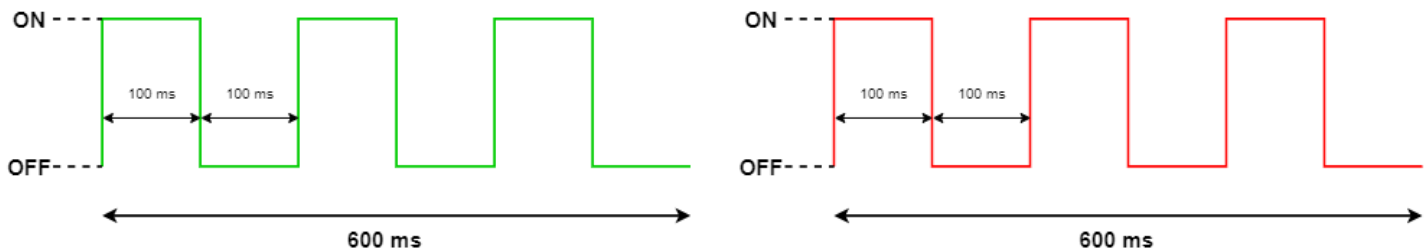


Figure 3-2: The GREEN POST Pass (left) and RED POST Failure (right) LED Patterns

4. After a successful POST, both **GREEN** and **RED** are turned off. Immediately following this, the sensor will enter JOIN mode and begin attempting to join the network. For the first hour⁷:
 - a. **GREEN** is toggled ON and OFF every 50 ms for the first hour.
 - 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 [4].
 - 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.

If the sensor has been unsuccessfully trying to join for more than an hour, it enters *join back-off* to conserve power. While the sensor still attempts to join, **GREEN** stops flashing and **RED** flashes twice (ON time: 10 ms, OFF time: 10 ms) every 8 s. The JOIN LED pattern is shown in Figure 3-3

⁷ The very first time a sensor is activated out of the box or after a battery replacement, there might be some ramp-up time required due to battery passivation. See §6.5.1.2 for details.

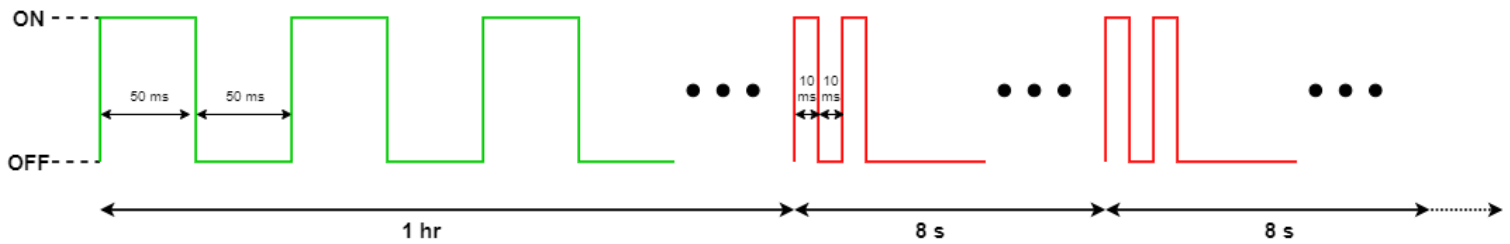


Figure 3-3: The LED Patterns During JOIN Mode

3.3.2 Normal Operation Patterns

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 100 ms right after receiving a downlink.

3.3.3 DEEP SLEEP and Magnetic Reset Patterns

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 turned ON for 75 ms, then turned OFF.
2. After a 100-500 ms pause while the device resets, the normal Power-On and Network Join LED patterns described in §3.3.1 occur.

There is another LED pattern for when the device is put back into DEEP SLEEP. The following LED pattern is displayed about 3 s after the pattern is applied:

1. After a 100-500 ms pause while the device resets, the Power-On POST LED patterns described in steps 1-3 in §3.3.1 occur.
2. Immediately, **RED** is toggled ON and OFF 3 times: every 100 ms for 0.6 sec as shown in Figure 3-4.

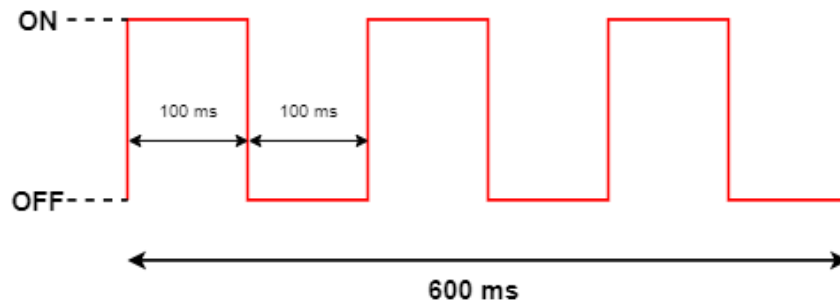


Figure 3-4: The RED LED Pattern Before Entering DEEP SLEEP

3.3.4 Entering and Exiting PANIC Mode Patterns

The sensor displays an LED indication when it enters PANIC mode and when it exits PANIC mode by applying the pushbutton pattern. The following LED pattern is displayed when the sensor is entering PANIC mode:

1. **RED** is turned ON for 500 ms, then turned OFF for 1000 ms.
2. This LED pattern is repeated three times.

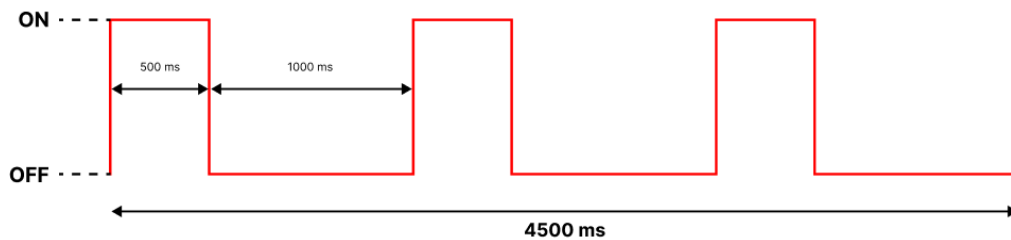


Figure 3-5: The LED Pattern for Entering PANIC Mode

The following LED pattern is displayed when the sensor is exiting PANIC mode:

1. **GREEN** is turned ON for 200 ms, then turned OFF for 200 ms.
2. This LED pattern is repeated three times.

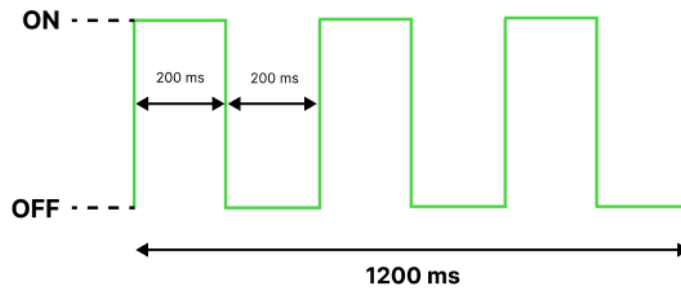


Figure 3-6: The LED Pattern for Exiting PANIC Mode

3.4 Buzzer and Vibration Motor

The sensor is equipped with a buzzer and vibration motor. The buzzer provides audible feedback whereas the vibration motor provides haptic feedback. These configurations are not user configurable.

The buzzer and vibration motor are normally off. Their patterns reflect different actions and states of the sensor. At a high level, the main patterns are summarized in Table 3-1. The detailed sequence and timings for each are described in the following subsections.

Table 3-2: Summary of Buzzer and Vibration Motor Patterns

Buzzer and Vibration Motor Pattern	Meaning
ON for 500 ms, OFF for 1000 ms, three times	Entering PANIC Mode
ON for 200 ms, OFF for 200 ms, three times	Exiting PANIC Mode

3.4.1 Entering and Exiting PANIC Mode Patterns

The sensor exhibits a buzzer and vibration motor pattern when it enters PANIC mode and when it exits PANIC mode by applying the pushbutton pattern. The following buzzer and vibration motor pattern is displayed when the sensor is entering PANIC mode:

- Both the buzzer and vibration motor are turned ON for 500 ms, then turned OFF for 1000 ms.
- This buzzer and vibration motor pattern is repeated three times.

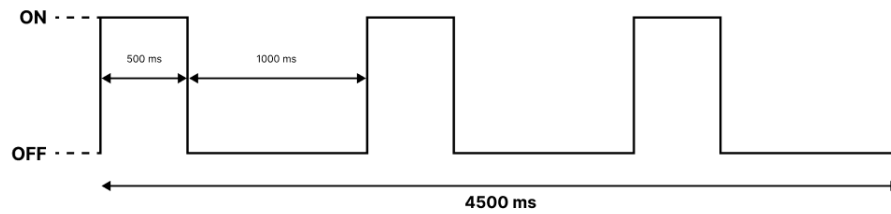


Figure 3-7: The Buzzer and Vibration Motor Pattern for Entering PANIC Mode

The following LED pattern is displayed when the sensor is exiting PANIC mode:

3. Both the buzzer and vibration motor are turned ON for 200 ms, then turned OFF for 200 ms.
4. This buzzer and vibration motor pattern is repeated three times.

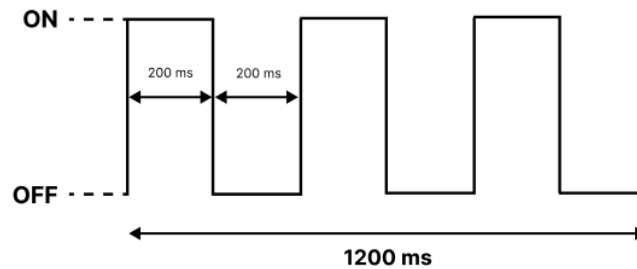


Figure 3-8: The LED Pattern for Exiting PANIC Mode

4 General Sensor and LoRaWAN Communication Information

The FINCH and ROBIN communicate with the NS using LoRaWAN packets [3]. The communication behaviour as well as other sensor functional behaviour is configurable through SW settings. The following subsections describe the general configuration settings format, communication streams, and packet formats supported by the FINCH and ROBIN.

There is an online application called *KONA ATLAS* available as a comprehensive packet codec tool [1]. It supports encoding DL payloads and decoding UL payloads for FINCH and ROBIN.

4.1 Configuration Settings Format

Configuration settings are saved in the flash storage *configuration registers*. Each register has an *address* that is assigned to a particular setting or action. These addresses are bound between 0x 00 and 0x 7F, inclusive. The bit indexing scheme for register addresses and values is as shown in Figure 4-1.

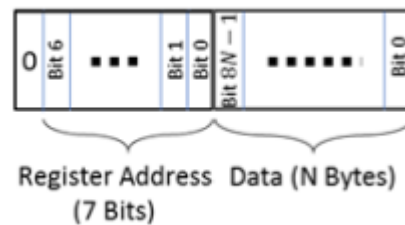


Figure 4-1: Bit Indexing Scheme for Configuration Registers

Possible register access permission options are read/write (R/W), read only (RO), or executable (X). Accessing each register involves *configuration and control commands*. The general formats are described in §4.3.2.

Not every register is utilized in the current FINCH/ROBIN SW; some are RFU. These registers are locked and inaccessible for R/W.

For a complete summary list of the specific definitions, sizes, and default values of all configuration registers, see Table 0-1 in Appendix 2.

4.2 UL Payload Formats

UpLinks (ULs) are LoRaWAN packets sent from the sensor to the NS. They are used for the following purposes in FINCH/ROBIN⁸:

1. **Reports** containing real time sensor data.
2. **Responses** to configuration and control commands⁹.

Each UL is sent on the appropriate upstream port. The UL streams supported by the FINCH/ROBIN SW are shown in Table 4-1.

Table 4-1: UL Information Streams

Data Type	Sent on <i>LoRaWAN Port</i> [decimal]
Reports containing sensor data from the MCU, battery, and/or accelerometer	10
Reports containing discovered BLE device data	25
Responses to configuration and control commands	100/101

4.2.1 Sensor Data Reports (*LoRaWAN Port 10* and *25* ULs)

All sensor data report ULs sent on ***LoRaWAN port 10*** and ***LoRaWAN port 25*** fall into one of the following reporting categories:

- **Periodic Reporting:** Scheduled reporting of sensor data at regular, configurable intervals. The reporting intervals are configured using the tick registers as described in §6.3.2.
- **Event-Based Reporting:** Various external events can trigger unscheduled sensor data reports outside of the periodic reporting schedule. These external events include function button presses, magnetic switch actuation, and motion of the device above the accelerometer threshold. Each event elicits a different type of response from the Sensor. Not all event-based reporting is enabled by default.

The following sensor data types are sent in reports on ***LoRaWAN port 10***:

- Remaining battery capacity
- Acceleration alarm
- Device status
- Remaining battery lifetime
- Acceleration vector
- MCU temperature

⁸ Here “ULs” means “SW application-level ULs;” there are other MAC-level ULs that the sensor may send to perform LoRaMAC operations. LoRaMAC behaviour occurs according to the LoRaWAN specifications and is outside the scope of this document [9].

⁹ See §4.3.2 for a description of configuration and control commands.

The **LoRaWAN port 10** UL report payload is encoded in a general frame format shown in Figure 4-2. A big-endian format (MSb/MSB first) is always followed.

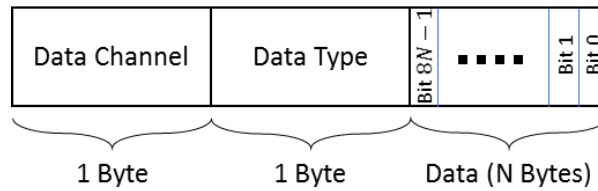


Figure 4-2: The UL Frame Format for a Sensor Data Report

A single sensor data report payload can include multiple data frames from different sensing components and these frames can be arranged in any order.

Discovered BLE device data is sent in reports on **LoRaWAN port 25**. BLE reporting payload structures differ from other sensor data formats described above. See §7.1.2 for a complete description of BLE reporting behaviour.

For a complete summary list of **LoRaWAN port 10** UL data channels, data types, and data encoding definitions, refer to Table 0-1 in Appendix 1. These frame format details are also included in the corresponding function-specific sections: §6.4.2 (MCU data), §6.5.2 (battery data), §8.2.2 (accelerometer vector), and §8.3.2 (accelerometer alarms).

See Figure 0-2 and Figure 0-3 in Appendix 1 for a summary of the **LoRaWAN port 25** frame formats.

4.2.2 Responses to Commands (LoRaWAN Port 100 and 101 ULs)

The sensor sends ULs in response to any DLs received which contain configuration and control commands¹⁰. These UL responses include 2 types:

- **Read response:** returning the value of a configuration register in response to a query from a DL read command block.
- **Write/Error response:** Returning an acknowledgement after a successful reconfiguration of a register(s) through a DL write command block, or an error message informing of an unsuccessful reconfiguration.

¹⁰ See §4.3.2 for a description of configuration and control commands.

The form that these response ULs take depends on which 1 of 2 *command response options* is enabled:

1. **Legacy LoRaWAN port 100 response option.**
2. **Enhanced LoRaWAN port 100/101 response option.**

The response behaviours for both options and both types are explained in the following subsections and summarized in Table 4-2.

Table 4-2: Command Response Option Behaviour

Response Type	Command Response Option	
	<u>Legacy (default)</u> <i>LoRaWAN Port 100 only</i>	<u>Enhanced</u> <i>LoRaWAN Port 100 and 101</i>
Read Response	Stream: LoRaWAN port 100 Contents: Register addresses and values requested in DL command Payload Format: 1 or more frames shown in Figure 4-1	
Write/Error Response	Stream: LoRaWAN port 100 Contents: CRC32 of DL command Payload Format: 4-byte CRC	Stream: LoRaWAN port 101 Contents: Command success/fail status Payload Format: Figure 4-3

The legacy option is enabled by default. To switch options, see the information in §6.2.

NOTE: The enhanced option offers improvements compared to the legacy option such as:

- More information is transmitted in the cases where write commands failed to execute. The CRC32 response is only an acknowledgement, whereas the enhanced option payloads specify success or failure, and in the case of failure the registers which failed to write are identified.
- Decoder tools can more easily handle the enhanced write response payloads since they are on a separate port.
- Decoder tools cannot differentiate the legacy write response 4-byte CRC32 from a legacy read response where the register address is equivalent to the first byte of the CRC32.

Despite these improvements, legacy is the default option to support backwards compatibility for already-fielded use-cases which predate the enhanced option. However, it is strongly recommended that users switch to the enhanced option and update their data decoders to accommodate the changes.

4.2.2.1 Legacy Option

Received command blocks in a DL payload are processed one-at-a-time in order from MSB to LSB.

Read Response: A message UL is sent back on **LoRaWAN port 100** containing the addresses and values of each of the registers under query (this can be in one or more consecutive UL packets depending on the maximum frame payload size allowed at the current data rate [4]). The bit indexing scheme is the same as for configuration registers, which is shown in Figure 4-1.

Write Response: A message UL is sent back on **LoRaWAN port 100** containing a 4-byte CRC32 of the entire DL payload (which may comprise more than one command block). If the DL payload also had read commands, the first 4 bytes of the UL response payload are the CRC32, and the rest consists of the read responses in the format described above.

The CRC is sent regardless of whether the write successfully executed (it will not execute in the cases of trying to access a register address designated as RFU or trying to write an invalid value to an accessible register).

4.2.2.2 Enhanced Option

Received command blocks in a DL payload are processed one-at-a-time in order from MSB to LSB but are generally not executed right away. Only write commands, if valid, are executed at the time they are processed. Other types of commands are queued to execute later.

If all commands were processed successfully, the following happens in order:

1. Any queued Command-and-Control (C&C) operations¹¹ are executed in the following order.
 - a. Save configuration settings to flash.
 - b. Reset configuration settings to factory default.
 - c. Restart sensor.
2. If the sensor did not reset due to C&C operations, any queued BLE mode switch commands¹² are executed.
3. If the sensor did not execute a BLE mode switch command,
 - a. If any read commands were queued, they are executed and a **LoRaWAN port 100 read response** is sent.
 - b. If any write commands were executed, a **LoRaWAN port 101 write response** is sent.

¹¹ C&C operations are defined as commands accessing register 0x 70 or 0x 72 with bit 7 set to 1 (i.e. if a command block begins with 0x F0 or 0x F2). Refer to §6.7 for more details.

¹² Refer to §7.3 for information about switching between BLE modes.

As soon as an invalid command¹³ is processed, the following happens in order:

5. No further command blocks are processed.
6. Any queued BLE mode switch commands are executed.
7. If the sensor did not execute a BLE mode switch command,
 - a. If any read commands were queued, they are executed and a **LoRaWAN port 100 read response** is sent.
 - b. A **LoRaWAN port 101 write/error response** is sent.

Read Response: In the case of a valid read command block, the enhanced behaviour is the same as the read response using the legacy option. That is, a message UL is sent back on **LoRaWAN port 100** containing the addresses and values of each of the registers under query. The bit indexing scheme is the same as for configuration registers, which is shown in Figure 4-1.

If the sensor receives a read command trying to access a register that is designated as RFU, the address is included in the error response as described below.

Write/Error Response: A message UL is sent on **LoRaWAN port 101** with the frame format as shown in Figure 4-3.

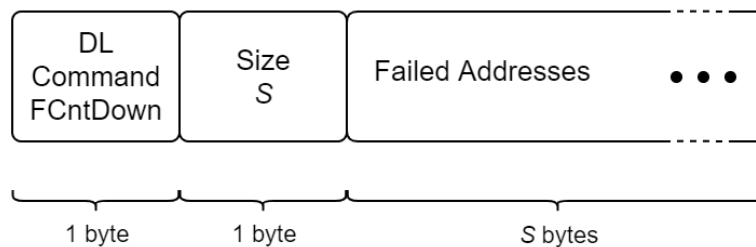


Figure 4-3: The LoRaWAN port 101 Write/Error Response UL Frame Format

The contents of the frame include:

- **DL Command FCntDown:** the last byte (LSB) of the LoRaWAN frame count down number of the DL payload which contained the command block that elicited this response [3].
- **Size:** the number of registers, *S*, that were not successfully written to, and therefore the size of the rest of the payload. *S* can range from 0 to 255.
- **Failed Addresses:** the address(es) of the register(s) where the command(s) failed.

If all commands were successful, *S* = 0x 00 and no failed addresses are included. This includes if a redundant write command was issued (i.e., the value of that register did not change).

¹³ An invalid command is one that either tries to access a register designated as RFU or write an invalid value to an accessible register.

As soon as the sensor encounters an invalid command block (read or write), the address of that command block is added to the **LoRaWAN port 101** response and no further command blocks are processed.

NOTES:

- If anti-bricking is activated, this is considered an unsuccessful command and register address 0x 21 will be added to the **LoRaWAN port 101** response. See §6.3.1.1. for more details about anti-bricking.
- If there were any C&C operations queued to be executed but 1 or more command blocks in the payload were unsuccessful, the C&C commands are not executed, and their addresses are added to the **LoRaWAN port 101** response.
- If the DL payload had a mix of read and write command blocks, the read responses are sent separately on **LoRaWAN port 100** as described above. In this case, the read responses are sent first and the write/error responses after.

4.2.2.3 Example UL Payloads

- **LoRaWAN port 100: 0x 20 00 00 02 58 21 00 01**
 - Register 0x 20 (seconds per core tick)
 - Value read = 0x 00 00 02 58 = 600 s per core tick (10 minutes)
 - Register 0x 21 (ticks per battery report)
 - Value read = 0x 00 01 = 1 tick per battery report
- **LoRaWAN port 100: 0x 81 03 E8 5B**
 - 4B CRC in response to a write command DL
- **LoRaWAN port 101: 0x 0F 00**
 - 0x 0F → Response to write command in DL with FCntDown ending in 15
 - 0x 00 → Size = 0; no failed write commands
- **LoRaWAN port 101: 0x 03 07 15 16 17 18**
 - 0x 0F → Response to write command in DL with FCntDown ending in 3
 - 0x 00 → Size = 4; 4 failed write commands
 - 0x 15 16 17 18 → The write commands attempting to overwrite registers 0x 15, 0x 16, 0x 17, and 0x 18 all failed.

4.3 DL Payload Formats and Configuration Settings

DownLinks (DLs) are LoRaWAN packets sent from the NS to the sensor. They are used for the following purposes in FINCH and ROBIN¹⁴:

1. To read the current configuration settings of the sensor (R).
2. To change the current configuration settings of the sensor (W).
3. To cause the sensor to perform an operation, such as reset or enter DEEP SLEEP (X).

Each DL must be sent on the appropriate downstream port. The DL streams supported by the FINCH/ROBIN SW are shown in Table 4-3.

Table 4-3: DL Information Streams

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

4.3.1 Entering DEEP SLEEP Command (LoRaWAN Port 99 DLs)

Only the ROBIN supports DEEP SLEEP mode.

A DL can be sent to remotely put the ROBIN into DEEP SLEEP mode. The frame payload must contain the single byte **0x 00** and the DL must be sent on **LoRaWAN port 99**.

See §2.1 for more information about DEEP SLEEP.

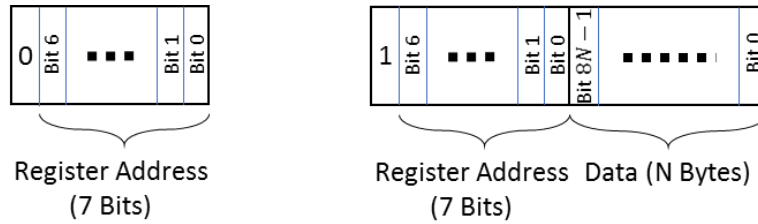
4.3.2 Configuration and Control Commands (LoRaWAN Port 100 DLs)

All DL configuration and control commands are sent on **LoRaWAN port 100**. These commands involve accessing one or more configuration register to perform R, W, or X operations¹⁵.

A single configuration and control DL payload can contain multiple command blocks, with a possible mix of R/W/X commands. Each message block is formatted as shown in Figure 4-4. A big-endian format (MSb/MSB first) is always followed.

¹⁴ Here “DLs” means “SW application-level DLs;” there are other MAC-level DLs that the NS may send to perform LoRaMAC operations. LoRaMAC behaviour occurs according to the LoRaWAN specifications and is outside the scope of this document [9].

¹⁵ See §4.1 for general information about configuration registers.



a) Read Command Block

(b) Write or Executable Command Block

Figure 4-4: The DL Formats for Configuration and Control Message Blocks

Bit 7 of the first byte determines whether a read or write action is being performed, as shown in Figure 4-4.

- **Read commands** are 1-byte where bit 7 is set to 0 and bits 0-6 are the register address being accessed. Data following a read access command will be interpreted as a new command block.
- **Write or executable commands** begin with 1-byte where bit 7 is set to 1 and bits 0-6 are the register address being accessed. The first byte is followed by N bytes with the user-specified value to write to that register. The value N depends on the size of the value attributed to each register.

NOTE: When forming DL payloads with more than 1 command block, sometimes the order of the commands will affect how the sensor responds overall to the whole payload. Refer to §4.2.2 for a description of how the sensor processes, executes, and responds to configuration and control commands.

For a complete summary list of the specific definitions, sizes, and default values of all configuration registers that are accessible by configuration and control commands, see Table 0-1 in Appendix 2. These details are also included in the corresponding function-specific sections:

- LoRaMAC Options | §6.1.2
- Command Response Options | §6.2.2
- Periodic Report Scheduling | §6.3.2
- MCU Data Reporting | §6.4.3
- Battery Management | §6.5.3
- Function Button | §6.6.2
- General Command-and-control Operations | §6.7.1
- BLE Tracker Mode | §7.1.3
- BLE Beacon Mode | §7.2.2
- Toggling BLE Modes | §7.3.1
- Accelerometer | §8

4.3.2.1 Example DL Payloads

- Read registers 0x 10, 0x 11, and 0x 12:
 - DL payload: **0x 10 11 12**
 - Register 0x 10 with bit 7 set to 0 = 0x 10
 - Register 0x 11 with bit 7 set to 0 = 0x 11
 - Register 0x 12 with bit 7 set to 0 = 0x 12
- Read register 0x 13 and write value 0x 80 00 to register 0x 10:
 - DL payload: **0x 13 90 80 00**
 - Register 0x 13 with bit 7 set to 0 = 0x 13
 - Register 0x 10 with bit 7 set to 1 = 0x 90
 - Value 0x 80 00

5 State Machine Implementation – PANIC vs. NORMAL State

5.1 Operational Description

The FINCH and ROBIN devices support a state machine once they have joined a LoRaWAN network. This state machine allows for the transition between the following two states:

1. PANIC State
2. NORMAL State

Figure 5-1 shows the state machine architecture and the transition options between different states. A summary of the transition abbreviations and their conditions are described in Table 5-1 below.

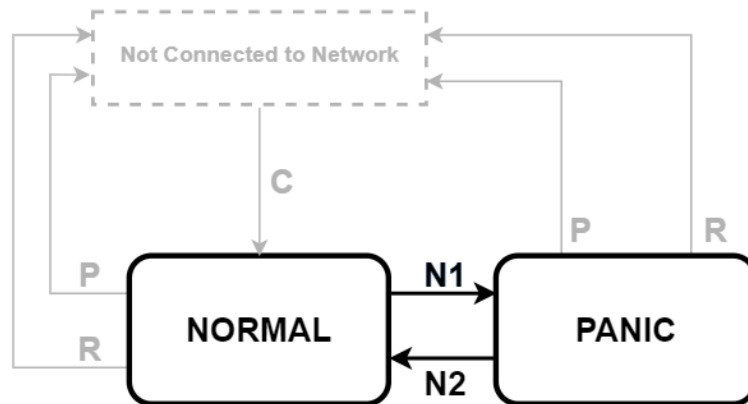


Figure 5-1: FINCH/ROBIN State Machine While Connected to Network (Normal Operation)

Table 5-1: FINCH/ROBIN State Machine Transition Summary

Transition	From	To	Conditions for Triggering	Actions Associated
C	JOIN	NORMAL	Sensor receives a valid JOIN ACCEPT message DL at any time during JOIN.	N/A
N1	NORMAL	PANIC	Any of the enabled triggers (as defined by register 0x 60) occur. With default settings, a N1 transition occurs once: <ul style="list-style-type: none"> Function button pressed and held for 3 s (duration defined by register 0x 2C), OR: DL received containing a Panic App Request payload, 	<ul style="list-style-type: none"> Duty cycle disabled. Panic status values changed appropriately. Device status report UL sent. Tick schedule reset to 0. Buzzer and vibration patterns begin (as

			<ul style="list-style-type: none"> OR: Accelerometer event detected (<i>disabled by default</i>). 	configured by registers 0x 61 to 0x 65)
N2	PANIC	NORMAL	<p>All of the enabled triggers (as defined by register 0x 60) have cleared.</p> <p>With default settings, a N2 transition occurs once:</p> <ul style="list-style-type: none"> Panic button status = 0 (this gets set to 0 by pressing and holding the function button for 5 s), AND: Panic App Request status = 0 (this gets set to 0 by receiving a Normal App Request DL), AND: Accelerometer event cleared (<i>disabled by default</i>). 	<ul style="list-style-type: none"> Duty cycle enabled. Panic status values changed appropriately. Device status report UL sent. Tick schedule reset to 0. Buzzer and vibration patterns stop
P	Any state	POWER OFF	Power is removed from the sensor during any state.	N/A
R	Any state	STARTUP	The MCU is reset without losing power to the board during any state.	N/A

Upon both N1 and N2 transitions, the sensor always does the following general actions:

- Sends a device status report message (see definition in *Uplink Message Format* section below)
- Conducts a buzzer/vibration feedback pattern (if enabled using registers 0x 60 to 0x 65)
- Conducts a single (periodic) BLE scan and report UL

The N1 and N2 state transitions are controlled by three possible triggers: a panic button press, a panic app request (downlink command), or accelerometer motion detection. Each trigger can be independently configured to be either considered or ignored for state transitions, allowing any combination of triggers to be enabled. If no triggers are enabled, state transitions are disabled and the sensor will remain in its current state. All trigger configurations are managed through the *Panic State Trigger Options* register (0x60).

At a high-level, the following statements summarize the conditions for N1 and N2 transitions:

- N1 transition (NORMAL → PANIC) occurs as soon as ANY of the triggers enabled by register 0x 60 are registered.

- N2 transition (PANIC → NORMAL) occurs as soon as ALL of the triggers enabled by register 0x 60 are cleared.

The triggers are registered in SW by setting and clearing the status flags as defined as follows:

- **Button Status = 0/1**
 - = 0 by default upon join
 - Set to = 1 as soon as function button is pressed and held for the current value of *Panic Press Pattern Configuration* (register 0x 2C)
 - Set to = 0 as soon as function button is pressed and held for 5 s or sensor is reset
 - For more information, please refer to §6.6
- **Panic App Request Status = 0/1**
 - = 0 by default upon join
 - Set to = 1 as soon as Panic App Request (EAR) DL message is received
 - Set to = 0 as soon as Normal App Request (NAR) DL message is received or sensor is reset
 - For more information, please refer to §5.4
- **Accelerometer Status = 0/1**
 - = 0 by default upon join
 - Set to = 1 as soon as an accelerometer event is registered (motion is detected)
 - Set to = 0 as soon as an accelerometer clear is registered (motion is cleared) or sensor is reset
 - For more information, please refer to §8

The SW must keep track of these statuses at all times. Depending on the configuration of 0x 60 and the current state, the values of these status flags will determine when an N1 or N2 transition should occur. This is described in detail in the following corresponding subsections.

In addition to the 3 status flags, the SW also keeps track of a 4th flag which simply indicates the current state:

- **Panic Status = 0/1**
 - Set to = 0 upon starting NORMAL
 - Set to = 1 upon starting PANIC

5.2 NORMAL State Operation

The NORMAL state is the sensor's default operating mode once it has successfully joined the LoRaWAN network. In this state, the device functions according to its standard reporting and sensing configuration.

Key characteristics of NORMAL state:

- BLE scan interval: Conducted and reported at the interval defined by Register 0x25 (Ticks per BLE Location Update in NORMAL state).
- Sensor activity: Periodic, interrupt-driven, and background sampling occur according to the current configured settings.
- Trigger monitoring: Panic triggers (button, app request, accelerometer) are actively monitored but not active.
- Feedback: No buzzer or vibration patterns are active.

Figure 5-2 below illustrates which processes occur and in which order when the device is beginning an N1 transition. At any stage, P or R transitions can be triggered.

This N1 transition (NORMAL → PANIC) occurs as soon as ANY of the triggers enabled by register 0x 60 are registered.

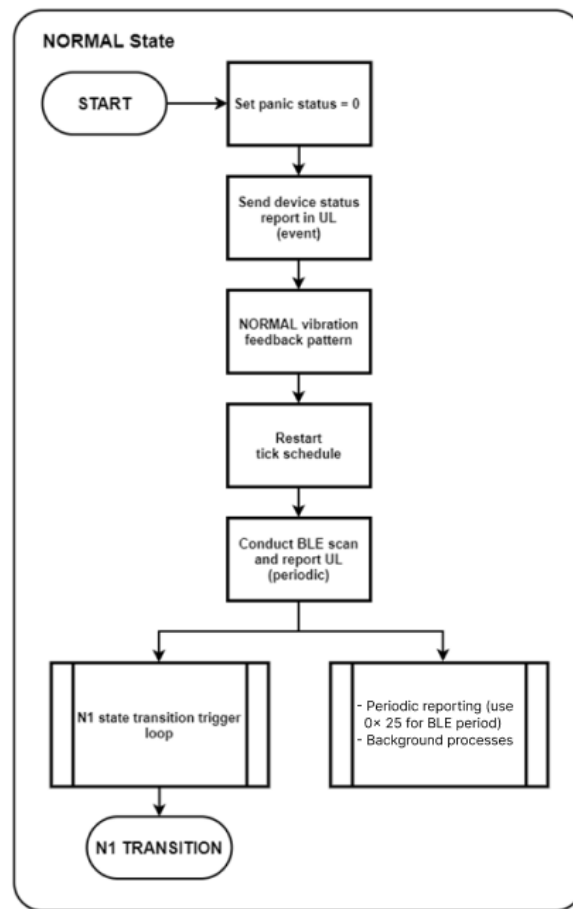


Figure 5-2: NORMAL Process Flow

5.3 PANIC State Operation

The PANIC state represents an emergency condition in which the device prioritizes accelerated BLE location update reporting. The device remains in this state until all enabled panic triggers are cleared.

Key characteristics of PANIC state:

- BLE scans: Conducted and reported at the interval defined by Register 0x23 (Ticks per BLE Location Update in PANIC state).
- Sensor activity: Periodic, interrupt-driven, and background sampling continue according to the configured settings.
- User feedback: Buzzer and/or vibration patterns are activated (if enabled through Registers 0x60–0x65).
- Persistent condition: The device will remain in PANIC until all enabled triggers have been cleared.

The sensor can only enter PANIC via an N1 transition. The flowchart below illustrates which processes occur and in which order following an N1 transition. At any stage, P or R transitions can be triggered.

The N2 transition occurs as soon as ALL of the triggers enabled by register 0x 60 are cleared.

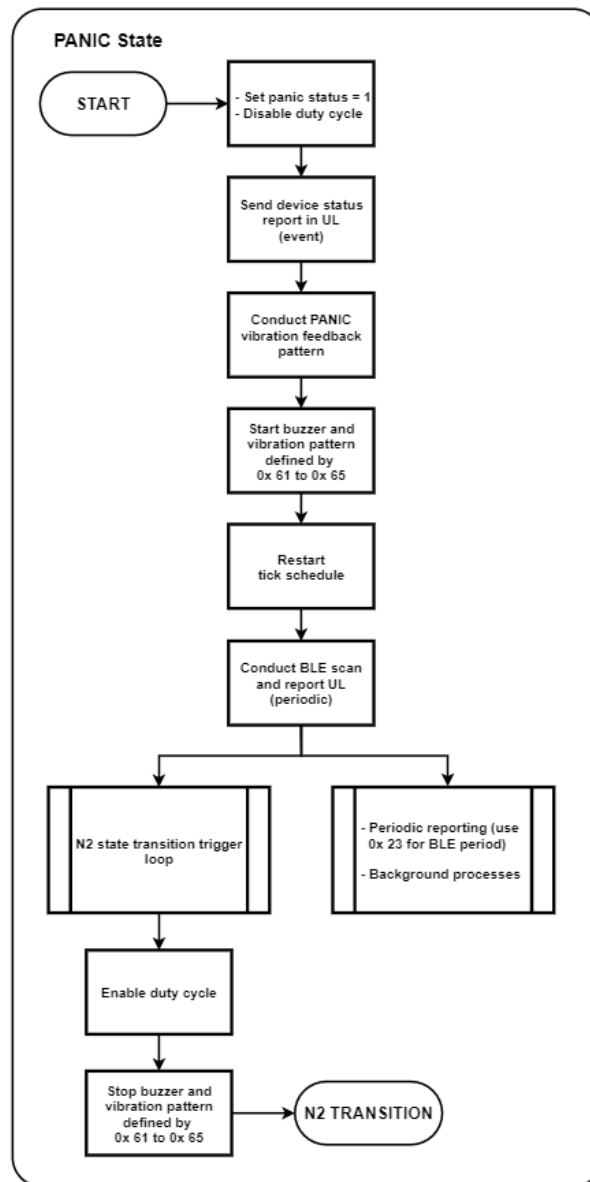


Figure 5-3: PANIC Process Flow

5.4 State Change Request DLs (PARs and NARs)

FINCH/ROBIN supports the ability for DL commands to influence sensor status and behaviour. This allows an application or network administrator to remotely indicate a state of panic to the sensor.

An example use-case would be a building security system that queues up a *state change request* DL upon detection of a breach. Security guards wearing FINCHes or ROBINs would hear and/or feel their sensors enter PANIC mode, and a system administrator would be able to track their locations more frequently.

There are 2 types of state change request DLs:

- **Panic App Request (PAR):** A payload indicating a PANIC state. When a sensor receives this command, the PAR status is set to 1. If the PAR trigger is enabled on register 0x 60, this will result in a transition to PANIC (if currently in NORMAL).
- **Normal App Request (NAR):** A payload indicating there is no longer a state of panic. When a sensor receives this command, the PAR status is set to 0. If the PAR trigger is enabled on register 0x 60, and all other enabled triggers are clear, this will result in a transition to NORMAL (if currently in PANIC).

State change request DLs are sent on **LoRaWAN port 10**. Their formats and actions are summarized in Table 5-2 below.

Table 5-2: Request Type DLs

Request Type	Frame Payload	Action
PAR	0x 00 01 FF	Set PAR status = 1
NAR	0x 00 01 00	Set PAR status = 0

5.5 UL Report Frame Format

The device status message is reportable in a sensor data report UL on **LoRaWAN port 10**. This message is formatted like all other sensor data report ULs with the bitmapping and headers defined in the below table.

This message is sent every time the sensor transitions from NORMAL to PANIC or vice versa (N1 or N2 transitions). This is non-configurable.

By default, this message is also sent every time the user presses the button to force a UL (button press type 1). This behaviour is configurable using register 0x 2A.

Please see for more information on BLE scan uplink format details. As by default, the device will send a BLE scan report every time a new state begins, and occurs periodically after that according to the periods configurable by register 0x 23 and 0x 25.

Table 5-3: Device Status Data Report UL Frame Format Details

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Device Status	0x 02	0x 95	1 B	Bitmap Input	<ul style="list-style-type: none"> Bit 0: Button status 0/1 = Panic Button currently not activated/activated Bit 3: PAR status 0/1 = Panic App Request currently inactive/active Bit 5: Accelerometer status 0/1 = Accelerometer currently detects stillness/motion Bit 7: Panic status 0/1 = Currently in NORMAL/PANIC state Bits 1-2,4,6: RFU 	<pre>device_status: { button_status: <value>, (string/no unit) par_status: <value>, (string/no unit) accl_status: <value> (string/no unit) current_state: <value> (string/no unit) }</pre>

5.6 Configuration Settings

All configuration registers that control the device status reporting behaviour are listed in Table 6-6. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 20	Seconds per Core Tick	4 B	Access R/W <ul style="list-style-type: none"> Tick value for periodic events, T Acceptable values: 15, 16, 17, ..., 86400 $B \times T \leq 86\,400$, otherwise invalid¹⁶ Other values: Invalid 	60 seconds 0x 00 00 00 3C	<pre>seconds_per_core_tick: <value> (unsigned/s)</pre>

¹⁶ This restriction is due to the anti-bricking strategy. See §6.3.1.1.

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 23	Ticks per BLE Location Update in PANIC State	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between BLE location update in PANIC state • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic BLE reports 	1 tick = 1-minute period 0x 00 01	<i>ticks_per_ble_panic_state: <value> (unsigned/no unit)</i>
0x 25	Ticks per BLE Location Update in NORMAL State	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between BLE location update in NORMAL state • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic BLE reports 	60 ticks = 1-hour period 0x 00 3C	<i>ticks_per_ble_normal_state: <value> (unsigned/no unit)</i>
0x 2A	Data Press Report Configuration	2 B	Access: R/W <ul style="list-style-type: none"> • Bit 0: 0/1 = Battery report disabled/enabled • Bit 1: 0/1 = Acceleration vector report disabled/enabled • Bit 2: 0/1 = MCU temperature report disabled/enabled • Bit 3: 0/1 = BLE location update disabled/enabled • Bit 3 set to 1 is unavailable in Beacon Mode • Bit 4: 0/1 = Device status report disabled/enabled • Bits 5-15: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • BLE report enabled • Device status report enabled 0x 00 18	<i>data_press_report_options: {</i> <i>battery_report: <value>, (unsigned/no unit)</i> <i>accl_vector: <value>, (unsigned/no unit)</i> <i>mcu_temp: <value>, (unsigned/no unit)</i> <i>ble_scan: <value>, (unsigned/no unit)</i> <i>device_status: <value> (unsigned/no unit),</i> <i>}</i>

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 2B	Data Press Pattern Configuration	1 B	Access: R/W <ul style="list-style-type: none"> Bits 0-3: Value of m [1 press/LSb] Acceptable values: 1, 2, 3, ..., 15, as long as $m \leq 2n$ 0: Invalid Bits 4-7: Value of n [1 s/LSb] Acceptable values: 1, 2, 3, ..., 15, as long as $m \leq 2n$ 0: Invalid 	<ul style="list-style-type: none"> $m = 1$ press $n = 1$ s 0x 11	<i>data_press_pattern:</i> { <i>m_value:</i> <value>, <i>(unsigned/presses)</i> <i>n_value:</i> <value>, <i>(unsigned/s)</i> }
0x 2C	Panic Press Pattern Configuration	1 B	Access: R/W <ul style="list-style-type: none"> Duration of press and hold to trigger transition to PANIC [1 s/LSb] Acceptable values: 1, 2, 3, 4 Other values: Invalid 	4 s 0x 04	<i>panic_press_pattern:</i> <value> <i>(unsigned/s)</i>
0x 42	Acceleration Trigger Threshold Count	2 B	Access: R/W <ul style="list-style-type: none"> Number of above-threshold acceleration samples before an acceleration trigger is registered Acceptable values: ≥ 1 0: Invalid 	1 trigger 0x 00 01	<i>accl_trigger_threshold_count:</i> <value> <i>(unsigned/no unit)</i>
0x 43	Acceleration Trigger Threshold Period	2 B	Access: R/W <ul style="list-style-type: none"> Period over which acceleration samples are counted for trigger detection [1s/LSb] Acceptable values: ≥ 5 0-4: Invalid 	10 seconds 0x 00 0A	<i>accl_trigger_threshold_period:</i> <value> <i>(unsigned/s)</i>
0x 44	Acceleration Trigger Threshold	2 B	Access: R/W <ul style="list-style-type: none"> Acceleration magnitude threshold for acceleration trigger [1 mg/LSb] Acceptable values: 200, 201, 202, ..., 16 000 Other values: Invalid 	2 g 0x 07 D0	<i>accl_trigger_threshold:</i> <value> <i>(unsigned/mg)</i>

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 45	Acceleration Event Grace Period	2 B	Access: R/W <ul style="list-style-type: none"> Time to pass after the last accelerometer trigger before the event is cleared [1s/LSb] Acceptable values: ≥ 15 0-14: Invalid 	5 min 0x 01 2C	<i>accl_event_grace_period: <value> (unsigned/s)</i>
0x 60	Panic State Trigger Options	1 B	Access: R/W <ul style="list-style-type: none"> Bit 0: 0/1 = Panic Button Trigger disabled/enabled Bit 3: 0/1 = Panic App Request Trigger disabled/enabled Bit 5: 0/1 = Accelerometer Trigger disabled/enabled Bits 0, 3, and 5 all set to 0: State transitions disabled Bits 1-2, 4, 6-7: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> Panic button trigger enabled Panic App Request trigger enabled 0x 00 09	<i>state_trigger_opts: { button_trigger: <value>, (unsigned/no unit) par_trigger: <value>, (unsigned/no unit) accl_trigger: <value> (unsigned/no unit) }</i>

6 Basic Operation

The basic functionality of the FINCH and ROBIN can be broken down into the following categories:

- **LoRaMAC Options:** LoRaWAN general parameters and behaviour as defined by the LoRaWAN Specifications [3].
- **Command Response Options:** Response to configuration and control command behaviour.
- **Periodic Report Scheduling:** Scheme for scheduling regular sensor data reports.
- **MCU Data Reporting:** Reporting the MCU temperature.
- **Battery Management:** Keeping track of consumed battery charge.
- **Function Button Operation:** How to configure and use the function button.
- **General Command-and-control Operations:** Reading SW metadata, saving configuration settings, resetting to factory default, and sensor restart.

In the following subsections, the operational descriptions, report formats, and configurable settings for each category are explained.

6.1 LoRaMAC Options

6.1.1 Operational Description

The LoRaMAC options control certain LoRaWAN-specified MAC configuration parameters that the sensor loads on start-up and uses during run-time. The definitions for these parameters are stipulated by the LoRaWAN Specifications and Regional Parameters [3], [4]. Refer to these sources for detailed descriptions of these parameters and expected behaviour, as this is outside the scope of this TRM.

6.1.2 Configuration Settings

Table 6-1 shows the MAC configuration registers. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 6-1: LoRaMAC Options Configuration Registers

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 10	Join Mode	2 B	Access: R/W <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: <value></i> (unsigned/no unit)
0x 11	Options	2 B	Access: R/W (bit 1 is RO) <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 = Duty Cycle disabled/enabled¹⁷ Bit 3: 0/1 = Disable/Enable ADR Bits 4-15: Ignored 	<ul style="list-style-type: none"> Unconfirmed UL Public Sync Word Duty cycle enabled¹⁷ ADR enabled 0x 00 0E	<i>loramac_opts: {</i> <i>confirm_mode: <value></i> , <i>(unsigned/no unit)</i> <i>sync_word: <value></i> , <i>(unsigned/no unit)</i> <i>duty_cycle: <value></i> , <i>(unsigned/no unit)</i> <i>adr: <value></i> <i>(unsigned/no unit)</i> <i>}</i>

¹⁷ **WARNING:** Disabling the duty cycle in certain regions makes the sensor non-compliant with the LoRaWAN Specifications [9]. It is recommended that the duty cycle remains enabled. In the LoRa RF regions where there is no duty cycle limitation, the “enabled duty cycle” configuration is ignored.

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 12	DR and Tx Power ¹⁸	2 B	Access: R/W • Bits 8-11: Default DR number • Bits 0-3: Default Tx power number • Bits 4-7, 12-15: Ignored	• DR4 • Tx Power 0 (max power; see Table 6-2) 0x 04 00	<i>loramac_dr_tx</i> : { <i>dr_number</i> : <value>, (unsigned/no unit) <i>tx_power_number</i> : <value> (unsigned/no unit) }
0x 13	Rx2 Window	5 B	Access: R/W • Bits 8-39: Channel frequency in Hz for Rx2 • Bits 0-7: DR for Rx2	As per Table 6-2	<i>loramac_rx2</i> : { <i>frequency</i> : <value>, (unsigned/Hz) <i>dr_number</i> : <value> (unsigned/no unit) }

Table 6-2: Default Max Tx Power, Rx2 Channel Frequency, and Rx 2 DR Number by Regions

RF Region	Max Tx EIRP ¹⁹ [dBm]	Channel Frequency for Rx2 Window [Hz]	DR Number for Rx2 Window
EU868	16	869525000	0
US915	30	923300000	8
AS923	16	923200000	2
AU915	30	923300000	8
IN865	30	866550000	2
KR920	14	921900000	0
RU864	16	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.

¹⁸ Tx power number m translates to the maximum Tx power, which is a function of the LoRaWAN RF region, minus $2 \times m$ dB [8].

¹⁹ These are Tx power *setpoints* and are the maximum allowable by the LoRaWAN regional parameters specification [8]. The actual Tx power of the sensor is limited by the radio transceiver, which is 15 dBm. Therefore, a setpoint greater than 15 dBm will be equivalent to a setpoint of 15 dBm or the Max Tx EIRP for the region, whichever is lower.

6.1.2.1 Example DL Payloads

- Switch Device to ABP Mode:
 - DL payload: **0x 90 00 00**
 - Register 0x 10 with bit 7 set to 1 = 0x 90
 - Desired option: register value with bit 15 set to 0 = 0x 00 00
- Disable ADR, keep Duty Cycle enabled, and use confirmed ULs:
 - DL payload: **0x 91 00 07**
 - Register 0x 11 with bit 7 set to 1 = 0x 91
 - Desired options: register value with bit 3 set to 0, bit 2 set to 1, bit 1 set to 1, and bit 0 set to 1 = 0x 00 07
- Set default DR number to 3, default Tx power number to 4, and read the current Rx2 settings:
 - DL payload: **0x 92 03 04 13**
 - Register 0x 12 with bit 7 set to 0 = 0x 92
 - DR3 = 0x 03
 - Tx 4 = 0x 04
 - Register 0x 13 with bit 7 set to 0 = 0x 13

6.2 Command Response Options

6.2.1 Operational Description

The sensor sends ULs in response to any DLs received which contain configuration and control commands²⁰. The response payload formats and **LoRaWAN ports** differ depending on which 1 of 2 *command response options* is enabled:

1. **Legacy LoRaWAN port 100 response option** (default).
2. **New LoRaWAN port 100/101 response option**.

The details for both options are considered general sensor LoRaWAN reporting behaviour and are described in §4.2.2.

6.2.2 Configuration Settings

Table 6-3 shows the register that determines which command response option is enabled. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 6-3: Command Response Option Configuration Register

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 6F	Command Response Option	1 B	Access: R/W • Bit 0: 0: Enhanced LoRaWAN port 100/101 1: Legacy LoRaWAN port 100 • Bits 1-7: Ignored	Legacy LoRaWAN port 100 option 0x 01	<i>command_response_opt: <value> (string/no unit)</i>

NOTE: Although the legacy format is enabled by default, the new option offers improvements in behaviour (see §4.2.2). It is recommended that users switch to the new option.

6.2.2.1 Example DL Payload

- Switch command response option to new **LoRaWAN port 100/101**:
 - DL payload: **0x EF 00**
 - Register 0x 6F with bit 7 set to 1 = 0x EF
 - Value bit 0 set to 0 = 0x 00

²⁰ See §4.3.2 for a description of configuration and control commands.

6.3 Periodic Report Scheduling

6.3.1 Operational Description

All periodic reporting of sensor data 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 between data transmissions is configurable. These reporting periods are defined by the following equation:

$$\text{<Data Type> Reporting Period} = \text{Seconds per Core Tick} \times \text{Ticks per <Data Type>}$$

The available options for periodically reported data types are listed below. That is, <Data Type> can be:

- **Battery:** Remaining capacity [%], remaining lifetime [days], or any combination. See §6.5 for battery management details.
- **Accelerometer:** Acceleration vector [*g*]. See §8 for accelerometer operation details.
- **BLE Scan:** Discovered BLE device MAC addresses and RSSIs [dBm]. See §7.1 for BLE scanning and reporting details.
- **MCU Temperature:** Temperature of the MCU [°C]. See §6.4 for MCU data reporting details.

Seconds per Core Tick cannot be set to 0; periodic transmissions cannot be globally disabled.

If <Data Type> *Reporting Period* equals 0, it means that periodic reporting is disabled for that data type. Since *Seconds per Core Tick* cannot be set to 0, the above equation can only equal 0 when *Ticks per <Data Type>* is equal to 0. Therefore, to disable the periodic reporting of a specific data type, set its *Ticks per <Data Type>* to 0. *Ticks per Battery* is the exception that cannot be set to 0 (see §6.3.1.1).

The default reporting behaviour is 1 battery report per day and 1 BLE scan report per hour.

NOTE 1: These settings only control the scheduling of reporting data, not *what* is reported; the format and/or content of the reported payloads may depend on other configuration settings. Additionally, the periodic report scheduling settings only affect *periodic* reporting behaviour and do not affect *event-based* reporting behaviour. To configure behaviour not related to the scheduling of reports, refer to the relevant sections for the subsystem or transducer being used.

NOTE 2: For best results it is not recommended to set the BLE scan report period to less than 60 s. Refer to §7.1.3.4 for best practices regarding BLE reporting period configuration.

NOTE 3: The first periodic report for every enabled report type occurs right after the sensor successfully joins the network. That is, tick 1 occurs right after successful join. A consequence of this is that, using the default battery reporting configuration as an example, the first battery report will occur immediately after join but the next one will occur 23 hours later (every one thereafter will occur at the expected 24-hour intervals).

6.3.1.1 Anti-Bricking Strategy

As a Class-A LoRaWAN end-device, the sensor can only be receptive to a DL in the short period after sending an UL. Therefore, if the sensor is configured to send periodic ULs very infrequently or not at all, it could become impossible to send a DL command. A sensor in a “stranded” state like this is referred to as *bricked*.

FINCH and ROBIN both have at least 1 way to manually force a UL (function button, magnet sensor, etc.) which cannot be disabled, so it is impossible to *completely* brick a sensor. 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 many devices at once. In these use-cases, a strategy to avoid bricking the sensor is beneficial and therefore included as a SW feature.

The anti-bricking strategy is summarized by the following statements:

1. *Periodic reporting can never be completely disabled.*
2. *The Battery Reporting Period cannot be set to a value greater than 1 day.*

These statements are accomplished through the use of restricting acceptable values on the tick registers. Specifically:

1. Register 0x 20: Seconds per Core Tick cannot be set to 0.

This ensures that all periodic reporting cannot be disabled at once.

2. The equation in §6.3.1 must be nonzero and less than or equal to 1 day (86 400 s) for the battery report. This ensures that at a minimum, the sensor will send a battery report UL once per week. That is, the following must be true:

$$\text{Battery Reporting Period} = \text{Seconds per Core Tick} \times \text{Ticks per Battery}$$

$$0 < \text{Battery Reporting Period} \leq 1 \text{ day}$$

$$0 < \text{Battery Reporting Period} \leq 86\,400 \text{ s}$$

If the SW detects that a configuration has been set which does not satisfy the above condition, the *Ticks per Battery* is automatically set to $\left\lfloor \frac{86\,400\text{ s}}{\text{Seconds per Core Tick}} \right\rfloor$.

6.3.2 Configuration Settings

Table 6-4 lists the registers used to configure the periodic reporting periods. *Seconds per Core Tick* is configured using register 0x 20, and the *Ticks per <Data Type>* are configured using registers 0x 21 through 0x 28. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 6-4: Periodic Transmission Configuration Registers

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 20	Seconds per Core Tick	4 B	Access R/W <ul style="list-style-type: none"> • Tick value for periodic events, T • Acceptable values: 15, 16, 17, ..., 86400 • $B \times T \leq 86\,400$, otherwise invalid²¹ • Other values: Invalid 	60 seconds 0x 00 00 00 3C	<i>seconds_per_core_tick: <value></i> (unsigned/s)
0x 21	Ticks per Battery	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between battery reports, B • Acceptable values: 1, 2, 3, ..., 5760 • $B \times T \leq 86400$, otherwise invalid²¹ • Other values: Invalid 	1440 ticks = 1-day period 0x 05 A0	<i>ticks_per_battery: <value></i> (unsigned/no unit)
0x 23	Ticks per BLE Location Update in PANIC State	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between BLE location update in PANIC state • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic BLE reports 	1 tick = 1-minute period 0x 00 01	<i>ticks_per_ble_panic_state: <value></i> (unsigned/no unit)

²¹ This restriction is due to the anti-bricking strategy. See §6.3.1.1.

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 24	Ticks per Accelerometer	2 B	Access R/W • 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> <i><value></i> <i>(unsigned/no unit)</i>
0x 25	Ticks per BLE Location Update in NORMAL State	2 B	Access R/W • Ticks between BLE location update in NORMAL state • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic BLE reports	60 ticks = 1-hour period 0x 00 3C	<i>ticks_per_ble_normal_state:</i> <i><value></i> <i>(unsigned/no unit)</i>
0x 28	Ticks per MCU Temperature	2 B	Access R/W • Ticks between temperature reports • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic MCU temperature reports	Periodic reporting disabled 0x 00 00	<i>ticks_per_mcu_temp:</i> <i><value></i> <i>(unsigned/no unit)</i>

6.3.2.1 Example DL Payloads

- Read current value of Seconds per Core Tick:
 - DL payload: **0x 20**
 - Register 0x 20 with bit 7 set to 0 = 0x 20
- Change settings to report BLE scan results every 15 minutes and MCU temperature every hour:
 - DL payload: **0x A0 00 00 01 2C A5 00 01 A8 00 04**
 - Register 0x 20 with bit 7 set to 1 = 0x A0
 - Seconds per core tick = 300 s = 0x 00 00 01 2C
 - Register 0x 25 with bit 7 set to 1 = 0x A5
 - Ticks per BLE Location Update in NORMAL State = 1 tick = 0x 00 01
 - Register 0x 28 with bit 7 set to 1 = 0x A8
 - Ticks per MCU Temperature = 4 ticks = 0x 00 04

6.4 MCU Data Reporting

6.4.1 Operational Description

The MCU has a built-in thermometer that can be polled for a temperature reading. This temperature represents the *on-board temperature*, not the ambient temperature of the surrounding environment.

The sensor can be configured to report MCU temperature periodically and/or on demand by pressing the function button. By default, the MCU temperature is not reported periodically nor by function button press. The UL report format is always as described in §6.4.2, regardless of report type.

6.4.2 UL Report Frame Format

MCU temperature reports are sent on **LoRaWAN port 10** and have the frame format as shown in Figure 4-2. The specific details for the MCU temperature frame format are listed in Table 6-5. For the general description of sensor data report formats and behaviour, see §4.2.1.

Table 6-5: MCU Data Report UL Frame Format Details

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
MCU Temperature	0x 00	0x 67	2 B	Temperature	0.1°C/LSb (signed)	<i>mcu_temperature</i> (signed/°C)

6.4.2.1 Example UL Payloads

- **0x 00 67 00 EB**
 - Channel ID = 0x 00, Type ID = 0x 67 → MCU temperature data report
 - 0x 00 EB = $235 \times 0.1^{\circ}\text{C} = 23.5^{\circ}\text{C}$
- **0x 00 67 00 FE A1**
 - Channel ID = 0x 00, Type ID = 0x67 → MCU temperature data report
 - 0x FE A1 = $-351 \times 0.1^{\circ}\text{C} = -35.1^{\circ}\text{C}$

6.4.3 Configuration Settings

All configuration registers that control MCU reporting behaviour are listed in Table 6-6. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 6-6: MCU Data Reporting Configuration Settings

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 20	Seconds per Core Tick	4 B	Access R/W <ul style="list-style-type: none"> • Tick value for periodic events, T • Acceptable values: 15, 16, 17, ..., 86400 • $B \times T \leq 86\,400$, otherwise invalid²² • Other values: Invalid 	60 seconds 0x 00 00 00 3C	<i>seconds_per_core_tick: <value></i> (unsigned/s)
0x 28	Ticks per MCU Temperature	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between temperature reports • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic MCU temperature reports 	Periodic reporting disabled 0x 00 00	<i>ticks_per_mcu_temp: <value></i> (unsigned/no unit)
0x 2A	Data Press Report Configuration	2 B	Access: R/W <ul style="list-style-type: none"> • Bit 0: 0/1 = Battery report disabled/enabled • Bit 1: 0/1 = Acceleration vector report disabled/enabled • Bit 2: 0/1 = MCU temperature report disabled/enabled • Bit 3: 0/1 = BLE location update disabled/enabled • Bit 3 set to 1 is unavailable in Beacon Mode • Bit 4: 0/1 = Device status report disabled/enabled • Bits 5-15: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • BLE report enabled • Device status report enabled 0x 00 18	<i>data_press_report_options: {</i> <i>battery_report: <value>, (unsigned/no unit)</i> <i>accl_vector: <value>, (unsigned/no unit)</i> <i>mcu_temp: <value>, (unsigned/no unit)</i> <i>ble_scan: <value>, (unsigned/no unit)</i> <i>device_status: <value> (unsigned/no unit),</i> <i>}</i>

²² This restriction is due to the anti-bricking strategy. See §6.3.1.1.

6.4.3.1 Periodic Reporting Configuration

The MCU temperature report period can be configured using registers 0x 20 and 0x 28 according to the details described in §6.3.2. That is:

$$\text{MCU Temperature Reporting Period} = \text{Seconds per Core Tick} \times \text{Ticks per MCU Temperature}$$

Setting the MCU temperature report period to 0 (default) disables periodic reporting of the MCU temperature.

6.4.3.2 Event-Based Reporting Configuration

The MCU temperature can be configured to be reported upon a function button press by setting bit 2 to 1 in the *Function Button Mode* register, 0x 2A. For details about how the function button works and is configured, see §6.6.

6.4.3.3 Example DL Payloads

- Report MCU temperature every 60 min:
 - DL payload: **0x A0 00 00 0E 10 A8 00 01**
 - Register 0x 20 with bit 7 set to 1 = 0x A0
 - 3600 s/core tick = 0x 00 00 0E 10
 - Register 0x 28 with bit 7 set to 1 = 0x A8
 - Report every 1 tick = 0x 00 01
- Report only MCU temperature upon function button press:
 - DL payload: **0x AA 04**
 - Register 0x 2A with bit 7 set to 1 = 0x AA
 - Value bit 2 set to 1 = 0x 04

6.5 Battery Management

6.5.1 Operational Description

The FINCH and ROBIN have battery management systems that monitor battery energy depletion.

The *remaining battery capacity* is the percentage of battery energy still available to the sensor relative to a fully charged battery. The SW always keeps track of this value, which gradually drops throughout normal operation. The rate at which the capacity drops will depend on configuration; for example, a sensor configured to send a UL report every 15 min will have a larger energy consumption rate than one that is configured to send a UL report every 60 min.

The *remaining battery lifetime* is the estimated number of days remaining before the battery dies. The SW estimates this value, which may fluctuate up and down throughout the normal operation of the sensor. This is because configuration settings can be changed at any time during normal operation, thus changing the energy consumption rate.

The remaining battery lifetime is estimated using the past energy consumption rate trends over a timeframe called the *Average Energy Trend Window*. This is a rolling window of configurable size that contains the most recently collected energy consumption rate data. The window size is defined in units of core ticks and the default value is 10 core ticks. In other words, by default, the remaining battery lifetime is estimated based on energy consumption rates over the last 10 core ticks only, which is visualized in Figure 6-1.

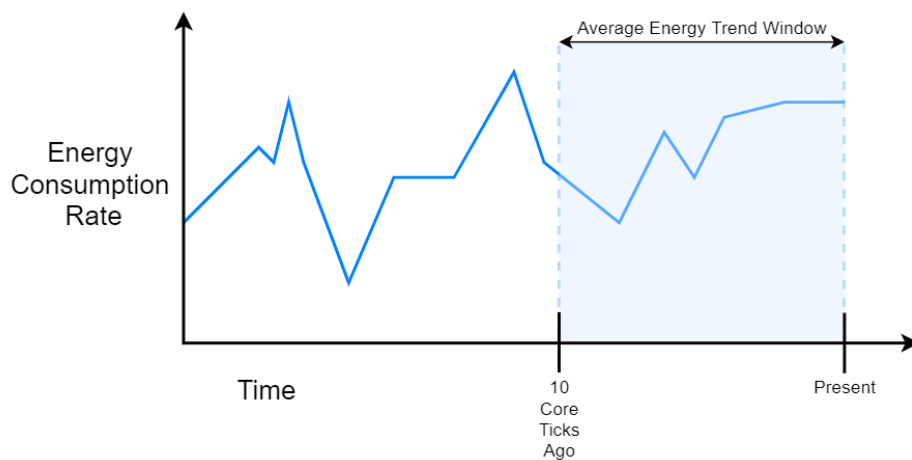


Figure 6-1: Example Visualization of the Default Average Energy Trend Window

The sensor can be configured to send a battery report periodically and/or on demand by pressing the function button. By default, a battery report is sent periodically every 24 hours, and no battery report is sent by function button press.

The data type reported is configurable and can be the remaining battery capacity, remaining battery lifetime, or both. By default, both values are reported. The UL report format is always as described in §6.5.2, regardless of whether it is a periodic or event-based report.

NOTE 1: The JOIN procedure consumes energy at a higher rate than default normal operation, so the remaining lifetime value reported will be skewed for some time after the sensor joins the network. It will take approximately 1 average energy trend window over which “steady state” energy consumption occurs before the remaining battery lifetime values stabilize.

NOTE 2: Battery voltage used to be an optional reportable value for Gen1 and Gen2 devices. Voltage measurement is not supported on Gen2.5 sensors. See Table 1-1 for how to determine generation.

6.5.1.1 Resets and Battery Replacement

The battery management system bases calculations on the average nominal battery capacity of a new battery. When the battery is replaced, the remaining battery capacity and lifetime values are automatically reset to reflect a fully charged battery. Any hard reset (i.e., any complete loss of power to the battery contacts) will result in the battery management system resetting.

Battery management data will not reset when a soft reset occurs (i.e., when a magnet reset, reset button press, OTA reset command, or Tracker/Beacon Mode switch occurs).

6.5.1.2 Battery Passivation

Due to the chemistry of the batteries (LTC), a *passivation layer* can build up internally during periods when the battery has little to no charge flowing out. This layer can prevent high pulse current draws for a few minutes at the time of first-use [5]. As the sensor begins drawing current from the battery, the passivation begins to break down. The longer the dormant state of the battery, the longer it takes to break the passivation layer down.

At the user-level, passivation means that the first time a sensor is woken up or powered on with a new battery, it is possible that there is some ramp-up time required before it can complete the join procedure. Until the device detects that it can get enough current from the battery, it will constantly reset in attempt to break the passivation layer down. If this occurs, the LEDs will go through the normal power-on patterns and begin the join pattern (flashing green) for about 1 s before a reset occurs (i.e., steps 1-4a in §3.3.1).

Some example circumstances which may lead to battery passivation include:

- The battery is replaced with a new one, including new devices from the factory, where the battery may have been unused for more than a month.

- The sensor was in DEEP SLEEP for longer than a month, such as while being in long-term storage or warehouse stock.

6.5.2 UL Report Frame Formats

Battery reports are sent on **LoRaWAN port 10** and have the frame format as shown in Figure 4-2. The specific details for the battery report frame formats are listed in Table 6-7. For the general description of sensor data report formats and behaviour, see §4.2.1.

Table 6-7: Battery Report UL Frame Formats

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Remaining Battery Capacity	0x 00	0x D3	1 B	Percentage	1% / LSb (unsigned)	<i>rem_batt_capacity: <value> (unsigned/%)</i>
Remaining Battery Lifetime	0x 00	0x BD	2 B	Days	1 day / LSb	<i>rem_batt_days: <value> (unsigned/days)</i>

6.5.2.1 Example UL Payloads

- **0x 00 D3 32 00 BD 01 E6**
 - Channel ID = 0x 00, Type ID = 0x D3 → remaining battery capacity data report
 - 0x 32 = $50 \times 1\% = 50\%$
 - Channel ID = 0x 00, Type ID = 0x BD → remaining battery lifetime data report
 - 0x 01 E6 = $486 \times 1 \text{ day} = 486 \text{ days}$

6.5.3 Configuration Settings

All configuration registers that control battery management behaviour are listed in Table 6-8. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 6-8: Battery Management Configuration Registers

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 20	Seconds per Core Tick	4 B	Access R/W <ul style="list-style-type: none"> • Tick value for periodic events, T • Acceptable values: 15, 16, 17, ..., 86400 • $B \times T \leq 86400$, otherwise invalid²³ • Other values: Invalid 	60 seconds 0x 00 00 00 3C	<i>seconds_per_core_tick:</i> <i><value></i> <i>(unsigned/s)</i>
0x 21	Ticks per Battery	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between battery reports, B • Acceptable values: 1, 2, 3, ..., 5760 • $B \times T \leq 86400$, otherwise invalid²³ • Other values: Invalid 	1440 ticks = 1-day period 0x 05 A0	<i>ticks_per_battery:</i> <i><value></i> <i>(unsigned/no unit)</i>
0x 2A	Data Press Report Configuration	2 B	Access: R/W <ul style="list-style-type: none"> • Bit 0: 0/1 = Battery report disabled/enabled • Bit 1: 0/1 = Acceleration vector report disabled/enabled • Bit 2: 0/1 = MCU temperature report disabled/enabled • Bit 3: 0/1 = BLE location update disabled/enabled • Bit 3 set to 1 is unavailable in Beacon Mode • Bit 4: 0/1 = Device status report disabled/enabled • Bits 5-15: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • BLE report enabled • Device status report enabled 0x 00 18	<i>data_press_report_options:</i> { <i>battery_report:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>accl_vector:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>mcu_temp:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>ble_scan:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>device_status:</i> <i><value></i> <i>(unsigned/no unit)</i> , }

²³ This restriction is due to the anti-bricking strategy. See §6.3.1.1.

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 4A	Battery Report Options	1 B	Access: R/W <ul style="list-style-type: none"> • Bit 0: Ignored • 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 • Bits 3-7: Ignored 	Remaining battery capacity [%] and remaining battery lifetime [days] reported 0x 06	<pre> battery_tx: { report_capacity_enabled: <value>, (unsigned/no unit) report_lifetime_enabled: <value> (unsigned/no unit) }</pre>
0x 4B	Average Energy Trend Window	1 B	Access: R/W <ul style="list-style-type: none"> • Window size over which remaining battery lifetime is estimated [1 core tick/LSB] • Acceptable values: 1, 2, 3, ..., 255 • 0: Invalid 	10 core ticks 0x 0A	<pre> avg_energy_trend_window: <value> (unsigned/no unit)</pre>

6.5.3.1 Periodic Reporting Configuration

The battery reporting period can be configured using registers 0x 20 and 0x 21 according to the equation described in §6.3.2. That is:

$$\text{Battery Reporting Period} = \text{Seconds per Core Tick} \times \text{Ticks per Battery}$$

Setting the battery reporting period to 0 disables periodic reporting of battery management data.

6.5.3.2 Event-Based Reporting Configuration

The battery can be configured to be reported upon a function button press by setting bit 0 to 1 in the *Function Button Mode* register, 0x 2A. For details about how the function button works and is configured, see §6.6.

6.5.3.3 Operational Configuration

Register 0x 4A determines what type of data is reported at the time a battery report is due.

The average energy trend window size is configured using register 0x 4B.

6.5.3.4 Guidelines for Best Practice

It is recommended that the average energy trend window be configured in relation to the core tick setting. 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.

The optimal settings may need to be tuned for particular use cases.

6.5.3.5 Example DL Payloads

- Schedule a battery report every 48 hours:
 - DL payload: **0x A0 00 00 0E 10 A1 00 30**
 - Register 0x 20 with bit 7 set to 1 = 0x A0
 - 3600 s/core tick = 0x 00 00 0E 10
 - Register 0x 21 with bit 7 set to 1 = 0x A1
 - Report every 48 ticks = 0x 00 30
- Report battery data and MCU temperature upon function button press:
 - DL payload: **0x AA 05**
 - Register 0x 2A with bit 7 set to 1 = 0x AA
 - Value bit 0 and bit 2 set to 1 = 0x 05
- Only include remaining battery capacity in battery reports:
 - DL payload: **0x CA 02**
 - Register 0x 4A with bit 7 set to 1 = 0x CA
 - Value bit 1 set to 1 = 0x 02

6.6 Function Button Operation

6.6.1 Operational Description

The function button is externally accessible and located on the bottom of the FINCH and ROBIN enclosure as shown in Figure 3-1. It is actuated by pressing until the haptic click feedback is felt.

The button supports four distinct press types: **Data Press**, **Panic Press**, **Normal Press**, and **Device Reset Press**. Each type is defined by a unique press pattern and results in specific actions such as triggering uplink messages, transitioning to or from PANIC state, or performing a soft reset. Patterns and resulting behaviors are configurable through dedicated registers.

Refer to Table 6-9 for a summary of the function button.

Table 6-9: Function Button Behavior

Press Type	Default Function Button Pattern	Result
Data Press	Button is pressed once and released within 1 second	<ul style="list-style-type: none"> While in tracker mode: Perform and report a BLE scan, report the current device status While in beacon mode: Report the current device status
Panic Press	Button is pressed and held for 4 seconds	Device transitions from NORMAL state to PANIC state
Normal Press	Button is pressed and held for 5 seconds	Device transitions from PANIC state to NORMAL state
Device Reset Press	Button is pressed and held for 8 seconds	Device will perform a soft reset

The button cannot be disabled and requires at least one data type to be configured for transmission. Further details are described in the following subsections.

For more information on the transition between NORMAL and PANIC state (N1 and N2 transitions), please refer to §5.

6.6.1.1 Data Press

The purpose of the Data Press button pattern is to force a non-periodic data report transmission. The actuation pattern is defined by:

- The value of the *Data Tx Press Pattern Configuration* register, 0x 2B: press the button m times in n seconds.

- By default, $m = n = 1$, which means as soon as the button is pressed once and released within 1 second, a data press is registered.

Once this pattern is completed, a data report UL is queued for transmission. Data content is defined by the *Data Press Report Configuration* register, 0x 2A. The default configuration for this register is:

- While in tracker mode: Perform and report a BLE scan, report the current device status.
- While in beacon mode: Report the current device status.

After this button press is registered, the message UL must be queued to transmit before any other button presses (of any type) can be registered.

6.6.1.2 *Panic Press*

The purpose of the Panic Press button pattern is to, if enabled according to register 0x 60, transition the device from the NORMAL state to the PANIC state. The actuation pattern is defined by:

- The values of the *Panic Press Pattern Configuration* register, 0x 2C: press registered as soon as the button is pressed and held for (0x 2C) seconds.
- By default, the value is 4 seconds, which means as soon as the button is held for 4 seconds, a panic press is registered.

Once this pattern is completed, the button status is set to 1 and if the button press trigger is enabled on register 0x 60, this type of press will also result in a transition to the PANIC state. While transitioning to the PANIC state, the LEDs, buzzer and vibration motor will perform a pattern as described in §3.3.4 and §3.4.1.

After this button press is registered, the device ignores all further button presses until:

- The button has been released,
- AND the next time the start of either the N1 or N2

6.6.1.3 *Normal Press*

The purpose of the Normal Press button pattern is to transition the device from the PANIC state to the NORMAL state. The actuation pattern is defined by:

- Pressing and holding the button for 5 seconds or more. This value is non-configurable.

Once this pattern is completed, the button status is set to 0 and if all other triggers enabled on register 0x 60 are cleared, this type of press will result in a transition to the NORMAL state. While transitioning to the NORMAL state, the LEDs, buzzer and vibration motor will perform a pattern as described in §3.3.4 and §3.4.1.

After this button press is registered, the device ignores all further button presses until:

- The button has been released,
- AND the next time the start of either the N1 or N2

6.6.1.4 Device Reset Press

The purpose of the Device Reset Press button pattern is to force the device to perform a soft reset. The actuation patter is defined by:

- Pressing and holding the button for 8 seconds. This value is non-configurable.

Note that while holding the button for 8 seconds, the device may begin a N1 or N2 transition indicated by the LEDs, buzzer and vibration motor as described in §3.3.4 and §3.4.1.

6.6.2 Configuration Settings

Table 6-10 shows the list of configuration registers which control function button behaviour. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 6-10: Function Button Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 2A	Data Press Report Configuration	2 B	Access: R/W <ul style="list-style-type: none"> • Bit 0: 0/1 = Battery report disabled/enabled • Bit 1: 0/1 = Acceleration vector report disabled/enabled • Bit 2: 0/1 = MCU temperature report disabled/enabled • Bit 3: 0/1 = BLE location update disabled/enabled • Bit 3 set to 1 is unavailable in Beacon Mode • Bit 4: 0/1 = Device status report disabled/enabled • Bits 5-15: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • BLE report enabled • Device status report enabled 0x 00 18	<pre>data_press_report_options: { battery_report: <value>, (unsigned/no unit) accl_vector: <value>, (unsigned/no unit) mcu_temp: <value>, (unsigned/no unit) ble_scan: <value>, (unsigned/no unit) device_status: <value> (unsigned/no unit), }</pre>

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 2B	Data Press Pattern Configuration	1 B	Access: R/W <ul style="list-style-type: none"> Bits 0-3: Value of m [1 press/LSb] Acceptable values: 1, 2, 3, ..., 15, as long as $m \leq 2n$ 0: Invalid Bits 4-7: Value of n [1 s/LSb] Acceptable values: 1, 2, 3, ..., 15, as long as $m \leq 2n$ 0: Invalid 	<ul style="list-style-type: none"> $m = 1$ press $n = 1$ s 0x 11	<code>data_press_pattern: { m_value: <value>, (unsigned/presses) n_value: <value>, (unsigned/s) }</code>
0x 2C	Panic Press Pattern Configuration	1 B	Access: R/W <ul style="list-style-type: none"> Duration of press and hold to trigger transition to PANIC [1 s/LSb] Acceptable values: 1, 2, 3, 4 Other values: Invalid 	4 s 0x 04	<code>panic_press_pattern: <value> (unsigned/s)</code>
0x 60	Panic State Trigger Options	1 B	Access: R/W <ul style="list-style-type: none"> Bit 0: 0/1 = Panic Button Trigger disabled/enabled Bit 3: 0/1 = Panic App Request Trigger disabled/enabled Bit 5: 0/1 = Accelerometer Trigger disabled/enabled Bits 0, 3, and 5 all set to 0: State transitions disabled Bits 1-2, 4, 6-7: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> Panic button trigger enabled Panic App Request trigger enabled 0x 00 09	<code>state_trigger_opts: { button_trigger: <value>, (unsigned/no unit) par_trigger: <value>, (unsigned/no unit) accl_trigger: <value> (unsigned/no unit) }</code>

6.6.2.1 Operational Configuration

Bits 0-4 of the *Data Press Report Configuration* register determine what the sensor action is and what is transmitted when an event is registered.

Registers 0x 2B and 0x 2C configure the *Data Press Pattern Configuration* and *Panic Press Pattern Configuration*, respectively. Note that trying to set m or n to 0 is invalid and ignored by the SW. Also, for the Data Press Pattern, 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 s.

6.6.2.2 Example DL Payloads

- Enable BLE report and battery report for the data press report:
 - DL payload: **0x AA 00 09**
 - Register 0x 2A with bit 7 set to 1 = 0x AA
 - Value bits 0 and 3 set to 1 = 0x 00 09
- Read current value of Panic Press Pattern Configuration:
 - DL payload: **0x 2C**
 - Register 0x 2C with bit 7 set to 0 = 0x 2C

6.7 General Command-and-Control Operations

6.7.1 List of Operations and Register Values

The general command-and-control operations supported by the FINCH and ROBIN are:

- Saving the current configuration settings to flash memory.
- Restarting the sensor (soft reset).
- Factory reset of configuration settings.

The above are executable operations. There are also read-only operations supported for reading FW and BLE metadata (SW version numbers).

To perform a command-and-control operation, the appropriate register must be accessed. Table 6-11 lists the details of the command-and-control registers. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in the following subsections and in §4.3.2.

Table 6-11: Command & Control Register Details

Address	Access	Name	Size	Description and Options	JSON Variable (Type/Unit)
0x 70	X	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 Sensor • Bits 1-12, 15: Must be set to 0, otherwise INVALID 	<pre>write_to_flash { app_config: <value>, (unsigned/no unit) lora_config: <value>, (unsigned/no unit) restart_sensor: <value> (unsigned/no unit) }</pre>

Address	Access	Name	Size	Description and Options	JSON Variable (Type/Unit)
0x 71	RO	FW 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 number²⁴ 	<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>
0x 72	X	Reset Configuration to Factory Defaults	1 B	<ul style="list-style-type: none"> • 0x 0A: Reset app configuration • 0x B0: Reset LoRaMAC configuration • 0x BA: Reset both App and LoRaMAC configurations • Any other value: Invalid 	<pre>config_factory_reset { app_config: <value>, (unsigned/no unit) loramac_config: <value> (unsigned/no unit) }</pre>
0x 73	RO	BLE Metadata	6 B	<ul style="list-style-type: none"> • Bits 16-23: BLE stack version major • Bits 8-15: BLE stack version minor • Bits 0-7: BLE stack version revision • Bits 24-47: RFU 	<pre>ble_metadata { stack_ver_major: <value>, (unsigned/no unit) stack_ver_minor: <value>, (unsigned/no unit) stack_ver_revision: <value>, (unsigned/no unit) }</pre>

6.7.2 Save Current Configuration Settings

Configuration changes are not retained after a power cycle (soft or hard reset) unless they are saved in the non-volatile flash memory. To do so, the *Flash Write Command* register, 0x 70, must be accessed to execute the save-to-flash operation.

²⁴ Defined by Table 6-12.

The DL payload structure is as shown in Figure 4-4. That is, with the first byte being the register address with bit 7 set to 1 (i.e., 0x F0) and the data indicating which options are selected of those listed in Table 6-11. Specifically, the payloads for the different save options (without restarting the sensor) are:

- **0x F0 20 00:** Save current configuration settings of all FW application registers (0x 20 through 0x 6F) to flash.
- **0x F0 40 00:** Save current configuration settings of all FW LoRaMAC Option registers (0x 10 through 0x 13) to flash.
- **0x F0 60 00:** Save current configuration settings of both FW application and LoRaMAC Options registers to flash.

The save-to-flash command can be sent in a separate DL at any time or be included in the same payload as other read and write command blocks. In the latter case, all other command blocks are always executed first, so that settings can be changed and saved in a single payload.

Register 0x 70 also supports a reset option, described in §6.7.3. When this option is not selected, the sensor will send a write response (as described in §4.2.2) after receiving the flash write command.

6.7.3 Sensor Restart

The *Flash Write Command* register, 0x 70, is used to restart the device via soft reset.

This is done by setting bit 0 to 1. This can be used alone or in conjunction with any of the save-to-flash operation options listed in §6.7.2 above. In the former case, the explicit payload is **0x F0 00 01**.

Immediately after receiving the reset command in a DL, the sensor will reset.

NOTE: Do not send the reset command as a confirmed DL. The reset command causes the sensor to restart before it can send the acknowledgement UL in response. The sensor will rejoin the network but then get the command sent again from the NS, causing a loop of continual rebooting. Some network servers, including TEKTELIC's KONA Core, have an optional setting to clear the DL queue upon receiving a JOIN REQUEST for a new session. This is an alternative solution if confirmed DLs are required.

6.7.4 Read FW and BLE Metadata

The *FW Metadata* register, 0x 71, can be accessed to read the *application version number*, *LoRaMAC version number*, and *LoRaMAC region number*. The *BLE Metadata* register, 0x 73, can similarly be accessed to read the *BLE application module version number*, and *BLE stack version number*.

The read metadata command is formulated as a regular read command as shown in Figure 4-4. Explicitly, the command blocks in the payload would be **0x 71** for FW and **0x 73** for BLE.

After receiving one of these commands, the sensor will respond with a UL message containing the following:

- For FW metadata:
 - The first byte is the register address: 0x 71.
 - Bits 32 to 55 of the value contain the application revision numbers which define the FW version. The FW version is reported in the format as shown in Figure 6-2, which is shown using the example FW v1.0.15 (value 0x 01 00 0F).

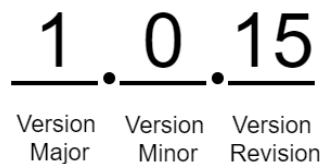


Figure 6-2: Example FW Version Format

- Bits 8 to 31 of the value contain the LoRaMAC version numbers. The format is the same as shown in Figure 6-2. 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 last byte contains the LoRaMAC region number. Current LoRaMAC regions and corresponding region numbers for the sensor are listed in Table 6-12.

Table 6-12: LoRaMAC Regions and Region Numbers

LoRaMAC Region	Region Number
EU868	1
US915	2
AU915	5
AS923-1	7
AS923-2	8
AS923-3	9
KR920	10
IN865	11

LoRaMAC Region	Region Number
RU864	12
AS923-4	13

- For BLE metadata:
 - The first byte is the register address: 0x 73.
 - Bits 0 to 23 of the value contain the BLE stack version numbers. The BLE stack is the SW published by Bluetooth and not developed by TEKTELIC. The version is reported in the format as shown in Figure 6-2.

6.7.5 Factory Reset

The *Reset Configuration to Factory Defaults* register, 0x 72, is used to reset all the configuration register values (0x 10 to 0x 6F) back to the default settings.

The DL payload structure is as shown in Figure 4-4. That is, with the first byte being the register address with bit 7 set to 1 (i.e., 0x F2) and the data indicating which options are selected of those listed in Table 6-11. Specifically, the payloads for the different factory reset options are:

- **0x F2 0A:** Restore configuration settings of all FW application registers (0x 20 to 0x 5C) to factory default values.
- **0x F2 B0:** Restore configuration settings of all FW LoRaMAC Options registers (0x 10 to 0x 13) to factory default values.
- **0x F2 BA:** Restore configuration settings of both FW application and LoRaMAC Options registers to factory default values.

The factory command can be sent in a separate DL at any time or be included in the same payload as the other read and write command blocks. In the latter case, only the factory command block is executed while all other commands are discarded.

After receiving the factory reset command, the sensor always restarts immediately.

NOTE: Do not send the reset command as a confirmed DL. The reset command causes the sensor to restart before it can send the acknowledgement UL in response. The sensor will rejoin the network but then get the command sent again from the NS, causing a loop of continual rebooting. Some network servers, including TEKTELIC's KONA Core, have an optional setting to clear the DL queue upon receiving a JOIN REQUEST for a new session. This is an alternative solution if confirmed DLs are required.

7 BLE Operation

BLE operation is the main function of the ROBIN/FINCH devices and forms the basis for asset tracking through TEKTELIC's LOCUS application. In LOCUS, asset localization works by analyzing the strength of Bluetooth signals between ROBIN/FINCH and nearby beacons to estimate position with an accuracy of about 2–5 meters. For more detail about LOCUS please visit the [Knowledge Hub: LOCUS](#).

To support this, ROBIN/FINCH can operate in two BLE modes:

- **Tracker Mode (default):** The sensor conducts BLE Rx scanning to discover nearby BLE peripherals, determine their signal strengths, and relay this information via LoRaWAN UL to the NS. The primary use case is for indoor positioning/asset tracking.
- **Beacon Mode:** The sensor broadcasts BLE Tx advertisements so it is discoverable to nearby BLE scanning devices. The primary use case is for setting up a beacon network for indoor positioning/asset tracking.

In the following subsections, the operational descriptions, report formats, and configurable settings for each mode are explained. The procedure for switching between modes is also explained below.

The sensor supports BLE as specified by Bluetooth 5.3 and uses only the 3 default advertising channels: 37, 38, and 39 [6].

NOTE: The terminology throughout this section is chosen to be clear on whether the described operation relates to tracker mode, beacon mode, or both. When referring to parameters specifically pertaining to tracker mode operation only, the sensor 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. All information in this section is applicable to both FINCH and ROBIN.

7.1 Tracker Mode (BLE Rx)

7.1.1 Operational Description

The default mode of the sensor is tracker mode.

When in tracker mode, the BLE operates in Rx only; the tracker only scans and does not advertise, meaning it is not discoverable by other BLE-capable devices.

During each scan, other nearby 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 LoRaWAN data report UL. This UL is normally reported immediately after the scan concludes but may be delayed due to LoRaWAN duty cycle limitations²⁵ [4].

The tracker can be configured to conduct a BLE scan and report periodically. By default, a BLE scan and report is conducted every 60 min in tracker mode.

The tracker can also be configured to conduct additional BLE scans and reports upon the following event-based triggers:

- Function button press: Enabled by default.
- Motion detected and motion cleared through the *Accelerometer Assist*²⁶ feature: Enabled by default.

The following subsections detail the scanning scheme, data preprocessing options, and device filtering options.

7.1.1.1 Scanning Scheme

A single BLE scan lasts for a configurable *scan duration*.

As illustrated in Figure 7-1, each scan duration is divided into regular *scan intervals*. The beginning of each scan interval marks the beginning of scanning on a different BLE channel frequency. The channels are cycled through in order; during the first scan interval the tracker receives on channel 37, the second on channel 38, the third on channel 39, the fourth on channel 37, and so on. The scan interval is configurable.

²⁵ If a new BLE scan occurs before the results of the previous scan have been sent, the old scan results will be discarded.

²⁶ See §8.4 for a complete description of the Accelerometer Assist feature.

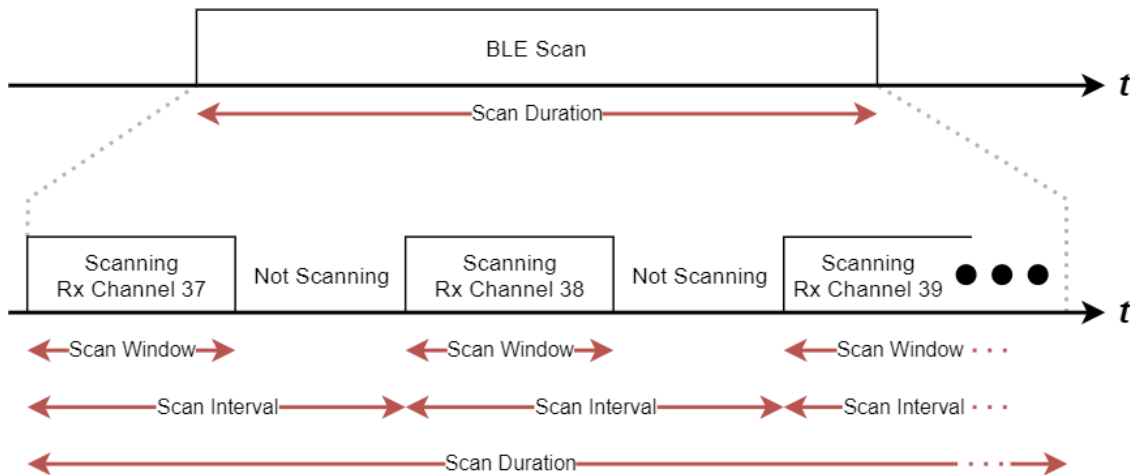


Figure 7-1: The BLE Scan Scheme

Also illustrated in Figure 7-1, the actual BLE scanning only occurs during the *scan window* portion of the scan interval. The scan window is configurable, which allows for the *scan duty cycle* to be implemented:

$$\text{scan duty cycle} = \frac{\text{scan window}}{\text{scan interval}} \times 100\%$$

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 BLE scan can be disabled entirely and re-enabled at any time in tracker mode. BLE advertising never occurs during tracker mode.

The BLE scan is *passive*. This means the tracker listens to surrounding beacons but does not transmit to them to request additional information.

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

7.1.1.2 Pre-processing Discovered BLE Device Data (Averaging Mode)

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*. For each individual advertisement packet received, the SW records:

- The 6-byte *BLE MAC address* of the advertising device, and
- The 1-byte *Received Signal Strength Indicator (RSSI)*, in [dBm], of the packet.

The tracker reports up to N MAC address + RSSI pairs following each scan, where N is configurable via register 0x 50.

During a single scan, it is possible to detect more than one advertisement from a single discovered device if that device is advertising with a period less than the tracker's scan duration. Because of this, the tracker supports the ability to report either the complete, "raw," discovered device data, or apply some pre-processing before reporting. In both cases, a maximum of N MAC address + RSSI data pairs are reported:

- **Raw Data:** No pre-processing is done on the packets received during each scan before reporting via UL; the last N discovered devices are sent. This means it is possible to have repeated devices in the same UL (one for each received advertisement).
- **Averaging Mode:** The RSSIs from each unique discovered device are averaged over all packets received from that device to get a single value. Then the devices are sorted in order from strongest average RSSI to weakest. The strongest N devices are sent in the UL report. Averaging mode is enabled by default.

7.1.1.3 Filtering Discovered BLE Devices

The tracker supports inclusion filtering of discovered devices based on BLE Device MAC address, BD_{ADDR} .

Each 6-byte BD_{ADDR} consists of an *Organizationally Unique Identifier (OUI)* comprising the 3 MSBs followed by a *Lower Address Part (LAP)* comprising the 3 LSBs in the form:

$$BLE_{ADDR} = OUI:LAP$$

The *OUI* is unique for a batch of devices; most commonly this means that all devices manufactured by the same company will share the same *OUI*²⁷. The *LAP* is unique for each device.

The tracker can filter by single BD_{ADDR} or, more generally, by a range of BD_{ADDR} s if a common *OUI* is shared. This keeps undesired devices out of the OTA report, thus reducing OTA time and saving

²⁷ TEKTELIC's OUI is 64:7F:DA.

battery life. An example application is for an indoor BLE beacon network; only the beacon signals are of interest and not those from other devices like smartphones.

Up to 4 separate filter ranges can be defined and used at the same time. The user must specify the start and end bounds of each range. That is, if a discovered device has a BD_{ADDR} that falls inside the range $OUI:LAP_{start}$ to $OUI:LAP_{end}$ its data is kept. If not, its data is discarded immediately. The range is inclusive, meaning that a BD_{ADDR} equivalent to the start or end bounds is considered in-range. Each range is defined in the following 9-byte format:

$$BD_{ADDR} \text{ range} = OUI|LAP_{start}|LAP_{end}$$

The following rules describe the behaviour of the filter ranges and reporting:

1. A range set to all 0s is an *inactive range*. Otherwise, an *active range*. All ranges are inactive by default.
2. All 4 ranges inactive means **basic reporting** is enabled. Following each scan, any and all discovered devices (up to a maximum N) are reported. Otherwise, **filtered reporting** is enabled. Basic reporting is the default behaviour.
3. Following any BLE scan, there will be at least 1 report corresponding to each active range. Each report is sent in a separate UL. Even if no devices from a certain range are discovered, an empty list is always sent for that range. See §7.1.2 for more details on reporting.
4. If a range is defined with $LAP_{start} > LAP_{end}$, the range is active but empty. I.e., an empty list is always reported for that range.
5. If an active range is defined with $LAP_{start} = LAP_{end}$, the range has only one BD_{ADDR} in it. This is *single device filtering*.
6. Ranges are allowed to overlap. A BD_{ADDR} that falls into more than one active range will be reported in each of those ranges.

NOTE: Averaging/raw modes and basic/filtered reporting are mutually compatible; that is, any combination of the four options is acceptable. Filtering always occurs during the scan, and averaging always occurs after.

7.1.2 UL Report Frame Formats

Discovered BLE devices are reported on **LoRaWAN port 25**. As described in §7.1.1.3, either basic or filtered reporting can be done.

NOTE 1: 0, 1, 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 [4]. For all BLE reporting described below, if the discovered device data cannot fit into one message, more than one UL will be subsequently transmitted to report all N devices.

NOTE 2: The scan results are normally reported in a UL immediately after the scan concludes but may be delayed due to duty cycle limitations [4]. If a new BLE scan occurs before the results of the previous scan have been sent, the old scan results will be discarded.

7.1.2.1 Basic Reporting

Basic reporting occurs when no filter ranges are active. In this case, following each scan, the data from all discovered devices (up to a maximum N) are reported. The UL payload format is shown in Figure 7-2. Basic reporting is the default behaviour.

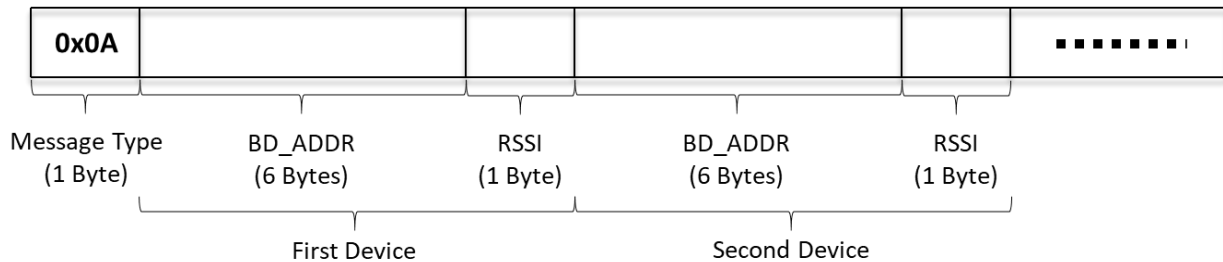


Figure 7-2: The BLE Report UL Payload Format for Basic Reporting

Each payload consists of a 1-byte header (0x 0A) followed by the data from each discovered device. If no devices were discovered during a scan, an empty list (i.e., only the header) is sent.

The data from each device consists of a 7-byte BD_{ADDR} and RSSI pair, where the BD_{ADDR} is the full 6-byte BLE Device MAC Address and the RSSI is a signed 1-byte number in units of [dBm].

The BD_{ADDR} and RSSI pairs for any other discovered devices are concatenated into the payload. The order of the listed devices depends on whether averaging mode is enabled or disabled (see §7.1.1.2).

7.1.2.2 Filtered Reporting

Filtered reporting occurs when one or more filter range is active. In this case, following each scan, only discovered devices (up to a maximum N) from specified ranges of favorable BD_{ADDRS} are included in the report. The UL payload formats are shown in Figure 7-3.

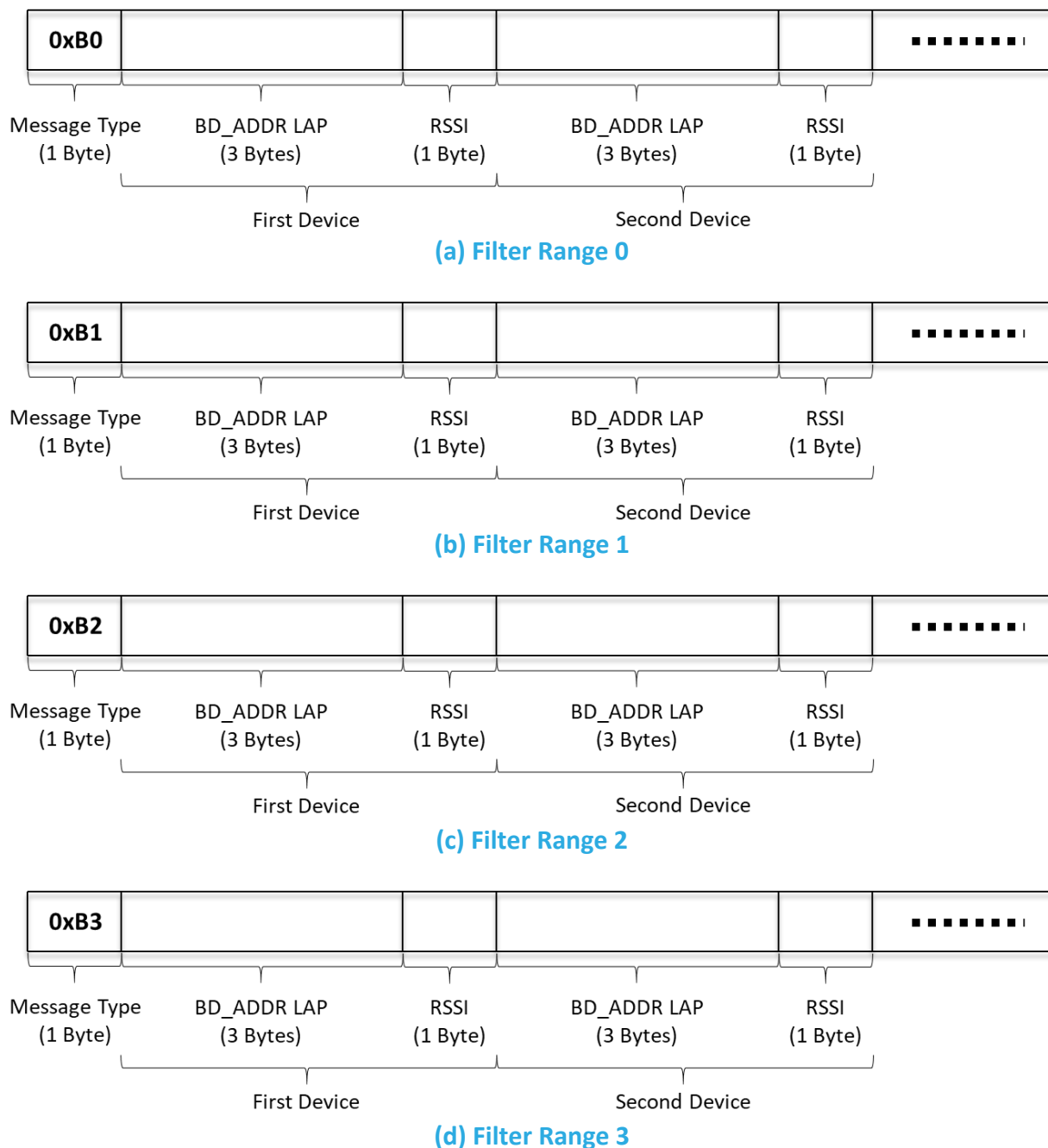


Figure 7-3: BLE Report UL Payload Formats for Filtered Reporting

Each payload consists of a 1-byte header followed by the data from each discovered device. The header denotes the filter range corresponding to these devices. Explicitly, BLE reports for ranges 0, 1, 2, and 3 have the headers 0x B0, 0x B1, 0x B2, and 0x B3, respectively. If no devices were discovered during a scan for a particular filter range, an empty list (i.e., only the header) is sent.

The data from each device consists of a 4-byte *LAP* and RSSI pair, where the *LAP* is the 3 LSBs of the *BD_ADDR* and the RSSI is a signed 1-byte number in units of [dBm]. The *OUI* is implicitly known due to the definition of the filter range, so only the *LAP* is necessary for unique identification. See §7.1.1.3 for more details about filter operation.

The *LAP* and RSSI pairs for any other discovered devices are concatenated into the payload. The order of the listed devices depends on whether averaging mode is enabled or disabled (see §7.1.1.2).

There will always be at least 1 report corresponding to each active range following every BLE scan. Each report is sent in a separate UL.

7.1.2.3 Example UL Payloads

- **0x 0A 64 7F DA 00 00 01 C9**
 - Header = 0x 0A → basic BLE data report
 - 0x 64 7F DA 00 00 01 = *BD_ADDR* = 64:7F:DA:00:00:01
 - 0x C9 = RSSI = $-55 \times 1 \text{ dBm} = -55 \text{ dBm}$
- **0x B0 12 34 56 C4 AB CD EF 3F**
 - Header = 0x B0 → BLE data report for filter range 0
 - 0x 12 34 56 = device 1 *BD_ADDR LAP* = 12:23:56
 - 0x C4 = device 1 RSSI = $-60 \times 1 \text{ dBm} = -60 \text{ dBm}$
 - 0x AB CD EF = device 2 *BD_ADDR LAP* = AB:CD:EF
 - 0x 3F = device 2 RSSI = $-63 \times 1 \text{ dBm} = -63 \text{ dBm}$

7.1.3 Configuration Settings

Table 7-1 shows the list of configuration registers which affect BLE Rx behaviour. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 7-1: Tracker Mode Configuration Registers

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 20	Seconds per Core Tick	4 B	Access R/W <ul style="list-style-type: none"> • Tick value for periodic events, T • Acceptable values: 15, 16, 17, ..., 86400 • $B \times T \leq 86\,400$, otherwise invalid²⁸ • Other values: Invalid 	60 seconds 0x 00 00 00 3C	<i>seconds_per_core_tick: <value></i> (unsigned/s)
0x 23	Ticks per BLE Location Update in PANIC State	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between BLE location update in PANIC state • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic BLE reports 	1 tick = 1-minute period 0x 00 01	<i>ticks_per_ble_panic_state: <value></i> (unsigned/no unit)
0x 25	Ticks per BLE Location Update in NORMAL State	2 B	Access R/W <ul style="list-style-type: none"> • Ticks between BLE location update in NORMAL state • Acceptable values: 0, 1, 2, ..., 65535 • 0 disables periodic BLE reports 	60 ticks = 1-hour period 0x 00 3C	<i>ticks_per_ble_normal_state: <value></i> (unsigned/no unit)

²⁸ This restriction is due to the anti-bricking strategy. See §6.3.1.1.

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 2A	Data Press Report Configuration	2 B	<p>Access: R/W</p> <ul style="list-style-type: none"> • Bit 0: 0/1 = Battery report disabled/enabled • Bit 1: 0/1 = Acceleration vector report disabled/enabled • Bit 2: 0/1 = MCU temperature report disabled/enabled • Bit 3: 0/1 = BLE location update disabled/enabled • Bit 3 set to 1 is unavailable in Beacon Mode • Bit 4: 0/1 = Device status report disabled/enabled • Bits 5-15: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • BLE report enabled • Device status report enabled <p>0x 00 18</p>	<pre>data_press_report_options: { battery_report: <value>, (unsigned/no unit) accl_vector: <value>, (unsigned/no unit) mcu_temp: <value>, (unsigned/no unit) ble_scan: <value>, (unsigned/no unit) device_status: <value> (unsigned/no unit), }</pre>
0x 46	Acceleration Event Report Configuration	1 B	<p>Access: R/W</p> <ul style="list-style-type: none"> • Bit 0: 0/1 = Acceleration alarm UL report disabled/enabled • Bit 1: 0/1 = Accelerometer Assist BLE scan report disabled/enabled • Bit 4: 0/1 = MCU temperature report disabled/enabled • Bit 5: 0/1 = Battery report disabled/enabled • Bits 2-3, 6-7: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • Acceleration alarm report UL enabled • Accelerometer Assist enabled <p>0x 03</p>	<pre>accl_event_report: { accl_alarms: <value>, (unsigned/no unit) accl_assist_ble: <value>, (unsigned/no unit) mcu_temperature: <value>, (unsigned/no unit) battery_report: <value>, (unsigned/no unit) }</pre>

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 50	BLE Rx Mode	1 B	Access: R/W <ul style="list-style-type: none"> Bits 0-6: N Number of reported devices (1–127) 0: Disables BLE Bit 7: 0/1 = Averaging mode off/on 	<ul style="list-style-type: none"> Up to 8 reported devices Averaging mode on 0x 88	<pre>ble_mode: { num_reported_devices: <value>, (unsigned/no unit) averaging_mode: <value> (unsigned/no unit) }</pre>
0x 51	BLE Scan Duration	2 B	Access: R/W <ul style="list-style-type: none"> Bits 0-7: Scan duration for periodic reports [1 s/LSb] Acceptable values: 1, 2, ..., 10 Other values: Invalid Bits 8-15: Scan duration for event-based reports [1 s/LSb] Acceptable values: 1, 2, ..., 10 Other values: Invalid 	<ul style="list-style-type: none"> 3 s for periodic scans 3 s for event-based scans 0x 03 03	<pre>ble_scan_duration: { periodic: <value>, (unsigned/no unit) event-based: <value>, (unsigned/no unit) }</pre>
0x 52	BLE Scan Interval	2 B	Access: R/W <ul style="list-style-type: none"> Scan interval [1 ms/LSb] Acceptable values: “Scan Window”, ..., 10 000 Other values: Invalid 	30 ms 0x 00 1E	<pre>ble_scan_interval: <value> (unsigned/sec)</pre>
0x 53	BLE Scan Window	2 B	Access: R/W <ul style="list-style-type: none"> Scan window [1 ms/LSb] Acceptable values: 3, ..., “Scan Interval” Other values: Invalid 	30 ms 0x 00 1E	<pre>ble_scan_window: <value> (unsigned/sec)</pre>
0x 54	Filter Range 0	9 B	Access: R/W <ul style="list-style-type: none"> Range 0 for filtered BD_{ADDRS} $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> 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	<pre>filter_range_0: { oui: <value>, (unsigned/no unit) lap_start: <value>, (unsigned/no unit) lap_end: <value> (unsigned/no unit) }</pre>

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 55	Filter Range 1	9 B	Access: R/W • Range 1 for filtered BD_{ADDRS} • $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> ▪ $OUI = B_0:B_1:B_2$ ▪ $LAP_{start} = B_3:B_4:B_5$ ▪ $LAP_{end} = B_6:B_7:B_8$ 	Range inactive 0x 00 00 00 00 00 00 00 00 00 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	Access: R/W • Range 2 for filtered BD_{ADDRS} • $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> ▪ $OUI = B_0:B_1:B_2$ ▪ $LAP_{start} = B_3:B_4:B_5$ ▪ $LAP_{end} = B_6:B_7:B_8$ 	Range inactive 0x 00 00 00 00 00 00 00 00 00 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	Access: R/W • Range 3 for filtered BD_{ADDRS} • $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> ▪ $OUI = B_0:B_1:B_2$ ▪ $LAP_{start} = B_3:B_4:B_5$ ▪ $LAP_{end} = B_6:B_7:B_8$ 	Range inactive 0x 00 00 00 00 00 00 00 00 00 00	$filter_range_3: \{$ $oui: <value>$, (unsigned/no unit) $lap_start: <value>$, (unsigned/no unit) $lap_end: <value>$ (unsigned/no unit) }

7.1.3.1 Periodic Scanning and Reporting Configuration

The BLE scan period can be configured using registers 0x 20 and 0x 25 according to the equation described in §6.3.2. That is:

$$BLE\ Scan\ Period = Seconds\ per\ Core\ Tick \times Ticks\ per\ BLE\ Scan$$

The BLE scan period is equal in value to the BLE report period but scanning and reporting necessarily happen in sequence. As illustrated in Figure 7-4, the BLE scan period specifically defines the period between the starts of consecutive scans and the scan results are reported after each scan concludes. The results are always sent in a LoRaWAN UL within 1 s of to conclusion of the scan, except in some cases when regional duty cycle limitations are in effect²⁹.

²⁹ See notes in §7.1.2.

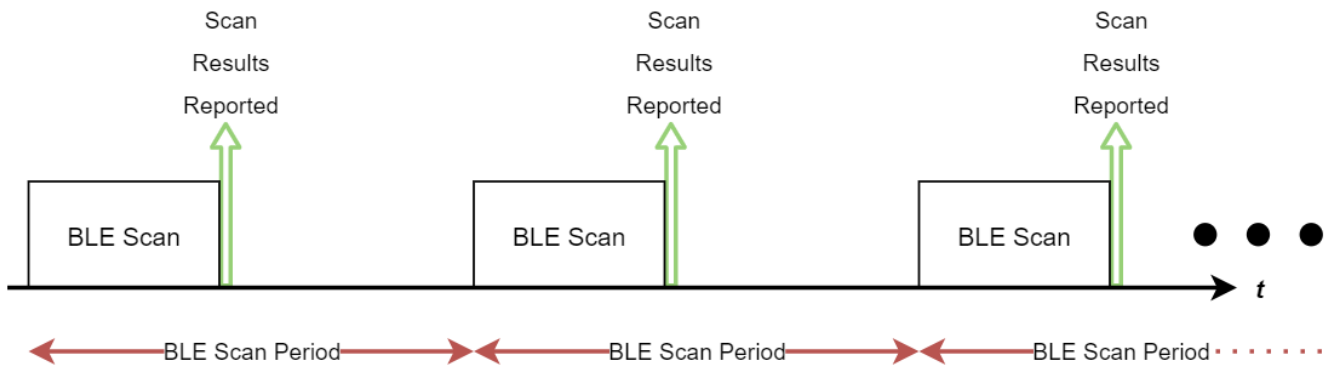


Figure 7-4: Periodic BLE Scanning

Setting the BLE scan period to 0 disables periodic BLE scanning and reporting.

7.1.3.2 Event-Based Scanning and Reporting Configuration

The tracker supports additional, non-periodic BLE scans and reports upon the following event-based triggers:

- Function button press:
 - Setting bit 3 to 1 in the *Function Button Mode* register, 0x 2A, causes a single BLE scan and report to be conducted following a button press detection.
 - Enabled by default.
 - For details about how the function button works and is configured, see §6.6.
- *Accelerometer Assist* feature:
 - Setting bit 1 to 1 in the *Acceleration Event Value to Tx* register, 0x 46, causes a single BLE scan and report to be conducted when the accelerometer detects the beginning of motion, and another single scan when the accelerometer detects that the motion has cleared.
 - Enabled by default.
 - See §8.4 for details about Accelerometer Assist.

NOTE 1: Periodic and event-based scans have scan durations which are independently configurable. See §7.1.3.3 for details.

NOTE 2: BLE scanning must be enabled on register 0x 50 in order for the function button press to cause event based BLE scans and reports. See §6.6 for details.

NOTE 3: The accelerometer must be enabled in order for Accelerometer Assist to cause event based BLE scans and reports. See §8.4.1 for details.

7.1.3.3 Operational Configuration

The *BLE Rx Mode* register, 0x 50, controls 3 settings:

- *BLE scan enable/disable*: setting bits 0 through 6 to all 0s disables all BLE scanning, periodic and event-based. NOTE: this value is ignored if the device is in beacon mode; it is impossible to conduct any scanning while in beacon mode.
- *Number of reported devices, N*: The results of each scan will only include up to *N* discovered devices. If *N* or fewer devices are discovered, all devices are reported. If more than *N* devices are discovered, *N* devices are reported and they are chosen based on whether averaging mode is enabled (see §7.1.1.2).
- *Averaging mode enable/disable*: setting bit 7 to 1 enables averaging mode.

The scan parameters, namely the *BLE scan durations*, *BLE scan interval*, and *BLE scan window*, are all fined by registers 0x 51, 0x 52, and 0x53, respectively. The periodic scan duration can be configured independently of the event-based scan duration, but the scan interval and scan window apply to both types of scans.

Registers 0x 54 through 0x 57 define filter ranges 0 through 3, respectively.

7.1.3.4 Guidelines for BLE Scanning and Reporting Configuration

For periodic BLE scanning, although the scan period, scan duration, and number of devices to report can be freely configured to different values, some combinations 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 s [1].

Table 7-2 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 7-2, 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 7-2: 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.

7.1.3.5 Example DL Payloads

- Schedule a BLE scan and report every 15 min:
 - DL payload: **0x A0 00 00 00 3C A5 00 0F**
 - Register 0x 20 with bit 7 set to 1 = 0x A0
 - 60 s/core tick = 0x 00 00 00 3C
 - Register 0x 25 with bit 7 set to 1 = 0x A1
 - Scan and report every 15 ticks = 0x 00 0F
- Configure the tracker to report the strongest beacon only:
 - DL payload: **0x D0 81**
 - Register 0x 50 with bit 7 set to 1 = 0x D0
 - Number of reported devices = 1, Averaging mode enabled
 - Value bits 0 and 7 set to 1 = 0x 81
- Decrease the scan duty cycle to 50% from default while keeping scan durations the same to save battery life:

- DL payload: **0x D3 00 0F**
 - Register 0x 53 with bit 7 set to 1 = 0x D3
 - Scan window 15 ms = 0x 00 0F
- Set filters to only report devices with MAC addresses ABCDEF000001 and ABCDEF500000 to ABCDEF999999:
 - DL payload: **0x D4 AB CD EF 00 00 01 00 00 01 D5 AB CD EF 50 00 00 99 99 99**
 - Register 0x 54 with bit 7 set to 1 = 0x D4
 - Filter range 0 = ABCDEF000001 to ABCDEF000001 = 0x AB CD EF 00 00 01 00 00 01
 - Register 0x 55 with bit 7 set to 1 = 0x D5
 - Filter range 1 = ABCDEF500000 to ABCDEF999999 = 0x AB CD EF 50 00 00 99 99 99
- Filter for only TEKTELIC BLE devices:
 - DL payload: **0x D4 64 7F DA 00 00 00 FF FF FF**
 - Register 0x 54 with bit 7 set to 1 = 0x D4
 - Filter range 0 = 647FDA000000 to 647FDAFFFFFF = 0x 64 7F DA 00 00 00 FF FF FF

7.2 Beacon Mode (BLE Tx)

7.2.1 Operational Description

The FINCH and ROBIN sensors support a *beacon mode* function as an alternative to tracker mode. The default mode of the sensor is tracker mode, so it must be switched into beacon mode as described in §7.3.

When in beacon mode, the BLE operates in Tx only. It sends out periodic BLE *advertisements* which are small packets of data. These packets are discoverable by other FINCHs and ROBINS operating in tracker mode, as well as any other device capable of BLE scanning.

When in beacon mode, the sensor is still LoRaWAN-backhauled. That is, it can still send sensor data in LoRaWAN ULs and be reconfigured through LoRaWAN DLs. Furthermore, all other transducer functions are accessible in either beacon or tracker mode.

When switching to beacon mode, some of the default configuration settings are changed such that the behaviour reflects the standard beacon use-case³⁰. Namely, the accelerometer is disabled, the function button press triggers a battery report, and a periodic battery report is sent once every 24 hours.

The following subsections detail the advertising scheme and packet formats.

7.2.1.1 Advertising Scheme

After a beacon joins the LoRaWAN network, it begins broadcasting BLE advertisements. This continues throughout normal operation as a background process.

The *advertising interval* is the time between the beginnings of consecutive advertisement transmissions as shown in Figure 7-5. It is user-configurable in units of [ms].

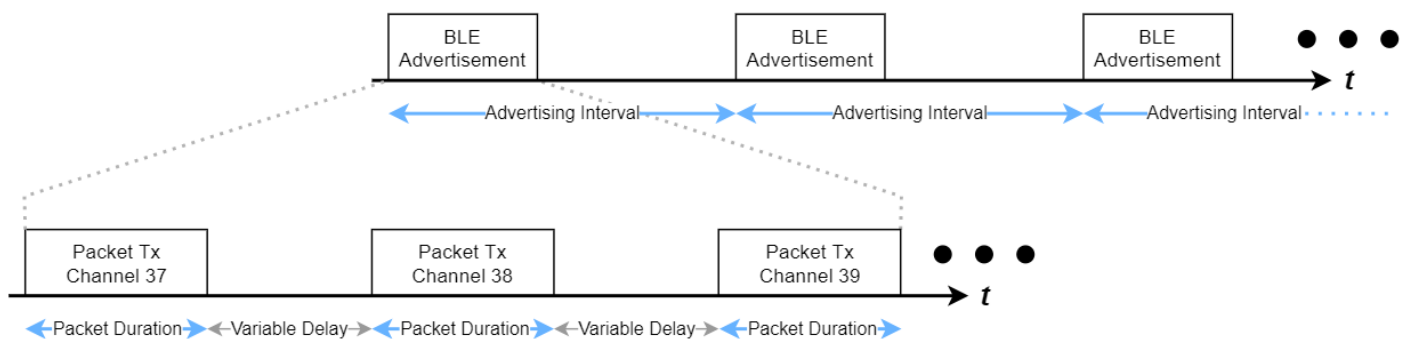


Figure 7-5: The BLE Advertising Scheme

³⁰ The standard beacon is a stationary device mounted on a wall that periodically broadcasts BLE advertisements.

Figure 7-5 also shows that each single BLE advertisement comprises 3 individual packet transmissions, sent one after another on BLE channels 37, 38, and 39 [6]. This is to maximize the chances of a BLE device scanning on a single channel receiving 1 packet per advertising interval.

Each packet is identical in contents (see §7.2.1.2). Each *packet duration* is approximately 400 μ s and is always constant for a given packet format.

Each packet is separated by a delay of between 0 and 6 ms. The BLE module adds some random offset to the start of each packet transmission to mitigate synchronization issues, so the actual interval between transmissions will vary. The combined offsets translate to a total advertisement duration of between 2 and 8 ms and total error in the advertising interval of ± 10 ms.

In addition to the advertising interval, the advertisement *Tx power* level is also a configurable operational parameter.

The BLE advertisement and LoRa radio transmission are mutually exclusive and never overlap. If any reporting becomes due, the BLE advertisements are paused while the LoRa activity is occurring.

NOTE: The advertising interval is independent from the application tick settings. The sensor data LoRaWAN reporting intervals and BLE advertising interval are independently configurable.

7.2.1.2 Advertisement Packet Formats

The BLE advertising packet formatting supports 3 major BLE standards: iBeacon, Eddystone UID, and Eddystone TLM. By default, only iBeacon is enabled. Figure 7-6 shows the frame breakdown of each of these packet formats.

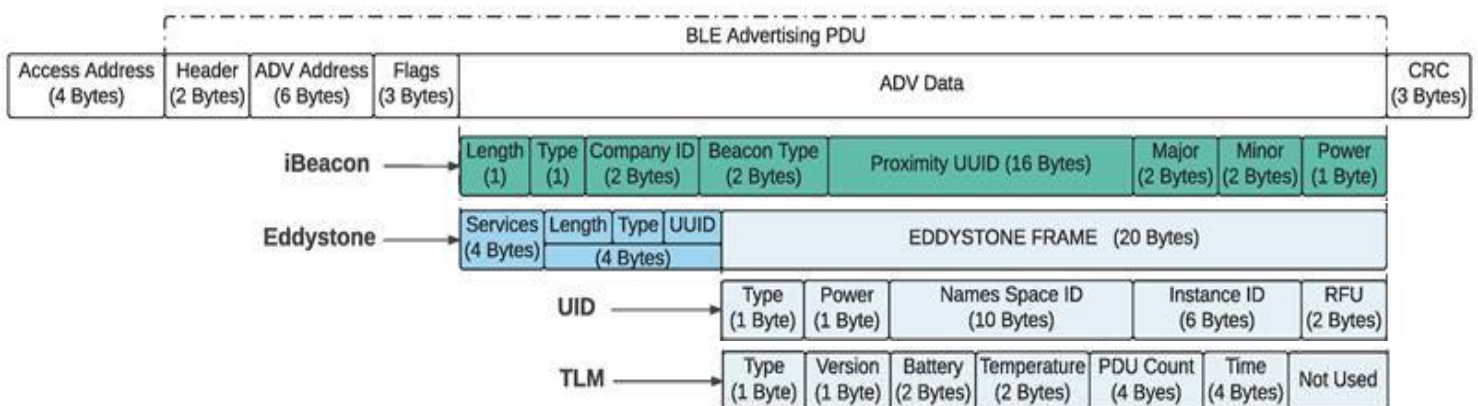


Figure 7-6: Supported Standard BLE Beacon Advertisement Packet Formats

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

- **ADV Address:** This is the 6-byte BLE Device MAC Address, BD_{ADDR} , and is included in all packet format types.
- **iBeacon <Proximity UUID>:** A 16-byte unique identifier. This field is populated by the beacon's LoRaWAN DevEUI twice. For example, if the DevEUI is 647FDA0000001234, then the iBeacon Proximity UUID would be 647FDA0000001234-647FDA0000001234.
- **Eddystone UID <Namespace ID>:** A 10-byte unique identifier. This field is populated by the beacon's LoRaWAN DevEUI, followed by 2-bytes of 0s. For example, if the DevEUI is 647FDA0000001234, then the Eddystone UID Namespace ID would be 647FDA00000012340000.
- **Eddystone TLM <Battery>:** This field is populated with battery report data. The data type is always remaining 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.
- **Eddystone TLM <Temperature>:** This field is populated with MCU temperature data. 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 <PDU Count>:** A running counter that increments every time the beacon emits a new advertisement frame (of any type). This counter is reset to 0 upon hard and soft resets.
- **Eddystone TLM <Time>:** A counter that represents the time since the beacon is powered on or rebooted. This is represented in 0.1 s/LSb. This counter is reset to 0 upon hard and soft 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 5.3 Core Specification [6].

Any combination of the 3 packet formats can be enabled at once. If more than 1 format is enabled, the packets are interleaved with each advertising interval. In other words, the packet format changes with each advertisement and cycles through all enabled options.

For example, consider the case with all 3 formats enabled and an advertising interval of 100 ms. In this case, at 100 ms, 3 iBeacon packets are transmitted, one on each channel. At 200 ms, 3 Eddystone UID packets are transmitted. At 300 ms, 3 Eddystone TLM packets are transmitted. At 400 ms, 3 iBeacon packets are transmitted. Etc.

7.2.2 Configuration Settings

Table 7-3 shows the list of configuration registers which affect BLE Tx behaviour. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 7-3: BLE Advertising Configuration

Address	Value	Size	Description and Options	Default Value ³¹	JSON Variable (Type/Unit)
0x 58	BLE Advertising Enable / Disable	1 B	Access: R/W • Bits 7-1: Ignored • Bit 0: 0/1 = Advertising disabled/enabled	Advertising disabled 0x 00	<i>advertising_enabled: <value> (unsigned/no unit)</i>
0x 59	BLE Advertising Interval	2 B	Access: R/W • Advertising interval [1 ms/LSb] • Acceptable values: 30 ms – 10000 ms • Other values: Invalid	100 ms 0x 00 64	<i>min_advertising_interval: <value> (unsigned/ms)</i>

³¹ The default values in this (and all other tables like it throughout this document) are the sensor default. Since the default sensor mode is tracker mode, all default values represent tracker mode defaults. When the sensor is switched to beacon mode, some default values change automatically. For more information, see §7.3. For a summary of all register default values in both tracker and beacon mode, see Table 0-2 in Appendix 2.

Address	Value	Size	Description and Options	Default Value ³¹	JSON Variable (Type/Unit)
0x 5B	BLE Tx Advertising Power	1 B	Access: R/W • BLE Tx Power [dBm] ³² <ul style="list-style-type: none"> 0x 00 = 0 dBm 0x 01 = -0.85 dBm 0x 02 = -1.8 dBm 0x 03 = -3.15 dBm 0x 04 = -4 dBm 0x 05 = -4.95 dBm 0x 06 = -5.9 dBm 0x 07 = -6.9 dBm 0x 08 = -7.8 dBm 0x 09 = -8.85 dBm 0x 0A = -9.9 dBm 0x 0B = -12.05 dBm 0x 0C = -14.1 dBm 0x 0D = -16.5 dBm 0x 0E = -20.85 dBm 0x 0F = -40 dBm • Other values: Invalid	0 dBm 0x 00	<i>tx_advertising_power: <value></i> <i>(unsigned/no unit)</i>
0x 5C	BLE Advertisement Packet Format	1 B	Access: R/W • 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 disabled • Eddystone TLM disabled 0x 01	<i>advertising_packet_format: {</i> <i> ibeacon: <value>,</i> <i> (unsigned/no unit)</i> <i> eddytone_uid: <value>,</i> <i> (unsigned/no unit)</i> <i> eddytone_tlm: <value></i> <i> (unsigned/no unit)</i> <i>}</i>
0x 5F	BLE MAC Address	6 B	Access: RO • The 6-Byte BLE Device MAC address, <i>BD_ADDR</i> , of the sensor.	Unique for every sensor	<i>mac_address: <value></i> <i>(unsigned/no unit)</i>

³² These are the supported setpoints on Gen2.5 devices. Gen2 devices have different options. See Table 1-1 for how to determine generation.

7.2.2.1 Operational Configuration

The *BLE Advertising Enable/Disable* register, 0x 58, allows advertising to start and stop as needed in beacon mode. By default, the sensor is in tracker mode, so advertising is disabled. When switched to beacon mode, this register value gets changed to enable advertising. In tracker mode, the value of this register is ignored; it is impossible to transmit BLE advertisements while in tracker mode.

Registers 0x 59 and 0x 5B determine the advertising interval and the advertisement Tx power, respectively. The higher these values, the more power the sensor consumes while in beacon mode. These values should be configured with respect to balancing the desired battery lifetime and beacon detectability for each particular use-case.

The packet format(s) are enabled or disabled by the value of register 0x 5C.

7.2.2.2 BLE Device 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 (see §7.1.2.2).

Register 0x 5F 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 switched into Beacon mode (described in §7.3), upon rejoining the LoRaWAN network, the first UL with an application frame 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 (see §4.2.2).

7.2.2.3 Example DL Payloads

- Change the Beacon's BLE advertisement power to -3.15 dBm:
 - DL payload: **0x DB 03**
 - Register 0x 5B with bit 7 set to 1 = 0x DB
 - Power setpoint 3 = 0x 03
- Change the advertising interval to 5 s:
 - DL payload: **0x D9 13 88**
 - Register 0x 59 with bit 7 set to 1 = 0x D9
 - Advertising interval = 5000 ms = 0x 13 88
- Read MAC address and change the advertising packet format to Eddystone TLM:
 - DL payload: **0x 5F DC 04**

- Register 0x 5F with bit 7 set to 0 = 0x 5F
- Register 0x 5C with bit 7 set to 1 = 0x DC
- Value bit 2 set to 1 = 0x 04

7.3 Toggling Tracker Mode ↔ Beacon Mode

7.3.1 Toggling Procedure and Commands

The device can operate in 1 of 2 modes at a time:

- **Tracker mode:** BLE Rx only, see §7.1 for detailed operation descriptions.
- **Beacon mode:** BLE Tx only, see §7.2 for detailed operation descriptions.

The mode is defined by register 0x 0A. Switching between modes is done by sending a DL command to reconfigure the register according to the details described in §4.3.2. Table 7-4 shows the details and values for register 0x 0A. In this table, the bit indexing scheme is as shown in Figure 4-1.

Table 7-4: Tracker/Beacon Mode Configuration Register

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

The write command to reconfigure register 0x 0A doubles as an executable command. That is, when a DL containing a command block to reconfigure register 0x 0A is received, a set of actions occur:

1. Configuration DL received containing a mode switch command. The value of register 0x 0A is overwritten with the new value.
2. Sensor resets.
3. Sensor enters JOIN mode, attempting to rejoin the LoRaWAN network.
4. Once rejoined, the sensor operates in the mode specified by the DL in step 1.

The explicit commands and procedures are as follows.

- **Switch to Beacon Mode:**
 - DL payload **0x 8A 01**.
 - Sensor behaviour upon rejoin: the first LoRaWAN UL with an application frame payload will be sent on **LoRaWAN port 100** and contain the MAC address as described in §7.2.2.2.
- **Switch to Tracker Mode:**
 - DL payload **0x 8A 00**.
 - Sensor behaviour upon rejoin: a BLE scan is conducted, and the results of that scan are sent in a UL on **LoRaWAN port 25**.

NOTE 1: Do not include any other command blocks in the same DL payload as a BLE mode switch command. Since the mode switch commands includes a device restart and reset to factory default configuration settings, any other command blocks will not be executed and therefore are irrelevant.

NOTE 2: Do not send the mode switch command as a confirmed DL. The mode switch command causes the sensor to restart before it can send the acknowledgement UL in response. The sensor will rejoin the network but then get the command sent again from the NS, causing a loop of continual rebooting. Some network servers, including TEKTELIC's KONA Core, have an optional setting to clear the DL queue upon receiving a JOIN REQUEST for a new session. This is an alternative solution if confirmed DLs are required.

NOTE 3: The current mode (current value of register 0x 0A) is does not impact the behaviour upon receiving a mode switch command. E.g., if a command to switch to tracker mode is received when already in tracker mode, the sensor will still restart, reset, and rejoin.

7.3.2 Differences Between Modes

Tracker and beacon modes differ only by the values of certain configuration registers.

The default mode is tracker mode, so default sensor configuration settings is equivalent to default tracker mode settings. When switching to beacon mode, a different set of *default beacon mode* configuration values are loaded and used by the sensor after it resets and rejoins the LoRaWAN network.

The only default register values that differ between modes are listed in Table 7-5. The default values of all other registers not listed in Table 7-5 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, regardless of the current mode.

Table 7-5: Default Register Values for Tracker/Beacon Mode

Name	Register Address	Default Value, Tracker Mode (Factory Default)	Default Value, Beacon Mode
LoRaMAC DR and Tx Power	0x 12	0x 04 00	0x 00 00
Ticks per BLE Location Update in PANIC State	0x 23	0x 00 01	0x 00 00
Ticks per BLE Location Update in NORMAL State	0x 25	0x 00 3C	0x 00 00
Data Press Report Configuration	0x 2A	0x 00 18	0x 00 01

Name	Register Address	Default Value, Tracker Mode (Factory Default)	Default Value, Beacon Mode
Accelerometer Mode	0x 40	0x 87	0x 07
Acceleration Event Report Configuration	0x 46	0x 03	0x 01
Advertising Enable / Disable	0x 58	0x 00	0x 01

For a complete list of both default tracker mode and default beacon mode configuration settings, see Table 0-2 in Appendix 2 – Lists of Configuration Registers and DL Payload Formats.

8 Accelerometer Operation

The FINCH and ROBIN are each equipped with an accelerometer that can measure the direction and magnitude of acceleration on up to 3 individual axes: $\pm X$, $\pm Y$, and $\pm Z$.

The accelerometer is enabled by default. When enabled, it operates constantly in the background during normal operation, making measurements at a configurable *sample rate*.

The accelerometer measurement sampling can be used for 4 main sensor functions:

- **Orientation Detection:** the accelerations on each axis can be reported periodically and/or upon function button press.
- **Motion Detection Alarms:** if the acceleration samples meet certain threshold criteria, “motion detected” and “motion clear” alarm reports can be generated.
- **Accelerometer Assist:** if acceleration samples meet certain threshold criteria, non-periodic BLE scans and reports can be triggered upon the beginning and end of motion.
- **Device Status Accelerometer Trigger:** if enabled via register 0x 60, an accelerometer event can result in the transition between the NORMAL and PANIC state. A “motion detected” event would transition the device to PANIC (if currently in NORMAL) and a “motion cleared” event would transition the device to NORMAL (if currently in PANIC).

By default, orientation reporting and accelerometer trigger are disabled, but motion detection alarms and accelerometer assist are enabled in tracker mode. The accelerometer is disabled entirely by default in beacon mode.

The general behaviour and configuration, as well as the descriptions of the 4 sensor accelerometer functions above are detailed in the following subsections.

8.1 General Sampling Behaviour

8.1.1 Operational Description

Regardless of which sensor accelerometer function is being used, the basic measurement scheme is the same. A single accelerometer measurement sample is comprised of the direction and magnitude of acceleration on up to 3 individual axes: $\pm X$, $\pm Y$, and $\pm Z$.

The acceleration magnitude is measured in units of *acceleration due to gravity*, g , where 1 g is equivalent to the acceleration experienced by a body at rest on earth's surface: 9.810 m/s^2 . By definition, measuring acceleration means detecting changes in movement.

The accelerometer is enabled by default. When enabled, it operates constantly in the background during normal operation, making measurements at a configurable *sample rate*. As with any physical sampling, any real acceleration value must be sustained longer than the sample period to be accurately measured. Quicker sample rates have a shorter period and can therefore resolve shorter physical acceleration events. However, sampling the transducer at a quicker rate increases the power consumption, impacting the battery life.

The *measurement range* is configurable and defines the full dynamic range of accelerations that can be monitored on any enabled axis. Since the accelerometer output is always an 8-bit signed number, a larger measurement range means less precision (i.e., a larger g unit per LSb). Explicitly, the supported measurement ranges of $\pm 2 g$, $\pm 4 g$, $\pm 8 g$, $\pm 16 g$ correspond to typical output precisions of 16 mg, 32 mg, 64 mg, 192 mg, respectively³³. If the physical acceleration magnitude is outside the current measurement range at the time of sampling, it will not be registered.

8.1.2 Configuration Settings

Table 8-1 shows the list of configuration registers which control general accelerometer behaviour. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

³³ The magnitudes are always reported by the sensor with $0.001 g = 1 \text{ mg}$ resolution, regardless of attainable precision given the current measurement range.

Table 8-1: General Accelerometer Configuration Registers

Address	Value	Size	Description	Default Value	JSON Variable (Type/Unit)
0x 40	Accelerometer Mode	1 B	Access: R/W • 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	<pre>accl_mode: { xaxis_enabled: <value>, (unsigned/no unit) yaxis_enabled: <value>, (unsigned/no unit) zaxis_enabled: <value>, (unsigned/no unit) power_on: <value> (unsigned/no unit) }</pre>
0x 41	Accelerometer Sensitivity	1 B	Access: R/W • Bits 0-2: Sample Rate: 1/2/3/4/5/6/7 = 1/10/25/50/100/200/400 Hz 0: Invalid and ignored • Bits 4-5: Measurement Range: 0/1/2/3 = ±2/±4/±8/±16 g • Bits 3, 6, 7: Ignored	• Sample Rate = 10 Hz • Measurement Range = ±8 g 0x 22	<pre>accl_sensitivity: { sample_rate: <value>, (unsigned/Hz) measurement_range: <value> (unsigned/g) }</pre>

8.1.2.1 Operational Configuration

The *Accelerometer Mode* register, 0x 40, controls 2 settings:

- *Accelerometer on/off*: setting bit 7 to 0 turns the accelerometer off, thereby disabling all accelerometer functions (motion detection is off).
- *X/Y/Z Axis enable/disable*: individual axes can be enabled or disabled by setting the appropriate bits. When an axis is disabled, the value reported for that axis is always 0.

The *sample rate* and *measurement range* are both defined by the *accelerometer sensitivity* register, 0x 41. The options for both parameters are listed in Table 8-1.

These settings apply in general for all 3 sensor accelerometer functions (orientation detection, motion alarms, and accelerometer assist).

8.1.2.2 Example DL Payloads

- Power on accelerometer and enable all axes:
 - DL payload: **0x C0 87**

- Register 0x 40 with bit 7 set to 1 = 0x C0
 - Value bits 0, 1, 2, and 7 set to 1 = 0x 87
- Set sample rate to 400 Hz and measurement range to 0:
 - DL payload: **0x C1 07**
 - Register 0x 41 with bit 7 set to 1 = 0x C1
 - Sample rate option 7, measurement range option 0 = 0x 07

8.2 Orientation Detection

8.2.1 Operational Description

The sensor orientation is expressed quantitatively as an *acceleration vector*, \vec{a} :

$$\vec{a} = (x, y, z)$$

Where x , y , and z represent the measured acceleration components on each of the 3 axes X, Y, and Z, respectively. These components can be positive or negative, depending on the direction of net acceleration on each axis. The directions of the axes are shown in Figure 8-1.



Figure 8-1: Directions of Axes for ROBIN (left) and FINCH (right).

For a stationary sensor, the total acceleration *magnitude*³⁴ should be approximately 1 *g*; the only net acceleration experienced by a non-moving body is gravity. Therefore, the vector components can be used to determine the direction of gravity and therefore the sensor orientation.

The sensor can be configured to send an accelerometer vector report periodically and/or on demand by pressing the function button. By default, neither periodic nor event-based vector reports are enabled in tracker or beacon mode.

NOTE: The accelerometer must be turned on via register 0x 40 for valid vectors to be reported for both periodic reporting and function button press.

³⁴ Acceleration magnitude is defined as $a = \|\vec{a}\| = \sqrt{x^2 + y^2 + z^2}$

8.2.2 UL Report Frame Formats

Accelerometer vector reports are sent on **LoRaWAN port 10** and have the frame format as shown in Figure 4-2. The specific details for the accelerometer vector report frame format are listed in Table 8-2. For the general description of sensor data report formats and behaviour, see §4.2.1.

Table 8-2: Accelerometer Vector Report UL Frame Format

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Acceleration Vector	0x 00	0x 71	6 B	Acceleration	<ul style="list-style-type: none"> • 1 mg/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>

8.2.2.1 Example UL Payloads

- **0x 00 71 00 00 00 03 E8**
 - Channel ID = 0x 03, Type ID = 0x 72 → acceleration vector data report
 - 0x 00 00 = $0 \times 1 \text{ mg} = 0.000 \text{ g}$ in X-direction
 - 0x 00 00 = $0 \times 1 \text{ mg} = 0.000 \text{ g}$ in Y-direction
 - 0x 03 E8 = $1000 \times 1 \text{ mg} = 1.000 \text{ g}$ in Z-direction
- **0x 00 71 00 00 FB 50 00 00**
 - Channel ID = 0x 03, Type ID = 0x 72 → acceleration vector data report
 - 0x 00 00 = $0 \times 1 \text{ mg} = 0.000 \text{ g}$ in X-direction
 - 0x FB 50 = $-1200 \times 1 \text{ mg} = -1.200 \text{ g}$ in Y-direction
 - 0x 00 00 = $0 \times 1 \text{ mg} = 0.000 \text{ g}$ in Z-direction

8.2.3 Configuration Settings

Table 8-3 shows the list of configuration registers which affect accelerometer vector reporting behaviour. In addition to these registers, the configuration parameters listed in Table 8-1 also affect general accelerometer behaviour. In this table, the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 8-3: Accelerometer Vector Report Configuration Registers

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 20	Seconds per Core Tick	4 B	Access R/W <ul style="list-style-type: none"> • Tick value for periodic events, T • Acceptable values: 15, 16, 17, ..., 86400 • $B \times T \leq 86\,400$, otherwise invalid³⁵ • Other values: Invalid 	60 seconds 0x 00 00 00 3C	<i>seconds_per_core_tick:</i> <i><value></i> <i>(unsigned/s)</i>
0x 24	Ticks per Accelerometer	2 B	Access R/W <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> <i><value></i> <i>(unsigned/no unit)</i>
0x 2A	Data Press Report Configuration	2 B	Access: R/W <ul style="list-style-type: none"> • Bit 0: 0/1 = Battery report disabled/enabled • Bit 1: 0/1 = Acceleration vector report disabled/enabled • Bit 2: 0/1 = MCU temperature report disabled/enabled • Bit 3: 0/1 = BLE location update disabled/enabled • Bit 3 set to 1 is unavailable in Beacon Mode • Bit 4: 0/1 = Device status report disabled/enabled • Bits 5-15: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • BLE report enabled • Device status report enabled 0x 00 18	<i>data_press_report_options:</i> { <i>battery_report:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>accl_vector:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>mcu_temp:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>ble_scan:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>device_status:</i> <i><value></i> <i>(unsigned/no unit)</i> , }

³⁵ This restriction is due to the anti-bricking strategy. See §6.3.1.1.

8.2.3.1 Periodic Reporting Configuration

The accelerometer vector reporting period can be configured using registers 0x 20 and 0x 24 according to the equation described in §6.3.2. That is:

$$\text{Accelerometer Reporting Period} = \text{Seconds per Core Tick} \times \text{Ticks per Accelerometer}$$

Setting the accelerometer reporting period to 0 disables periodic reporting of vector orientation data.

8.2.3.2 Event-Based Reporting Configuration

The accelerometer vector can be configured to be reported upon a function button press by setting bit 1 to 1 in the *Function Button Mode* register, 0x 2A. For details about how the function button works and is configured, see §6.6.

8.2.3.3 Example DL Payloads

- Schedule a vector report every 48 hours:
 - DL payload: **0x A0 00 00 0E 10 A4 00 30**
 - Register 0x 20 with bit 7 set to 1 = 0x A0
 - 3600 s/core tick = 0x 00 00 0E 10
 - Register 0x 24 with bit 7 set to 1 = 0x A4
 - Report every 48 ticks = 0x 00 30
- Report acceleration vector upon function button press:
 - DL payload: **0x AA 02**
 - Register 0x 2A with bit 7 set to 1 = 0x AA
 - Value bit 1 set to 1 = 0x 02

8.3 Motion Detection Alarms

8.3.1 Operational Description

The accelerometer measurements can be used to generate event-based reports when:

- New motion is detected. This is called an ***acceleration event***.
- The motion has ended. This is called an ***acceleration clear***.

Motion Detected:

To determine when motion is detected, ***acceleration triggers*** are used. An acceleration trigger is registered when a certain number of acceleration samples are collected that exceed the trigger threshold, within a certain period after the first above-threshold sample is detected.

Explicitly, 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 Trigger Threshold*) 1 time (register 0x 42, *Acceleration Trigger Threshold Count*) in less than 10 seconds (register 0x 43, *Acceleration Trigger Threshold Period*), an acceleration trigger is registered.

The first trigger to occur after the last motion was cleared is interpreted as new motion detected. When this occurs, an acceleration event is registered and a “motion detected” alarm report is sent in a LoRaWAN UL if alarms are enabled.

Motion Cleared:

To determine when the motion has ended, the configurable *Acceleration Event Grace Period* timer (register 0x 45) is employed. With each newly registered accelerometer trigger, the grace period timer is reset and begins counting down. If the timer elapses without any further registered triggers, the motion is considered ended; an accelerometer clear is registered and a “motion cleared” alarm report is sent in a LoRaWAN UL if alarms are enabled.

No additional “motion detected” alarms are reported before the first acceleration clear is registered, even if more acceleration triggers are registered.

Alarm reporting is enabled by default in tracker mode but can be disabled without turning the accelerometer off. The accelerometer is off by default in beacon mode.

All alarm reporting is event-based; no periodic reporting behaviour is supported.

An example sequence of detected motion and generated alarm ULs is illustrated in Figure 8-2.

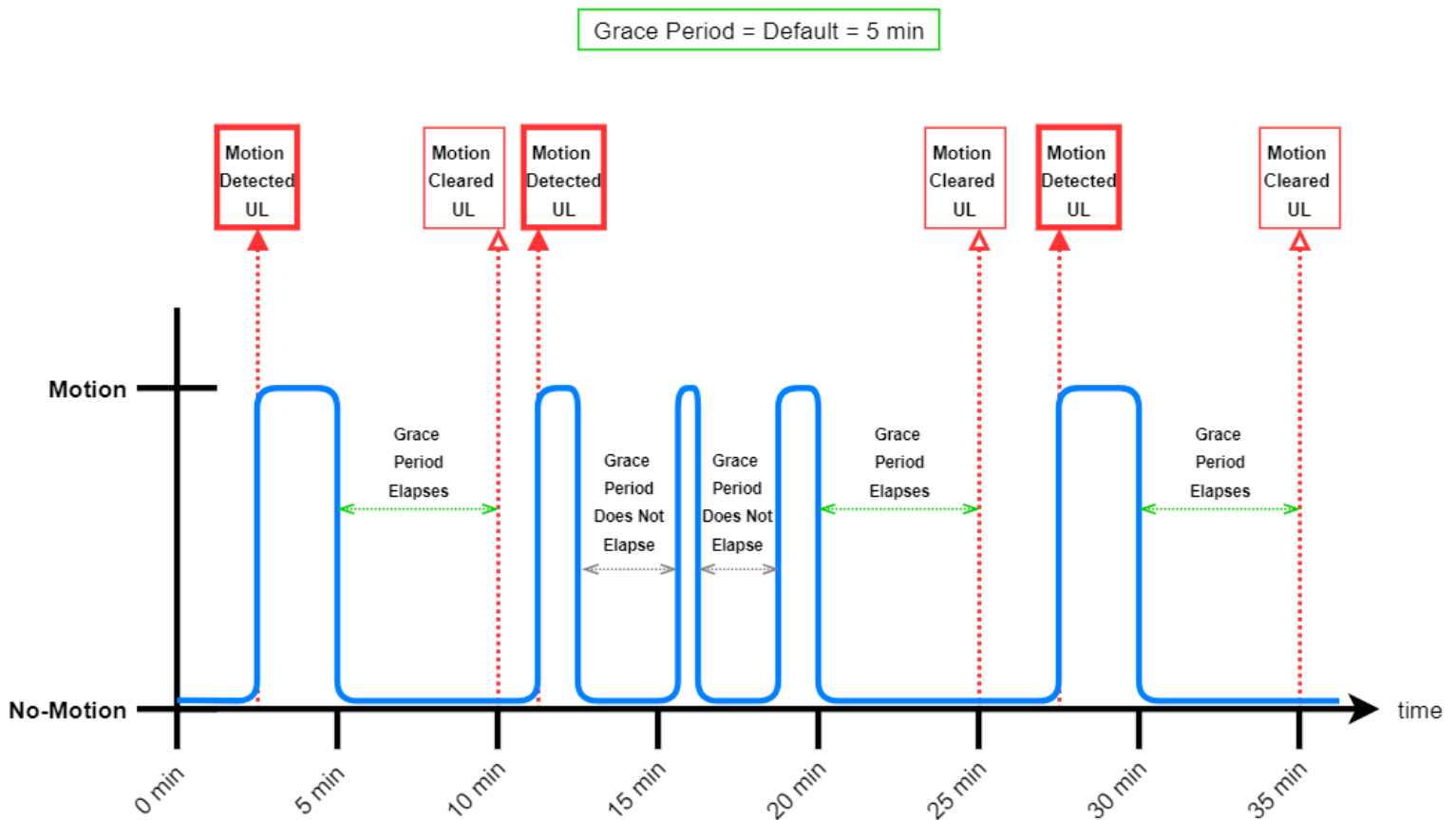


Figure 8-2: Example Sequence of Accelerometer Events and UL Reports

8.3.2 UL Report Frame Formats

Accelerometer alarm reports are sent on **LoRaWAN port 10** and have the frame format as shown in Figure 4-2. The specific details for the alarm report frame format are listed in Table 8-4. For the general description of sensor data report formats and behaviour, see §4.2.1.

Table 8-4: Accelerometer Vector Report UL Frame Format

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Acceleration Alarm Status	0x 00	0x 00	1 B	Digital Input	<ul style="list-style-type: none"> 0x 00 = Alarm inactive (motion no longer detected) 0x FF = Alarm active (motion detected) 	<i>acceleration_alarm</i> (unsigned/no unit)

8.3.2.1 Example UL Payloads

- **0x 00 00 FF**
 - Channel ID = 0x 00, Type ID = 0x 00 → acceleration alarm status report
 - 0x FF = Motion Detected

- **0x 00 00 00**

- Channel ID = 0x 00, Type ID = 0x 00 → acceleration alarm status report
- 0x 00 = Motion Cleared

8.3.3 Configuration Settings

Table 8-5 shows the list of configuration registers which affect accelerometer alarm reporting behaviour. In addition to these registers, the configuration parameters listed in Table 8-1 also affect general accelerometer behaviour. In these tables the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 8-5: Accelerometer Motion Detection Alarm Configuration Registers

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 42	Acceleration Trigger Threshold Count	2 B	Access: R/W <ul style="list-style-type: none"> • Number of above-threshold acceleration samples before an acceleration trigger is registered • Acceptable values: ≥ 1 • 0: Invalid 	1 trigger 0x 00 01	<i>accl_trigger_threshold_count: <value> (unsigned/no unit)</i>
0x 43	Acceleration Trigger Threshold Period	2 B	Access: R/W <ul style="list-style-type: none"> • Period over which acceleration samples are counted for trigger detection [1s/LSb] • Acceptable values: ≥ 5 • 0-4: Invalid 	10 seconds 0x 00 0A	<i>accl_trigger_threshold_period: <value> (unsigned/s)</i>
0x 44	Acceleration Trigger Threshold	2 B	Access: R/W <ul style="list-style-type: none"> • Acceleration magnitude threshold for acceleration trigger [1 mg/LSb] • Acceptable values: 200, 201, 202, ..., 16 000 • Other values: Invalid 	2 g 0x 07 D0	<i>accl_trigger_threshold: <value> (unsigned/mg)</i>

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 45	Acceleration Event Grace Period	2 B	Access: R/W • Time to pass after the last accelerometer trigger before the event is cleared [1s/LSb] • Acceptable values: ≥ 15 • 0-14: Invalid	5 min 0x 01 2C	<i>accl_event_grace_period:</i> <i><value></i> <i>(unsigned/s)</i>
0x 46	Acceleration Event Report Configuration	1 B	Access: R/W • Bit 0: 0/1 = Acceleration alarm UL report disabled/enabled • Bit 1: 0/1 = Accelerometer Assist BLE scan report disabled/enabled • Bit 4: 0/1 = MCU temperature report disabled/enabled • Bit 5: 0/1 = Battery report disabled/enabled • Bits 2-3, 6-7: Must be set to 0, otherwise INVALID	• Acceleration alarm report UL enabled • Accelerometer Assist enabled 0x 03	<i>accl_event_report:</i> { <i>accl_alarms:</i> <value>, <i>(unsigned/no unit)</i> <i>accl_assist_ble:</i> <value>, <i>(unsigned/no unit)</i> <i>mcu_temperature:</i> <value>, <i>(unsigned/no unit)</i> <i>battery_report:</i> <value>, <i>(unsigned/no unit)</i> }

8.3.3.1 Operational Configuration

The *Acceleration Event Value to Tx* register, 0x 46, controls whether motion detection alarm reporting is enabled or disabled. Setting bit 0 to 1 enables alarm reporting. This register also controls whether accelerometer behaviour should result in event-based BLE scans through enabling/disabling the *Accelerometer Assist* feature. See §8.4 for a description of this feature.

The threshold criteria for accelerometer triggers, events, and clears are defined by the *Acceleration Trigger Threshold Count*, *Acceleration Trigger Threshold Period*, *Acceleration Trigger Threshold*, and *Acceleration Event Grace Period* registers. These are registers 0x 42, 0x 43, 0x 44, and 0x 45, respectively.

8.3.3.2 Example DL Payloads

- Change threshold value to 800 mg:
 - DL payload: 0x **C4 03 20**
 - Register 0x 44 with bit 7 set to 1 = 0x C4

- 800 mg = 0x 03 20
- Read *Acceleration Event Value to Tx*:
 - DL payload: **0x 46**
 - Register 0x 46 with bit 7 set to 0 = 0x 46

8.4 Accelerometer Assist for BLE Asset Tracking

8.4.1 Operational Description

In addition to motion detection alarm reporting as described in §8.3, *accelerometer events* and *acceleration clears* can trigger non-periodic BLE scans. That is, when the **Accelerometer Assist** function is enabled, event-based BLE scans and reports are triggered when:

- New motion is detected.
- The motion has ended.

In the indoor asset tracking use-case, Accelerometer Assisted BLE scans applied in conjunction with periodic scans to help get BLE indoor location updates at 2 critical times; when assets leave old locations and settle in new ones.

Accelerometer Assist is triggered under the same configurable conditions as acceleration alarm reporting (namely, the threshold criteria for accelerometer triggers, events, and clears as defined by the *Acceleration Trigger Threshold Count*, *Acceleration Trigger Threshold Period*, *Acceleration Trigger Threshold*, and *Acceleration Event Grace Period* registers).

If Accelerometer Assist is enabled, a single BLE scan and report is triggered by each accelerometer event and by each accelerometer clear. See §8.3 for details on how these are registered. See §7.1 for details about BLE scanning and reporting operation.

Accelerometer Assist is enabled by default in tracker mode and permanently disabled in beacon mode.

NOTE 3: The accelerometer must be enabled via register 0x 40 in order for Accelerometer Assist to cause event based BLE scans and reports. BLE scanning must be enabled via register 0x 50 in order for Accelerometer Assist to work. See §7.1.3 for more details.

8.4.2 UL Report Frame Formats

The BLE results from an Accelerometer Assisted BLE scan have the same report format as all other BLE reports. See §7.1.2 for details.

8.4.3 Configuration Settings

Table 8-6 shows the list of configuration registers which affect accelerometer assist behaviour. In addition to these registers, the configuration parameters listed in Table 8-1 also affect general accelerometer behaviour. In these tables the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 8-6: Accelerometer Assist Configuration Registers

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 42	Acceleration Trigger Threshold Count	2 B	Access: R/W <ul style="list-style-type: none"> Number of above-threshold acceleration samples before an acceleration trigger is registered Acceptable values: ≥ 1 0: Invalid 	1 trigger 0x 00 01	<i>accl_trigger_threshold_count: <value> (unsigned/no unit)</i>
0x 43	Acceleration Trigger Threshold Period	2 B	Access: R/W <ul style="list-style-type: none"> Period over which acceleration samples are counted for trigger detection [1s/LSb] Acceptable values: ≥ 5 0-4: Invalid 	10 seconds 0x 00 0A	<i>accl_trigger_threshold_period: <value> (unsigned/s)</i>
0x 44	Acceleration Trigger Threshold	2 B	Access: R/W <ul style="list-style-type: none"> Acceleration magnitude threshold for acceleration trigger [1 mg/LSb] 	2 g 0x 07 D0	<i>accl_trigger_threshold: <value> (unsigned/mg)</i>
0x 45	Acceleration Event Grace Period	2 B	Access: R/W <ul style="list-style-type: none"> Time to pass after the last accelerometer trigger before the event is cleared [1s/LSb] Acceptable values: ≥ 15 0-14: Invalid 	5 min 0x 01 2C	<i>accl_event_grace_period: <value> (unsigned/s)</i>

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 46	Acceleration Event Report Configuration	1 B	<p>Access: R/W</p> <ul style="list-style-type: none"> • Bit 0: 0/1 = Acceleration alarm UL report disabled/enabled • Bit 1: 0/1 = Accelerometer Assist BLE scan report disabled/enabled • Bit 4: 0/1 = MCU temperature report disabled/enabled • Bit 5: 0/1 = Battery report disabled/enabled • Bits 2-3, 6-7: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> • Acceleration alarm report UL enabled • Accelerometer Assist enabled <p>0x 03</p>	<pre>accl_event_report: { accl_alarms: <value>, (unsigned/no unit) accl_assist_ble: <value>, (unsigned/no unit) mcu_temperature: <value>, (unsigned/no unit) battery_report: <value>, (unsigned/no unit) }</pre>

8.4.3.1 Operational Configuration

The *Acceleration Event Value to Tx* register, 0x 46, controls whether Accelerometer Assist is enabled or disabled. Setting bit 1 to 1 enables Accelerometer Assist. This register also controls whether accelerometer behaviour should result in motion detection reports through the alarm reporting feature. See §8.3 for a description of this feature.

The threshold criteria for accelerometer triggers, events, and clears are defined by the *Acceleration Trigger Threshold Count*, *Acceleration Trigger Threshold Period*, *Acceleration Trigger Threshold*, and *Acceleration Event Grace Period* registers. These are registers 0x 42, 0x 43, 0x 44, and 0x 45, respectively.

8.4.3.2 Example DL Payloads

- Change threshold value to 800 mg:
 - DL payload: 0x **C4 03 20**
 - Register 0x 44 with bit 7 set to 1 = 0x C4
 - 800 mg = 0x 03 20
- Read *Acceleration Event Value to Tx*:
 - DL payload: **0x 46**
 - Register 0x 46 with bit 7 set to 0 = 0x 46

8.5 Device Status Accelerometer Trigger

8.5.1 Operational Description

FINCH/ROBIN supports the ability for accelerometer thresholds to influence the sensor status and behaviour.

Longstanding accelerometer features included in the SW are motion detection alarms and accelerometer assist for BLE. In addition to the legacy features, the threshold can be configured to be used as a trigger for changing device state. This is controlled by bit 5 of register 0x 60.

Accelerometer triggers are disabled by default on FINCH/ROBIN.

- When an accelerometer event is registered according to current configuration settings, the accelerometer status is set to 1, in all cases. If the accelerometer trigger is enabled on register 0x 60, this will also result in a transition to PANIC (if currently in NORMAL).
- When an accelerometer clear is registered according to current configuration settings, the accelerometer status is set to 0. If the accelerometer trigger is enabled on register 0x 60, and all other enabled triggers are clear, this will result in a transition to NORMAL (if currently in PANIC).

Accelerometer events and clears are defined and configured as usual in normal accelerometer behaviour context.

NOTE: Enabling/disabling the accelerometer trigger is mutually exclusive of enabling/disabling accelerometer assist. That is, they any combination of configurations is possible. If both are enabled, then any registered accelerometer clears and events only result in a single BLE scan, rather than 2: 1 from accelerometer assist and 1 from the (possible) state change.

8.5.2 UL Report Frame Formats

The device status reports from an accelerometer trigger have the same report format as all other device status reports. See §5.5 for details.

8.5.3 Configuration Settings

Table 8-7 shows the list of configuration registers which affect accelerometer assist behaviour. In addition to these registers, the configuration parameters listed in Table 8-1 also affect general accelerometer behaviour. In these tables the bit indexing scheme is as shown in Figure 4-1. To access these registers, a command must be formatted and sent according to the details described in §4.3.2.

Table 8-7: Device Status Accelerometer Trigger Configuration Registers

Address	Value	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 42	Acceleration Trigger Threshold Count	2 B	Access: R/W <ul style="list-style-type: none"> • Number of above-threshold acceleration samples before an acceleration trigger is registered • Acceptable values: ≥ 1 • 0: Invalid 	1 trigger 0x 00 01	<i>accl_trigger_threshold_count: <value> (unsigned/no unit)</i>
0x 43	Acceleration Trigger Threshold Period	2 B	Access: R/W <ul style="list-style-type: none"> • Period over which acceleration samples are counted for trigger detection [1s/LSb] • Acceptable values: $\geq 5a$ • 0-4: Invalid 	10 seconds 0x 00 0A	<i>accl_trigger_threshold_period: <value> (unsigned/s)</i>
0x 44	Acceleration Trigger Threshold	2 B	Access: R/W <ul style="list-style-type: none"> • Acceleration magnitude threshold for acceleration trigger [1 mg/LSb] 	2 g 0x 07 D0	<i>accl_trigger_threshold: <value> (unsigned/mg)</i>
0x 45	Acceleration Event Grace Period	2 B	Access: R/W <ul style="list-style-type: none"> • Time to pass after the last accelerometer trigger before the event is cleared [1s/LSb] • Acceptable values: ≥ 15 • 0-14: Invalid 	5 min 0x 01 2C	<i>accl_event_grace_period: <value> (unsigned/s)</i>

Appendix 1 – List of Sensor Data Report UL Frame Formats

LoRaWAN Port 10 UL Frame Formats

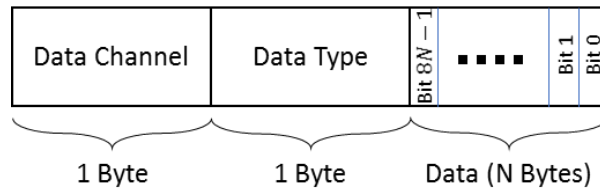


Figure 0-1: The UL Frame Format for a Sensor Data Report

Table 0-1: Complete List of LoRaWAN Port 10 Sensor Data Frame Formats

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Remaining Battery Capacity	0x 00	0x D3	1 B	Percentage	<ul style="list-style-type: none"> 1% / LSB (unsigned) 	<i>rem_batt_capacity: <value> (unsigned/%)</i>
Remaining Battery Lifetime	0x 00	0x BD	2 B	Days	<ul style="list-style-type: none"> 1 day / LSB 	<i>rem_batt_days: <value> (unsigned/days)</i>
Acceleration Alarm Status	0x 00	0x 00	1 B	Digital Input	<ul style="list-style-type: none"> 0x 00 = Alarm inactive (motion no longer detected) 0x FF = Alarm active (motion detected) 	<i>acceleration_alarm (unsigned/no unit)</i>
Acceleration Vector	0x 00	0x 71	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 	<i>acceleration_vector {</i> <i> xaxis: <value>, (signed/g)</i> <i> yaxis: <value>, (signed/g)</i> <i> zaxis: <value> (signed/g)</i> <i>}</i>
MCU Temperature	0x 00	0x 67	2 B	Temperature	<ul style="list-style-type: none"> 0.1°C/LSb (signed) 	<i>temperature (signed/°C)</i>
Device Status	0x 02	0x 95	1 B	Bitmap Input	<ul style="list-style-type: none"> Bit 0: Button status 0/1 = Panic Button currently not activated/activated Bit 3: PAR status 0/1 = Panic App Request currently inactive/active Bit 5: Accelerometer status 0/1 = Accelerometer currently detects stillness/motion Bit 7: Panic status 0/1 = Currently in NORMAL/PANIC state Bits 1-2,4,6: RFU 	<i>device_status: {</i> <i> button_status: <value>, (string/no unit)</i> <i> par_status: <value>, (string/no unit)</i> <i> accl_status: <value> (string/no unit)</i> <i> current_state: <value> (string/no unit)</i> <i> }</i>

LoRaWAN Port 25 UL Frame Formats

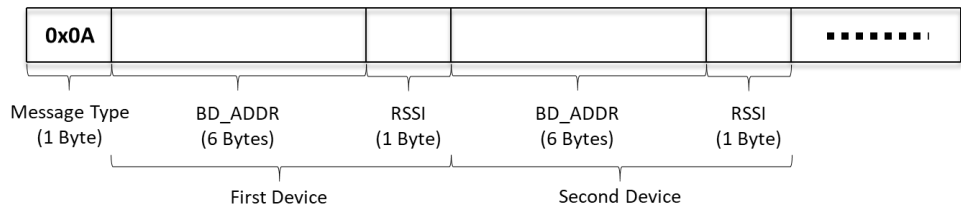


Figure 0-2: The BLE Report UL Payload Format for Basic Reporting

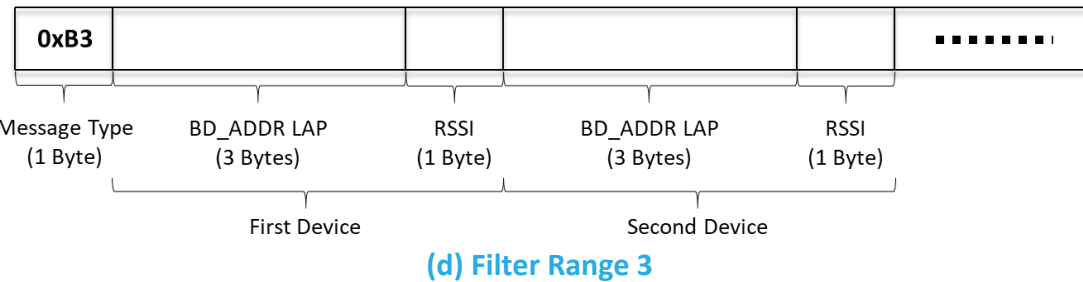
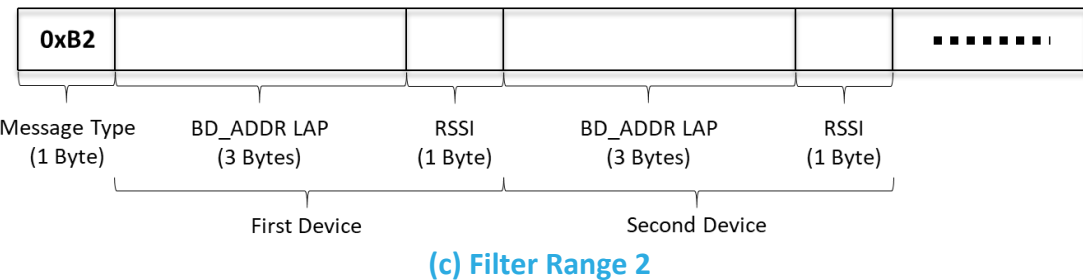
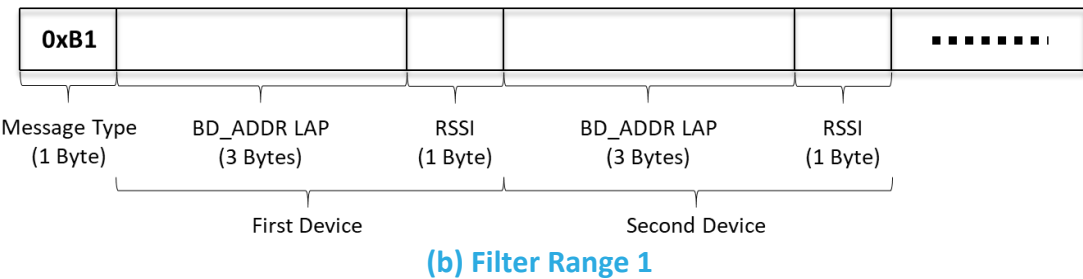
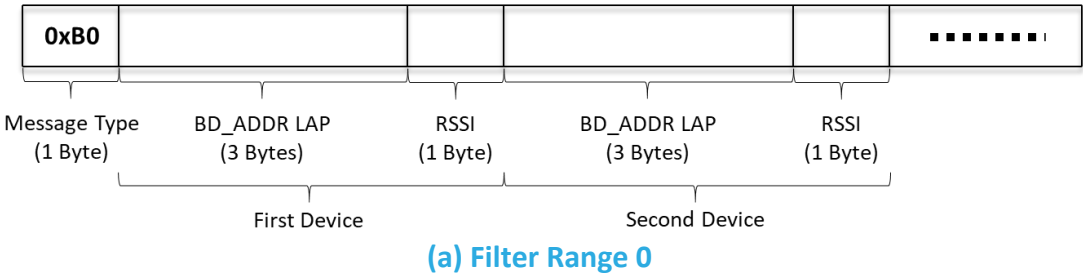


Figure 0-3: BLE Report UL Payload Formats for Filtered Reporting

LoRaWAN Port 100 and 101 UL Frame Formats

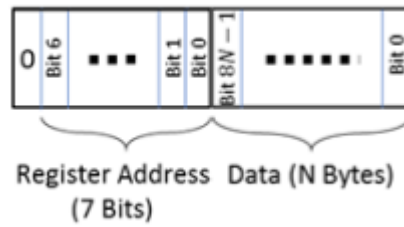


Figure 0-4: The LoRaWAN Port 100 Read Response Format

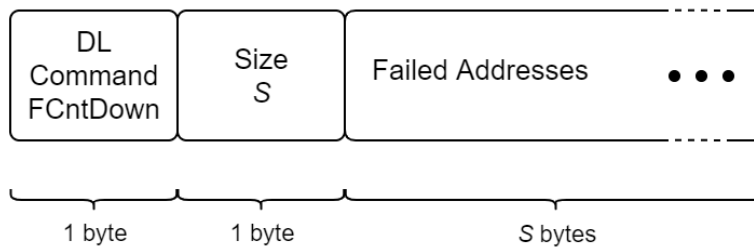
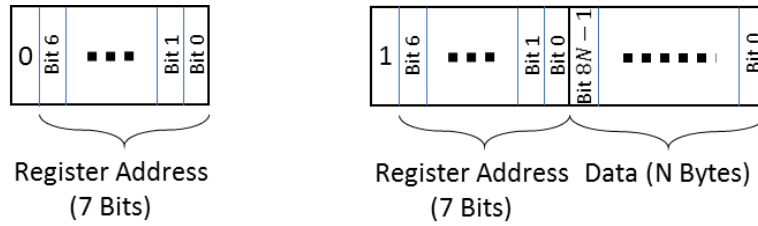


Figure 0-5: The LoRaWAN port 101 Write Response Format

Appendix 2 – Lists of Configuration Registers and DL Payload Formats



a) Read Command Block

(b) Write or Executable Command Block

Figure 0-1: The LoRaWAN Port 100 DL Formats for Configuration and Control Message Blocks

Table 0-1: Complete list of Configuration Registers

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 0A	Tracker / Beacon Mode	1 B	Access: W/X <ul style="list-style-type: none"> 0x 00 = Tracker Mode 0x 01 = Beacon Mode 	Tracker Mode 0x 00	<i>tracker_beacon_mode: <value></i> (unsigned/no unit)
0x 10	Join Mode	2 B	Access: R/W <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: <value></i> (unsigned/no unit)
0x 11	Options	2 B	Access: R/W (bit 1 is RO) <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 = Duty Cycle disabled/enabled³⁶ Bit 3: 0/1 = Disable/Enable ADR Bits 4-15: Ignored 	<ul style="list-style-type: none"> Unconfirmed UL Public Sync Word Duty cycle enabled³⁶ ADR enabled 0x 00 0E	<i>loramac_opts: {</i> <i>confirm_mode: <value></i> , <i>(unsigned/no unit)</i> <i>sync_word: <value></i> , <i>(unsigned/no unit)</i> <i>duty_cycle: <value></i> , <i>(unsigned/no unit)</i> <i>adr: <value></i> <i>(unsigned/no unit)</i> <i>}</i>

³⁶ WARNING: Disabling the duty cycle in certain regions makes the sensor non-compliant with the LoRaWAN Specifications [9]. It is recommended that the duty cycle remains enabled. In the LoRa RF regions where there is no duty cycle limitation, the “enabled duty cycle” configuration is ignored.

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 12	DR and Tx Power ³⁷	2 B	Access: R/W <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"> DR4 Tx Power 0 (max power; see Table 6-2) 0x 04 00	<i>loramac_dr_tx: {</i> <i>dr_number: <value>, (unsigned/no unit)</i> <i>tx_power_number: <value> (unsigned/no unit)</i> <i>}</i>
0x 13	Rx2 Window	5 B	Access: R/W <ul style="list-style-type: none"> Bits 8-39: Channel frequency in Hz for Rx2 Bits 0-7: DR for Rx2 	As per Table 6-2	<i>loramac_rx2: {</i> <i>frequency: <value>, (unsigned/Hz)</i> <i>dr_number: <value> (unsigned/no unit)</i> <i>}</i>
0x 20	Seconds per Core Tick	4 B	Access R/W <ul style="list-style-type: none"> Tick value for periodic events, T Acceptable values: 15, 16, 17, ..., 86400 $B \times T \leq 86400$, otherwise invalid³⁸ Other values: Invalid 	60 seconds 0x 00 00 00 3C	<i>seconds_per_core_tick: <value> (unsigned/s)</i>
0x 21	Ticks per Battery	2 B	Access R/W <ul style="list-style-type: none"> Ticks between battery reports, B Acceptable values: 1, 2, 3, ..., 5760 $B \times T \leq 86400$, otherwise invalid³⁸ Other values: Invalid 	1440 ticks = 1-day period 0x 05 A0	<i>ticks_per_battery: <value> (unsigned/no unit)</i>
0x 23	Ticks per BLE Location Update in PANIC State	2 B	Access R/W <ul style="list-style-type: none"> Ticks between BLE location update in PANIC state Acceptable values: 0, 1, 2, ..., 65535 0 disables periodic BLE reports 	1 tick = 1-minute period 0x 00 01	<i>ticks_per_ble_panic_state: <value> (unsigned/no unit)</i>
0x 24	Ticks per Accelerometer	2 B	Access R/W <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: <value> (unsigned/no unit)</i>

³⁷ Tx power number m translates to the maximum Tx power, which is a function of the LoRaWAN RF region, minus $2 \times m$ dB [8].

³⁸ This restriction is due to the anti-bricking strategy. See §6.3.1.1.

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 25	Ticks per BLE Location Update in NORMAL State	2 B	Access R/W <ul style="list-style-type: none"> Ticks between BLE location update in NORMAL state Acceptable values: 0, 1, 2, ..., 65535 0 disables periodic BLE reports 	60 ticks = 1-hour period 0x 00 3C	<i>ticks_per_ble_normal_state:</i> <i><value></i> <i>(unsigned/no unit)</i>
0x 28	Ticks per MCU Temperature	2 B	Access R/W <ul style="list-style-type: none"> Ticks between temperature reports Acceptable values: 0, 1, 2, ..., 65535 0 disables periodic MCU temperature reports 	Periodic reporting disabled 0x 00 00	<i>ticks_per_mcu_temp:</i> <i><value></i> <i>(unsigned/no unit)</i>
0x 2A	Data Press Report Configuration	2 B	Access: R/W <ul style="list-style-type: none"> Bit 0: 0/1 = Battery report disabled/enabled Bit 1: 0/1 = Acceleration vector report disabled/enabled Bit 2: 0/1 = MCU temperature report disabled/enabled Bit 3: 0/1 = BLE location update disabled/enabled Bit 3 set to 1 is unavailable in Beacon Mode Bit 4: 0/1 = Device status report disabled/enabled Bits 5-15: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> BLE report enabled Device status report enabled 0x 00 18	<i>data_press_report_options:</i> { <i>battery_report:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>accl_vector:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>mcu_temp:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>ble_scan:</i> <i><value></i> , <i>(unsigned/no unit)</i> <i>device_status:</i> <i><value></i> , <i>(unsigned/no unit)</i> , }
0x 2B	Data Press Pattern Configuration	1 B	Access: R/W <ul style="list-style-type: none"> Bits 0-3: Value of <i>m</i> [1 press/LSb] Acceptable values: 1, 2, 3, ..., 15 0: Invalid Bits 4-7: Value of <i>n</i> [1 s/LSb] Acceptable values: 1, 2, 3, ..., 15 0: Invalid 	<ul style="list-style-type: none"> <i>m</i> = 1 press <i>n</i> = 1 s 0x 11	<i>data_press_pattern:</i> { <i>m_value:</i> <i><value></i> , <i>(unsigned/presses)</i> <i>n_value:</i> <i><value></i> , <i>(unsigned/s)</i> }
0x 2C	Panic Press Pattern Configuration	1 B	Access: R/W <ul style="list-style-type: none"> Duration of press and hold to trigger transition to PANIC [1 s/LSb] Acceptable values: 1, 2, 3, 4 Other values: Invalid 	4 s 0x 04	<i>panic_press_pattern:</i> <i><value></i> , <i>(unsigned/s)</i>

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 60	Panic State Trigger Options	1 B	<p>Access: R/W</p> <ul style="list-style-type: none"> Bit 0: 0/1 = Panic Button Trigger disabled/enabled Bit 3: 0/1 = Panic App Request Trigger disabled/enabled Bit 5: 0/1 = Accelerometer Trigger disabled/enabled Bits 0, 3, and 5 all set to 0: State transitions disabled Bits 1-2, 4, 6-7: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> Panic button trigger enabled Panic App Request trigger enabled <p>0x 00 09</p>	<pre>state_trigger_opts: { button_trigger: <value>, (unsigned/no unit) par_trigger: <value>, (unsigned/no unit) accl_trigger: <value> (unsigned/no unit) }</pre>
0x 40	Accelerometer Mode	1 B	<p>Access: R/W</p> <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 on <p>0x 87</p>	<pre>accl_mode: { xaxis_enabled: <value>, (unsigned/no unit) yaxis_enabled: <value>, (unsigned/no unit) zaxis_enabled: <value>, (unsigned/no unit) power_on: <value> (unsigned/no unit) }</pre>
0x 41	Accelerometer Sensitivity	1 B	<p>Access: R/W</p> <ul style="list-style-type: none"> Bits 0-2: Sample Rate: 1/2/3/4/5/6/7 = 1/10/25/50/100/200/400 Hz 0: Invalid and ignored Bits 4-5: Measurement Range: 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 <p>0x 22</p>	<pre>accl_sensitivity: { sample_rate: <value>, (unsigned/Hz) measurement_range: <value> (unsigned/g) }</pre>
0x 42	Acceleration Trigger Threshold Count	2 B	<p>Access: R/W</p> <ul style="list-style-type: none"> Number of above-threshold acceleration samples before an acceleration trigger is registered Acceptable values: ≥ 1 0: Invalid 	<p>1 trigger</p> <p>0x 00 01</p>	<pre>accl_trigger_threshold_count: <value> (unsigned/no unit)</pre>
0x 43	Acceleration Trigger Threshold Period	2 B	<p>Access: R/W</p> <ul style="list-style-type: none"> Period over which acceleration samples are counted for trigger detection [1s/LSb] Acceptable values: ≥ 5 0-4: Invalid 	<p>10 seconds</p> <p>0x 00 0A</p>	<pre>accl_trigger_threshold_period: <value> (unsigned/s)</pre>

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 44	Acceleration Trigger Threshold	2 B	Access: R/W <ul style="list-style-type: none"> Acceleration magnitude threshold for acceleration trigger [1 mg/LSb] 	2 g 0x 07 D0	<i>accl_trigger_threshold: <value> (unsigned/mg)</i>
0x 45	Acceleration Event Grace Period	2 B	Access: R/W <ul style="list-style-type: none"> Time to pass after the last accelerometer trigger before the event is cleared [1s/LSb] Acceptable values: ≥ 15 0-14: Invalid 	5 min 0x 01 2C	<i>accl_event_grace_period: <value> (unsigned/s)</i>
0x 46	Acceleration Event Report Configuration	1 B	Access: R/W <ul style="list-style-type: none"> Bit 0: 0/1 = Acceleration alarm UL report disabled/enabled Bit 1: 0/1 = Accelerometer Assist BLE scan report disabled/enabled Bit 4: 0/1 = MCU temperature report disabled/enabled Bit 5: 0/1 = Battery report disabled/enabled Bits 2-3, 6-7: Must be set to 0, otherwise INVALID 	<ul style="list-style-type: none"> Acceleration alarm report UL enabled Accelerometer Assist enabled 0x 03	<i>accl_event_report: {</i> <i>accl_alarms: <value>, (unsigned/no unit)</i> <i>accl_assist_ble: <value>, (unsigned/no unit)</i> <i>mcu_temperature: <value>, (unsigned/no unit)</i> <i>battery_report: <value>, (unsigned/no unit)</i> <i>}</i>
0x 4A	Battery Report Options	1 B	Access: R/W <ul style="list-style-type: none"> Bit 0: Ignored 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 Bits 3-7: Ignored 	Remaining battery capacity [%] and remaining battery lifetime [days] reported 0x 06	<i>battery_tx: {</i> <i>report_capacity_enabled: <value>, (unsigned/no unit)</i> <i>report_lifetime_enabled: <value> (unsigned/no unit)</i> <i>}</i>
0x 4B	Average Energy Trend Window	1 B	Access: R/W <ul style="list-style-type: none"> Window size over which remaining battery lifetime is estimated [1 core tick/LSB] Acceptable values: 1, 2, ..., 255 0: Invalid 	10 core ticks 0x 0A	<i>avg_energy_trend_window: <value> (unsigned/no unit)</i>

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 50	BLE Rx Mode	1 B	Access: R/W <ul style="list-style-type: none"> Bits 0-6: N Number of reported devices (1–127) 0: Disables BLE Bit 7: 0/1 = Averaging mode off/on 	<ul style="list-style-type: none"> Up to 8 reported devices Averaging mode on 0x 88	<i>ble_mode</i> : { <i>num_reported_devices</i> : <value>, (unsigned/no unit) <i>averaging_mode</i> : <value> (unsigned/no unit) }
0x 51	BLE Scan Duration	2 B	Access: R/W <ul style="list-style-type: none"> Bits 0-7: Scan duration for periodic reports [1 s/LSb] Acceptable values: 1, 2, ..., 10 0: Invalid Bits 8-15: Scan duration for event-based reports [1 s/LSb] Acceptable values: 1, 2, ..., 10 0: Invalid 	<ul style="list-style-type: none"> 3 s for periodic scans 3 s for event-based scans 0x 03 03	<i>ble_scan_duration</i> : { <i>periodic</i> : <value>, (unsigned/no unit) <i>event-based</i> : <value> (unsigned/no unit) }
0x 52	BLE Scan Interval	2 B	Access: R/W <ul style="list-style-type: none"> Scan interval [1 ms/LSb] Acceptable values: “Scan Window”, ..., 10000 Other values: Invalid 	30 ms 0x 00 1E	<i>ble_scan_interval</i> : <value> (unsigned/sec)
0x 53	BLE Scan Window	2 B	Access: R/W <ul style="list-style-type: none"> Scan window [1 ms/LSb] Acceptable values: 3, ..., “Scan Interval” Other values: Invalid 	30 ms 0x 00 1E	<i>ble_scan_window</i> : <value> (unsigned/sec)
0x 54	Filter Range 0	9 B	Access: R/W <ul style="list-style-type: none"> Range 0 for filtered BD_{ADDRS} $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> 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 00	<i>filter_range_0</i> : { <i>oui</i> : <value>, (unsigned/no unit) <i>lap_start</i> : <value>, (unsigned/no unit) <i>lap_end</i> : <value> (unsigned/no unit) }
0x 55	Filter Range 1	9 B	Access: R/W <ul style="list-style-type: none"> Range 1 for filtered BD_{ADDRS} $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> 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 00	<i>filter_range_1</i> : { <i>oui</i> : <value>, (unsigned/no unit) <i>lap_start</i> : <value>, (unsigned/no unit) <i>lap_end</i> : <value> (unsigned/no unit) }

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 56	Filter Range 2	9 B	Access: R/W <ul style="list-style-type: none"> Range 2 for filtered BD_{ADDRS} $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> 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	<i>filter_range_2: {</i> <i>oui: <value>,</i> <i>(unsigned/no unit)</i> <i>lap_start: <value>,</i> <i>(unsigned/no unit)</i> <i>lap_end: <value></i> <i>(unsigned/no unit)</i> <i>}</i>
0x 57	Filter Range 3	9 B	Access: R/W <ul style="list-style-type: none"> Range 3 for filtered BD_{ADDRS} $B_0:B_1:B_2:B_3:B_4:B_5$ to $B_0:B_1:B_2:B_6:B_7:B_8$ <ul style="list-style-type: none"> 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	<i>filter_range_3: {</i> <i>oui: <value>,</i> <i>(unsigned/no unit)</i> <i>lap_start: <value>,</i> <i>(unsigned/no unit)</i> <i>lap_end: <value></i> <i>(unsigned/no unit)</i> <i>}</i>
0x 58	BLE Advertising Enable / Disable	1 B	Access: R/W <ul style="list-style-type: none"> Bits 7-1: Ignored Bit 0: 0/1 = Advertising disabled/enabled 	Advertising disabled 0x 00	<i>advertising_enabled: <value></i> <i>(unsigned/no unit)</i>
0x 59	BLE Advertising Interval	2 B	Access: R/W <ul style="list-style-type: none"> Advertising interval [1 ms/LSb] Acceptable values: 30 ms – 10 000 ms Other values: Invalid 	100 ms 0x 00 64	<i>min_advertising_interval: <value></i> <i>(unsigned/ms)</i>

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 5B	BLE Tx Advertising Power	1 B	Access: R/W <ul style="list-style-type: none"> BLE Tx Power [dBm]³⁹ <ul style="list-style-type: none"> 0x 00 = 0 dBm 0x 01 = -0.85 dBm 0x 02 = -1.8 dBm 0x 03 = -3.15 dBm 0x 04 = -4 dBm 0x 05 = -4.95 dBm 0x 06 = -5.9 dBm 0x 07 = -6.9 dBm 0x 08 = -7.8 dBm 0x 09 = -8.85 dBm 0x 0A = -9.9 dBm 0x 0B = -12.05 dBm 0x 0C = -14.1 dBm 0x 0D = -16.5 dBm 0x 0E = -20.85 dBm 0x 0F = -40 dBm Other values: Invalid 	0 dBm 0x 00	<i>tx_advertising_power: <value> (unsigned/no unit)</i>
0x 5C	BLE Advertisement Packet Format	1 B	Access: R/W <ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> iBeacon enabled Eddystone UID disabled Eddystone TLM disabled 0x 01	<i>advertising_packet_format: { ibeacon: <value>, (unsigned/no unit) eddytone_uid: <value>, (unsigned/no unit) eddytone_tlm: <value> (unsigned/no unit) }</i>
0x 5F	BLE MAC Address	6 B	Access: RO The 6-Byte BLE Device MAC address, <i>BD_ADDR</i> , of the sensor.	Unique for every sensor	<i>mac_address: <value> (unsigned/no unit)</i>
0x 6F	Command Response Option	1 B	Access: R/W <ul style="list-style-type: none"> Bit 0: <ul style="list-style-type: none"> 0: Enhanced LoRaWAN port 100/101 1: Legacy LoRaWAN port 100 Bits 1-7: Ignored 	Legacy LoRaWAN port 100 option 0x 01	<i>command_response_opt: <value> (string/no unit)</i>

³⁹ These are the supported setpoints on Gen2.5 devices. Gen2 devices have different options. See Table 1-1 for how to determine generation.

Address	Name	Size	Description and Options	Default Value	JSON Variable (Type/Unit)
0x 70	Flash Write Command	2 B	Access: X <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 Sensor Bits 1-12, 15: Must be set to 0, otherwise INVALID 	N/A	<pre>write_to_flash { app_config: <value>, (unsigned/no unit) lora_config: <value>, (unsigned/no unit) restart_sensor: <value> (unsigned/no unit) }</pre>
0x 71	FW Metadata	7 B	Access: RO <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 	N/A	<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>
0x 72	Reset Configuration to Factory Defaults	1 B	Access: X <ul style="list-style-type: none"> 0x 0A: Reset App Configuration 0x B0: Reset LoRaMAC Configuration 0x BA: Reset both App and LoRaMAC Configurations Any other value: Invalid 	N/A	<pre>config_factory_reset { app_config: <value>, (unsigned/no unit) loramac_config: <value> (unsigned/no unit) }</pre>
0x 73	BLE Metadata	6 B	Access: RO <ul style="list-style-type: none"> Bits 24-47: RFU Bits 16-23: BLE stack version major Bits 8-15: BLE stack version minor Bits 0-7: BLE stack version revision 	N/A	<pre>ble_metadata { stack_ver_major: <value>, (unsigned/no unit) stack_ver_minor: <value>, (unsigned/no unit) stack_ver_revision: <value>, (unsigned/no unit) }</pre>

Table 0-2: Comparison of Default Configuration Values for Tracker and Beacon Mode

Name	Register Address [Hex]	Default Value, Tracker Mode [Hex]	Default Value, Beacon Mode [Hex]	Category
Tracker/Beacon Mode	0A	00: Tracker Mode is Default		BLE
Join Mode	10	80 00	80 00	LoRaMAC
LoRaMAC Options	11	00 0E	00 0E	
LoRaMAC DR and Tx Power	12	04 00	00 00	
LoRaMAC Rx2 Window	13	As per Table 6-2		
Seconds per Core Tick	20	00 00 00 3C	00 00 00 3C	Periodic Reporting
Ticks per Battery	21	05 A0	05 A0	
Ticks per BLE Location Update in PANIC State	23	00 01	00 00	
Ticks per Accelerometer	24	00 00	00 00	
Ticks per BLE Location Update in NORMAL State	25	00 3C	00 00	
Ticks per MCU Temperature	28	00 00	00 00	
Data Press Report Configuration	2A	00 18	00 01	Function Button
Data Press Pattern Configuration	2B	11	11	
Panic Press Pattern Configuration	2C	04	04	
Panic State Trigger Options	60	00 09	00 00	
Mode	40	87	07	Accelerometer
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	01	
Battery Report Options	4A	06	06	Battery Management
Average Energy Trend Window	4B	0A	0A	
Mode	50	88	88	BLE Scanning (Tracker Mode)
Scan Duration	51	03 03	03 03	
Scan Interval	52	00 1E	00 1E	
Scan Window	53	00 1E	00 1E	
Filter Range 0	54	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
Filter Range 1	55	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
Filter Range 2	56	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
Filter Range 3	57	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	
Advertising Enable / Disable	58	00	01	BLE Advertising (Beacon Mode)
Advertising Interval	59	00 64	00 64	
Tx Advertising Power	5B	00	00	
Advertisement Packet Format	5C	01	01	
BLE MAC Address	5F	Unique for every device		

Command Response Option	6F	01	01	Command Response Options
-------------------------	----	----	----	--------------------------------

References

- [1] TEKTELIC Communications Inc., "KONA ATLAS," TEKTELIC Communications Inc., August 2023. [Online]. Available: <https://www.atlas.tektelic.com/>. [Accessed August 2023].
- [2] TEKTELIC Communications Inc., "Knowledge Hub," TEKTELIC Communications Inc., 2024. [Online].
- [3] LoRa Alliance, "TS001-1.0.4 LoRaWAN® L2 1.0.4 Specification," October 2020. [Online]. Available: <https://lora-alliance.org/wp-content/uploads/2021/11/LoRaWAN-Link-Layer-Specification-v1.0.4.pdf>.
- [4] LoRa Alliance, "LoRaWAN® Regional Parameters RP002-1.0.4," September 2022. [Online]. Available: <https://resources.lora-alliance.org/technical-specifications/rp002-1-0-4-regional-parameters>.
- [5] Jauch Marketing Team, "Passivation of Lithium Thionyl Chloride Batteries," Jauch Quartz GmbH, 20 May 2020. [Online]. Available: <https://www.jauch.com/blog/en/passivation-lithium-thionyl-chloride-batteries/>. [Accessed 28 11 2022].
- [6] Bluetooth SIG, "Core Specification 5.3," 13 July 2021. [Online]. Available: <https://www.bluetooth.com/specifications/specs/core-specification-5-3/>.
- [7] TEKTELIC Communications Inc., "Gen2 BLE Sensor Family User Guide," TEKTELIC, Calgary, 2021.
- [8] LoRa Alliance, "LoRaWAN Regional Parameters," v1.0.2, rev. B, Feb 2017.
- [9] LoRa Alliance, Inc., "LoRaWAN Specification 1.0.2," July 2016. [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/lorawan1_0_2-20161012_1398_1.pdf.