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

Smart Room Sensor

Technical Reference Manual

DOCUMENT TYPE:	Technical Reference Manual
DOCUMENT NUMBER:	T0006338_TRM
DOCUMENT VERSION:	2.3
PRODUCT NAME:	Smart Room Sensor
PRODUCT CODE:	T0006115 (Base Module, NA) T0006116 (PIR Module, NA) T0006117 (Base Module, EU) T0006118 (PIR Module, EU) T0006161 (Base Module, CN) T0006162 (PIR Module, CN) T0006163 (Base Module, DN) T0006164 (PIR Module, DN)
RELEASE DATE:	March 26, 2021

Revision History

Version	Date	Editor	Comments
0.1	Jun 10, 2019	Reza Nikjah	Initial draft for Smart Room Sensors (Gen 3 Home Sensors).
0.2	Jun 13, 2019	Reza Nikjah	Modified based on feedback, including, • typos and oversights; • 2 bytes instead of 1 byte for analog input threshold; • moving Accelerometer Configuration section ahead of Threshold Configuration section, so the register addresses appear in a reasonable order; • change of “Sample Rate & Measurement Range” for the Accelerometer to “Sensitivity” • change of register name “Threshold” to “Threshold Control” for Light; • change of bitmapping of the Light “Threshold Control” register; • change of bitmapping of the PIR “Mode” register.
0.3	Jun 19, 2019	Reza Nikjah	• Minor edits: ○ Categorized accelerometer events into “acceleration events” and “impact alarm events” ○ Corrected description of the accelerometer events debounce time ○ Corrected possible light threshold values and number of bits devoted to the light threshold ○ Corrected Rx2 frequency for DN915 ○ Removed the default register value for configuration registers, and instead, *described* the default values
0.4	Jul 10, 2019	Reza Nikjah	• Edits based on feedback, including, ○ Clarified the grouping in Table 2-1 ○ Clarified the External Connector configuration ○ Clarified Accelerometer configuration ○ showed current draws for different accelerometer sample rates and moisture sample periods
0.5	Jul 15, 2019	Reza Nikjah	• Corrected typos • Edits based on feedback, including, ○ Added register 0x11 Sync Word bit ○ Corrected Acceleration Event Debounce Time description ○ Explained limits on the Sample Periods
0.6	Oct 16, 2019	Reza Nikjah	• Minor edits (Table 2-1) • Clarified that moisture detection is specific to Base
1.0	Jan 2, 2020	Reza Nikjah, Mark Oevering	• Updated acceleration vector example • Updated the document format • Changed the bit indexing scheme from bytes and bits to only bits, with the rightmost bit indexed 0, for more clarity • Minor edits to read/write examples

			• Added acceptable values, wherever applicable, to the tables of configuration registers
1.1	May 6, 2020	Reza Nikjah	• Added Jason variables for UL and DL encoding and decoding • Some editorial changes
1.2	Nov 23, 2020	Lukas Morrison	• Updated expected current draws for moisture transducer sampling (Section 3.2.8.1).
1.3	Nov 24, 202	Reza Nikjah	• Added description of register 0x54 to control PIR hold-off intervals.
2.0	Nov 26, 2020	Reza Nikjah	• Enhanced accelerometer configuration to support the toilet leakage use case.
2.1	Dec 7, 2020	Reza Nikjah	• Removed analytical event (AE) alarm uplinks. In the case of threshold being exceeded, the AE value is just reported (i.e. a threshold-based reporting, rather than an alarm reporting). • Rearranged bits and bytes in registers 0x64 and 0x65.
2.2	Jan 4, 2021	Reza Nikjah	• Redefined register 0x32 to decouple reporting values for accelerometer periodic reporting and acceleration event reporting
2.3	Mar 26, 2021	Reza Nikjah	• Added configuration commands for the water flow detection use case. • Removed duration median from enhanced accelerometer configuration due to its limited use and challenging calculation for streaming data. • Removed duration range from enhanced accelerometer configuration due to its redundancy in view of duration min and duration max. • Removed duration mean from enhanced accelerometer configuration due to its redundancy in view of duration total and duration count. • Replaced CIC filter with a simple IIR filtering that only has a single recall factor parameter.

Table of Contents

List of Tables

List of Figures

Acronyms and Glossary

ABP .... activation by personalization
ADC .... analog-to-digital converter
ADR .... adaptive data rate
AE .... analytical event
CRC .... cyclic redundancy check
DL .... downlink
DLim .... duration limit
DR .... data rate
DRan .... duration range
EIRP .... effective isotropic radiated power
Flash memory .... Non-volatile memory on the Room Sensor (contains application & configuration settings)
g .... gravity (unit of acceleration ≈ 9.8 m/s^2)
HPF .... high-pass filter
ID .... identity
IoT .... Internet of things
LoRa .... a patented “long-range” IoT technology acquired by Semtech
LoRAMAC .... LoRaWAN MAC
LoRaWAN .... LoRa wide area network (a network protocol based on LoRa)
LoRaWAN Commissioning..... The unique device identifiers and encryption keys used for LoRaWAN communication (see LoRaWAN Specification [1] for more details).
LPF .... low-pass filter
LSB .... least significant bit
MAC .... medium access control
MCU .... microcontroller unit
min .... minute(s)
ML .... magnitude level
MSB .... most significant bit
NS .... network server
OTA .... over-the-air
OTAA .... OTA activation
PIR .... passive infrared
RH .... relative humidity
RFU .... reserved for future use
RO .... read-only
Room Sensor .... A Smart Room Sensor module
R/W .... read/write
Rx .... receiver

SD .... standard deviation
sec .... second(s)
Sensor .... Room Sensor
transducer .... sensing element on the Room Sensor (e.g. PIR or temperature transducers)
TRM .... technical reference manual (this document)
Tx .... transmitter
UL .... uplink
WO .... write-only

1 Overview

This TRM describes the user accessible configuration settings (pseudo registers) supported by the Lora IoT Smart Room Sensor (or Generation 3 of All-in-One LoRa IoT Home Sensors), referred to as the Room Sensor or the Sensor henceforth. This document is intended for a technical audience, such as application developers, with an understanding of the NS and its command interfaces.

The Room Sensor is a multi-purpose LoRaWAN IoT sensor packed into a very small form factor. The Room Sensor is ideal for monitoring and reporting temperature (ambient, remote through a probe, or MCU), humidity, light, shock, and open/closed doors and windows in the home environment. Additional sensing features such as leak and motion detection, as well as counting pulses from an external device are also supported with the appropriate Room Sensor model. Table 1-1 presents the current generation of the Room Sensor family. Also, Table 1-2 lists the Room Sensor variants for regions identified by the LoRa Alliance [2]—see [2] for the Tx and Rx bands in each LoRaWAN region. The DN915 regional variant is similar to US915 in Tx band, but in Rx it works in the 722-728 MHz band.

Table 1-1: Smart Room Sensor Family

Part Number			Description
Level 1	Level 2	Level 3	
T0006115			Smart Room Sensor Module, LoRa IoT, NA Base
T0006116			Smart Room Sensor Module, LoRa IoT, NA PIR
T0006117			Smart Room Sensor Module, LoRa IoT, EU Base
T0006118			Smart Room Sensor Module, LoRa IoT, EU PIR
T0006161			Smart Room Sensor Module, LoRa IoT, CN Base
T0006162			Smart Room Sensor Module, LoRa IoT, CN PIR
T0006163			Smart Room Sensor Module, LoRa IoT, DN Base
T0006164			Smart Room Sensor Module, LoRa IoT, DN PIR
	T0006003		Smart Room Sensor Bottom, LoRa IoT
	T0006107		Smart Room Sensor Top, LoRa IoT, Base
	T0006149		Smart Room Sensor Top, LoRa IoT, PIR
	T0006132		Smart Room Sensor PCBA, LoRa IoT, NA Base
	T0006133		Smart Room Sensor PCBA, LoRa IoT, NA PIR
	T0006151		Smart Room Sensor PCBA, LoRa IoT, EU Base
	T0006152		Smart Room Sensor PCBA, LoRa IoT, EU PIR
	T0006156		Smart Room Sensor PCBA, LoRa IoT, CN Base
	T0006157		Smart Room Sensor PCBA, LoRa IoT, CN PIR
	T0006154		Smart Room Sensor PCBA, LoRa IoT, DN Base
	T0006155		Smart Room Sensor PCBA, LoRa IoT, DN PIR
		T0006131	Smart Room Sensor PCB, LoRa IoT

Table 1-2: Smart Room Sensor Region Specific Variants

LoRaWAN RF Variant	Functional Variant	HW Variant	Order Code
EU868	Base	EU Base	SMTBBEU868
EU868	PIR	EU PIR	SMTPBEU868
US915	Base	NA Base	SMTBBUS915
US915	PIR	NA PIR	SMTPBUS915
AS923	Base	NA Base	SMTBBAS923
AS923	PIR	NA PIR	SMTPBAS923
AU915	Base	NA Base	SMTBBAU915
AU915	PIR	NA PIR	SMTPBAU915
IN865	Base	EU Base	SMTBBIN865
IN865	PIR	EU PIR	SMTPBIN865
CN470	Base	CN Base	SMTBBCN470
CN470	PIR	CN PIR	SMTPBCN470
KR920	Base	NA Base	SMTBBKR920
KR920	PIR	NA PIR	SMTPBKR920
RU864	Base	EU Base	SMTBBRU864
RU864	PIR	EU PIR	SMTPBRU864
DN915	Base	DN Base	SMTBBDISH
DN915	PIR	DN PIR	SMTPBDISH

Regarding communication direction (UL or DL) and LoRaWAN ports, all information streams currently supported by the SW are as follows:

- Readings obtained from on-board transducers (***sent in UL, LoRaWAN port 10***)
- Configuration and control commands from the NS used to change the Sensor's behavior or inquire the Sensor for the values of registers (***sent in DL, LoRaWAN port 100***)
- Response to configuration and control commands from the NS (***sent in UL, LoRaWAN port 100***)

The default configuration of the Sensor for reporting transducer readings includes the following:

- Report the battery voltage every hour
- Report the ambient temperature every hour
- Report the ambient RH every hour
- Report actuation (an open-to-close or close-to-open event) of the reed switch every 1 (one) actuation
- Report actuation (an open-to-close or close-to-open event) of the digital input every 1 (one) actuation
- Report motion after one PIR event (PIR model only)
- Clear motion after 5 min of no motions (PIR model only)

In the following sections, the UL (departing from the Sensor) and DL (destined to the Sensor) payload formats are explained. Refer to the *Smart Room Sensor Uplink and Downlink Frame Payloads* spreadsheet [3] for a thorough tool to build any UL or DL frame payload by varying parameter values, toggling read/write actions, and enabling/disabling different fields as desired.

2 UL Payload Formats

The UL streams (from the Sensor to the NS) include the following:

- The readings obtained from on-board transducers (***sent on LoRaWAN port 10***)
- Response to configuration and control commands from the NS (***sent on LoRaWAN port 100***)

These topics are explained in Sections 2.1 and 2.2, respectively.

2.1 Frame Payload to Report Transducers Data

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

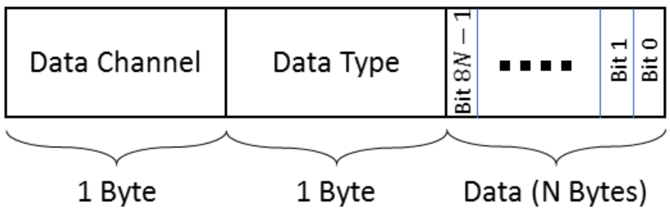


Figure 2-1: The UL frame payload format.

A Sensor message payload can include multiple transducer data frames. The ordering of frames is not guaranteed (they can be in any order). A single payload may include data from any given transducer. The Room Sensor payload frame values are shown in Table 2-1. In this table, the bit indexing scheme is as shown in Figure 2-1. Payload frame values in Table 2-1 has been grouped by bolded boundaries. This grouping is only to indicate which payloads are related to the same physical transducer; it *does not imply* that the payloads within the same group are uplinked together.

Transducer data in the UL are sent through ***LoRaWAN port 10***.

Table 2-1: UL Frame Payload Values for Transducer Data

Information Type	Channel ID	Type ID	Size	Data Type	Data Format	JSON Variable (Type/Unit)
Battery Voltage	0x00	0xFF	2 B	Analog	• 10 mV / LSB (signed)	*battery_voltage: <value> (signed/volt)*
Reed Switch State	0x01	0x00	1 B	Digital	• 0x00 = Low—magnet present • 0xFF = High—magnet absent	*reed_state: <value> (unsigned/no unit)*
Reed Switch Count	0x08	0x04	2 B	Counter	• Number	*reed_count: <value> (unsigned/no unit)*

Impact Alarm	0x0C	0x00	1 B	Digital	• 0x00 = Impact alarm inactive • 0xFF = Impact alarm active	*impact_alarm: <value>* *(unsigned/no unit)*
Acceleration Magnitude	0x05	0x02	2 B	Analog	• 1 milli-g/LSB (unsigned)	*impact_magnitude: <value>* *(unsigned/g)*
Acceleration Vector	0x07	0x71	6 B	Acceleration	• 1 milli-g/LSB (signed) • Bits 32-47: X-axis acceleration • Bits 16-31: Y-axis acceleration • Bits 0-15: Z-axis acceleration	*acceleration {* *xaxis: <value>,* *(signed/g)* *yaxis: <value>,* *(signed/g)* *zaxis: <value>* *(signed/g)* *}*
External Connector: Digital Input State	0x0E	0x00	1 B	Digital	• 0x00 = Low—Connector short-circuited • 0xFF = High—Connector open-circuited	*extconnector_state: <value>* *(unsigned/no unit)*
External Connector: Digital Input Count	0x0F	0x04	2 B	Counter	• Number	*extconnector_count: <value>* *(unsigned/no unit)*
External Connector: Analog Input[1]	0x11	0x02	2 B	Analog	• 1 mV/LSB (signed)	*extconnector_analog: <value>* *(signed/V)*
MCU Temperature	0x0B	0x67	2 B	Temperature	• 0.1°C / LSB (signed)	*mcu_temperature: <value>* *(signed/°C)*
Ambient Temperature	0x03	0x67	2 B	Temperature	• 0.1°C / LSB (signed)	*ambient_temperature: <value>* *(signed/°C)*
Ambient RH	0x04	0x68	1 B	RH	• 0.5% / LSB	*relative_humidity: <value>* *(unsigned/1%)*
Ambient Light State	0x02	0x00	1 B	Digital	• 0x00 = Dark • 0xFF = Bright	*light_detected: <value>* *(unsigned/no unit)*

[1] Voltage value, to be converted to temperature for a remote temperature probe using a conversion table or formula.

Ambient Light Intensity	0x10	0x02	1 B	Analog	• Uncalibrated digitized light intensity • Values: 0, 1, …, 64	*light_intensity: <value> (unsigned/no unit)*
Motion (PIR) Event State	0x0A	0x00	1 B	Digital	• 0x00 = No motion • 0xFF = Motion detected	*motion_event_state: <value> (unsigned/no unit)*
Motion (PIR) Event Count	0x0D	0x04	2 B	Counter	• Number	*motion_event_count: <value> (unsigned/no unit)*
Moisture	0x09	0x00	1 B	Digital	• 0x00 = Dry • 0xFF = Wet	*moisture: <value> (unsigned/no unit)*
Analytical Event†	$b_8b_7b_6b_5$ $b_4b_3b_2b_1$ †	0x10	2 B	Duration Total	• Total duration, 1 sec/LSB	*ae_duration_total: <value> (unsigned/sec)*
		0x11	2 B	Duration Min	• Min duration, 1 sec/LSB	*ae_duration_min: <value> (unsigned/sec)*
		0x12	2 B	Duration Max	• Max duration, 1 sec/LSB	*ae_duration_max: <value> (unsigned/sec)*
		0x13	2 B	Duration SD	• Duration SD, 1 sec/LSB	*ae_duration_sd: <value> (unsigned/sec)*
		0x14	2 B	Count	• Count, 1 count/LSB	*ae_count: <value> (unsigned/no unit)*

† An “analytical event (AE)” is identified by a magnitude level (ML) and a duration range (DRan)—see Section 3.2.9. The ML is addressed by $b_8b_7b_6b_5$ (1 for ML_1, 2 for ML_2, …, 15 for ML_{15}), DRan by $b_4b_3b_2b_1$ (0 for $DRan_0$, 1 for $DRan_1$, 2 for $DRan_2$, 3 for $DRan_3$). For example, channel ID 0x11 is the channel for the $AE_{1,1}$, i.e., the AE of (ML_1, $DRan_1$).

Examples:

In the following example payloads, the data channel ID and data type ID are boldfaced:

- 0x **03 67** 00 0A **04 68** 28
  - 0x **03 67** (Ambient Temperature) = (0x 00 0A) × 0.1°C = 1°C
  - 0x **04 68** (Ambient RH) = (0x 28) × 0.5% = 20%
- 0x **04 68** 14 **01 00** FF **08 04** 00 05
  - 0x **04 68** (Ambient RH) = (0x 14) × 0.5% = 10%
  - 0x **01 00** (Reed Switch State) = 0x FF = Magnet absent
  - 0x **08 04** (Reed Switch Count) = 0x 00 05 = 5 switch triggers
- 0x **04 68** 2A **03 67** FF FF **00 FF** 01 2C
  - 0x **04 68** (Ambient RH) = (0x 2A) × 0.5% = 21%

- $0x\ \mathbf{03\ 67}\ (\text{Ambient Temperature}) = (0x\ \text{FF FF}) \times 0.1°\text{C} = -0.1°\text{C}$
- $0x\ \mathbf{00\ FF}\ (\text{Battery Voltage}) = (0x\ 01\ 2\text{C}) \times 0.01\ \text{V} = 3.00\ \text{V}$

- 0x **02 00** FF **0E 00** 00
  - $0x\ \mathbf{02\ 00}\ (\text{Light State}) = 0x\ \text{FF} = \text{Bright}$
  - $0x\ \mathbf{0E\ 00}\ (\text{Digital Input State}) = 0x\ 00 = \text{Connector short–circuited}$
- 0x **0D 04** 00 02
  - $0x\ \mathbf{0D\ 04}\ (\text{Motion Event Count}) = 0x\ 00\ 02 = 2$ motion events
- 0x **30 10** 00 42
  - $0x\ \mathbf{30\ 10}\ (\text{Duration Total for analytical event of } \text{ML}_3 \text{ and } \text{DRan}_0) = 0x\ 00\ 42 = 66$ sec

2.2 Response to Configuration and Control Commands

Sensor responses to DL configuration and control commands (which are sent on ***LoRaWAN port 100***; see Section 3) are sent in the UL on ***LoRaWAN port 100***. These responses include the following:

- Returning the value of a configuration register in response to an inquiry from the NS.
- Writing to a configuration register.

In the former case, the Sensor responds by the address and value of each of the registers under inquiry (this can be in one or more consecutive UL packets depending on the maximum frame payload size allowed). In the latter case, the Sensor responds with a CRC32 of the entire DL payload (which may be a combination of read and write commands) as the first 4 bytes of the UL frame. If the DL payload has also had read commands, the 4 CRC32 bytes are followed by the address and value of each of the registers under inquiry (similar to the Sensor response in the former case).

3 DL Payload Formats

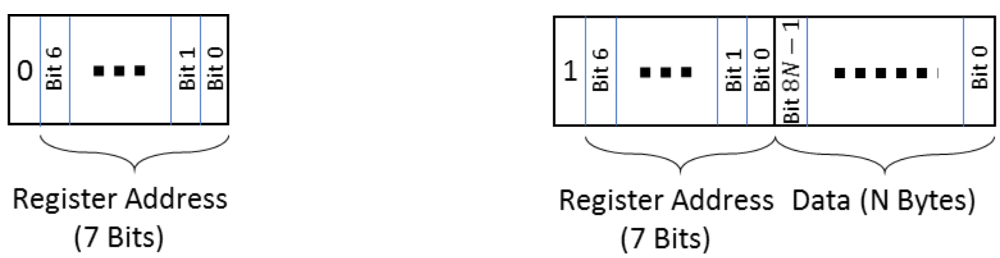
The only DL message (from the NS to the Sensor) supported by the SW includes the following:

- Configuration and control commands used to change the Sensor's behavior or inquire the Sensor for the values of registers (***sent on LoRaWAN port 100***).

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

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

Bit 7 of the first byte determines whether a read or write action is being performed, as shown in Figure 3-1. All read commands are one-byte long. Data following a read access command will be interpreted as a new command block. Read commands are processed last. For example, in a single DL message, if there is a read command from a register and a write command to the same register, the write command is executed first.



(a) The read command. (b) The write command.

Figure 3-1: The format of a DL configuration and control message block.

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

When a write command is sent to the Sensor, the Sensor immediately responds with a CRC32 of the entire DL payload as the first 4 bytes of the UL frame on ***LoRaWAN port 100*** (also see Section 2.2).

DL configuration and control commands fall into one of the following four categories and are discussed in Sections 3.1, 3.2, and 3.2.9, respectively:

- LoRaMAC Configuration
- Application Configuration
- Command and Control

3.1 LoRaMAC Configuration

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

Table 3-1: LoRaMAC Configuration Registers

Address	Value	Size	Description	JSON Variable (Type/Unit)
0x10	Join Mode	2 B	• Bit 15: 0/1 = ABP/OTAA mode • Bits 0-14: Ignored	*loramac_join_mode: <value>* *(unsigned/no unit)*
0x11	Options	2 B	• Bit 0: 0/1 = Unconfirmed/Confirmed UL • Bit 1 = 1 (RO): 0/1 = Private/Public Sync Word • Bit 2: 0/1 = Disable/Enable Duty Cycle • Bit 3: 0/1 = Disable/Enable ADR • Bits 4-15: Ignored	*loramac_opts {* *confirm_mode: <value>,* *(unsigned/no unit)* *sync_word: <value>,* *(unsigned/no unit)* *duty_cycle: <value>,* *(unsigned/no unit)* *adr: <value>* *(unsigned/no unit)* *}*
0x12	DR and Tx Power[2]	2 B	• Bits 8-11: Default DR number • Bits 0-3: Default Tx power number • Bits 4-7, 12-15: Ignored	*loramac_dr_tx {* *dr_number: <value>,* *(unsigned/no unit)* *tx_power_number: <value>,* *(unsigned/no unit)* *}*
0x13	Rx2 Window	5 B	• Bits 8-39: Channel frequency in Hz for Rx2 • Bits 0-7: DR for Rx2	*loramac_rx2 {* *frequency: <value>,* *(unsigned/Hz)* *dr_number: <value>* *(unsigned/no unit)* *}*

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

Examples:

- Switch Sensor to ABP Mode:

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

    - DL payload: { 0x **90** 00 00 }
- Set ADR enabled, no duty cycle, and confirmed UL payloads:
    - DL payload: { 0x **91** 00 0B }
- Set default DR number to 1 and default Tx power number to 2:
    - DL payload: { 0x **92** 01 02 }

3.1.1 Default Configuration

Table 3-2 and Table 3-4 show the default values for the LoRaMAC configuration registers (cf. [1]).

Table 3-2: Default Values of LoRaMAC Configuration Registers

Address	Default Value
0x10	• OTAA mode
0x11	• Unconfirmed UL • Duty cycle enabled[3] • ADR enabled
0x12	• DR0 • Tx Power 0 (max power; see Table 3-3)
0x13	• As per Table 3-4

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

RF Region	Max Tx EIRP [dBm]
EU868	16
US915	30
AS923	16
AU915	30
IN865	30
CN470	19.15
KR920	14
RU864	16
DN915	30

Table 3-4: Default Values of Rx2 Channel Frequency and DR Number in Different Regions

RF Region	Channel Frequency [Hz]	DR Number
EU868	869525000	0
US915	923300000	8
AS923	923200000	2

[3] In the LoRa RF regions where there is no duty cycle limitation, such as US915, the “enabled duty cycle” configuration of the Sensor is ignored.

AU915	923300000	8
IN865	866550000	2
CN470	505300000	0
KR920	921900000	0
RU864	869100000	0
DN915	725900000	8

3.2 Application Configuration

This section lists all possible application configurations (as part of DL configuration and control commands), including periodic Tx configuration and configurations of the different transducers.

3.2.1 Periodic Tx Configuration

All periodic transducer reporting is synchronized around *ticks*. A *tick* is simply a user configurable time-base that is used to schedule transducer measurements. For each transducer, the number of elapsed *ticks* before transmitting can be defined as shown in Table 3-5. All the registers in this table have R/W access.

Note: Certain transducer types, such as accelerometer and light, need to be enabled for periodic reporting. Details are available in each transducer's respective section.

Table 3-5: Periodic Transmission Configuration Registers

Address	Value	Size	Description	JSON Variable (Type/Unit)
0x20	Seconds per Core *Tick*	4 B	• *Tick* value for periodic events • Acceptable values: 0, 60, 61, …, 86400 • 0 disables all periodic transmissions • Other values: Invalid and ignored	*seconds_per_core_tick: <value>* *(unsigned/sec)*
0x21	*Ticks* per Battery	2 B	• *Ticks* between battery reports • 0 disables periodic battery reports	*tick_per_battery: <value>* *(unsigned/no unit)*
0x22	*Ticks* per Ambient Temperature	2 B	• *Ticks* between ambient temperature reports • 0 disables periodic ambient temperature reports	*tick_per_ambient_temperature: <value>* *(unsigned/no unit)*
0x23	*Ticks* per Ambient RH	2 B	• *Ticks* between ambient RH reports • 0 disabled periodic ambient RH reports	*tick_per_relative_humidity: <value>* *(unsigned/no unit)*
0x24	*Ticks* per Reed Switch	2 B	• *Ticks* between reed switch reports • 0 disables periodic reed switch reports	*tick_per_reed_switch: <value>* *(unsigned/no unit)*
0x25	*Ticks* per Ambient Light	2 B	• *Ticks* between ambient light reports • 0 disables periodic ambient light reports	*tick_per_light: <value>* *(unsigned/no unit)*
0x26	*Ticks* per Accelerometer	2 B	• *Ticks* between accelerometer reports	*tick_per_accelerometer: <value>* *(unsigned/no unit)*

	(both Acceleration and Impact Alarm)		• 0 disables periodic accelerometer reports	
0x27	*Ticks* per MCU Temperature	2 B	• *Ticks* between MCU temperature reports • 0 disables periodic MCU temperature reports	*tick_per_mcu_temperature: <value>* *(unsigned/no unit)*
0x28	*Ticks* per Motion (PIR)/Moisture	2 B	• *Ticks* between motion (PIR)/moisture reports • 0 disables periodic motion (PIR)/moisture reports	*tick_per_pir: <value>* *(unsigned/no unit)*
0x29	*Ticks* per External Connector (Digital/Analog Input)	2 B	• *Ticks* between external connector (digital/analog input) reports • A value of 0 disables periodic external connector (digital/analog input) reports	*tick_per_external_connector: <value>* *(unsigned/no unit)*

3.2.1.1 Seconds per Core Tick

All periodic Tx events are scheduled in *ticks*. This allows for transducer reads to be synchronized, reducing the total number of ULs required to transmit Sensor data. The minimum seconds per *tick* is 60 sec, and the maximum is 86,400 sec (one day). Values from 1 sec to 59 sec and values above 86,400 sec are invalid and ignored. A value of 0 (zero) disables all periodic reporting.

3.2.1.2 Ticks per <Transducer>

This register sets the reporting period for a transducer in terms of *ticks*. Once the configured number of *ticks* has expired, the Sensor polls the specified transducer and reports the data in an UL message. A setting of 0 (zero) disables periodic reporting for the specified transducer.

3.2.1.3 Default Configuration

Table 3-6 shows the default values for the periodic transmission configuration registers.

Table 3-6: Default Values of Periodic Transmission Configuration Registers

Seconds per Core *tick*	3600 (1 hour)
***Ticks* per Battery**	1 (thus 1-hour period)
***Ticks* per Ambient Temperature**	1 (thus 1-hour period)
***Ticks* per Ambient RH**	1 (thus 1-hour period)
***Ticks* per Reed Switch**	0 (periodic Tx disabled)
***Ticks* per Ambient Light**	0 (periodic Tx disabled)
***Ticks* per Accelerometer**	0 (periodic Tx disabled)
***Ticks* per MCU Temperature**	0 (periodic Tx disabled)

***Ticks* per PIR/Moisture**	0 (periodic Tx disabled)
***Ticks* per Digital/Analog Input**	0 (periodic Tx disabled)

Examples:

- Disable all periodic events:
  - DL payload: { 0x **A0** 00 00 00 00 }
    - Register 0x20 with the write bit set to true
    - Seconds per *Tick* set to 0 (zero)—i.e. disable periodic transmissions
- Read current value of Seconds per *Tick*:
  - DL payload: { 0x **20** }
    - Register 0x20 with the write bit set to false
- Report Temperature every *tick* and RH every two *ticks*:
  - DL payload: { 0x **A2** 00 01 **A3** 00 02 }
    - Registers 0x22 and 0x23 with their write bits set to true
    - Temperature *Ticks* set to 1 (one)
    - RH *Ticks* set to 2 (two)

3.2.1.4 Anti-Bricking Strategy

Care has been taken to avoid stranding (hard or soft bricking) the Sensor during reconfiguration. Hard bricking refers to the condition that the Sensor does not transmit anymore as all periodic and event-based reporting (see subsequent sections) have been disabled and the configuration has been saved to the Flash memory. Soft bricking refers to the condition where the Sensor has been configured such that all event-based reporting is disabled and any periodic reporting is either disabled or has a period of larger than a week.

To avoid these situations, for any reconfiguration command sent to the Sensor, the following algorithm is executed:

After the reconfiguration is applied, if all event-based reporting (as explained in subsequent sections) is disabled, then periodic reporting is checked. If all periodic reporting is disabled or the minimum non-zero period is greater than a week, then to avoid bricking the Sensor, the core *tick* is set to 86,400 (i.e. one day), and the battery *tick* is set to 1 (one).

3.2.2 Reed Switch Configuration

Table 3-7 shows a list of Reed Switch configuration registers. All registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-7: Reed Switch Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x2A	Mode	1 B	• Bit 0:	*reed_mode {* *rising_edge_enabled: <value>,*

			0/1 = Rising edge disabled/enabled • Bit 1: 0/1 = Falling edge disabled/enabled • Both bits 0 and 1 set to 0: Invalid and ignored Bits 2-7: Ignored	*(unsigned/no unit)* *falling_edge_enabled: <value>* *(unsigned/no unit)* *}*
0x2B	Count Threshold	2 B	• Number of triggers for event transmission • 0 disables event transmission	*reed_switch_count_threshold: <value>* *(unsigned/no unit)*
0x2C	Value to Tx	1 B	• Bit 0: 0/1 = Input state not reported/reported • Bit 1: 0/1 = Counter value not reported/reported • Both bits 0 and 1 set to 0: Invalid and ignored • Bits 2-7: Ignored	*reed_tx {* *report_state_enabled: <value>,* *(unsigned/no unit)* *report_count_enabled: <value>,* *(unsigned/no unit)* *}*

3.2.2.1 Mode

The Reed Switch is edge-triggered and can be set to trigger to rising-edge trigger (Low or Closed to High or Open), falling-edge triggered (High or Open to Closed or Low) or both. An attempt to set the Mode to 0x00 (i.e. to disable both rising and falling edges) is ignored by the Sensor.[4]

Application Example:

Door Open/Close detection would use both rising and falling triggers to detect when the door was opened and when it was closed.

3.2.2.2 Count Threshold

The Count Threshold determines when the Sensor transmits after seeing an event on the Reed Switch. A value of 0 (zero) disables the event driven transmission, while a value of 1 (one) or greater triggers an event-based transmission after the configured number of events has occurred, which is when the event “counter” reaches the value of the Count Threshold. Whenever such event-based transmission occurs, the event counter is automatically reset to 0 and starts incrementing as events occur until the counter reaches the threshold again and another event-based transmission occurs.

Application Example:

[4] Input pulse frequency must be less than 5 Hz.

If a sensor is intended to monitor room utilization, it may be configured either to disable event-based transmission in favor of getting hourly reports from the sensor, or to only transmit after 50 "events" logged in the room. The latter may be useful for alerting cleaning staff that room requires attention.

3.2.2.3 Value to Tx

The Value to Tx determines what information is transmitted whenever an event or periodic digital transmission is required. If the value is "Counter Value", the transmission contains the number of times the Reed Switch was triggered since the last transmission, while the value of "Input State" causes a transmission of the current input state of the switch (i.e. Open or Closed).

3.2.2.4 Default Configuration

Table 3-8 shows the default values for the Reed Switch configuration registers.

Table 3-8: Default Values of Reed Switch Configuration Registers

Mode	Rising and falling edges enabled
Threshold	1 (one)
Value to Tx	State and count reported

Examples:

- Have Reed Switch be triggered only on rising edges:
  - DL payload: { 0x **AA** 01 }
    - Register 0x2A with write bit set to true
    - "Rising Edge" enabled, "Falling Edge" disabled
- Read current value of Count Threshold:
  - DL payload: { 0x **2B** }
    - Register 0x2B with write bit set to false
- Transmit the Reed Switch "state" as soon as the Reed Switch is tripped 10 times:
  - DL payload: { 0x **AB** 00 0A **AC** 01 }
    - Registers 0x2B and 0x2C with their write bits set to true
    - Count Threshold set to 10
    - Value to Tx set to "Input State"
- Disable the Reed Switch event-driven transmission, but report the number of times the Reed Switch has been triggered whenever a report is inquired (i.e. in the case of periodic reporting):
  - DL payload: { 0x **AB** 00 00 **AC** 02 }
    - Count Threshold set to 0 (zero)
    - Value to Tx set to "Counter Value"

3.2.3 External Connector Configuration

Only the Base variant of the Room Sensor is equipped with an external connector. This connector can be configured as either a digital input (having only two values or states of “open” and “closed”), or analog input. The input mode (digital or analog) is determined by bit 7 of register 0x2D (see Table 3-9). The input mode is digital by default.

In the digital input mode, the external connector has only two values or states: open (open-circuited) with a value of 0xFF, and closed (short-circuited) with a value of 0x00. For example, in this mode of operation, the external connector can be used for leak detection. This mode of operation supports periodic (Section 3.2.1) and event-based (edge triggered) reporting (see the following subsections).

In the analog input mode, one pin is grounded, and the other pin is pulled up to VMCU (1.8 V) by a 68.1-*k*Ω resistor. The analog input has values in units of mV from 0 to VMCU (the precision is 1 mV[5]), and is suitable for connection to a thermistor (recommended 10-*k*Ω) as a remote temperature probe. The actual temperature can be obtained from the value of the analog input and a provided conversion table or formula (see [4] for such conversion methods and formulas). This mode of operation supports periodic (Section 3.2.1) and threshold-based reporting (Section 3.2.5).

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

Table 3-9: External Connector Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x2D	Mode	1 B	• Bit 0: 0/1 = Rising edge disabled/enabled • Bit 1: 0/1 = Falling edge disabled/enabled • Both bits 0 and 1 set to 0: Invalid and ignored • Bits 2-6: Ignored. • Bit 7: 0/1 = Digital/Analog Input mode	*external_connector {* *rising_edge_enabled: <value>, (unsigned/no unit)* *falling_edge_enabled: <value>, (unsigned/no unit)* *mode: <value> (unsigned/no unit)* *}*
0x2E	Count Threshold	2 B	• Number of triggers for event transmission • 0 disables event transmission	*external_connector_count_threshold: <value> (unsigned/no unit)*
0x2F	Value to Tx	1 B	• Bit 0: 0/1 = Digital Input state not reported/reported	*external_connector_tx {* *report_state_enabled: <value>, (unsigned/no unit)*

---

[5] The actual ADC output has a resolution of 0.61 mV.

			• Bit 1: 0 = Digital Input count not reported/reported • Both bits 0 and 1 set to 0: Invalid and ignored • Bits 2-7: Ignored	*report_count_enabled: <value> (unsigned/no unit)* *}*

3.2.3.1 Mode

In the Digital Input mode (bit 7 = 0), the input is edge-triggered and can be set to be triggered by the rising edge (Low or Closed to High or Open), falling edge (High or Open to Closed or Low), or both. An attempt to set the Mode to 0x00 (i.e. to disable both rising and falling edges) is ignored by the Sensor.[6]

In the Analog Input mode (bit 7 = 1), bits 0-6 of register 0x2D, and the entire registers 0x2E and 0x2F are irrelevant and ignored. The configuration registers in the Analog Input mode only include 0x29 (see Section 3.2.1), 0x44, 0x45, 0x46, and 0x4A (see Section 3.2.4).

Application Examples for Digital Input Mode:

- Door Open/Close detection would use both rising and falling triggers to detect when the door was opened and when it was closed.
- Pulse counting from a water meter would use a single edge trigger, depending on the resting state of the connected device (positive pulse would use rising edge, negative pulse would use falling edge).

3.2.3.2 Count Threshold

The Count Threshold is only applicable in the Digital Input mode, and determines when the Sensor transmits after seeing an event on Digital Input. A value of 0 (zero) disables the event driven transmission, while a value of 1 (one) or greater triggers an event-based transmission after the configured number of events has occurred, which is when the event "counter" reaches the value of the Count Threshold. Whenever such event-based transmission occurs, the event counter is automatically reset to 0 and starts incrementing as events occur until the counter reaches the threshold again and another event-based transmission occurs.

Application Examples:

If a sensor is intended to pulse count from a high-volume water meter, it may be configured to disable event-based transmission in favor of getting hourly reports from the sensor.

- If a sensor is intended to monitor room utilization it may be configured to only transmit after 100 "events" logged in the room. This may be useful for alerting cleaning staff that room requires attention.

[6] Input pulse frequency must be less than 3 Hz.

3.2.3.3 Value to Tx

The Value to Tx is only applicable in the Digital Input mode, and determines what information is transmitted whenever a Digital Input event or periodic transmission is required. If the value is Digital Input state, the transmission contains the current Digital Input state of the switch (i.e., 0xFF for open or 0x00 for closed). If the value is the Digital Input count, the transmission contains the number of times that the Digital Input was triggered since the last transmission.

3.2.3.4 Default Configuration

Table 3-10 shows the default values for the External Connector configuration registers.

Table 3-10: Default Values of External Connector Configuration Registers

Mode	Digital Input mode with rising and falling edges enabled
Threshold	1 (one)
Value to Tx	State and count reported for Digital Input

Examples:

- Have Digital Input be triggered only on falling edges:
    - DL payload: { 0x **AD** 02 }
        - Register 0x2D with write bit set to true
        - "Rising Edge" disabled, "Falling Edge" enabled
- Read current value of Count Threshold:
    - DL payload: { 0x **2E** }
        - Register 0x2E with write bit set to false
- Transmit the Digital Input state as soon as the Digital Input is tripped 20 times:
    - DL payload: { 0x **AE** 00 14 **AF** 01 }
        - Registers 0x2E and 0x2F with their write bits set to true
        - Count Threshold set to 20
        - Value to Tx set to "Input State"
- Disable the Digital Input event-driven transmission, but report the number of times the Digital Input has been triggered whenever a report is inquired (i.e. in the case of periodic reporting):
    - DL payload: { 0x **AE** 00 00 **AF** 02 }
        - Count Threshold set to 0 (zero)
        - Value to Tx set to "Digital Input count"

3.2.4 Accelerometer Configuration

The accelerometer transducer offers a threshold for an "impact alarm event"[7], and a threshold for an "acceleration event". It can also be polled periodically for applications where the Sensor orientation may be of interest. Table 3-11 shows a list of accelerometer configuration registers. All registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Some terminology in this section is as follows:

- Accelerometer (transducer) refers to the accelerometer transducer component.
- Impact alarm (event) refers to an accelerometer event based on exceeding an impact alarm event threshold. Impact alarm events are reported with an impact alarm.
- Acceleration (event) refers to an accelerometer event, independent of the impact alarm event, and based on exceeding an acceleration event threshold. Acceleration events are reported with the acceleration magnitude, acceleration vector, or both.

Table 3-11: Accelerometer Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x30	Impact Alarm Event Threshold	2 B	• Unsigned, 1 milli-g/LSB	*impact_event_threshold* *(unsigned/g)*
0x31	Acceleration Event Threshold	2 B	• Unsigned, 1 milli-g/LSB	*acceleration_event_threshold: <value>* *(unsigned/g)*
0x32	Value to Tx	1 B	• Bit 0 (applicable to periodic reporting[8]): 0/1 = Impact alarm not reported/reported • Bit 1 (applicable to periodic reporting): 0/1 = Acceleration magnitude not reported/reported • Bit 2 (applicable to periodic reporting): 0/1 = Acceleration vector not reported/reported	*accelerometer_tx {* *report_periodic_alarm_enabled: <value>,* *(unsigned/no unit)* *report_periodic_magnitude_enabled: <value>,* *(unsigned/no unit)* *report_periodic_vector_enabled : <value>* *(unsigned/no unit)*

---

[7] Here "impact" generally refers to a Sensor motion event (i.e., not necessarily an *impact* to the Sensor).

[8] This bit only controls whether the impact alarm status (i.e., raised or cleared) will be present in periodic reporting when such accelerometer periodic reporting is enabled (see Section 3.2.1). This bit does not control reporting of the impact alarm status for impact alarm events. If the impact alarm event threshold is enabled (register 0x34, bit 0), an impact alarm is always raised (reported) when the impact alarm event threshold (register 0x30) is exceeded, and is cleared after an impact alarm event grace period (register 0x36) elapses without any impact alarm events (see Section 3.2.4.7).

			• Bit 4 (applicable to acceleration event reporting): 0/1 = Acceleration magnitude not reported/reported • Bit 5 (applicable to acceleration event reporting): 0/1 = Acceleration vector not reported/reported • Bits 3, 6, 7: Ignored	*report_event_magnitude_enabled: <value>,* *(unsigned/no unit)* *report_event_vector_enabled: <value>* *(unsigned/no unit)* *}*
0x33	Acceleration Event Debounce Time	2 B	• Seconds to wait before possibly reporting an acceleration event again • Acceptable values: 1, 2, …, 65535 • 0: Invalid and ignored	*acceleration_impact_grace_period: <value>* *(unsigned/seconds)*
0x34	Mode	1 B	• Bit 0: 0/1 = Impact alarm event threshold disabled/enabled • Bit 1: 0/1 = Acceleration event threshold disabled/enabled • Bits 2, 3: Ignored • Bit 4/5/6: 0/1 = X/Y/Z-axis disabled/enabled • Bit 7: 0/1 = Accelerometer power off/on	*accelerometer {* *impact_threshold_enabled: <value>,* *(unsigned/no unit)* *acceleration_threshold_enabled: <value>,* *(unsigned/no unit)* *xaxis_enabled: <value>,* *(unsigned/no unit)* *yaxis_enabled: <value>,* *(unsigned/no unit)* *zaxis_enabled: <value>,* *(unsigned/no unit)* *poweron: <value>* *(unsigned/no unit)* *}*
0x35	Sensitivity	1 B	• Bits 0-2 (Sample Rate): 0: Invalid and ignored 1/2/3/4/5/6/7 = 1/10/25/50/100/200/400 Hz	*sensitivity {* *accelerometer_sample_rate: <value>,* *(unsigned/Hz)* *accelerometer_measurement_range: <value>,*

			• Bits 4-5 (Measurement Range[9]): 0/1/2/3 = ±2 *g*/±4 *g*/±8 *g*/±16 *g* • Bits 3, 6, 7: Ignored	*(unsigned/g)* *}*
0x36	Impact Alarm Event Grace Period	2 B	• Impact alarm grace period in sec (time to pass after the last impact alarm before the alarm can be cleared) • Acceptable values: 15, 16, …, 65535 • Other values: Invalid and ignored	*impact_alarm_grace_period:* *<value>* *(unsigned/seconds)*
0x37	Impact Alarm Event Threshold Count	2 B	• Number of impact alarm events before an impact alarm is raised • Acceptable values: 1, 2, …, 65535 • 0: Invalid and ignored	*impact_alarm_threshold_count:* *<value>* *(unsigned/no unit)*
0x38	Impact Alarm Event Threshold Period	2 B	• Period in sec over which impact alarm events are counted for threshold detection • Acceptable values: 5, 6, …, 65535 • Other values: Invalid and ignored	*impact_alarm_threshold_period:* *<value>* *(unsigned/seconds)*

3.2.4.1 Impact Alarm Event Threshold

This parameter is the g-threshold for an impact alarm event. Impact alarm events are reported only if,

- the impact alarm event threshold (bit 0 of register 0x34) is enabled; and
- the impact alarm event threshold is exceeded on at least one of the enabled axes (X, Y, Z) within a period (Impact Alarm Event Threshold Period—register 0x38) for more than a number of times (Impact Alarm Event Threshold Count—register 0x37).

3.2.4.2 Acceleration Event Threshold

This parameter is the g-threshold for an acceleration event. Provided that the acceleration threshold is enabled (bit 1 of register 0x34), acceleration events are reported as soon as the Acceleration Event Threshold is exceeded on at least one of the enabled axes (X, Y, Z). However, acceleration event interrupts are totally ignored (not registered) for a time period equal to the Acceleration Event Debounce Time (register 0x33) after a registered (and thus reported) acceleration event.

[9] Measurement ranges ±2 *g*, ±4 *g*, ±8 *g*, ±16 *g* correspond to typical transducer output precisions of 16 m*g*, 32 m*g*, 64 m*g*, 192 m*g*, respectively. Note that if a threshold configured in register 0x30 or register 0x31 is equal to or greater than the configured measurement full scale (2 *g*, 4 *g*, 8 *g*, 16 *g*), then the corresponding event (impact alarm or acceleration event) will never be triggered.

3.2.4.3 Value to Tx

Determines what is reported (transmitted) in the case of accelerometer periodic transmission or an acceleration event. The parameters to report include the status of the impact alarm (alarm on/off), the acceleration magnitude $\|\langle x, y, z\rangle\| = \sqrt{x^2 + y^2 + z^2}$, and the acceleration vector $\langle x, y, z\rangle$.

3.2.4.4 Acceleration Event Debounce Time

Interrupts due to acceleration events are disabled for a configurable time frame, called the Acceleration Event Debounce Time, after an acceleration event is registered. This is done to prevent a single acceleration event from being transmitted as multiple events. The minimum debounce time is 1 (one) sec. Value 0 is invalid and ignored.

3.2.4.5 Mode

When not being used in an end-user application, the accelerometer transducer can be put in the power-down mode to save battery life. Otherwise, the accelerometer is put in the low-power mode, which is an active and operational, but a low consumption, mode for the accelerometer.

Additionally, impact alarm and acceleration event thresholds can be enabled/disabled. Disabling a threshold prevents the Sensor from generating the corresponding event. It is also possible to enable/disable X, Y, Z axes independently. When an axis is disabled, it is not considered in monitoring impact alarm or acceleration events.

3.2.4.6 Sensitivity

When powered on, the accelerometer always samples the transducer element at a fixed rate, called the Sample Rate. To capture an impact alarm or acceleration event, the physical event needs to last longer than the sample period. Larger sample rates have a shorter period and can therefore resolve shorter impacts. However, sampling the transducer at a larger rate increases the power usage, impacting the battery life. Table 3-12 shows how much continuous current draw is expectable to be drawn from a 3.2-V battery for the different sample rates when the accelerometer is powered on. For example, the sample rate of 1 Hz would translate to about 15 mAh/year battery usage, while a sample rate of 50 Hz would triple that usage.

Table 3-12: Typical Current Draws at 3.2 V for Different Accelerometer Sample Rates

Sample Rate [Hz]	1	10	25	50	100	200	400
Current Draw [μA]	1.6	2.3	3.1	4.7	7.8	14.1	28.1

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

3.2.4.7 Impact Alarm Event Grace Period

The Grace Period determines how long the Room Sensor waits before the previously reported impact alarm event is considered clear. For example, a Grace Period of 5 (five) min results in the sensor transmitting "Impact Detected" when there is movement, and "Impact Alarm Cleared" 5 (five) min after the Sensor has been still.

The minimum acceptable value for this register is 15. Values smaller than 15 are invalid and ignored.

3.2.4.8 Impact Alarm Event Threshold Count

The accelerometer generates an impact alarm event each time it detects movement. Depending on the customer use case, it may be desirable to increase the threshold count to reduce sensitivity. This feature is to allow customers to filter out short impact events, while still allowing longer impact events to be reported.

The minimum acceptable value for this register is 1. Value 0 is invalid and ignored.

3.2.4.9 Impact Alarm Event Threshold Period

The Impact Alarm Event Threshold Period is the amount of time that impact alarm events are accumulated for threshold detection. For example, an Impact Alarm Event Threshold Period of 10 (ten) sec accumulates impact alarm events over a 10 (ten)-sec period from the time of first detection. If the Impact Alarm Event Threshold Count is reached before the time expires, the sensor reports "Impact Detected", otherwise it does not report.

The minimum acceptable value for this register is 5. Values smaller than 5 are invalid and ignored.

3.2.4.10 Default Configuration

Table 3-13 shows the default values for the accelerometer configuration registers.

Table 3-13: Default Values of Accelerometer Configuration Registers

Impact Alarm Event Threshold	1500 milli-g
Acceleration Event Threshold	3000 milli-g
Value to Tx	Acceleration vector for periodic and acceleration-event reports
Acceleration Event Debounce Time	2 sec
Mode	• Impact alarm threshold disabled • Acceleration threshold disabled • X-axis, Y-axis, and Z-axis enabled • Accelerometer power off
Sensitivity	• Sample rate 1 Hz • Measurement range ±8 g
Impact Alarm Event Grace Period	300 sec (5 min)
Impact Alarm Event Threshold Count	1
Impact Alarm Event Threshold Period	15 sec

3.2.5 Temperature/RH/Analog Input Threshold Configuration

The Room Sensor supports threshold transmission on four different transducer values:

- Ambient temperature: Measured by the Temperature/RH transducer
- Ambient RH: Measured by the Temperature/RH transducer
- MCU Temperature: Measured by the MCU (with lower accuracy compared to the Ambient Temperature)
- Analog Input Voltage: When the External Connector is in the Analog Input mode.

When a threshold on a transducer is enabled, the Sensor reports the transducer value when it leaves the configured threshold window, and once again when the transducer value re-enters the threshold window[10]. The Threshold mode is compatible with periodic reporting. Table 3-14 shows a list of configuration registers for the temperature/RH/Analog Input threshold setting. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-14: Temperature/RH/Analog Input Threshold Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x39	Ambient Temperature/RH Sample Period: Idle	4 B	• Sample period of Ambient Temperature/RH transducer: Idle state (sec) • Acceptable values: 30, 31, …, 86400 • Other values: Invalid and ignored	*temperature_relative_humidity_sample_period_idle: <value> (unsigned/sec)*
0x3A	Ambient Temperature/RH Sample Period: Active	4 B	• Sample period of Ambient Temperature/RH transducer: Active state (sec) • Acceptable values: 30, 31, …, 86400 • Other values: Invalid and ignored	*temperature_relative_humidity_sample_period_active: <value> (unsigned/sec)*
0x3B	Low/High Ambient Temperature Thresholds	2 B	• Bits 8-15: High temperature threshold (signed, 1°C / LSB) • Bits 0-7: Low temperature threshold (signed, 1°C / LSB) • High threshold ≤ Low threshold: Invalid and ignored	*ambient_temperature_threshold { high: <value> (signed/°C) low: <value> (signed/°C) }*
0x3C	Ambient Temperature Thresholds Enabled	1 B	• Bit 0: 0/1 = Thresholds disabled/enabled • Bits 1-7: Ignored	*ambient_temperature_threshold_enabled: <value> (unsigned/no unit)*

[10] Note that the threshold window here is defined as the open interval "(Low Threshold, High Threshold)", not e.g. the closed interval "[Low Threshold, High Threshold]"; i.e. even if the transducer value is equal to Low Threshold or High Threshold, the Sensor is considered to have left the threshold window.

0x3D	Low/High Ambient RH Thresholds	2 B	• Bits 8-15: High RH threshold (unsigned, 1% RH / LSB) • Bits 0-7: Low RH threshold (unsigned, 1% RH / LSB) • High threshold ≤ Low threshold: Invalid and ignored	*relative_humidity_threshold {* *high: <value>,* *(unsigned/%)* *low: <value>* *(unsigned/%)* *}*
0x3E	Ambient RH Thresholds Enabled	1 B	• Bit 0: 0/1 = Thresholds disabled/enabled • Bits 1-7: Ignored	*relative_humidity_threshold_enabled: <value>* *(unsigned/no unit)*
0x40	MCU Temperature Sample Period: Idle	4 B	• Sample period of MCU temperature transducer: Idle state (sec) • Acceptable values: 30, 31, …, 86400 • Other values: Invalid and ignored	*mcu_temperature_sample_period_idle: <value>* *(unsigned/sec)*
0x41	MCU Temperature Sample Period: Active	4 B	• Sample period of MCU temperature transducer: Active state (sec) • Acceptable values: 30, 31, …, 86400 • Other values: Invalid and ignored	*mcu_temperature_sample_period_active: <value>* *(unsigned/sec)*
0x42	Low/High MCU Temperature Thresholds	2 B	• Bits 8-15: High MCU temperature threshold (signed, 1°C / LSB) • Bits 0-7: Low MCU temperature threshold (signed, 1°C / LSB) • High threshold ≤ Low threshold: Invalid and ignored	*mcu_temperature_threshold {* *high: <value>,* *(signed/°C)* *low: <value>* *(signed/°C)* *}*
0x43	MCU Temperature Thresholds Enabled	1 B	• Bit 0: 0/1 = Thresholds disabled/enabled • Bits 1-7: Ignored	*mcu_temperature_threshold_enabled: <value>* *(unsigned/no unit)*
0x44	Analog Input Sample Period: Idle	4 B	• Sample period of analog input: Idle state (sec) • Acceptable values: 30, 31, …, 86400 • Other values: Invalid and ignored	*analog_sample_period_idle: <value>* *(unsigned/sec)*
0x45	Analog Input Sample Period: Active	4 B	• Sample period of analog input: Active state (sec) • Acceptable values: 30, 31, …, 86400 • Other values: Invalid and ignored	*analog_sample_period_active: <value>* *(unsigned/sec)*
0x46	Low/High Analog Input Thresholds	4 B	• Bits 16-31: High analog input threshold (unsigned, 1 mV/LSB)	*analog_input_threshold {* *high: <value>,* *(unsigned/V)*

			• Bits 0-15: Low analog input threshold (unsigned, 1 mV/LSB) • High threshold ≤ Low threshold: Invalid and ignored	*low: <value> (unsigned/V)* *}*
0x4A	Analog Input Thresholds Enabled	1 B	• Bit 0: 0/1 = Thresholds disabled/enabled • Bits 1-7: Ignored	*analog_input_threshold_enabled: <value> (unsigned/no unit)*

3.2.5.1 Temperature/RH/Analog Input Sample Period: Idle

The idle sample period determines how often the transducer is checked when the reported value is within the threshold window. When first enabled, the transducer starts in the Idle state.

The minimum Sample Period in the Idle state is 30 sec, and the maximum is 86,400 sec (one day). Values smaller than 30 for this register are invalid and ignored.

3.2.5.2 Temperature/RH/Analog Input Sample Period: Active

The active sample period determines how often the transducer is checked when the reported value is outside the threshold window.

The minimum Sample Period in the Active state is 30 sec, and the maximum is 86,400 sec (one day). Values smaller than 30 for this register are invalid and ignored.

3.2.5.3 Temperature/RH/Analog Input Thresholds

The thresholds are stored in a single 2-byte register, with the MSB storing the upper threshold, and the LSB storing the lower threshold. Ambient or MCU Temperature thresholds have a precision of 1°C per bit, and are stored/transmitted as 2-s complement numbers. The RH thresholds have a precision of 1% per bit, and are stored/transmitted as unsigned numbers. The Analog Input thresholds are also unsigned numbers, and have a precision of 1 mV per bit.

In all cases, the upper threshold must be greater than the lower threshold. Otherwise, the configuration is considered invalid and ignored.

3.2.5.4 Temperature/RH/Analog Input Thresholds Enabled

The Thresholds Enabled registers enable and disable the threshold reporting on the specified transducer. Thresholds and Sample Periods can be configured but are not activated unless the Thresholds Enabled bit is set.

3.2.5.5 Default Configuration

Table 3-15 shows the default values for the threshold configuration registers.

Table 3-15: Default Values of Threshold Configuration Registers

Ambient Temperature/RH Sample Period: Idle	60 sec
Ambient Temperature/RH Sample Period: Active	30 sec
Ambient Temperature Threshold: High	30°C
Ambient Temperature Threshold: Low	15°C
Ambient Temperature Thresholds Enabled	Disabled
Ambient RH Threshold: High	80%
Ambient RH Threshold: Low	15%
Ambient RH Thresholds Enabled	Disabled
MCU Temperature Sample Period: Idle	300 sec
MCU Temperature Sample Period: Active	60 sec
MCU Temperature Threshold: High	30°C
MCU Temperature Threshold: Low	15°C
MCU Temperature Thresholds Enabled	Disabled
Analog Input Sample Period: Idle	60 sec
Analog Input Sample Period: Active	30 sec
Analog Input Threshold: High	1200 mV
Analog Input Threshold: Low	600 mV
Analog Input Thresholds Enabled	Disabled

Examples:

- Set Ambient Temperature Thresholds:
  - DL payload: { 0x **BB** 19 0A }
    - Register 0x3B with write bit set to true
    - High threshold set to 25°C
    - Low threshold set to 10°C
- Read Ambient Temperature/RH Sample Periods:
  - DL payload: { 0x **39 3A** }
    - Registers 0x39 and 0x3A with their write bits set to false
- Set and enable Ambient RH thresholds:
  - DL payload: { 0x **BD** 3C 14 **BE** 01 }
    - Registers 0x3D and 0x3E with their write bits set to true
    - High RH thresholds set to 60% RH
    - Low RH threshold set to 20% RH
    - RH thresholds enabled

3.2.6 Light Sensing Configuration

The Room Sensor light sensing allows for the detection of the presence or absent of light based on the built-in light sensing transducer. The sensing element light pipe is visible on the top surface of the Room Sensor. The orientation of the Room Sensor relative to the light source impacts the measured level of light intensity. The Room Sensor light sensing capability supports both periodic and threshold-based transmissions. Table 3-16 shows a list of light transducer configuration registers. All the registers have R/W access.

Table 3-16: Light Transducer Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x47	Sample Period	4 B	• Sample period of the light transducer (sec) • Acceptable values: 0, 30, 31, …, 86400 • 0: Disables the light sensing element • 1, 2, …, 29: Invalid and ignored	*light_sample_period: <value>* *(unsigned/second)*
0x48	Threshold Control	1 B	• Bits 0-5: 0: Invalid and ignored 1-63: Threshold level (1: darkest, 63: brightest) • Bit 6: Ignored • Bit 7: • 0/1 = Threshold-based reporting disabled/enabled	*light {* *threshold: <value>,* *(unsigned/no unit)* *threshold_enabled: <value>* *(unsigned/no unit)* *}*
0x49	Value to Tx	1 B	• Bit 0: 0/1 = State (dark or bright) not reported/reported • Bit 1: 0/1 = Intensity (a value between 0 and 64, inclusive) not reported/reported • Both bits 0 and 1 set to 0: Invalid and ignored • Bits 2-7: Ignored	*light_tx {* *state_reported: <value>,* *(unsigned/no unit)* *intensity_reported: <value>* *(unsigned/no unit)* *}*

3.2.6.1 Sample Period

The light transducer is held turned off to preserve energy, but turned on periodically by the MCU to take samples. The light sensing sample period determines how often the light sensing transducer is powered on and checked for the presence of light. Shorter sample periods result in an improved detection time but result in additional battery usage.

Acceptable values for the sample period are 0, 30–86,400 sec. Setting the sample period to 0 (zero) disables the light sensing element. Values 1 to 29 sec or values greater than 86,400 sec are invalid and ignored.

Note: The light sensing sample period needs to be enabled for periodic transmission. Otherwise, in every transmission a repetitive light value residing in the MCU memory is reported.

3.2.6.2 Threshold Control

Bits 0-5 of the Threshold Control register is used to set the dark/bright transition point for the Sensor, and can be set to any value from 1 to 63. A light value smaller than or equal to the threshold is interpreted as "dark", and values greater than the threshold as "bright". Therefore, a threshold setting of 1 (one) corresponds to the darkest threshold, and 63 to the brightest threshold. When first enabled, the Sensor begins in the "dark" state.

Bit 7 of the Threshold is used to enable or disable the threshold-based reporting. If the threshold-based reporting is enabled, the Sensor transmits whenever the threshold is crossed (i.e. when the current and previous samples lie both sides of the threshold). If the threshold-based reporting is disabled, the threshold defined in bits 0-5 is only used to determine the "state" (dark or bright) in possible periodic transmissions.

3.2.6.3 Value to Tx

The Value to Tx determines the value that is reported in periodic or threshold-based transmissions. The light state is either dark or bright (based on a comparison of the light intensity value with the light threshold). The light intensity is a value between 0 and 64, inclusive.

3.2.6.4 Default Configuration

Table 3-17 shows the default values for the light transducer configuration registers.

Table 3-17: Default Values of Light Transducer Configuration Registers

Sample Period	Light transducer disabled
Threshold Control	• Threshold-based reporting enabled • Light threshold = 32 (i.e. a mid-range threshold)
Value to Tx	Light state reported only

3.2.7 Motion Transducer Configuration

The motion transducer (detector) is *on PIR Room Sensor models only* and uses a PIR array sensor for the detection of human motion in a room. Due to the sensitive electronics used in the PIR motion detector, the Room Sensor is designed to behave as follows:

- For **2 (two) min** after power is first applied to the device, the PIR motion detector is disabled (*post-turn on hold-off interval*). This is required for the PIR transducer output to stabilize and avoids false detections. The post-turn on hold-off interval is configurable through register 0x54 (see Table 3-18).
- For approximately **15 (fifteen) sec** after a radio transmission or after sampling the temperature/RH transducer, the PIR motion detector is disabled (*post-disturbance hold-off interval*). The operation of the radio or the temperature/RH transducer causes the PIR transducer to produce false positives so a

"cool down" period is required after each Tx. The post-disturbance hold-off interval is also configurable through register 0x54 (see Table 3-18).

The Room Sensor runs a simple state machine for reporting whether motion is detected. To conserve battery usage, the Room Sensor only reports motion when it is first detected and when motion has not been detected for a configurable Grace Period.

Note: The PIR transducer is designed to detect motion so if a room is occupied but the occupants are not moving, the sensor may report "No Motion" after the Grace Period (see Section 3.2.7.1) expires.

Table 3-18 shows a list of motion transducer configuration registers. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-18: Motion Transducer Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x50	Grace Period	2 B	• Grace period in sec (time before motion is no longer detected) • Acceptable values: 15, 16, …, 65535 • Other values: Invalid and ignored	*pir_grace_period: <value>* *(unsigned/second)*
0x51	Threshold Count	2 B	• Number of PIR events before motion is detected • Acceptable values: 1, 2, …, 65535 • 0: Invalid and ignored	*pir_threshold: <value>* *(unsigned/no unit)*
0x52	Threshold Period	2 B	• Period over which PIR events are counted for threshold detection • Acceptable values: 5, 6, …, 65535 • Other values: Invalid and ignored	*pir_threshold_period: <value>* *(unsigned/no unit)*
0x53	Mode	1 B	• Bit 0 (only applies to periodic Tx): 0/1 = Motion count not reported/reported • Bit 1 (only applies to periodic Tx): 0/1 = Motion state not reported/reported • Both bits 0 and 1 set to 0: Invalid and ignored • Bits 2-5: Ignored • Bit 6: 0/1 = PIR event-based transmission disabled/enabled • Bit 7: 0/1 = PIR transducer disabled/enabled	*pir_mode {* *motion_count_reported: <value>,* *(unsigned/no unit)* *motion_state_reported: <value>,* *(unsigned/no unit)* *event_transmission_enabled: <value>,* *(unsigned/no unit)* *transducer_enabled: <value>* *(unsigned/no unit)* *}*

0x54	Hold-Off Intervals	2 B	• Bits 8-15: Post-turn on hold-off interval (unsigned, 1 sec / LSB) 0 = Default value (120 sec) • Bits 0-7: Post-disturbance hold-off interval (unsigned, 1 sec / LSB) 0 = Default value (15 sec)	*pir_holdoff {* *post_turn_on: <value>, (unsigned/sec)* *Post_disturbance: <value>, (unsigned/sec)* *}*

3.2.7.1 Grace Period

The Grace Period determines how long the Room Sensor waits before the previously reported PIR motion event is considered clear. For example, a Grace Period of 5 (five) min results in the sensor transmitting "Motion Detected" when someone enters the room, and "Motion Not Detected" 5 (five) min after the room is empty. Values less than 15 sec are invalid and ignored.

3.2.7.2 Threshold Count

The PIR transducer generates an event each time it detects motion in its field of view. Depending on the customer use case it may be desirable to increase the Threshold to reduce sensitivity. This feature was designed to allow customers to filter out short motion events (such as a person quickly entering a room to pick-up a notebook), while still allowing longer motion events (a team meeting) to be reported.

3.2.7.3 Threshold Period

The Threshold Period is the amount of time that motion events will be accumulated for Threshold detection. For example, a Threshold Period of 10 (ten) sec accumulates motion detection events over a 10 (ten)-sec period from the time of first detection. If the Threshold is exceeded before the time expires, the sensor reports "Motion Detected", otherwise it does not report. Values less than 5 for the Threshold Period are invalid and ignored.

3.2.7.4 Mode

The Mode register allows the customer to disable/enable the motion transducer, as well as change the type of data that is transmitted by the Room Sensor. When the PIR transducer is disabled, no events from the PIR are monitored. When enabled, the motion transducer always reports the "motion state" (i.e. only the presence or absence of movement) in event-based reporting, if the event-based reporting is enabled. Bit 0 (motion count) and bit 1 (motion state) bits determine what values are transmitted when periodic reporting is enabled.

3.2.7.5 Hold-Off Intervals

PIR has two configurable hold-off intervals, post-turn on and post-disturbance, where the PIR is temporarily disabled to avoid false positives. This has been explained in more details at the beginning of Section 3.2.7.

The MSB of register 0x54 controls the post-turn on hold-off interval while the LSB controls the post-disturbance interval. Note that a value of 0 for any of these intervals is equivalent to the default value of that interval (120 sec for the post-turn on and 15 sec for the post-disturbance hold off).

3.2.7.6 Default Configuration

Table 3-19 shows the default values for the motion transducer configuration registers.

Table 3-19: Default Values of Motion Transducer Configuration Registers

Grace Period	300 sec (5 min)
Threshold	1
Threshold Period	15 sec
Mode	• PIR transducer enabled • Event-based transmission enabled • Motion count reported only, in the case of a periodic transmission
Hold-Off Intervals	• Post-turn on hold-off interval 120 sec (2 min) • Post-disturbance hold-off interval 15 sec

3.2.8 Moisture Configuration

The Base Room Sensor is equipped with a capacitance-based moisture detection system. *The moisture detection is not supported in the PIR Room Sensor.* This allows the Room Sensor to detect the pooling of water (water line leak, spills, etc.) and report moisture detection events. The moisture transducer (detector) is integrated into the Room Sensor enclosure base (screw side) and can sense moisture without making physical contact with the liquid. This transducer does not measure humidity in air. Table 3-20 shows a list of moisture transducer configuration registers. All the registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-20: Moisture Transducer Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x5A	Sample Period	1 B	• Period of moisture measurement • Bits 0-2: 1 = 16 sec 2 = 32 sec 3 = 64 sec 4 = 128 sec 0, 5-7: Invalid and ignored • Bits 3-7: Ignored	*moisture_sample_period: <value> (unsigned/second)*
0x5B	Threshold	1 B	Moisture detection threshold: • Acceptable values: 50, 51, …, 255 • Other values: Invalid and ignored	*moisture_threshold: <value> (unsigned/no unit)*

0x5C	Enable/Disable	1 B	• Bit 0: 0/1 = Moisture sensing disabled/enabled • Bits 1-7: Ignored	*moisture_sensing_enabled: <value> (unsigned/no unit)*
0x5D	Calibrate Baseline	1 B	• Command to calibrate the transducer as dry: 0: Ignored • 1-255: Recalibrate the dry baseline	*moisture_caliberation_dry: <value> (unsigned/no unit)*

3.2.8.1 Sample Period

The moisture transducer is activated periodically to determine if water is present. A smaller sample period results in a faster response from the Sensor in the event of a leak, however it results in higher battery usage than a larger sample period.

Table 3-21 shows how much continuous current draw is expectable to be drawn from the battery for different moisture sample periods when moisture sensing is enabled. For example, the default sample period of 32 seconds would translate to about 122 mAh/year battery usage.

Table 3-21: Typical Current Draws for Different Moisture Sample Periods[11]

Sample Period [sec]	16	32	64	128
Current Draw [μA]	14.1	13.9	13.8	13.8

Note: For the updates to the Sample Period to take effect, the moisture transducer requires to be de-initialized and then initialized.

3.2.8.2 Threshold

The Threshold of the moisture transducer determines the tripping point for various conditions. Nominally, a 1/4" of water below the Room Sensor results in a shift of about 300 units from the dry measurement baseline. The Threshold is tunable to allow the customer to set the desired sensitivity level. However, note that changing the threshold may desensitize the moisture transducer or increase the likelihood of a false positive.

Any value less than 50 for the Threshold is ignored.

3.2.8.3 Enable/Disable

The Enable/Disable register sets whether the moisture transducer is initialized (enabled) or de-initialized (disabled). This register is used to determine the default state of the moisture transducer when first powered on.

[11] Measured with an input voltage of 3.3V at the battery terminals.

3.2.8.4 Calibrate Baseline

Writing a non-zero value to this register forces the transducer to re-calibrate the dry baseline to the current value regardless of its actual state (wet or dry). It is recommended that this command is run when a Room Sensor is first deployed or relocated to ensure that the baseline is correctly set for the material under the Room Sensor. Any issued recalibration is performed as soon as the next moisture sample is taken. Therefore, as an example, if the sample period is 16 sec, it may take up to 16 sec for a recalibration command to be executed.

3.2.8.5 Default Configuration

Table 3-22 shows the default values for the moisture transducer configuration registers.

Table 3-22: Default Values of Moisture Transducer Configuration Registers

Sample Period	32 sec
Threshold	100
Enable/Disable	Disabled (De-initialized)

Examples:

- Set the Sample Period to 64 sec and read the Threshold
    - DL payload: { 0x **DA** 03 **5B** }
- Force the transducer to calibrate as being dry
    - DL payload: { 0x **DD** 01 }

3.2.8.6 Operation Algorithm

1. Whenever the moisture detector is enabled, it is recalibrated for a new dryness baseline.
2. In every sample, if the measured value goes up from the baseline by more than 10 counts (not user configurable), the moisture detector is recalibrated for a new dryness baseline.
3. In every sample, if the measured value goes down from the baseline by more than the Threshold (user configurable, see Section 3.2.8.2), the detector is tripped (signaling wetness).
4. Every 60 samples, if humidity changes by more than 10% (up or down–not user configurable), the moisture detector gets recalibrated for a new dryness baseline.

3.2.9 Accelerometer Enhanced Configuration

This section introduces enhanced configuration to expand the functionality of the accelerometer, as needed by some applications, such as toilet leakage or water flow detection. The enhanced functionality is related to enabling Sample Analysis from when an **acceleration event** is registered until when the **acceleration event debounce time** is expired (see Section 3.2.4).

The block diagram of such an analysis has been shown in Figure 3-2. The magnitudes of the accelerometer samples are low-pass filtered, then binned into a number of *magnitude levels (MLs)*, ML_0, ML_1, …, ML_p, where $1 \leq p \leq 15$. ML_0 is always the zero level, and not counted or kept track of for event reporting.

A whole run of the same ML is called an *ML event*, which has a length (duration) associated with it; e.g. a run of 3 ML_1 is an ML_1 event of duration 3. ML events can have length 1. That can happen due to noisy and quick jumps between MLs. Enabled hysteresis in binning can reduce noise.

An *analytical event (AE)* is defined by an ML (ML_1 to ML_{15}) and a *duration range (DRan)*. In other words, an AE is an ML event with a duration within some range. To avoid complexity, from 1 to only 4 DRans can be defined, and only for ML_1 and ML_2; ML_3 to ML_{15} always have 1 DRan. The case of 1 DRan, $DRan_0$, for ML_p $(1 \leq p \leq 15)$ means there is no duration limits for ML_p and any ML_p event (i.e. any whole run of ML_p) is considered an AE, denoted by $AE_{p,0}$, where index 0 means $DRan_0$.

The case of 2 DRans for ML_1 or ML_2 is defined by 1 *duration limit (DLim)*, $DLim_0$, where durations below $DLim_0$ constitute $DRan_0$ and those above $DLim_0$ constitute $DRan_1$, thus having 2 DRans below and above a single DLim. Similarly, the cases of 3 and 4 DRans for ML_1 or ML_2 ($DRan_0$ to $DRan_2$, and $DRan_0$ to $DRan_3$) are defined by 2 DLims ($DLim_0$ and $DLim_1$) and 3 DLims ($DLim_0$, $DLim_1$, and $DLim_2$), respectively. Now, $AE_{1,q}$ $(0 \leq q \leq 3)$ simply denotes an ML_1 event with the duration inside $DRan_q$ of ML_1. Similarly, $AE_{2,q}$ $(0 \leq q \leq 3)$ is an ML_2 event with the duration inside $DRan_q$ of ML_2.

The Count & Group block in Figure 3-2 counts and keeps track of statistics of all defined AEs. For example, it is possible that 15 MLs, ML_1 or ML_{15}, each with only one DRan, $DRan_0$, have been defined, e.g. during application training where DLims are not known yet. In this case, we have a total of 15 AEs, $AE_{1,0}$ to $AE_{15,0}$. After the training, the application may learn a DLim, and only define $AE_{1,0}$ and $AE_{1,1}$ for the sensor, i.e. AEs associated with a single ML, ML_1, and 2 DRans, $DRan_0$ and $DRan_1$, which are defined by only 1 DLam, $DLam_0$, for ML_1.

Simply put, $AE_{p,q}$ denotes an ML_p event $(p > 0)$ with duration within $DRan_q$, where $DRan_q$ is the DRan between $DLim_{q-1}$ and $DLim_q$. The Count & Group block in Figure 3-2 keeps track of all defined $AE_{p,q}$'s within the acceleration event debounce time, and update statistics for these AEs at the end of each debounce interval. At the periodic reporting time, the statistics asked for are reported for all defined $AE_{p,q}$'s. These statistics include total duration, min duration, max duration, duration SD, and count. All the statistics and counters for all defined AEs are cleared when a periodic report is made.

It is also possible to enable threshold-based reporting on one of the statistics, the duration total and duration count. As soon as that statistic for a defined $AE_{p,q}$ exceeds the $AE_{p,q}$'s corresponding threshold, the value of that statistic is reported for the $AE_{p,q}$. Such a threshold-based report does not reset the counters or statistics. It is just for periodic reports that all counters are reset.[12]

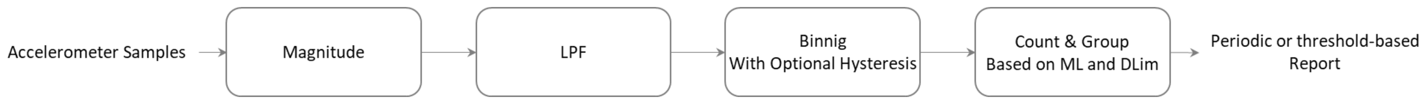


Figure 3-2: Block diagram of the Sample Analysis after an acceleration event.

Example: 8 MLs, ML_1 to ML_8, with 1 DRan ($DRan_0$) have been defined for the sensor. In a debounce time, the following MLs come out of the binning block:

ML_4 ML_4 ML_4 ML_0 ML_0 ML_5 ML_3 ML_3 ML_0 ML_7 ML_6 ML_3 ML_8 ML_8 ML_8 ML_8 ML_0 ML_0 ML_0 ML_0 ML_1 ML_1 ML_1 ML_1 ML_3 ML_6 ML_5 ML_5 ML_3 ML_6 ML_4 ML_7 ML_5 ML_2 ML_2 ML_2 ML_2 ML2 ML_2 ML_2 ML_0

Table 3-23 shows the AE analysis results for this example. It shows there are one $AE_{1,0}$, one $AE_{2,0}$, four $AE_{3,0}$, two $AE_{4,0}$, three $AE_{5,0}$, three $AE_{6,0}$, two $AE_{7,0}$, and one $AE_{8,0}$.

Note that ML_0 is the zero level and does not define an AE.

Table 3-23: Example of AEs and Their Statistics in an Acceleration Debounce Time

	ML1	ML2	ML3	ML4	ML5	ML6	ML7	ML8
AE Runs	4	7	2 1 1 1	3 1	1 2 1	1 1 1	1 1	4
Total Duration	4	7	5	4	4	3	2	4
Min Duration	4	7	1	1	1	1	1	4
Max Duration	4	7	2	3	2	1	1	4
Duration SD	0	0	0.5	1.4	0.6	0	0	0
Count	1	1	4	2	3	3	2	1

Table 3-24 shows the list of enhanced accelerometer configuration registers. All registers have R/W access. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-24: Enhanced Accelerometer Configuration Registers

Address	Name	Size	Description	JSON Variable (Type/Unit)
0x60	Sample Analysis Mode	1 B	• Bits 0-3: IIR filter recall factor • Bit 7: Sample analysis enabled	*sample_analysis_mode {* *iir_recall_factor: <value>,*

[12] However, if counters are not reset after a threshold-based report, the threshold keeps getting triggered. To avoid this issue, SW increases the thresholds appropriately each time a threshold is exceeded, but then resets all the thresholds to their original values at the time of a periodic report.

			• Bits 4-6: Ignored	*(unsigned/no unit)* *analysis_enabled: <value>, (unsigned/no unit)* *}*
0x61	HPF Configuration	1 B	• Bit 0: 0/1 = HPF disabled/enabled on accelerometer outputs • Bit 1: 0/1 = HPF disabled/enabled on accelerometer interrupts • Bit 2: 0/1 = HPF disabled/enabled on Sample Analysis • Bits 4-5: HPF cutoff frequency selector Acceptable values: 0, 1, 2, 3 Larger number = Smaller cutoff • Bits 3, 6, 7: Ignored	*hpf_mode {* *enabled_on_outputs: <value>, (unsigned/no unit)* *enabled_on_interrupts: <value>, (unsigned/no unit)* *enabled_on_sample_analysis: <value>, (unsigned/no unit)* *cutoff: <value> (unsigned/no unit)* *}*
0x62	RFU			
0x63	ML Configuration	7 B	• Bits 48-55: Number of nonzero MLs $(1 \le \cdot \le 15)$ • Bits 32-47: ML_1 upper limit, 1 mg/LSB • Bits 16-31: ML_1 lower limit, 1 mg/LSB • Bits 0-15: ML step, 1 mg/LSB • Number of nonzero MLs = 0 or > 15: Invalid and ignored • ML_1 upper limit < ML_1 lower limit: Invalid and ignored • ML_1 upper limit = ML_1 lower limit: Hysteresis disabled • Number of nonzero MLs > 1 & ML step = 0: Invalid and ignored	*ml_config {* *num: <value>, (unsigned/no unit)* *ml1upper: <value>, (unsigned/g)* *ml1lower: <value>, (unsigned/g)* *ml_step: <value>, (unsigned/g)* *}*
0x64	ML_1 DLim Configuration	14 B	• Bits 104-111: Number of DLims $(0 \le \cdot \le 3)$ • Bits 88-103: $DLim_0$, 1 sec/LSB • Bits 72-87: DLim step, 1 sec/LSB • Number of DLims > 3: Invalid and ignored • Number of DLims > 1 & DLim step = 0: Invalid and ignored • Bits 64-71:	*ml1_dlim_config {* *num_dlims: <value>, (unsigned/no unit)* *dlim0: <value>, (unsigned/sec)* *dlim_step: <value>, (unsigned/sec)* *thresholds_enabled: <value>, (unsigned/no unit)*

			0: Thresholds disabled 1: Thresholds enabled on duration total 2: Thresholds enabled on count 3-255: Invalid and ignored • Bits 48-63: $AE_{1,0}$ threshold, unsigned, 1 sec or 1 count/LSB • Bits 32-47: $AE_{1,1}$ threshold, unsigned, 1 sec or 1 count/LSB • Bits 16-31: $AE_{1,2}$ threshold, unsigned, 1 sec or 1 count/LSB • Bits 0-15: $AE_{1,3}$ threshold, unsigned, 1 sec or 1 count/LSB	*ae10_threshold: <value>, (unsigned/sec)* *ae11_threshold: <value>, (unsigned/sec)* *ae12_threshold: <value>, (unsigned/sec)* *ae13_threshold: <value> (unsigned/sec)* *}*
0x65	ML_2 DLim Configuration	14 B	• Bits 104-111: Number of DLims ($0 \le \cdot \le 3$) • Bits 88-103: $DLim_0$, 1 sec/LSB • Bits 72-87: DLim step, 1 sec/LSB • Number of DLims > 3: Invalid and ignored • Number of DLims > 1 & DLim step = 0: Invalid and ignored • Bits 64-71: 0: Thresholds disabled 1: Thresholds enabled on duration total 2: Thresholds enabled on count 3-255: Invalid and ignored • Bits 48-63: $AE_{2,0}$ threshold, unsigned, 1 sec or 1 count/LSB • Bits 32-47: $AE_{2,1}$ threshold, unsigned, 1 sec or 1 count/LSB • Bits 16-31: $AE_{2,2}$ threshold, unsigned, 1 sec or 1 count/LSB • Bits 0-15: $AE_{2,3}$ threshold, unsigned, 1 sec or 1 count/LSB	*ml2_dlim_config {* *num_dlims: <value>, (unsigned/no unit)* *dlim0: <value>, (unsigned/sec)* *dlim_step: <value>, (unsigned/sec)* *thresholds_enabled: <value>, (unsigned/no unit)* *ae20_threshold: <value>, (unsigned/sec)* *ae21_threshold: <value>, (unsigned/sec)* *ae22_threshold: <value>, (unsigned/sec)* *ae23_threshold: <value> (unsigned/sec)* *}*
0x66	RFU			
0x67	RFU			
0x68	Sample Analysis Report Mode (applicable to periodic transmissions)	1 B	• Bit 0: AE duration total enabled • Bit 1: AE duration min enabled • Bit 2: AE duration max enabled • Bit 3: AE duration SD enabled • Bit 4: AE count enabled	*accelerometer_analysis_tx {* *duration_total_enabled: <value>, (unsigned/no unit)* *duration_min_enabled: <value>, (unsigned/no unit)*

				duration_max_enabled: <value>, *(unsigned/no unit)* *duration_sd_enabled: <value>,* *(unsigned/no unit)* *count_enabled: <value>,* *(unsigned/no unit)* *}*

3.2.9.1 Sample Analysis Mode

This register configures the settings of Sample Analysis. Sample Analysis can be disabled or enabled. If disabled, none of bin related periodic or threshold-based transmissions will be active. Also, the register configures the recall factor of the IIR LPF. The forget factor of the IIR filter is related to the recall factor through $f = \frac{1}{2^r}$, where f and r are the forget and recall factors, respectively. The recall factor r is configured by the 4 LSBs of register 0x60, and can range from 0 to 15. The value $r = 0$ results in $f = 1$, which basically disables the IIR filter.

3.2.9.2 HPF Configuration

Accelerometer HPF to remove the DC component out of accelerometer samples can be disabled or enabled separately for accelerometer output vectors and for the samples triggering an accelerometer alarm. The HPF can also be controlled separately for the accelerometer output vectors outside or inside Sample Analysis. For example, DC removal may not be desired for periodic accelerometer vector reporting, but may be required for triggering the accelerometer or during Sample Analysis.

The cutoff frequency of the HPF can be adjusted on a four-level basis from 0 to 3, with 0 showing the largest and 3 indicating the smallest cutoff.

3.2.9.3 ML Configuration

This register defines the number and values of the nonzero MLs for the binning analysis (ML_0 is always level 0). 1 to 15 nonzero MLs can be defined where the upper and lower limits of consecutive MLs increase in equal steps. Setting the upper and lower limits equal disables the hysteresis effect on the binning.

3.2.9.4 ML_1 and ML_2 DLim Configuration

These registers define DLims for ML_1 and ML_2. From 0 to 3 DLims can be defined for each, so we can have $AE_{1,0}$, $AE_{1,1}$, $AE_{1,2}$, $AE_{1,3}$ based on ML_1, and $AE_{2,0}$, $AE_{2,1}$, $AE_{2,2}$, $AE_{2,3}$ based on ML_2. For simplicity and similar to MLs, we assume that for each ML, DLims are equidistance.

These registers also define what threshold reporting, if any, is enabled on ML_1 family of events ($AE_{1,0}$, $AE_{1,1}$, $AE_{1,2}$, $AE_{1,3}$) and ML_2 family of events ($AE_{2,0}$, $AE_{2,1}$, $AE_{2,2}$, $AE_{2,3}$). In addition, the thresholds for these AEs can be defined in these registers. When the threshold for an AE is exceeded, a report of the exceeded amount is created for that AE.

3.2.9.5 Sample Analysis Report Mode

This register determines what statistics of the durations of defined AEs are periodically reported. The example given in Section 3.2.9 shows how the statistics of the AEs are calculated.

3.2.9.6 Default Configuration

Table 3-13 shows the default values for the accelerometer configuration registers.

Table 3-25: Default Values of Accelerometer Configuration Registers

Sample Analysis Mode	• IIR filter recall factor set to 4 Sample Analysis disabled
HPF Configuration	• HPF disabled on accelerometer outputs • HPF enabled on accelerometer interrupts • HPF enabled on Sample Analysis • HPF cutoff frequency selector set to 0
ML Configuration	• Number of nonzero MLs = 8 • ML1 upper limit = 40 mg • ML1 lower limit = 20 mg • ML step = 30 mg
ML_1 DLim Configuration	• Number of DLims = 0 • $DLim_0$ = 0 • DLim step = 0 • Thresholds disabled • $AE_{1,0}$, $AE_{1,1}$, $AE_{1,2}$, $AE_{1,3}$ thresholds = 0
ML_2 DLim Configuration	• Number of DLims = 0 • $DLim_0$ = 0 • DLim step = 0 • Thresholds disabled • $AE_{2,0}$, $AE_{2,1}$, $AE_{2,2}$, $AE_{2,3}$ thresholds = 0
Sample Analysis Value to Tx	AE duration total enabled

3.2.9.7 Configuration Example for Water Flow Detection Use Case

In the water flow detection use case, the application server configures the sensor in a way that allows the sensor to collect data and uplink it to the application so the application learns how to set the sensor configuration to detect toilet leakages. We call this the training stage.

The training stage starts after the sensor joins the network, or after it gets triggered by a magnet so it sends an alarm, from which the application understands the sensor is ready and sends the sensor the proper training configuration.

The training configuration has two parts. The first part includes the following:

1. Set the core tick to 10 min, and accelerometer tick to 1 hour, but the other ticks to 0:
   0x A0 00 00 02 58 A1 00 00 A2 00 00 A3 00 00 A6 00 01
2. Set the acceleration event threshold to 0:
   0x B1 00 00
3. Enable the periodic accelerometer vector:
   0x B2 04
4. Se the acceleration event debounce time to 90 sec:
   0x B3 00 5A
5. Power on the accelerometer and enable the acceleration event threshold:
   0x B4 F2
6. Enable the accelerometer with sample rate of 10 Hz and measurement range of ±2 *g*:
   0x B5 02
7. Enable Sample Analysis and IIR filtering with recall factor of 4:
   0x E0 84
8. Disable HPF for periodic accelerometer vector reporting, but enable HPF with cutoff index 0 for accelerometer triggers and Sample Analysis:
   0x E1 06
9. Define 8 (nonzero) MLs with upper and lower limits of 20 m*g* to 160 m*g* in steps of 20 m*g:*
   0x E3 08 00 14 00 14 00 14
10. Enable AE duration total reporting:
   0x E8 01

The above configuration can be sent in one DL command as:

0x A0 00 00 02 58 A1 00 00 A2 00 00 A3 00 00 A6 00 01 B1 00 00 B2 04 B3 00 3C B4 F2 B5 02 E0 84 E1 06 E3 08 00 14 00 14 00 14 E8 01

or “oAAAAlihAACiAACjAACmAAGxAACyBLMAPLTytQLghOEG4wgAFAAUABToAQ==” in Base64.

With the above configuration, the application receives a histogram for 8 bins from the sensor after a few report periods. With this histogram, the application finds a suitable acceleration event threshold and ML_1 for leakage detection. In the second part of the training, the application tries to configure the sensor with only one (nonzero) ML and derives statistics on the durations of runs of ML_1. To this end, the application reconfigures the sensor as follows:

1. 1 ML with a hysteresis of 20 m*g* is defined for the sensor, with the ML upper limit derived from the histogram obtained from the first part of the training stage:

0x E3 01 <xx xx> <xx xx - 20> 00 00
where <xx xx> is the threshold in hex.

2. AE duration total and AE duration count are enabled for periodic reporting:
   0x E8 11

The above configuration can be sent in one DL command as:

0x E3 01 <xx xx> <xx xx - 20> 00 00 E8 11.

After a few reports of the sensor data, the application finds out the average duration range of a normal flush, thus determining $DLim_0$ and $DLim_1$, and defining 3 DRans for ML_1 ($DRan_0$ corresponding to hiccup flushes, $DRan_1$ corresponding to normal flushes, and $DRan_2$ corresponding to water running events). Now, the application configures the sensor as follows for normal operation:

1. Set the acceleration event threshold to the threshold obtained from the histogram analysis in the training stage, part 1:
   0x B1 <xx xx>
   where <xx xx> is the threshold.
2. Set the acceleration event debounce time 10 sec above the estimated average time of a normal flush:
   0x B3 <yy yy + 10>
   where <yy yy> is the estimated average of the duration of a normal flush.
3. Define two DLims with the obtained values from the training for ML_1, and enable threshold-based reporting on $AE_{1,0}$, $AE_{1,1}$, or $AE_{1,2}$, as desired:
   0x E4 02 <zz zz> <tt tt> 01 <aa aa> <bb bb> <cc cc> 00 00
   where <zz zz>, <tt tt>, <aa aa>, <bb bb>, and <cc cc> are $DLim_0$, DLim step, $AE_{1,0}$ threshold, $AE_{1,1}$ threshold, and $AE_{1,2}$ threshold, respectively.

The above configuration can be sent in one DL command as:

0x B1 <xx xx> B3 <yy yy + 10> E4 02 <zz zz> <tt tt> 01 <aa aa> <bb bb> <cc cc> 00 00.

3.3 Command and Control

Configuration changes are not retained after a power cycle unless they are saved in the Flash memory. Table 3-26 shows the structure of the Command & Control Register. In this table, the bit indexing scheme is as shown in Figure 3-1.

Table 3-26: Sensor Command & Control Register

Address	Access	Name	Size	Description	JSON Variable (Type/Unit)
0x70	W	Flash Write Command	2 B	• Bit 14: • 0/1 = Do not write/Write LoRaMAC Config • Bit 13:	*write_to_flash {* *app_configuration: <value>, (unsigned/no unit)*

				• 0/1 = Do not write/Write App Config • Bit 0: • 0/1 = Do not restart/Restart Tracker • Bits 1-12, 15: Ignored	*lora_configuration: <value>, (unsigned/no unit)* *restart_sensor: <value> (unsigned/no unit)* *}*
0x71	R	FW Version	7 B	• Bits 48-55: App version major • Bits 40-47: App version minor • Bits 32-39: App version revision • Bits 24-31: LoRaMAC version major • Bits 16-23: LoRaMAC version minor • Bits 8-15: LoRaMAC version revision • Bits 0-7: LoRaMAC region number (see Section 3.3.1)	*firmware_version {* *app_major_version: <value>, (unsigned/no unit)* *app_minor_version: <value>, (unsigned/no unit)* *app_revision: <value>, (unsigned/no unit)* *loramac_major_version: <value>, (unsigned/no unit)* *loramac_minor_version: <value>, (unsigned/no unit)* *loramac_revision: <value>, (unsigned/no unit)* *region: <value> (unsigned/no unit)* *}*
0x72	W	Reset Config Registers to Factory Defaults[13]	1 B	• 0x0A = Reset App Config • 0xB0 = Reset LoRa Config • 0xBA = Reset both App and LoRa Configs • Any other value: Invalid and ignored	*configuration_factory_reset: <value> (unsigned/no unit)*

[13] After sending the reset-to-factory-defaults command, the Sensor is automatically reset with corresponding default configuration values.

Note: The Command & Control Register is always executed after the full DL configuration message has been decoded. The reset command should always be sent as an "unconfirmed" DL message. Failure to do so may cause a poorly designed NS to continually reboot the Sensor.

3.3.1 LoRaMAC Region

The LoRaMAC region is indicated by B_6 in the FW Version register (register 0x71). Current LoRaMAC regions and corresponding region numbers are listed in Table 3-27.

Table 3-27: LoRaMAC Regions and Region Numbers

LoRaMAC Region	Region Number
EU868	0
US915	1
AS923	2
AU915	3
IN865	4
CN470	5
KR920	6
RU864	7
DN915	8

Examples:

- Write Application Configuration to Flash memory
    - DL payload: { 0x **F0** 20 00 }
- Write Application and LoRa Configurations to Flash memory
    - DL payload: { 0x **F0** 60 00 }
- Reboot Device
    - DL payload: { 0x **F0** 00 01 }
- Get FW version, and reset App Config to factory defaults
    - DL payload: { 0x **71 F2** 0A}

References

[1] LoRa Alliance, "LoRaWAN Specification," ver. 1.0.2, Jul 2016.

[2] LoRa Alliance, "LoRaWAN Regional Parameters," ver 1.0.2, Rev B, Feb 2017.

[3] TEKTELIC Communications Inc., "Smart Room Sensor Uplink and Downlink Frame Payloads," ver 0.1, Aug 2019.

[4] TEKTELIC Communications Inc., "Smart Room Sensor User Guide," ver 1.5, Jan 2020.