PTM 535BZ

Bluetooth® and Zigbee Green Power Pushbutton Transmitter

06.07.2022

Observe precautions! Electrostatic sensitive devices!

Patent protected:
WO98/36395, DE 100 25 561, DE 101 50 128,
WO 2004/051591, DE 103 01 678 A1, DE 10309334,
WO 04/109236, WO 05/096482, WO 02/095707,
US 6,747,573, US 7,019,241

PTM 535BZ – BLUETOOTH AND ZIGBEE GREEN POWER PUSHBUTTON TRANSMITTER

REVISION HISTORY
The following major modifications and improvements have been made to this document:

Version	Author	Reviewer	Date	Major Changes
1.0	MKA	RS, EG, MK	12.05.2021	First public release
1.1	MKA	MKA	26.05.2021	Added NFC PIN HASH description and calculation example
1.2	MKA	MKA	06.07.2022	Added FCC and ISED certification

Published by EnOcean GmbH, Kolpingring 18a, 82041 Oberhaching, Germany
www.enocean.com, info@enocean.com, phone +49 (89) 6734 6890

Important!
This information describes the type of component and shall not be considered as assured characteristics. No responsibility is assumed for possible omissions or inaccuracies. Circuitry and specifications are subject to change without notice. For the latest product specifications, refer to the EnOcean website: http://www.enocean.com.

As far as patents or other rights of third parties are concerned, liability is only assumed for modules, not for the described applications, processes and circuits.

EnOcean does not assume responsibility for use of modules described and limits its liability to the replacement of modules determined to be defective due to workmanship. Devices or systems containing RF components must meet the essential requirements of the local legal authorities. The modules must not be used in any relation with equipment that supports, directly or indirectly, human health or life or with applications that can result in danger for people, animals or real value.

Components of the modules are considered and should be disposed of as hazardous waste. Local government regulations are to be observed. Packing: Please use the recycling operators known to you.

TABLE OF CONTENT

1 General description

1.1 Basic functionality

PTM 535BZ enables the realization of energy harvesting wireless switches for lighting, building or industrial automation control systems using Bluetooth® Low Energy (BLE) or Zigbee Green Power (ZGP) technology.

PTM 535BZ is mechanically compatible with the established PTM 330 / PTM 430J / PTM 535 form factor enabling quick integration into existing designs for these products. Key applications are wall-mounted or portable pushbutton or position switches.

PTM 535BZ provides an NFC interface with integrated NFC antenna that can be used to configure certain product parameters.

PTM 535BZ pushbutton transmitters are intended for operation together with the ECO 200 kinetic harvester which generates the required energy based on an external action (such as a button press). The combination of ECO 200 with PTM 535BZ enables the implementation of self-powered (no batteries) and fully maintenance-free products. They can therefore be used in all environments including locations that are difficult to reach or within hermetically sealed housings.

When the ECO 200 kinetic energy harvester is actuated (pressed or released), electrical energy is generated and - depending on the device configuration – either a BLE or a ZGP radio telegram is transmitted. This radio telegram transmits the action of the energy generator (press or release) and the status of the two external inputs. PTM 535BZ radio telegrams are protected with AES-128 security based on a device-unique private key.

Figure 1 below shows the top side of PTM 535BZ (on the left side) and the bottom side of PTM 535BZ (on the right side).

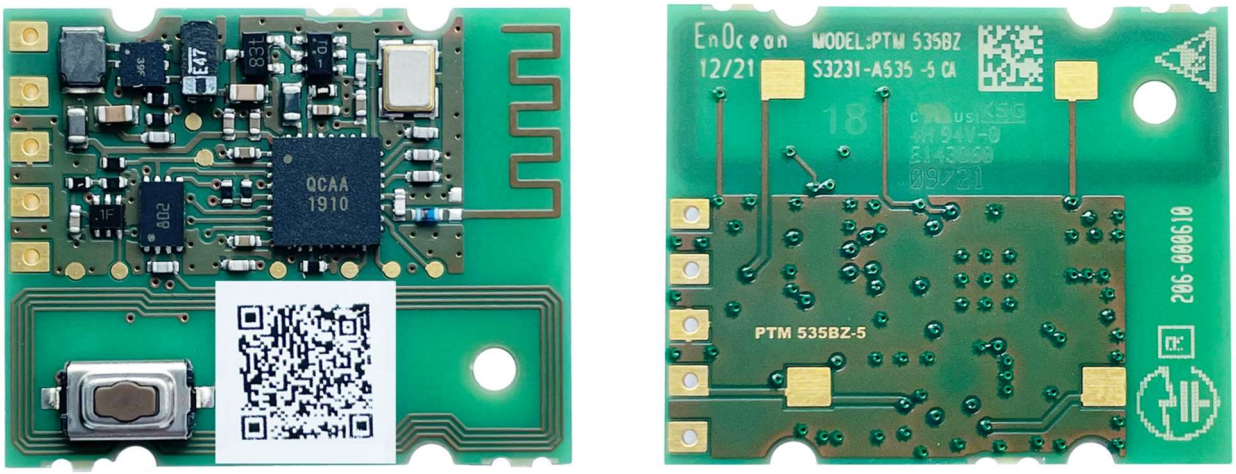

Figure 1 – PTM 535BZ top and bottom view

1.2 Technical data

Radio Standards	2.4 GHz Bluetooth Low Energy (default setting) 2.4 GHz Zigbee Green Power (optional setting via NFC)
Radio Channels	BLE Channel 37, 38 and 39 (BLE Advertising Channels) Zigbee Radio Channel 11 … 26 (NFC configurable, default Channel 11)
Data Encoding	EnOcean BLE Sensor Protocol (for BLE radio, default setting) Zigbee Green Power Generic Switch (for ZGP radio, NFC configurable)
Security	AES128 (CBC) authentication with sequence counter
Transmission Power	+4 dBm
Transmission Range (typ.)	30 m line of sight / 10 m indoor environment
Antenna	Integrated antenna
Power Supply	Kinetic harvester (ECO 200)
Configuration Interface	NFC (ISO15683 tag and integrated antenna)
User Interface	Learn button
Operating Conditions	-25°C ... +65°C / 0 ... 90 % r.h. Indoor use in dry rooms only
Dimensions	26.2 mm x 21.15 mm (same as PTM 535)

1.3 Environmental conditions

Operating Temperature	-25°C ... 65°C
Storage Temperature	-25°C ... 65°C
Humidity	0% to 95% r.h. (non-condensing)

1.4 Packaging information

Packaging Unit	100 units
Packaging Method	Tray / Box (10 units per tray, 10 trays per box)

1.5 Ordering information

Type	Ordering Code	Description
PTM 535BZ	S3231-A535	PTM 535BZ transmitter module
ECO 200	S3016-N200	ECO 200 kinetic energy generator

2 Functional information

2.1 Product overview

The transmitter module PTM 535BZ from EnOcean enables the implementation of wireless buttons and switches without batteries. It transmits Bluetooth Low Energy (BLE) or Zigbee Green Power (ZGP) data telegrams where the required energy is provided by an external electro-dynamic energy generator such as the kinetic harvester ECO 200.

2.2 Basic functionality

PTM 535BZ works in conjunction with an electro-dynamic energy converter (e.g. ECO 200) which is actuated (pressed and released) by external action such as a button press. The term "ECO 200" will be used throughout this document to describe a suitable energy converter.

When ECO 200 is actuated (pressed / pushed or released / pulled), electrical energy is generated and a BLE or ZGP radio telegram is transmitted which identifies the action (pressed or released) and the status of the two external input contacts.

When ECO 200 is actuated in the opposite direction (restored to its original position), it similarly generates energy which is used to transmit a different radio telegram. It is therefore possible to distinguish between radio telegrams sent when ECO 200 was pressed and radio telegrams sent when ECO 200 was released.

By identifying these different telegram types and measuring the time between pressing and releasing of the energy generator, it is possible to distinguish between "Long" and "Short" presses if required.

2.3 Functional block diagram

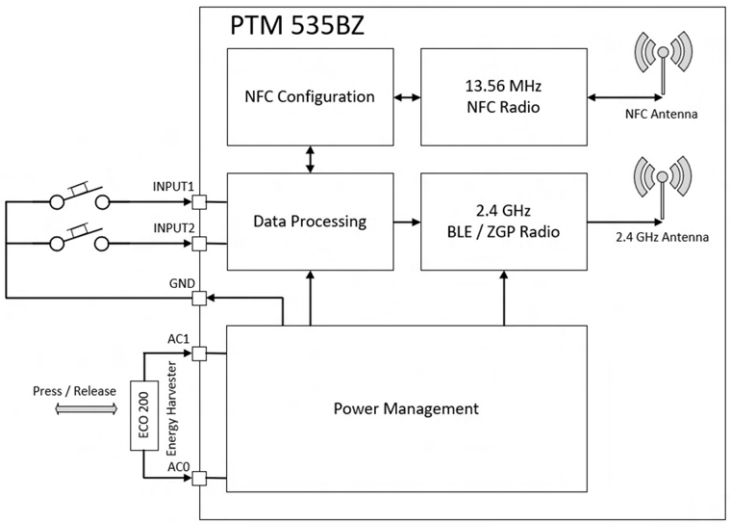

Figure 2 – Functional block diagram of PTM 535BZ working with ECO 200

2.3.1 Key components

PTM 535BZ uses the following main components to achieve the desired product functionality:

ECO 200 energy harvester
Converts the kinetic energy (press / release) into electrical energy and is used to power PTM 535BZ in self-powered applications. Alternatively, a power supply might be used.

Power management
Converts the energy of the power generator into a stable supply voltage suitable to power the device electronics. It also determines the polarity of the input voltage which allows identifying the direction of the ECO 200 action (press or release).

Data processing
Determines the status of the external inputs and the ECO 200 action, encodes this status into a data word, calculates the unique security signature, generates the proper radio telegram structure and sends it to the 2.4 GHz BLE / Zigbee radio transmitter

2.4 GHz BLE / Zigbee radio transmitter
Transmits the data in the form of a series of short 2.4 GHz Bluetooth Low Energy (BLE) or Zigbee Green Power (ZGP) radio telegrams using the integrated antenna

NFC configuration interface
Allows reading and writing certain product parameters using an NFC compliant reader / writer supporting NFC Forum tags (as specified by ISO/IEC 15693).

2.4 Product interface

Figure 3 below shows the product interface of PTM 535BZ seen from the top side (shown on the left) and the bottom side (shown on the right).

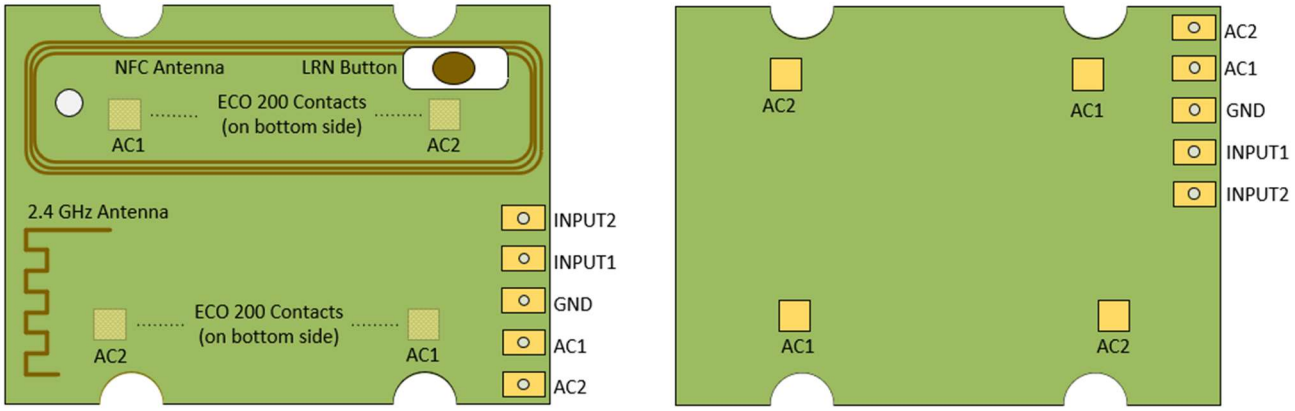

Figure 3 – PTM 535BZ product interface

The following chapters describe the key components of this product interface.

2.4.1 Energy harvester interface

PTM 535BZ is designed to operate based on the energy supplied by a kinetic energy harvester such as ECO 200. Refer to [1] for a description of ECO 200.

ECO 200 provides the harvested energy to PTM 535BZ using its AC1 and AC2 output pins. The polarity of the voltage identifies the direction of the ECO 200 action (press or release) which is transmitted by PTM 535BZ as part of data telegrams. For press actions, the voltage difference V(AC2) - V(AC1) is positive; for release actions this difference is negative. It is possible to reverse this press / release encoding (so that a press is treated as a release and vice versa) in PTM 535BZ data telegrams via the NFC interface.

PTM 535BZ provides the AC1 and AC2 supply input signals which have to be connected to the AC1 and AC2 supply output signals of the ECO 200 harvester or another suitable power source. Connection between PTM 535BZ and ECO 200 can either be made mechanically (direct connection between the ECO 200 contacts and the PTM 535BZ contacts) or by wiring.

For a mechanical connection, PTM 535BZ provides two pairs of AC1 and AC2 contact pads on the bottom of the PCB. Having two pairs of contacts enables the user to select the orientation of the ECO 200 harvester according to the mechanical design needs of the application. The AC1 and AC2 contact pads of PTM 535BZ can be mechanically connected to the AC1 and AC2 outputs of an ECO 200 kinetic harvester using a suitable fixation housing for ECO 200 and PTM 535BZ such as the one shown on the right in Figure 4.

For a wired connection, PTM 535BZ provides two boundary contact signals AC1 and AC2 (shown on the right side of Figure 3) which can be used for a wired connection to the AC1 and AC2 outputs of ECO 200.

Figure 4 below shows the ECO 200 harvester (on the left side) and an example for a mechanical integration with PTM 535BZ (on the right side). This figure also indicates the direction of movement which is a "Press" or "Release" action.

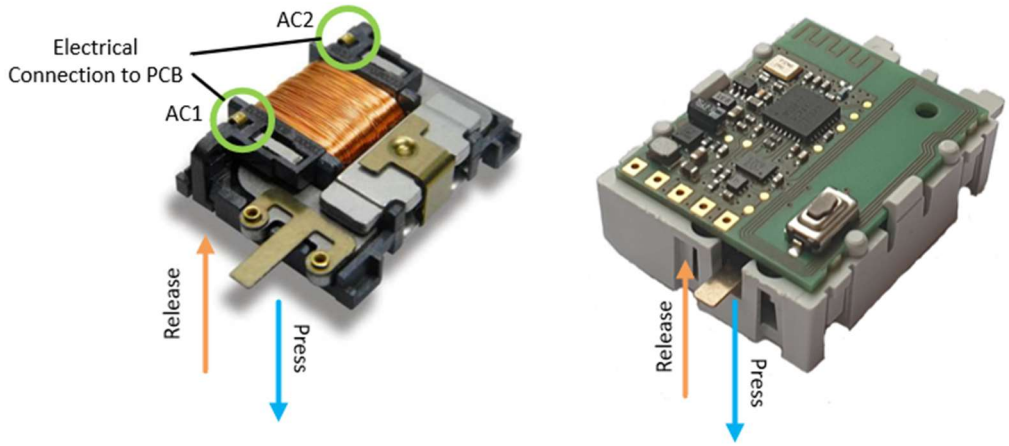

Figure 4 – ECO 200 kinetic harvester

2.4.2 External inputs

PTM 535BZ provides two electrical inputs called INPUT1 and INPUT2 and will report the status of these input pins as part of each data telegram. These signals can for instance be used for external contacts or switches.

INPUT1 and INPUT2 are active low, meaning that they will be considered to be active (connected) if they are connected to the GND signal of the PTM 535BZ product interface and inactive (not connected) if they are left open.

INPUT1 and INPUT2 must either be connected to the GND signal of the PTM 553BZ product interface or be left open. Do not connect these signals to a supply voltage!

2.4.3 LRN button

PTM 535BZ provides an LRN button which can be used to trigger the transmission of a commissioning (LRN) telegram and to execute the channel selection process in Zigbee Green Power.

2.4.4 Radio subsystem

PTM 535BZ integrates a radio transceiver including a 2.4 GHz antenna for the transmission of Bluetooth Low Energy (BLE) or Zigbee Green Power (ZGP) radio telegrams. The BLE radio functionality is described in Chapter 3; the ZGP radio functionality is described in Chapter 4.

By default, PTM 535BZ will transmit BLE telegrams. Transmission of ZGP telegrams can be selected using the NFC interface as described in Chapter 5.

2.4.5 NFC interface

PTM 535BZ provides an NFC interface with integrated NFC antenna which can be used to configure PTM 535BZ parameters. The NFC interface uses the ISO15693 standard and is described in Chapter 5.

PTM 535BZ uses the ISO15693 (long range) variant of the NFC standard to achieve the best possible NFC communication distance based on the very limited available NFC antenna space.
Other EnOcean NFC products (such as PTM 215B, STM 550B or EMDCB) use the ISO14443 (high speed) variant of the NFC standard. NFC-enabled smartphones typically support both NFC standard variants.

2.5 Security Keys

PTM 535BZ authenticates data telegrams based on an authentication signature as described in Chapter 3.4.2 for BLE data telegrams and in Chapter 4.4.4 for ZGP data telegrams.

In addition to that, PTM 535BZ provides for BLE data telegrams the option to obfuscate the sender identity by using Resolvable Private Addresses that are generated using an Identity Resolution Key as described in Chapter 3.3.5.2.

The authentication and obfuscation functionalities are based on a device-specific random key. PTM 535BZ provides SECURITY_KEY1 and SECURITY_KEY2 for this purpose.

SECURITY_KEY1 is programmed at manufacturing, can be changed by the user via the NFC interface and is NFC-readable. SECURITY_KEY1 will be reset to its factory-programmed value by a Factory Reset as described in Chapter 5.1.4.3.

SECURITY_KEY2 has to be programmed by the user via the NFC interface and is not NFC readable. SECURITY_KEY2 will be updated to a new random value upon Factory Reset as described in Chapter 5.1.4.3 or – if PTM 535BZ is transmitting ZGP data telegrams – upon a ZGP decommissioning request as described in Chapter 5.1.4.2.

It is user-selectable via NFC if SECURITY_KEY1 or SECURITY_KEY2 is used. By default, SECURITY_KEY1 is used. Use of SECURITY_KEY2 can be configured via the NFC interface as described in Chapter 5.6.4 for the case of BLE and Chapter 5.6.10 for the case of ZGP.

In addition to these two security keys, SECURITY_KEY3 is an additional security key intended for future use in ZGP applications as pre-shared key (or Install Code) to encrypt the actual security key that is transmitted in the ZGP commissioning telegram.

3 Bluetooth Low Energy (BLE) radio

By default, PTM 535BZ is configured to transmit BLE telegrams. The format of these telegrams is the similar to the format used by PTM 215B. Refer to [2] for a detailed description of the BLE telegram format.

PTM 535BZ can transmit two types of BLE telegrams:

- Data telegrams
  Data telegrams report the button status of PTM 535BZ

- Commissioning telegrams
  Commissioning telegrams provide PTM 535BZ device parameters necessary for the receiver to interpret and authenticate data telegrams

PTM 535BZ transmits BLE commissioning telegrams if the ECO 200 harvester is actuated and either the LRN button is pressed or transmission of a commissioning telegram has been requested via the NFC interface.

If the LRN button remains pressed, then commissioning telegrams will be transmitted whenever the same ECO action (press or release) is executed as when the LRN button became pressed and ECO 200 was actuated. For the other direction (release or press), no telegram will be transmitted.

If for instance the LRN button is pressed by the user and ECO 200 is then actuated into "press" direction, then PTM 535BZ will transmit commissioning telegrams as long as the LRN button remains pressed and ECO 200 is moved into the press direction. No telegrams will be transmitted as long as the LRN button remains pressed and ECO 200 is moved into the release direction.

If transmission of a commissioning telegram is requested via NFC as described in Chapter 5.1.4.1, then PTM 535BZ will transmit a commissioning telegram upon the next actuation (either press or release) of the ECO 200 harvester. After that, the request to transmit a commissioning telegram will be cleared.

If the LRN button is not pressed when ECO 200 is actuated and no transmission of a commissioning telegram is requested via the NFC interface, then PTM 535BZ transmits data telegrams if the ECO 200 harvester is actuated.

Data and commissioning telegrams share the same high-level telegram format and differ only in the payload as described in subsequent chapters.

3.1 Radio parameters

PTM 535BZ transmits Bluetooth Low Energy (BLE) advertising telegrams within the 2.4 GHz radio frequency band (2402MHz … 2480MHz) as defined in [2].

3.1.1 Bit rate

By default, PTM 535BZ uses a bit rate of 1 Mbit/s as defined in [2]. PTM 535BZ also supports a custom bit rate of 2 Mbit/s which can be selected via the NFC interface as described in Chapter 5.6.3. Note that the 2 Mbit/s bit rate uses custom radio settings and is intended for use only with certain partner applications.

3.1.2 Radio channels

By default, PTM 535BZ will use the three BLE advertising channels (BLE Channel 37, 38 and 39) defined for transmission. Use of different radio channels within the frequency band from 2402 MHz to 2480 MHz can be configured using the NFC configuration interface as described in Chapter 5.6.3 and Chapter 5.6.7.

Table 1 below summarizes the supported radio channels that can be selected via the NFC configuration interface.

Radio Channel	Frequency	Channel Type
BLE Radio Channels		
37	2402 MHz	BLE Advertising Channel
0	2404 MHz	BLE Data Channel
1	2406 MHz	BLE Data Channel
…		
10	2424 MHz	BLE Data Channel
38	2426 MHz	BLE Advertising Channel
11	2428 MHz	BLE Data Channel
12	2430 MHz	BLE Data Channel
…		
36	2478 MHz	BLE Data Channel
39	2480 MHz	BLE Advertising Channel
Custom Data Channels		
40	2403 MHz	Custom Radio Channel
41	2405 MHz	Custom Radio Channel
…		
77	2477 MHz	Custom Radio Channel
78	2479 MHz	Custom Radio Channel

Table 1 – PTM 535BZ supported BLE radio channels

Figure 5 below illustrates the BLE advertising channel, BLE data channel and custom data channel assignment within the 2.4 GHz ISM band.

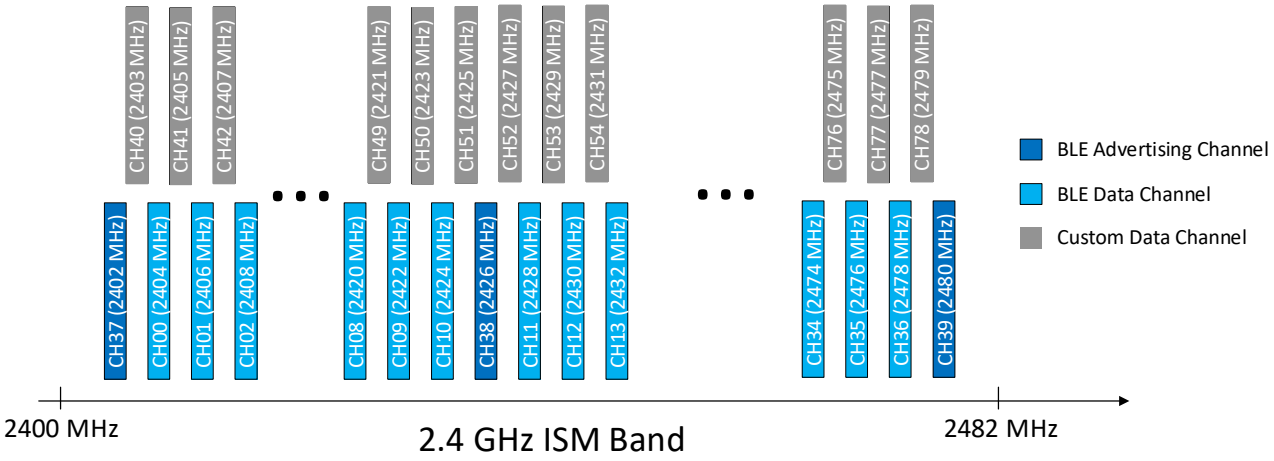

Figure 5 – PTM 535BZ BLE radio channel assignment within the 2.4 GHz ISM band

3.1.3 Data whitening

Data whitening prevents data with longs sequences of 0's and 1's from introducing a DC bias into the transmitted signal or from having a non-uniform power distribution over the occupied channel bandwidth.

To do so, the input data is reformatted based on defined rules and defined initialization values. The initialization value for data whitening is set as follows:

- For BLE data channels (0 … 36) and BLE advertising channels (37, 38 and 39)
  Initialization value is set according to specification (value = channel number)

- For custom data channels (40 … 78)
  Initialization value is equal to Center Frequency - 2400 MHz
  This means that custom channel 40 at 2403 MHz uses initialization value = 3

3.2 Radio transmission sequence

PTM 535BZ transmits BLE telegrams in its standard configuration by using so-called *BLE Advertising Events* as defined in [2]. Each data or commissioning telegram is transmitted as a sequence of redundant advertising events which all contain the same telegram payload.

For reliability reasons, PTM 535BZ will send each telegram using several (minimum two) BLE advertising events. This approach increases transmission reliability by providing redundancy in time (by transmitting the same telegram at different times) and redundancy in frequency (by transmitting the same telegram on different radio channels).

The default radio channels are the advertising channels (Channel 37, 38 and 39); they can be changed to different radio channels via the NFC configuration interface as described in Chapter 5.6.3 and Chapter 5.6.7.

The default interval between the BLE advertising events is 20 ms; this interval can be reduced to 10 ms via the NFC configuration interface as described in Chapter 5.6.3.

The resulting transmission sequence is shown in Figure 6 below for the case of data telegram transmission with default configuration parameters.

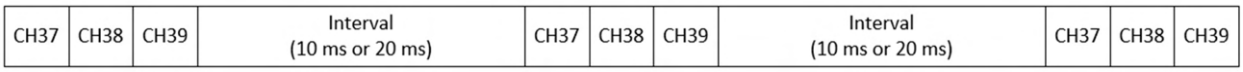

Figure 6 – BLE radio transmission sequence

3.3 Telegram format

PTM 535BZ transmits Bluetooth Low Energy (BLE) radio telegrams in the 2.4 GHz band. This chapter provides a summary of the BLE frame structure; for detailed information please refer to [2]. Figure 7 below summarizes the high-level BLE frame structure. The content of these fields is described in more detail below.

Figure 7 – BLE frame structure

3.3.1 Byte order

BLE uses little endian location meaning that if a data structure (e.g. Access Address, Header or Source Address) is bigger than one byte then the least significant byte is transmitted first.

Considering for instance the case of the four-byte Access Address `0x8E89BED6`, these 4 bytes will be transmitted (and received) in the order `0xD6` first, `0xBE` second, `0x89` third and `0x8E` last.

3.3.2 Preamble

The BLE Preamble is 1 byte long and identifies the start of the BLE frame. The value of the BLE Preamble is always set to `0xAA`.

3.3.3 Access Address

The four-byte BLE Access Address identifies the radio telegram type. For advertising frames, the value of the Access Address is always set to `0x8E89BED6`.

3.3.4 Advertising PDU Header

The Advertising PDU Header identifies certain radio telegram parameters. Figure 8 below shows the structure of the Advertising PDU header. The Advertising PDU Header is set to `0x1342` for data telegrams and `0x2442` for commissioning telegrams.

Advertising PDU Header (16 bit)									
BIT 15	...	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	...	BIT 0
Length			RxAdd	TxAdd	ChSel	RFU	PDU Type		
Data Telegram: 0x13 (19 byte follow) Commissioning: 0x24 (36 byte follow)			0b0: Unused	0b1: Random	0b0: Unused	0b0: Unused	0b0010: ADV_NONCONN_IND		

Figure 8 – Advertising PDU header structure

3.3.5 Source address

The six-byte source address identifies the originator (sender) of BLE telegrams.

PTM 535BZ supports using either static source addresses or resolvable private addresses. By default, PTM 535BZ uses static source addresses. PTM 535BZ can be configured to use resolvable private addresses as described in Chapter 5.6.4.

The type of address (static source address or resolvable private address) that is currently used can be determined by the two most significant bits of the address; both address types described in the following two chapters.

3.3.5.1 Static source address

Static source addresses are assigned during manufacturing and remain constant unless the user configures a different source address via NFC.

Static source addresses are identified by the two most significant bits (Bit 47 and Bit 46 in Figure 9) being set to `0b11`.

The structure of PTM 535BZ static source addresses is as follows:

- The upper 2 bytes of the source address are used to identify the device type and set to `0xE215` for all PTM 535BZ devices (to ensure telegram compatibility with EnOcean PTM 535BZ devices).
- The lower 4 bytes start with 0x1, are uniquely assigned to each PTM 535BZ during manufacturing and can be reconfigured via NFC as described in Chapter 5.6.6

PTM 535BZ static source addresses therefore have the format `0xE215:1xxx:yyyy`. This enables easy distinction (based on the static source address) between PTM 535BZ devices and PTM 215B devices (which use the static source address format `0xE215:0xxx:yyyy`).

Figure 9 below illustrates the static address structure used by PTM 535BZ.

STATIC SOURCE ADDRESS									
TYPE IDENTIFIER (fixed)				ADDRESS (configurable)					
BIT 47	BIT 46	...	BIT 32	BIT 31	...	BIT 28	BIT 27	...	BIT 0
0xE215				0x1			Variable		

Figure 9 – PTM 535BZ static source address structure

3.3.5.2 Resolvable private address

For some security-critical applications it might be desirable to prevent unauthorized tracking of PTM 535BZ devices based on the source address used for their radio transmissions. At the same time, PTM 535BZ devices must be unambiguously identifiable by the receiver. To address these requirements, PTM 535BZ can be configured via NFC to use resolvable private addresses (RPA) as defined by the BLE specification [2].

Using resolvable private addresses requires that PTM 535BZ and the receiver both know a common security key – the so-called Identity Resolution Key (IRK). This IRK is used to derive an authentication signature (hash) from a random, plaintext value (prand) as shown in Figure 10.

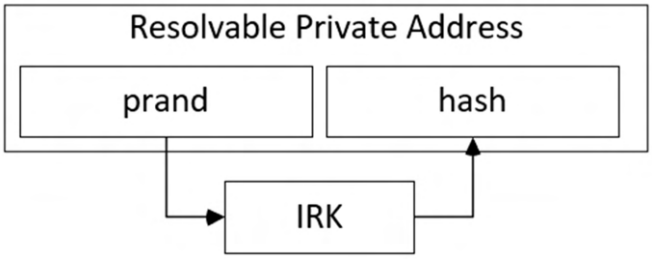

Figure 10 – Resolvable private address generation

The mechanism used to generate hash from prand and IRK is shown in Figure 11.

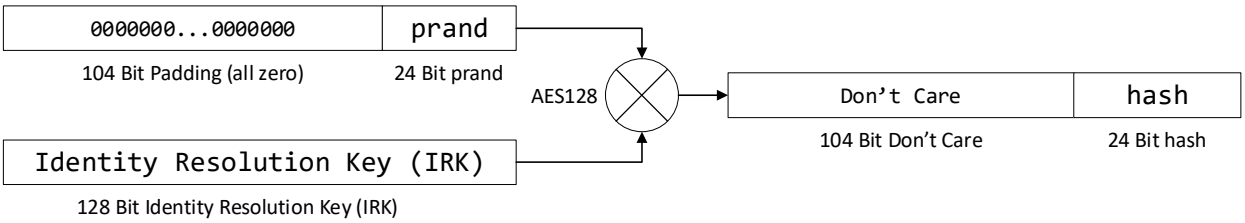

Figure 11 – Execution flow for resolving private addresses (RPA resolution)

The concatenation of 24 bit prand and 24 bit hash will then form the 48 bit resolvable private address. Resolvable private addresses are identified by the two most significant bits of prand being set to `0b01` as shown in Figure 12.

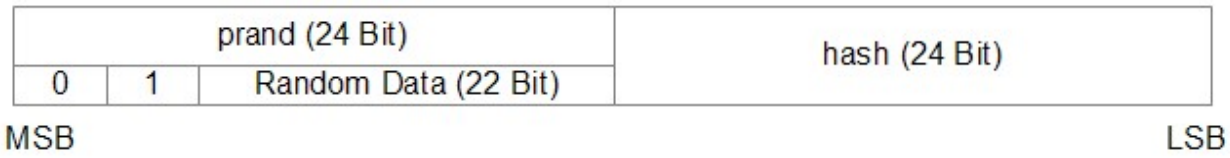

Figure 12 – BLE resolvable private address structure

The receiver will identify the transmitter based on the IRK that is used to generate the hash value from the prand value. When a transmitter is commissioned (learned in) to a receiver, the transmitter therefore has to communicate its IRK to the receiver. The receiver maintains a list of all IRK (one per transmitter) that have been commissioned.

To identify the originator of a message, the receiver will sequentially try all IRK from its list until it finds a matching IRK that derives the hash value from the prand value. This IRK then identifies the originator of the message.

Figure 13 below illustrates the address resolving scheme for resolvable private addresses. For an example of resolving a resolvable private address, please refer to Appendix E.

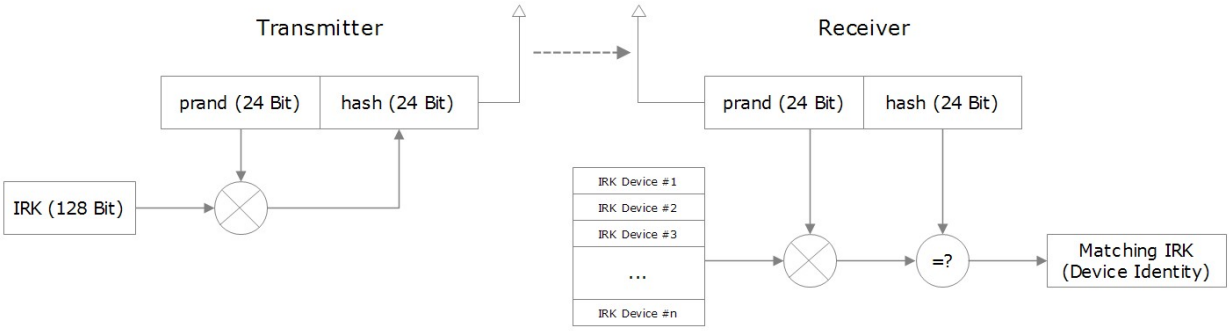

Figure 13 – Resolving of resolvable private addresses

By default, PTM 535BZ uses SECURITY_KEY1 as IRK. It is possible to configure PTM 535BZ via NFC to use SECURITY_KEY2 as IRK instead of SECURITY_KEY1 as described in Chapter 5.6.4.

3.4 Telegram payload

As described before, PTM 535BZ can transmit two types of BLE telegrams which use different telegram payloads:

- Data telegrams
  The payload of data telegrams contains the input status together with the current sequence counter value and the resulting authentication signature

- Commissioning telegrams
  The payload of commissioning telegrams contains the private security key as well as the current value of the sequence counter and the device address

The payload structure of both telegram types is described in the following chapters.

3.4.1 Data telegram payload

The payload of data telegrams is 13 byte long and consists of the following fields:

- Length (1 byte)
  The Length field specifies the combined length of the following fields. The content of the field is `0x0C` to identify 12 byte of payload that follow

- Type (1 byte)
  The Type field identifies the data type used for this telegram. For PTM 535BZ data telegrams, this field is always set to 0xFF to designate manufacturer-specific data

- Manufacturer ID (2 byte)
  The Manufacturer ID field is used to identify the manufacturer of BLE devices based on assigned numbers. EnOcean has been assigned `0x03DA` as manufacturer ID code. The Manufacturer ID can be changed via NFC as described in Chapter 5.6.5.

- Sequence Counter (4 byte)
  The Sequence Counter is a continuously incrementing counter used for security processing. It is initialized to `0` at the time of production and incremented for each telegram (data telegram or commissioning telegram) sent.

- Input Status (1 byte)
  The Input Status field reports the button action. The encoding of this field is described in Chapter 3.4.1.1.

- Security Signature (4 byte)
  The Security Signature is used to authenticate PTM 535BZ radio telegrams as described in Chapter 3.4.2

Figure 14 below illustrates the data telegram payload structure.

LEN	TYPE	MANUFACTURER_ID	SEQUENCE_COUNTER	INPUT_STATUS	AUTHENTICATION_SIGNATURE
0x0C	0xFF	0x03DA (EnOcean)	Variable (4 byte sequence counter)	Variable (1 byte status)	Variable (4 byte telegram signature)

Figure 14 – Data telegram payload structure

3.4.1.1 Input status encoding

The INPUT_STATUS field within the data telegram payload identifies the ECO 200 action (press or release) and the status of the external signals INPUT1 and INPUT2 (connected to GND or not connected to GND). PTM 535BZ uses the following sequence to identify and transmit button contact status:

1. Determine direction of the ECO 200 movement (press or release)
2. Read status of INPUT1 and INPUT2 signals
3. Calculate data payload and authentication signature

The ECO 200 action (press or release) is indicated by Bit 0 of the INPUT_STATUS field. As described in Chapter 2.4.1, the default behaviour is that a press action is a move of the ECO 200 spring away from the PCB and a release action is a move of the ECO 200 spring towards the PCB. This default behaviour can be inverted using the NFC interface if required (so that a press action would be a move of the ECO 200 spring towards the PCB and a release action would be a move of the ECO 200 spring away from the PCB).

If INPUT1 or INPUT2 are connected to GND while ECO 200 is actuated (press action or release action), then this is indicated by the according status bit set to '1'. The structure of the INPUT_STATUS field used in BLE data telegrams is shown Figure 15 in below.

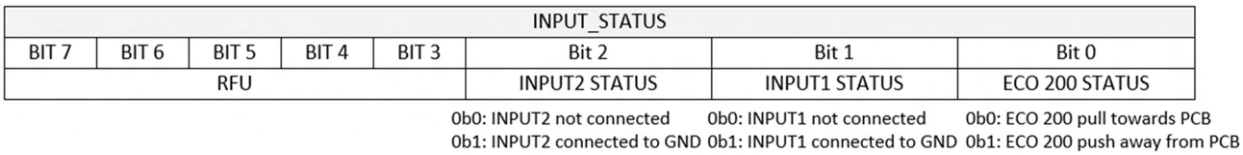

Figure 15 – INPUT_STATUS structure

Table 2 below summarizes the default INPUT_STATUS encoding for the eight possible combinations of ECO 200 action, INPUT1 status and INPUT2 status.

INPUT2 Status	INPUT1 Status	ECO 200	INPUT_STATUS
Not connected	Not connected	Press	0x01
Not connected	Not connected	Release	0x00
Not connected	Connected to GND	Press	0x03
Not connected	Connected to GND	Release	0x02
Connected to GND	Not connected	Press	0x05
Connected to GND	Not connected	Release	0x04
Connected to GND	Connected to GND	Press	0x07
Connected to GND	Connected to GND	Release	0x06

Table 2 – Default INPUT_STATUS encoding

3.4.1.2 Custom INPUT_STATUS encoding

PTM 535BZ enables the user to define (via the NFC interface) different values to be transmitted in the INPUT_STATUS field for some or all of the eight possible input events shown in Table 2 above.

To do so, PTM 535BZ provides an eight-entry custom encoding table in NFC where each entry contains a user-defined value that will be sent as INPUT_STATUS in BLE data telegrams instead of the default value if the corresponding input event occurs and the use of the custom encoding table has been enabled via NFC.

The user can configure any value between `0x00` and `0xFE` for each of the eight possible input combinations listed in Table 3. If the use of the custom encoding table has been enabled via NFC, then the configured value in the applicable table index will be transmitted in the INPUT_STATUS field of the BLE data telegram.

Setting an entry to `0xFF` means that PTM 535BZ will not transmit a data telegram if this particular input event occurs. This could for instance be useful if PTM 535BZ should send a data telegram only on button push, but no data telegram on button release.

The default values of the eight entries in the custom encoding table (shown in Table 3 below) are defined such that they reflect the standard INPUT_STATUS definition shown in Table 2 above.

Table Index	INPUT2 Status	INPUT1 Status	ECO 200 Action	INPUT_STATUS
BLE_INPUT_STATUS_0	Not connected	Not connected	Press	0x01
BLE_INPUT_STATUS_1	Not connected	Not connected	Release	0x00
BLE_INPUT_STATUS_2	Not connected	Connected to GND	Press	0x03
BLE_INPUT_STATUS_3	Not connected	Connected to GND	Release	0x02
BLE_INPUT_STATUS_4	Connected to GND	Not connected	Press	0x05
BLE_INPUT_STATUS_5	Connected to GND	Not connected	Release	0x04
BLE_INPUT_STATUS_6	Connected to GND	Connected to GND	Press	0x07
BLE_INPUT_STATUS_7	Connected to GND	Connected to GND	Release	0x06

Table 3 – Custom encoding table for INPUT_STATUS

3.4.2 BLE data telegram authentication

PTM 535BZ implements telegram authentication for transmitted BLE data telegrams to ensure that only telegrams from transmitters using a previously exchanged security key will be accepted by the receiver.

Authentication of BLE data telegrams relies on a 32 bit telegram signature which is calculated as shown in Figure 16 below and exchanged as part of the radio telegram. This mechanism is identical to the mechanism used in PTM 215B.

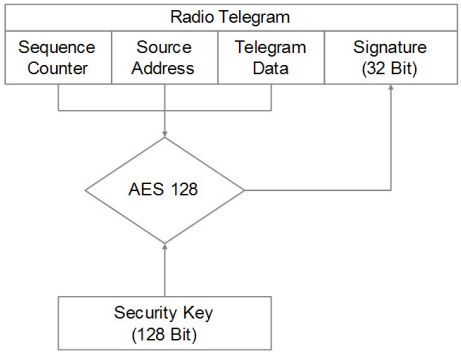

Figure 16 – Telegram authentication flow

Sequence counter, source address and the remaining telegram data together form the input data for the signature algorithm. Input data and the device-unique 128 bit security key are used as input to the RFC3610 algorithm [3] which generates a 32 bit signature. This signature which will be transmitted as part of the radio telegram.

The signature is therefore dependent both on the current value of the sequence counter, the device source address and the telegram payload. Changing any of these three parameters will therefore result in a different signature.

The receiver performs the same signature calculation based on sequence counter, source address and the remaining telegram data of the received telegram using the security key it received from PTM 535BZ during commissioning.

The receiver then compares the signature reported as part of the telegram with the signature it has calculated. If these two signatures match, then the receiver knows that the transmitter (PTM 535BZ) and receiver possess the same security key and that the message content (address, sequence counter, data) has not been modified.

In order to avoid message replay (capture and retransmission of a valid message), it is required that the receiver tracks the value of the sequence counter used by PTM 535BZ and only accepts messages with higher sequence counter values (i.e. not accepts equal or lower sequence counter values for subsequent telegrams).

By default, the factory programmed SECURITY_KEY1 is used for data telegram authentication and resolvable private address generation as described in Chapter 3.3.5.2. It is possible to configure PTM 535BZ via NFC (as described in Chapter 5.6.4) to use SECURITY_KEY2 instead of SECURITY_KEY1.

3.4.2.1 Authentication implementation

PTM 535BZ implements data telegram authentication as described in IETF RFC3610 [4].

The 13 Byte Nonce (number used once – unique) initialization value is constructed as concatenation of 6 byte Source Address, 4 byte Sequence Counter and 3 bytes of value `0x00` (for padding). Note that both Source Address and Sequence Counter use little endian format (least significant byte first).

Figure 17 below shows the structure of the Nonce.

Nonce												
STATIC SOURCE ADDRESS (Little Endian)						SEQUENCE COUNTER (Little Endian)				ZERO (Padding)		
BYTE0	BYTE1	BYTE2	BYTE3	0x15	0xE2	BYTE0	BYTE1	BYTE2	BYTE3	0x00	0x00	0x00

Figure 17 – Nonce structure

The Nonce and the 128 bit device-unique security key (by default SECURITY_KEY1, alternatively SECURITY_KEY2) are then used to calculate a 32 bit signature of the authenticated telegram payload shown in Figure 18 below.

AUTHENTICATED PAYLOAD								
LEN	TYPE	MANUFACTURER_ID (Little Endian)		SEQUENCE_COUNTER (Little Endian)				INPUT_STATUS
0x0C	0xFF	0xDA	0x03	BYTE0	BYTE1	BYTE2	BYTE3	BYTE0

Figure 18 – Authenticated payload

The calculated 32 bit signature is then appended to the data telegram payload as shown in Figure 14 in Chapter 3.4.1.

In addition to the RFC3610 standard [1] itself, please refer to Appendix D for a step-by-step description of the authentication process.

3.4.3 Commissioning telegram payload

The payload of commissioning telegrams is 30 bytes long and consists of the following fields:

- Length (1 byte)
  The Length field specifies the combined length of the following fields. For PTM 535BZ commissioning telegrams, this field is always set to `0x1D` to indicate that 29 byte of manufacturer-specific data follow.

- Type (1 byte)
  The Type field identifies the data type used for this telegram. This field is always set to `0xFF` to indicate a "Manufacturer-specific Data" field

- Manufacturer ID (2 byte)
  The Manufacturer ID field is used to identify the manufacturer of BLE devices based on assigned numbers. This field is by default set to `0x03DA` (EnOcean GmbH) but can be reconfigured using the NFC interface.

- Sequence Counter (4 byte)
  The Sequence Counter is a continuously incrementing counter used for security processing. It is initialized to `0x00000000` at the time of production and incremented for each telegram (data telegram or commissioning telegram) sent.

- Security Key (16 byte)
  Each PTM 535BZ device contains its own 16 byte device-unique random security key (SECURITY_KEY1) which is generated and programmed during manufacturing. It is transmitted during commissioning to enable the receiver to authenticate PTM 535BZ data telegrams and used as IRK for the case of resolvable private address mode.

- Static Source Address (6 byte)
  The Static Source Address is used to uniquely identify each BLE device. It is transmitted as part of the BLE frame as described in Chapter 3.3.5. Some receiver devices (most notably iOS-based products) however do not directly expose this address to their applications but rather assign a random value instead. The Static Source Address is therefore also transmitted as part of the commissioning telegram payload so that receivers can identify the source address of the sender.

Figure 19 below illustrates the commissioning telegram payload.

LEN	TYP	Manufacturer ID	Manufacturer-specific Data		
0x1D	0xFF	0x03DA	Sequence Counter (4 Byte)	Security Key (16 Byte)	Static Source Address (6 Byte)

Figure 19 – Commissioning telegram payload structure

4 Zigbee Green Power (ZGP) radio

PTM 535BZ can be configured via the NFC interface to transmit telegrams using the Zigbee Green Power (ZGP) standard [3] instead of using the BLE standard.

In this configuration, PTM 535BZ can transmit three types of ZGP telegrams:

- Data telegrams
  Data telegrams report the button status of PTM 535BZ

- Commissioning telegrams
  Commissioning telegrams provide PTM 535BZ device parameters necessary for the receiver to interpret and authenticate data telegrams

- Decommissioning telegrams
  Decommissioning telegrams signal to the receiver that PTM 535BZ data telegrams are no longer intended for the receiver. This can be helpful for instance if a switch is moved to a different room and therefore shall provide input to different receivers.

PTM 535BZ transmits ZGP decommissioning telegrams if transmission of a decommissioning telegram has been requested via the NFC interface.

PTM 535BZ transmits ZGP commissioning telegrams if the ECO 200 harvester is actuated and either the LRN button is pressed or transmission of a commissioning telegram has been requested via the NFC interface.

If the LRN button remains pressed, then commissioning telegrams will be transmitted whenever the same ECO action (press or release) is executed as when the LRN button became pressed and ECO 200 was actuated. For the other direction (release or press), no telegram will be transmitted.

If for instance the LRN button is pressed by the user and ECO 200 is then actuated into "press" direction, then PTM 535BZ will transmit commissioning telegrams as long as the LRN button remains pressed and ECO 200 is moved into the press direction. No telegrams will be transmitted as long as the LRN button remains pressed and ECO 200 is moved into the release direction.

If transmission of a commissioning telegram is requested via NFC as described in Chapter 5.1.4.1, then PTM 535BZ will transmit a commissioning telegram upon the next actuation (either press or release) of the ECO 200 harvester. After that, the request to transmit a commissioning telegram will be cleared.

If the LRN button is not pressed when ECO 200 is actuated and no transmission of a commissioning telegram or a decommissioning telegram has been requested via the NFC interface, then PTM 535BZ transmits data telegrams if the ECO 200 harvester is actuated.

All three telegram types share the same high-level telegram format and differ only in the payload as described in subsequent chapters.

4.1 Radio channels

PTM 535BZ transmits ZGP telegrams on one of the sixteen IEEE 802.15.4 radio channel within the 2.4 GHz radio frequency band (2402MHz … 2480MHz). The IEEE 802.15.4 radio channel assignment is shown in Table 4 below.

Primary channels (marked bold) are specified as preferred channels for the transmission of ZGP telegrams. When a ZGP network is formed, those primary channels will be evaluated first when selecting the radio channel used by the ZGP network. Most ZGP systems therefore operate on one of the primary channels.

Channel Number	Channel Type	Center Frequency
11 (default)	**Primary Channel**	**2405 MHz**
12	Standard Channel	2410 MHz
13	Standard Channel	2415 MHz
14	Standard Channel	2420 MHz
15	**Primary Channel**	**2425 MHz**
16	Standard Channel	2430 MHz
17	Standard Channel	2435 MHz
18	Standard Channel	2440 MHz
19	Standard Channel	2445 MHz
20	**Primary Channel**	**2450 MHz**
21	Standard Channel	2455 MHz
22	Standard Channel	2460 MHz
23	Standard Channel	2465 MHz
24	Standard Channel	2470 MHz
25	**Primary Channel**	**2475 MHz**
26	Standard Channel	2480 MHz

Table 4 – IEEE 802.15.4 radio channels

By default, PTM 535BZ uses IEEE 802.15.4 radio channel 11 (which is a primary channel) when transmitting ZGP telegrams. Other channels can be selected via the NFC configuration interface as described in Chapter 5.6.9 or using the LRN button as part of the commissioning process as described in Chapter 4.5.2.

4.2 Radio transmission sequence

PTM 535BZ transmits ZGP data telegrams as a set of redundant transmissions where the same data telegram is transmitted 3 times. The timing interval between the start of two consecutive redundant data telegrams is approximately 2.5 ms and varies by some random timing offset.

Figure 20 below shows the ZGP radio transmission sequence used by PTM 535BZ for data telegrams.

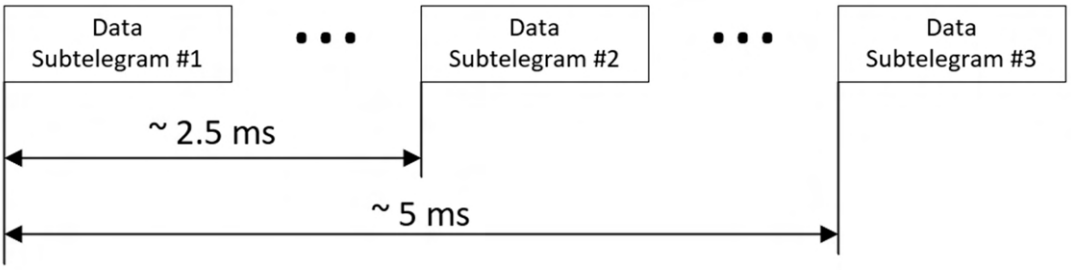

Figure 20 – ZGP radio transmission sequence for data telegrams

PTM 535BZ transmits ZGP commissioning telegrams (which are much longer than ZGP data telegrams) as a set of redundant transmissions where the same data telegram is transmitted 2 times. The timing interval between the start of the two consecutive redundant commissioning telegrams is approximately 2.5 ms and varies by some random timing offset.

Figure 21 below shows the ZGP radio transmission sequence used by PTM 535BZ for commissioning telegrams.

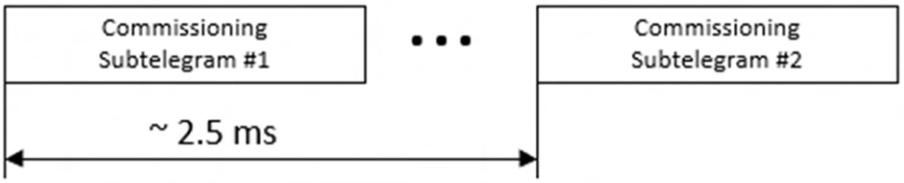

Figure 21 – ZGP radio transmission sequence for commissioning telegrams

4.3 Telegram format

PTM 535BZ transmits ZGP radio telegrams in the 2.4 GHz band according to the Zigbee Green Power specification [3] which uses IEEE 802.15.4 [5] as lower layer standard. Please refer to these specifications for detailed, up to date information.

Figure 22 below summarizes the high-level IEEE 802.15.4 / Zigbee Green Power frame structure. The content of these fields is described in more detail in the next chapters.

<table>
<tr><td rowspan="2">IEEE 802.15.4
PHY Header</td><td rowspan="2">IEEE 802.15.4
MAC Header</td><td colspan="3">IEEE 802.15.4 MAC Payload</td><td rowspan="2">IEEE 802.15.4
MAC Trailer</td></tr>
<tr><td>ZGP Network Header</td><td>ZGP Application Payload</td><td>ZGP Network Trailer</td></tr>
</table>

Figure 22 – IEEE 802.15.4 / Zigbee Green Power frame structure

4.3.1 Data integrity

Correct reception of the IEEE 802.15.4 frame is ensured using a 2 byte Cyclic Redundancy Check (CRC16) which forms the IEEE 802.15.4 MAC Trailer field.

4.3.2 Byte order

ZGP uses little endian byte order meaning that if a data structure (e.g. Source Address, Frame Control or Sequence Number) is bigger than 1 byte then the least significant byte is transmitted first. Considering the case of the 4 byte Source Address `0x01501234`, these 4 bytes will be transmitted (and received) in the order `0x34` first, `0x12` second, `0x50` third and `0x01` last.

4.3.3 IEEE 802.15.4 PHY Header

The IEEE 802.15.4 PHY header consists of the following fields:

- Preamble (4 byte long, always `0x0000:0000`)
- Start of Frame (1 byte long, always `0xA7`)
- Length of Frame (1 byte long, length depending on ZGP payload length)

The structure of the IEEE 802.15.4 PHY header is shown in Figure 23 below.

IEEE 802.15.4 PHY Header		
Preamble	Start of Frame	Frame Length
4 Byte	1 Byte	1 Byte
0x00000000	0xA7	Variable

Figure 23 – IEEE 802.15.4 PHY header structure

4.3.3.1 Frame Length

The *Frame Length* of the 802.15.4 frame depends on the telegram type (data telegram or commissioning telegram), the Device ID (identifying the device type as described in Chapter 4.4.1.1) and the length of the command list transmitted as part of the commissioning telegram (as discussed in Chapter 4.4.2.3). Table 5 below lists the telegram length for the supported telegram types.

Telegramm Type	Device ID	Command List / App Info	Length
Data Telegram	0x07 (default)	N.A.	25 byte (0x19)
	0x00 … 0x06, 0x10	N.A.	24 byte (0x18)
Commissioning Telegram	0x07 (default)	App Info always present	46 byte (0x2E)
	0x00 … 0x06, 0x10	Omitted (legacy)	42 byte (0x2A)
		Command list with 1 command	45 byte (0x2D)
		Command list with 2 commands	46 byte (0x2E)
		Command list with 3 commands	47 byte (0x2F)
		Command list with 4 commands	48 byte (0x30)
		Command list with 5 commands	49 byte (0x31)
		Command list with 6 commands	50 byte (0x32)
		Command list with 7 commands	51 byte (0x33)
		Command list with 8 commands	52 byte (0x34)

Table 5 – Telegram length for supported telegram types

4.3.4 IEEE 802.15.4 MAC Header

The IEEE 802.15.4 MAC Header contains the following fields:

- IEEE 802.15.4 Frame Control Field (1 byte)
  The *Frame Control Field* is `0x0801` for all ZGP telegram types supported by PTM 535BZ
- Sequence Number (1 byte)
  The *Sequence Number* is an incremental number used to identify the order of telegrams
- Address Field (4 byte)
  The *Address Field* is set to `0xFFFF:FFFF` for all PTM 535BZ ZGP telegrams

Figure 24 below shows the IEEE 802.15.4 MAC header structure.

IEEE 802.15.4 MAC Header		
IEEE 802.15.4 Frame Control	Sequence Number	Destination Addr 16 PAN | 16 Bit Addr
2 Byte	1 Byte	4 Byte
0x0801	Variable	0xFFFFFFFF

Figure 24 – IEEE 802.15.4 MAC header structure

4.4 IEEE 802.15.4 MAC payload (ZGP telegram)

The IEEE 802.15.4 MAC payload contains the ZGP telegram data. Figure 25 below shows the ZGP telegram format.

ZGP Network Header				ZGP Application Payload	ZGP Network Trailer
Frame Control	*Extended Frame Control*	Source Address	*Sequence Counter*	Content depends on Telegram Type	*Authentication Signature*
1 Byte	*0 or 1 Byte*	4 Byte	*0 or 4 Byte*	Size depends on Telegram Type	*0 or 4 Byte*

Figure 25 – ZGP telegram format

The content of the ZGP telegram data field depends on the telegram type (ZGP data telegram or ZGP commissioning telegram). Some of the fields shown in Figure 25 above are not used in all telegram types. These fields are marked in *italics*. The structure of data and commissioning telegrams is described in the subsequent chapters.

4.4.1 Data telegram structure

By default, PTM 535BZ transmits data telegrams. The payload of data telegrams is either 13 byte (Device ID = 0x07, default) or 12 byte (all other supported Device ID) long.

Figure 26 below shows the telegram structure for ZGP data telegrams.

ZGP Network Header				ZGP Application Data	ZGP Network Trailer
Frame Control	Extended Control	ZGP Source Address	ZGP Sequence Counter	ZGP Command	Authentication Signature
1 Byte	1 Byte	4 Byte	4 Byte	1 Byte / 2 Byte	4 Byte
0x8C	0x30	0x015x:xxxx	Variable	Variable	Variable

Figure 26 – Structure of ZGP data telegrams

ZGP data telegrams contain the following fields:

- Frame Control (1 byte)
  The *Frame Control* field is set to `0x8C`
- Extended Frame Control (1 byte)
  The *Extended Frame Control* field is set to `0x30`
- Source Address (4 byte)
  The *Source Address* uniquely identifies the originator (sender) of ZGP telegrams. This Source ID is assigned by Zigbee Alliance and cannot be changed by the user.
- Sequence Counter (4 byte)
  The *Sequence Counter* is a continuously incrementing counter used for security processing. It is initialized to `0` at the time of production and incremented for each telegram (data telegram or commissioning telegram) sent. The least significant byte of the Sequence Counter is used as Sequence Number in the IEEE 802.15.4 MAC Header.

- ZGP command (2 byte or 1 byte)
  The ZGP command identifies the switch action. Format and encoding used by the ZGP command depend on the selected device ID.

- Security Signature (4 byte)
  The Security Signature is used to authenticate PTM 535BZ ZGP radio telegrams as described in chapter 4.4.4

4.4.1.1 Device ID

The application data format within the data telegram is determined by the ZGP Device ID which identifies the device type as defined by the ZGP specification [3].

Table 6 below lists the ZGP Device ID that are supported by PTM 535BZ.

Device ID	Description	Payload size
0x00	Simple Generic 1-state Switch	1 byte
0x01	Simple Generic 2-state Switch	1 byte
0x02	ON / OFF Switch	1 byte
0x03	Level Control Switch	1 byte
0x05	Advanced Generic 1-state Switch	1 byte
0x06	Advanced Generic 2-state Switch	1 byte
0x07 (Default)	**Generic 8-contact Switch**	**2 byte**
0x10	GP Color Dimmer Switch	1 byte

Table 6 – Supported ZGP Device ID

The default Device ID used by PTM 535BZ is 0x07 (Generic 8-contact Switch) as described in Chapter 4.4.1.2.

It is possible to select another supported Device ID via the NFC interface. In that case, it is required to define the set of commands to be used for the different input events as described in Chapter 4.4.1.3.

4.4.1.2 Device ID 0x07: Generic 8-contact Switch

Device ID 0x07 (Generic 8-contact Switch) is the default configuration used by PTM 535BZ when transmitting ZGP data telegrams. Figure 27 below illustrates the ZGP command structure used by generic switch data telegrams.

ZGP Command	Button Status
1 Byte	1 Byte
0x69 (Press) / 0x6A (Release)	Variable

Figure 27 – ZGP payload structure for Device ID 0x07

Within this structure, the ZGP Command encodes the action of the ECO 200 harvester connected to PTM 535BZ (Press or Release) as described in Chapter 2.4.1.

The default behaviour is that a press action (ZGP command 0x69) is a move of the ECO 200 spring away from the PCB and a release action (ZGP command 0x6A) is a move of the ECO 200 spring towards the PCB.

This default behaviour can be inverted using the NFC interface if required (so that a press action would be a move of the ECO 200 spring towards the PCB and a release action would be a move of the ECO 200 spring away from the PCB).

The Button Status field following the 0x69 / 0x6A ZGP Command encodes the status of the input signals INPUT1 and INPUT2 as defined by the ZGP specification [3].

Table 7 below shows the resulting encoding of PTM 535BZ for Device ID 0x07.

INPUT2	INPUT1	ECO 200	ZGP Command	Button Status
Not connected	Not connected	Press action	0x69	0b00000001
Not connected	Not connected	Release action	0x6A	0b00000001
Not connected	Connected	Press action	0x69	0b00000010
Not connected	Connected	Release action	0x6A	0b00000010
Connected	Not connected	Press action	0x69	0b00000100
Connected	Not connected	Release action	0x6A	0b00000100
Connected	Connected	Press action	0x69	0b00000110
Connected	Connected	Release action	0x6A	0b00000110

Table 7 – ZGP command and button status encoding for Device ID 0x07

4.4.1.3 Device ID other than 0x07

It is possible to configure PTM 535BZ via NFC to use one of the alternative Device ID listed in Table 6 instead of the default Device ID 0x07.

The ZGP specification [3] defines the minimum set of commands that must be supported for each Device ID but leaves it up to the user to define which input actions trigger these commands. The specification also allows the user to define additional commands for the remaining input actions. The receiver is informed about the set of commands used by PTM 535BZ by means of the command list which is part of the commissioning telegram as described in Chapter 4.4.2.3.

PTM 535BZ therefore uses a command table with eight entries corresponding to each of the eight possible combinations of ECO 200 action (Press or Release), INPUT1 and INPUT2 status (Connected or Not Connected) to determine which ZGP command will be sent upon the corresponding input status when PTM 535BZ is configured to use a DEVICE ID other than 0x07.

It is NFC configurable which direction of ECO 200 movement is considered as Press action and which is considered as Release action. The default behaviour is that a Press action is a move of the ECO 200 spring away from the PCB and that a Release action is a move of the ECO 200 spring towards the PCB.

Table 8 below shows the structure of this command table together with the default ZGP command for each of the eight possible combinations.

Table Index	INPUT2 Status	INPUT1 Status	ECO 200 Action	Default Command
ZGP_COMMAND_0	Not connected	Not connected	Press action	0x22
ZGP_COMMAND_1	Not connected	Not connected	Release action	0x23
ZGP_COMMAND_2	Not connected	Connected	Press action	0x12
ZGP_COMMAND_3	Not connected	Connected	Release action	0x13
ZGP_COMMAND_4	Connected	Not connected	Press action	0x14
ZGP_COMMAND_5	Connected	Not connected	Release action	0x15
ZGP_COMMAND_6	Connected	Connected	Press action	0x16
ZGP_COMMAND_7	Connected	Connected	Release action	0x17

Table 8 – ZGP command table for Device ID other than 0x07

The default ZGP commands in the command table shown above have been defined to meet the requirements for DEVICE ID = 0x02 (ON / OFF Switch) and provide backwards compatibility to PTM 535Z.

The command table can be modified by the user according to application and specification requirements by writing the corresponding ZGP command via the NFC interface as described in Chapter 5.6.13. PTM 535BZ will accept (and transmit) one-byte values between `0x00` and `0xFE`. PTM 535BZ will not perform compliance checking on the defined commands.

If an entry in the table is set to `0xFF` (which is not a valid ZGP command for a ZGP device) then no data telegram will be transmitted for the corresponding input status. This can be useful if for instance a push button should only transmit data telegrams upon press.

4.4.2 Commissioning telegram

Transmission of a commissioning telegram can be selected either by pressing the LRN button or via the NFC interface. The commissioning telegram payload is shown in Figure 28 below.

Zigbee Green Power Protocol Payload for Commissioning Telegrams							
Valid for all Device ID							Device ID Specific
Command	DeviceID	Options	Ext Options	Encrypted Key	Key Hash	Sequence Counter	Application Info
1 Byte	1 Byte	1 Byte	1 Byte	16 Byte	4 Byte	4 Byte	Variable
0xE0	0x07	0x85	0xF2	Variable	Variable	Variable	Variable

Figure 28 – ZGP payload structure for commissioning telegrams

The commissioning telegram contains the following fields that are common to all Device ID:

- ZGP Network Header (6 byte)
  The ZGP Network Header is similar to that of data telegrams; the Sequence Counter field is omitted in the ZGP Network Header as it is part of the Application Payload

- Command (1 byte)
  The ZGP Command field is set to `0xE0` to identify a commissioning telegram

- Device ID (1 byte)
  The Device Type is set to the Device ID used by PTM 535BZ.
  By default, Device ID 0x07 (Generic Eight Button Switch) is used and therefore this field is set to `0x07`. If an alternative Device ID from the list of supported Device ID in Table 6 is selected by the user then this field will be set accordingly.

- Options (1 byte)
  The Option field provides information about the structure of the commissioning telegram. It is set to `0x85` if Application Info is present (default) and to `0x81` if Application Info is not present (optional setting via NFC).

- Extended Options (1 byte)
  The Extended Option field provides information about the security model. It is always set to `0xF2`.

- Encrypted Security Key (16 byte)
  The Encrypted Security Key field contains an encrypted representation of the 16 byte security key used by PTM 535BZ to authenticate its data telegrams.

- Key Hash (4 byte)
  The Key Hash can be used to verify if the security key was correctly decrypted

- Sequence Counter (4 byte)
  The Sequence Counter is a continuously incrementing counter used for security processing. It is initialized to 0 at the time of production and incremented for each telegram (data telegram or commissioning telegram) sent.

Size and format of the Application Information depends on the selected DEVICE ID as described below.

4.4.2.1 Application Information

PTM 535BZ provides Application Information according to the ZGP specification [3] as part of the commissioning telegram to describe the command set used by it.

It is possible to disable the transmission of the Application Information field for Device ID other than 0x07 via NFC to maintain backwards compatibility to older ZGP implementations. In this case, none of the fields listed below is present.

Application Information depends on the Device ID and consists of the following fields:

- Type (1 byte)
  The *Type* field identifies the type of the application information that follows
- Length (1 byte)
  The *Length* field indicates the size (number of bytes) of application information data that follows.
- Data (variable)
  The *Data* field contains either the application information data

Figure 29 below shows the structure of the Application Information field.

Application Information Type	Application Information Length	Application Information Data
1 Byte	1 Byte	Variable

Figure 29 – Application Information structure

4.4.2.2 Application Information for Device ID 0x07

The Application Information structure for Device ID 0x07 contains the *Switch Information* field information about the switch type (Generic Switch Configuration) and the input status that triggered the commissioning event (Current Contact).

Figure 30 below illustrates the Application Information structure for Device ID 0x07.

Application Information Type	Application Information Length	Application Information Data – Switch Information	
		Generic Switch Configuration	Current Contact
1 Byte	1 Byte	1 Byte	1 Byte
0x10	0x02	Variable	Variable

Figure 30 – Application Information structure for Device ID 0x07

The *Generic Switch Configuration* field which identifies the type of switch and the number of supported switch contacts. It is by default set to `0x01` (unknown switch type, one switch contact) as the majority of PTM 535BZ applications are single push buttons or position switches. The number of contacts can be changed to other values via the NFC interface as described in Chapter 5.6.11.

The *Current Contact* field identifies the input status when the commissioning telegram was triggered allowing different receivers to identify and respond to different button actions. Figure 31 below shows the encoding of the Current Contact field.

CURRENT CONTACT Button (Status at Commissioning)							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	Bit 2	Bit 1	Bit 0
RFU					INPUT2	INPUT1	ECO 200
					0b0: INPUT2 not connected 0b1: INPUT2 connected to GND	0b0: INPUT1 not connected 0b1: INPUT1 connected to GND	0b0: ECO 200 not only action (INPUT1 or INPUT2 connected) 0b1: ECO 200 is only action (INPUT1 and INPUT2 not connected)

Figure 31 – Current Contact encoding

4.4.2.3 Application Information for Device ID other than 0x07

The Application Information structure for Device ID other than 0x07 contains the list of supported commands. Figure 32 below illustrates the Application Information structure for Device ID other than 0x07.

Application Information Type	Application Information Length	Application Information Data - CommandID List		
		CommandID 1		CommandID *n*
1 Byte	1 Byte	1 Byte	...	1 Byte
0x04	0x02	Variable		Variable

Figure 32 – Application Information structure for Device ID other than ID 0x07

The *Application Information Type* field is set to `0x04` specifying that the CommandID List (list of supported commands) follows.

The *Application Information Length* field contains the number of commands that follow.

If the default command set (eight commands as listed in Table 8) is used, then the Application Information structure will have the content shown in Figure 33 below. If the commands in Table 8 are configured to different CommandID by the user then the CommandID List will change accordingly. If commands are set to non-active (0xFF) then these will be omitted from the list which will be shortened accordingly.

Application Information Type	Application Information Length	Application Info (CommandID List)							
		Command 1	Command 2	Command 3	Command 4	Command 5	Command 6	Command 7	Command 8
1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte
0x04	0x08	0x22	0x23	0x12	0x13	0x14	0x15	0x16	0x17

Figure 33 – Application Information content with default command list

4.4.3 Decommissioning telegram

Transmission of a decommissioning telegram can be selected only via the NFC interface. The payload of decommissioning telegrams is 12 byte long and shown in Figure 34 below.

Zigbee Green Power Protocol Payload for Decommissioning Telegrams					
Frame Control	Extended Control	ZGP Source Address	ZGP Sequence Counter	ZGP Command	Authentication Signature
1 Byte	1 Byte	4 Byte	4 Byte	1 Byte	4 Byte
0x8C	0x30	0x015x:xxxx	Variable	0xE1	Variable

Figure 34 – Structure of ZGP decommissioning telegrams

For security reasons, SECURITY_KEY2 will be automatically updated by PTM 535BZ to a new random value when a decommissioning request is transmitted. This prevents the case of using the same security key in different networks.

4.4.4 Data telegram authentication

PTM 535BZ implements telegram authentication for transmitted ZGP data telegrams as specified by the ZGP specification. This ensures that only telegrams from transmitters using a previously exchanged security key will be accepted by the receiver. Refer to the ZGP specification [3] for details about ZGP data telegram authentication.

4.5 Channel selection

ZGP uses the IEEE 802.15.4 radio standard [5] for telegram transmissions which defines 16 radio channels (designated as Channel 11 … Channel 26) as described in Chapter 4.1.

The radio channel used for communication is selected when a ZGP network is formed and usually remains the same throughout the lifetime of the network. The channel selection process is designed to ensure that a certain channel quality (low disturbances) is achieved.

Four of the 16 radio channels (Channel 11, Channel 15, Channel 20 and Channel 25) are designated as Primary Channels and will be tried first in the channel selection process. Most ZGP networks therefore operate on one of these four primary channels.

Devices within a ZGP network can receive radio telegrams from PTM 535BZ only if PTM 535BZ uses the same radio channel as the ZGP network. PTM 535BZ therefore has to be configured to use the right radio channel. This process is called channel selection.

Channel selection can either be executed via the NFC interface or via the LRN button. Both options are described below.

4.5.1 Channel selection via NFC

The radio channel used by PTM 535BZ can be selected via the NFC interface. To do so, the user first needs to determine the radio channel used by the ZGP network that shall receive the radio telegrams of PTM 535BZ. This is typically done by means of a commissioning application (such as an application on a smartphone) that can communicate with the devices in the ZGP network (for instance via a gateway).

In this process, the user will first determine the radio channel used by the ZGP network via the commissioning application and then configure PTM 535BZ via its NFC interface to use this radio channel. Smartphones with NFC interface allow combining these two steps within one application which enables quick and reliable configuration of PTM 535BZ.

4.5.2 Channel selection via LRN button

While channel selection via NFC is the preferred way of configuring PTM 535BZ, certain scenarios exist where this is not possible. For instance, a simple lighting control application might consist only of ZGP switches and ZGP receivers without a gateway that would allow a smartphone application to connect to the network.

For these cases, PTM 535BZ offers a manual mode of channel selection which is triggered by pressing and holding the LRN button while actuating the connected ECO 200 harvester.

In manual channel selection mode, PTM 535BZ will announce its identity (Source Address, Device ID, Application Information, Security Material) sequentially on different radio channels.

If a ZGP network operates on the currently used radio channel and is configured to accept new devices (for instance by pressing a dedicated button on the receiver) then it can signal to the installer (for instance by blinking the controlled light) that PTM 535BZ is now operating on the right radio channel, that its Commissioning Telegram has been received and that PTM 535BZ is now part of the ZGP network.

At this point, the installer has to release the LRN button to complete the channel selection sequence of PTM 535BZ and trigger a data telegram (by actuating the connected ECO 200 harvester after the LRN button has been released) to signal to the receiver that the configuration has been completed.

PTM 535BZ will continue to operate on the selected channel until the channel selection process is started again (for instance because the product using PTM 535BZ has been moved to a different room and should now be part of a different ZGP network).

Note that PTM 535BZ will change to a different channel if the LRN button is pressed during at least three consecutive ECO 200 actions (e.g. press – release - press). Therefore, it is strongly recommended to transmit a data telegram after completion of the channel selection sequence to avoid unintentional channel reconfiguration when the LRN button is pressed the next time.
Channel selection via the LRN button can be disabled via the NFC interface as described in Chapter 5.6.9 to prevent unintended change of the radio channel.

4.5.2.1 Channel selection sequence

The channel selection sequence will always start with the radio channel that is currently used by PTM 535BZ. This allows to communicate the identify of PTM 535BZ to additional devices in the same ZGP network (for instance to newly added devices or to devices that did not receive the initial Commissioning Telegram).

After that, PTM 535BZ will sequentially try the selected radio channels. PTM 535BZ can be configured via the NFC interface to use only use the current radio channel, to use only the Primary radio channels (11, 15, 20, 25) or to use all radio channels (11 … 26, default) as described in Chapter 5.6.9.

If PTM 535BZ is configured to only use the current radio channel, then all commissioning telegrams will be on that channel.

If PTM 535BZ is configured to use only the Primary channels, then commissioning telegrams will be sequentially transmitted on channels 11, 15, 20 and 25. If channel 25 has been reached then the sequence will be restarted at channel 11.

PTM 535BZ is by default configured to use all channels for manual channel selection. In this mode, commissioning telegrams will be sequentially transmitted on channels 11, 12, 13 … 25, 26. If channel 26 has been reached then the sequence will be restarted at channel 11.

5 NFC configuration

PTM 535BZ provides an NFC interface according to the ISO15693 standard for identifying and configuring device parameters. The NFC interface provides a textual description of key device parameters in NFC Data Exchange Format (NDEF), a register-based description of the current device configuration and a configuration update service with two permission levels.

5.1 Architecture

The NFC configuration architecture of PTM 535BZ is designed to provide configurable permissions so that a user with higher level access rights (for instance an OEM or System Integrator) can restrict the set of available configuration options for a user with lower level access rights (for instance an Installer) to avoid unintended reconfiguration of certain device parameters.

PTM 535BZ supports requests to change the value of configuration register(s) as well as requests for transmission of a commissioning telegram, for factory reset and for transmission of a ZGP decommissioning telegram (only when transmitting ZGP data telegrams).

If an update of configuration register(s) is requested, then the CONFIGURATION_SELECTION register is used to specify the configuration register(s) that shall be updated.

PTM 535BZ supports two users (USER1 and USER2) with different access rights (defined by USER1_CONFIGURATION_OPTIONS and USER2_CONFIGURATION_OPTIONS). Both users authenticate themselves using their individual PIN codes (USER1_PIN and USER2_PIN).

Figure **35** below shows the high-level NFC configuration architecture.

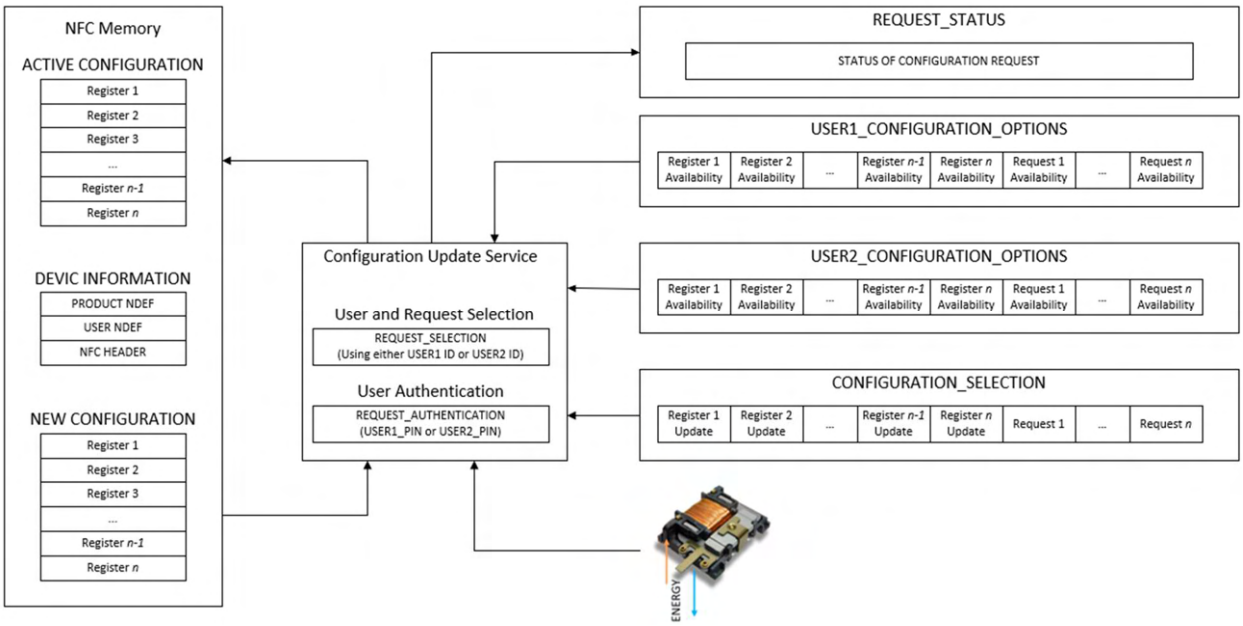

Figure 35 – NFC configuration architecture

5.1.1 Users

The NFC interface architecture of PTM 535BZ supports two different users with different levels of access rights. The user with the higher level access rights is called *USER1*; the user with the lower level access rights is called *USER2*.

USER1 has the right to change all available configuration options. The available configuration options for USER1 are defined by the register USER1_CONFIGURATION_OPTIONS.

USER2 can only change a subset of all available configuration options. The available configuration options for USER2 are defined by the register USER1_CONFIGURATION_OPTIONS.

USER1 can change (restrict) the available configuration options of USER2 so that USER2 cannot change certain configuration options or execute certain functional requests. This restriction is done by clearing the corresponding bits in the USER2_CONFIGURATION_OPTIONS register. Only USER1 can change this register.

5.1.2 PIN codes

USER1 authenticates requests by means of the 4 byte USER1_PIN while USER2 will authenticate requests by means of the 4 byte USER2_PIN.

USER1_PIN can only be changed by USER1 after providing the currently active USER1_PIN. The default value of USER1 PIN is `02:00:35:E5`.

USER2_PIN can be changed by USER1 after providing the currently active USER1_PIN or by USER2 after providing the currently active USER2_PIN.
The default value of USER2 PIN is `03:00:35:E5`.

USER1_PIN and USER2_PIN should be changed from their factory default values to prevent unauthorized access to the NFC configuration as described in Chapter 5.7.5.

Make sure that the new PIN code is properly noted especially when changing USER1_PIN. For security reasons, it is not possible to reset USER1_PIN after it has been changed.

Table 9 below summarizes the USER PIN codes.

User	PIN	PIN HASH	Default PIN	Permission to change PIN
USER1	USER1_PIN	USER1_PIN_HASH	02:00:35:E5	USER1
USER2	USER2_PIN	USER2_PIN_HASH	03:00:35:E5	USER1, USER2

Table 9 – NFC USER PIN codes

PTM 535BZ provides a 16 bit hash of the 32 bit USER1_PIN (USER1_PIN_HASH) and the 32 bit USER2_PIN (USER2_PIN_HASH). This hash can be used by a configuration tool to check if it possesses the correct USER1_PIN or USER2_PIN as described in Chapter 5.7.5.1.

5.1.3 NFC configuration

PTM 535BZ operation is configured using NFC configuration registers. PTM 535BZ will operate according to the settings of these registers. The currently active configuration registers are allocated in the ACTIVE CONFIGURATION area.

The structure of the ACTIVE CONFIGURATION area is replicated in the NEW CONFIGURATION area. This area contains a shadow register for each register in the ACTIVE CONFIGURATION area that can be changed by the user.

Registers in the NEW CONFIGURATION area are only used to update the registers of the ACTIVE CONFIGURATION area. The setting of these registers has no effect on the functionality of PTM 535BZ.

The available configuration options for USER1 or USER2 are listed in the USER1_CONFIGURATION_OPTIONS and the USER2_CONFIGURATION_OPTIONS registers. Each individual bit in that register corresponds to an individual NFC configuration register or an individual NFC functional request as described in Chapter 5.6.15.

The PTM 535BZ NFC architecture allows for a total of 32 configuration options and functional requests. In the current implementation, 20 of those are used (18 configuration options, 2 functional requests) while 12 are reserved for future use (RFU).

Each bit in the USER1_CONFIGURATION_OPTIONS that is set to `0b1` corresponds to a configuration register that is changeable or a functional request that can be made by USER1. Likewise, each bit in the USER2_CONFIGURATION_OPTIONS that is set to `0b1` corresponds to a configuration register that is changeable or a functional request that can be made by USER2.

USER1 can restrict the configuration options available to USER2 by setting the corresponding bits in the USER2_CONFIGURATION_OPTIONS to `0b0`. These configuration options are then not available to USER2 anymore and PTM 535BZ will issue a PERMISSION ERROR response to any configuration request from USER2 that contains such configuration option.

Currently unused bits (reserved for future use) which do not correspond to a changeable configuration setting are set to `0b0`. This allows USER1 and USER2 to identify which configuration options are available to them and to correctly treat different product revisions with different features sets (if for instance a newer product revision supports previously reserved configuration registers).

5.1.4 NFC functional requests

In addition to changing registers in the current configuration, USER1 and USER2 can also request the transmission of a commissioning telegram by PTM 535BZ, the transmission of a ZGP decommissioning telegram (if PTM 535BZ is configured to transmit ZGP telegrams) or request a factory reset of the PTM 535BZ configuration registers to their default values.

5.1.4.1 Commissioning request

PTM 535BZ will transmit a BLE commissioning telegram or a ZGP commissioning telegram upon pressing the LRN button while actuating the ECO 200 harvester. Additionally, the transmission of a commissioning telegram can also be requested by means of a commissioning request via the NFC interface.

If transmission of a commissioning telegram has been requested then a commissioning telegram will be transmitted upon the next actuation (press or release, whichever comes next) of the connected ECO 200 harvester. After that, PTM 535BZ will again transmit data telegrams.

5.1.4.2 ZGP decommissioning request

Transmission of a ZGP decommissioning telegram can be requested by means of a commissioning request via the NFC interface if PTM 535BZ is configured to transmit ZGP telegrams.

If transmission of a decommissioning telegram has been requested and PTM 535BZ is configured to transmit ZGP telegrams, then a ZGP decommissioning telegram will be transmitted upon the next actuation (press or release, whichever comes next) of the connected ECO 200 harvester. After that, PTM 535BZ will again transmit data telegrams.

If PTM 535BZ is configured to transmit BLE telegrams (this is the default configuration) then requesting the transmission of a ZGP decommissioning telegram will result in a PARAMETER ERROR response by PTM 535BZ as described in Chapter 5.7.1.

5.1.4.3 Factory reset request

The configuration of PTM 535BZ can either be changed via the NFC interface or – for the case of the radio channel used for the transmission of ZGP telegrams – also via the LRN button.

It is possible to reset configuration changes so that PTM 535BZ again uses its default configuration by means of a Factory Reset requested via the NFC interface. If a Factory Reset is executed, then SECURITY_KEY2 will be automatically updated to a new random value upon factory reset as described in Chapter 2.5.

Note that if USER2 issues a Factory Reset request, then this will affect only configuration parameters that USER2 can configure. If for instance USER1 has changed the radio protocol from BLE to ZGP and removed the authorization for USER2 to change the radio protocol, then a factory reset issued by USER2 will not reset the radio protocol from ZGP to BLE.

5.2 NFC memory map

The NFC memory is organized in pages (smallest addressable unit) where each page contains 4 byte of data. Several pages with similar functionality form an NFC memory area. PTM 535BZ uses the following areas:

- Device Identification NDEF string (Read access only)
  This area contains an NDEF string identifying key device parameters
- User Information NDEF string (Read / write access; no PIN required)
  This area allows any user to read or write information about the device such as the intended installation location or additional instructions
- NFC HEADER (Read access only)
  This area contains information about the NFC revision
- ACTIVE CONFIGURATION (Read access only)
  This area contains the currently used configuration
- NEW CONFIGURATION (Write access, PIN required to execute the update)
  This area is used to change configuration values.

The organization of the PTM 535BZ NFC memory map is shown in Table 10 below.

NFC Page Address	Memory Area	Content
0x00 … 0x17	PRODUCT NDEF	Device identification NDEF string (read-only)
0x18 … 0x27	USER NDEF	User information NDEF string (read / write)
0x28 … 0x2B	NFC HEADER	NFC memory revision (read-only)
0x2C … 0x4F	ACTIVE CONFIGURATION	Currently used device configuration (read only)
0x50 … 0x7B	NEW CONFIGURATION	New device configuration (write only, PIN protected)

Table 10 – PTM 535BZ NFC memory areas

5.3 PRODUCT NDEF

The PRODUCT NDEF area contains a device identification string using the NDEF (NFC Data Exchange Format) standard that is readable by most NFC-capable reader devices (including smartphones).

An example device identification string from the NDEF area of PTM 535BZ could be:

```
30SE21510000123+30PS3231-A535+2PAB04+12Z01234567891234+3C29+01000000
```

This NDEF string encodes the parameters shown in Table 11 below.

Identifier	Length of data (excl. identifier)	Value
30S	12 characters	BLE Source Address (6 byte, variable)
30P	10 characters	Ordering Code (“S3231-A535”)
2P	4 characters	Step Code and Revision (“AB04”)
3C	2 characters	Header Start Address (“29” = 0x29)
16S	8 characters	SW Version Example: “01000000” = 01.00.00.00

Table 11 – NDEF Parameters

5.4 USER NDEF

The USER NDEF area allows the user to store a string of up to 64 characters starting at page 0x18 and ending at page 0x27.

This area can for instance be used by the system integrator to provide information on the intended installation location or by the installer to leave information about the installation.

PTM 535BZ will neither modify nor interpret the content of this area.

5.5 NFC HEADER

The NFC HEADER area contains information about the NFC memory structure and can therefore be used to distinguish between different NFC memory layouts.

5.5.1 NFC HEADER structure

The structure of the NFC HEADER area is shown in Table 12 below.

NFC Page	Byte 0	Byte 1	Byte 2	Byte 3
0x29	START (0xE0)	LENGTH (0x0A)	VERSION (0x01)	OEM MSB (0x00)
0x30	OEM LSB (0x0B)	DEVICE_IDENTIFIER (0xCB:00:04)		
0x31	REVISION (0x01)	END (0xFE)	UNUSED (0x0000)	

Table 12 – NFC HEADER structure

The NFC HEADER contains the following fields:

- START
  This field identifies the start of the NFC header and is always set to `0xE0`
- LENGTH
  This field identifies the length of the NFC header.
  For PTM 535BZ, this field is set to `0x0A` since the header structure is 10 bytes long
- VERSION
  This field identifies the major revision and is set to 0x01 currently
- OEM
  The 16 bit OEM field identifies the manufacturer of the device so that manufacturer-specific layout implementations can be determined. For EnOcean GmbH this field is set to `0x000B`.
- DEVICE_IDENTIFIER
  The 24 bit DEVICE_IDENTIFIER field identifies an individual device from the range of devices manufactured by the manufacturer specified in the OEM field.
  For PTM 535BZ, the DEVICE_IDENTIFIER is set to `0xCB:00:04`
- REVISION
  The REVISION field identifies the exact revision of the NFC layout. This REVISION will be incremented whenever a change to the NFC layout is made.
- END
  The END field identifies the end of the NFC header and is always set to `0xFE`. The number of bytes from START to END must equal LENGTH, otherwise the NFC header is invalid.

5.6 ACTIVE CONFIGURATION

The ACTIVE CONFIGURATION area contains the currently active configuration settings and PTM 535BZ operates according to the register values in this area. Updating the register values is done via the Configuration Update process as described in Chapter 5.6.16.

The structure of ACTIVE CONFIGURATION area is shown in Table **13** below.

NFC Page	Content			
	Byte 0	Byte 1	Byte 2	Byte 3
0x2C	INPUT_CONFIG	RADIO_CONFIG		
0x2D				
0x2E				
0x2F				
0x30	BLE_TX_CONFIG	BLE_SEC_CONFIG	BLE_MANUFACTURER_ID	
0x31	BLE_SOURCE_ADDRESS			
0x32	CH_REG1	CH_REG2	CH_REG3	
0x33				
0x34	BLE_INPUT_STATUS_0	BLE_INPUT_STATUS_1	BLE_INPUT_STATUS_2	BLE_INPUT_STATUS_3
0x35	BLE_INPUT_STATUS_4	BLE_INPUT_STATUS_5	BLE_INPUT_STATUS_6	BLE_INPUT_STATUS_7
0x36				
0x37				
0x38	ZGP_TX_CONFIG	ZGP_SEC_CONFIG	ZGP_PROTOCOL_CONFIG	
0x39	ZGP_SOURCE_ID			
0x3A				
0x3B				
0x3C	ZGP_COMMAND_0	ZGP_COMMAND_1	ZGP_COMMAND_2	ZGP_COMMAND_3
0x3D	ZGP_COMMAND_4	ZGP_COMMAND_5	ZGP_COMMAND_6	ZGP_COMMAND_7
0x3E				
0x3F				
0x40 … 0x43	SECURITY_KEY1			
0x44				
0x45	USER1_CONFIGURATION_OPTIONS			
0x46	USER2_CONFIGURATION_OPTIONS			
0x47				
0x48				
0x49	USER1_PIN_HASH			
0x4A	USER2_PIN_HASH			
0x4B				
0x4C	SEQUENCE_COUNTER			
0x4D	REQUEST_STATUS	DEVICE_STATUS		

Table 13 – ACTIVE CONFIGURATION structure

5.6.1 INPUT_CONFIG

The ECO_DIRECTION field of the INPUT_CONFIG register is used to define which direction of the ECO 200 harvester is considered as a press event and which as a release event.

The default configuration is that a movement of the ECO 200 harvester spring away from the PTM 535BZ PCB is considered as a press event while a movement of the ECO 200 harvester spring towards the PTM 535BZ PCB is considered as a release event as described in chapter 2.4.1.

If ECO_DIRECTION status bit is set, then this logic is inverted meaning that a movement of the ECO 200 harvester spring away from the PTM 535BZ PCB is considered as release event while a movement of the ECO 200 harvester spring towards the PTM 535BZ PCB is considered as press event.

Additionally, the input signals INPUT1 and INPUT2 described in chapter 2.4.2 can be disabled using the corresponding status bits INPUT1 and INPUT2. If an input is disabled, then it will always be treated as if it is not connected.

Figure 36 below shows the structure of the INPUT_CONFIG register.

INPUT_CONFIG (Default Value 0x00)							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU	RFU	RFU	RFU	RFU	INPUT2	INPUT1	ECO_DIRECTION

Figure 36 – INPUT_CONFIG register

The bit fields within the INPUT_CONFIG register are shown in Table 14 below. The default settings are shown in bold.

Bit	Configuration Option	Supported Settings
0	ECO_DIRECTION Selects the ECO 200 direction considered as “Press”	**0b0: Standard (Press = Away from PCB)** 0b1: Inverted (Press = Towards PCB)
1	INPUT1 Enables / disables INPUT1	**0b0: Enabled** 0b1: Disabled (INPUT1 is considered disconnected)
2	INPUT1 Enables / disables INPUT2	**0b0: Enabled** 0b1: Disabled (INPUT2 is considered disconnected)
3	RFU	0b0 (Always set to 0b0)
4	RFU	0b0 (Always set to 0b0)
5	RFU	0b0 (Always set to 0b0)
6	RFU	0b0 (Always set to 0b0)
7	RFU	0b0 (Always set to 0b0)

Table 14 – INPUT_CONFIG settings

5.6.2 RADIO_CONFIG

The PROTOCOL field of the RADIO_CONFIG register is used to select the protocol. By default, PTM 535BZ will use the BLE radio standard for the transmission of telegrams. ZGP can be selected instead of BLE by setting the PROTOCOL bit is to `0b1`.

The TX_POWER field of the RADIO_CONFIG register is used to select the radio transmission power. By default, PTM 535BZ will use a transmission power of +4 dBm. The transmission power can be reduced to 0 dBm by setting the TX_POWER bit to `0b1`.

Figure 37 below shows the structure of the RADIO_CONFIG register.

RADIO_CONFIG (Default Value 0x00)							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU	RFU	RFU	TX_POWER	RFU	RFU	RFU	PROTOCOL

Figure 37 – RADIO_CONFIG register

The bit fields within the RADIO_CONFIG register are shown in Table 15 below. The default settings are shown in bold.

Bit	Configuration Option	Supported Settings
0	PROTOCOL Selects the radio protocol	**0b0: BLE** 0b1: ZGP
1	RFU	0b0 (Always set to 0b0)
2	RFU	0b0 (Always set to 0b0)
3	RFU	0b0 (Always set to 0b0)
4	TX_POWER Selects the transmission power	**0b0: +4 dBm** 0b1: 0 dBm
5	RFU	0b0 (Always set to 0b0)
6	RFU	0b0 (Always set to 0b0)
7	RFU	0b0 (Always set to 0b0)

Table 15 – RADIO_CONFIG settings

5.6.3 BLE_TX_CONFIG

The BLE_TX_CONFIG register is used to configure radio settings that are specific to BLE telegram transmission.

The CHANNEL_SELECTION field is used to select the BLE radio channels (as described in Chapter 3.1.2) for the transmission of BLE telegrams. By default, PTM 535BZ will use the three advertising channels (CH37, CH38 and CH39) for the transmission of both data and commissioning telegrams. The use of other radio channels (specified in the BLE channel registers CH1, CH2 and CH3) can be configured using this field.

The BLE_DATA_RATE field is used to select the data rate used for the transmission of BLE radio telegrams. The default setting is that a data rate of 1 Mbit/s is used; this can be increased to 2 Mbit/s by setting the BLE_DATA_RATE field to `0b1`.

The BLE_ADV_INTERVAL field is used to select the advertising interval between two advertising events as described in Chapter 3.2. The default setting is that an advertising interval of 20 ms is used; this can be reduced to 10 ms by setting the BLE_ADV_INTERVAL field to `0b1`.

The BLE_ADDRESS_MODE field is used to select the address mode as described in Chapter 3.3.5. The default setting is that a Static Source Address is used; a Resolvable Private Address (RPA) will be used instead if the BLE_ADDRESS_MODE field to `0b1`.

The BLE_INPUT_STATUS is used to select which encoding should be used to report the Input Status as discussed in Chapter 3.4.1. By default, PTM 535BZ will use the standard encoding for EnOcean BLE switches. User-defined commands will be used instead if the BLE_INPUT_STATUS field is set to `0b1`.

Figure 38 below shows the structure of the BLE_TX_CONFIG register.

BLE_TX_CONFIG (Default Value 0x00)							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU	BLE_INPUT_STATUS	BLE_ADDRESS_MODE	BLE_ADV_INTERVAL	BLE_DATA_RATE	BLE_CHANNEL_SELECTION		

Figure 38 – BLE_TX_CONFIG register

The bit fields within the RADIO_CONFIG register are shown in Table 16 below. The default settings are shown in bold.

Bit	Configuration Option	Supported Settings
2:0	BLE_CHANNEL_SELECTION Selects the radio channels used for telegram transmission	**0b000: Commissioning Telegrams on Advertising Channels** **Data Telegrams on Advertising Channels** 0b001: Commissioning Telegrams on Advertising Channels Data Telegrams on 3 custom channels (CH1, CH2, CH3) 0b010: Commissioning Telegrams on Advertising Channels Data Telegrams on 2 custom channels (CH1, CH2) 0b011: Commissioning Telegrams on Advertising Channels Data Telegrams on 1 custom channel (CH1) 0b100: Commissioning Telegrams 3 custom channels (CH1, CH2, CH3) Data Telegrams on 3 custom channels (CH1, CH2, CH3) 0b101: Commissioning Telegrams on 2 custom channels (CH1, CH2) Data Telegrams on 2 custom channels (CH1, CH2) 0b110: Commissioning Telegrams on 1 custom channel (CH1) Data Telegrams on 1 custom channel (CH1) 0b111: RFU
3	BLE_DATA_RATE Selects the data rate	**0b0: 1 Mbit/s** 0b1: 2 Mbit/s
4	BLE_ADV_INTERVAL Selects the advertising interval	**0b0: 20 ms** 0b1: 10 ms
5	BLE_ADDRESS_MODE Selects the address mode	**0b0: Static Source Address** 0b1: Resolvable Private Address
6	BLE_INPUT_STATUS Selects the input status encoding	**0b0: INPUT_STATUS uses default encoding** 0b1: INPUT_STATUS uses customer-defined encoding
7	RFU	0b0 (Always set to 0b0)

Table 16 – BLE_TX_CONFIG register settings

5.6.4 BLE_SEC_CONFIG

The BLE_SEC_CONFIG register is used to configure security settings that are specific to BLE telegram transmission.

The BLE_SECURITY_MODE field is intended for future implementation selecting the security mode used by PTM 535BZ when transmitting BLE telegrams. Currently, the supported security mode is using a 32 bit sequence counter to generate a 32 bit CMAC (signature) as described in Chapter 3.4.2.

The BLE_KEY_SELECTION field is used to select the security key used for the authentication of PTM 535BZ data telegrams as described in Chapter 3.4.2 and for the generation of Resolvable Private Addresses as described in Chapter 3.3.5.2. By default, SECURITY_KEY1 is used; SECURITY_KEY2 can be selected by setting the BLE_KEY_SELECTION field to to `0b1`. Note that SECURITY_KEY2 cannot be read via the NFC interface.

The BLE_LRN_BUTTON field is used to enable and disable transmission of a BLE commissioning telegram if the LRN button is pressed and SECURITY_KEY1 is selected. By default, a BLE commissioning telegram will be transmitted if the LRN button is pressed and the ECO 200 harvester is actuated. Transmission of a commissioning telegram can be disabled by setting the BLE_LRN_BUTTON field to `0b1`.

Figure 39 below shows the BLE_SEC_CONFIG register.

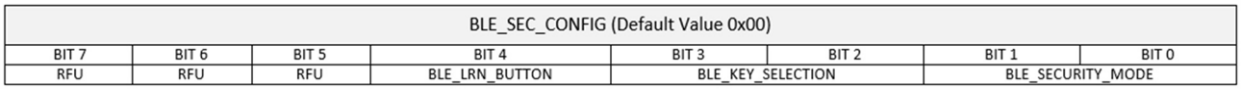

Figure 39 – BLE_SEC_CONFIG register

The bit fields within the RADIO_CONFIG register are shown in Table 17 below. The default settings are shown in bold.

Bit	Configuration Option	Supported Settings
1:0	BLE_SECURITY_MODE Selection of the security Mode used for BLE telegram transmission	**0b00: 32 bit Sequence Counter with 32 bit Signature** 0b01, 0b10, 0b11: RFU
3:2	BLE_KEY_SELECTION Selection of the security key used for authentication and RPA generation	**0b00: Use SECURITY_KEY1 for authentication and RPA generation** 0b01: Use SECURITY_KEY2 for authentication and RPA generation 0b10, 0b11: RFU
4	BLE_LRN_BUTTON Use of LRN button for commissioning telegram transmission	0b0: LRN Button press triggers commissioning telegram transmission 0b1: LRN Button press is ignored
7:5	RFU	0b0 (Always set to 0b0)

Table 17 – BLE_SEC_CONFIG settings

5.6.5 BLE_MANUFACTURER_ID

The register MANUFACTURER_ID identifies the manufacturer of the device using the 16 bit company identifier assigned by Bluetooth SIG [6]. The default setting of `0x03DA` identifies EnOcean GmbH as the manufacturer of the device.

Figure 40 below shows the structure of the BLE_MANUFACTURER_ID register.

BLE_MANUFACTURER_ID (Default Value 0x03DA)					
BIT 15	...	BIT 8	BIT 7	...	BIT 0
BLE MANUFACTURER ID LSB (0xDA)			BLE MANUFACTURER ID MSB (0x03)		

Figure 40 – BLE_MANUFACTURER_ID register

5.6.6 BLE_SOURCE_ADDRESS

Each PTM 535BZ module uses a unique 6 byte address (Static Source Address) to identify itself as the originator of BLE radio telegrams as described in chapter 3.3.5.1.

The two most significant byte of this address are always `0xE215`; i.e. the address always starts with `0xE215`. The four least significant byte of this address are assigned during manufacturing and are listed in the BLE_SOURCE_ADDRESS register. The resulting 6 byte Static Source Address used by PTM 535BZ for the transmission of BLE telegrams can then be calculated as (0xE215 << 32) + BLE_SOURCE_ADDRESS.

The factory-assigned address will always have the format 0x1nnn:nnnn which allows easy distinction between PTM 535BZ Static Source Addresses using the format 0xE215:1nnn:nnnn and PTM 215B Static Source Addresses using the format 0xE215:0nnn:nnnn.

The structure of the BLE_SOURCE_ADDRESS register is shown in Figure 41 below.

BLE_SOURCE_ADDRESS			
BYTE0	BYTE1	BYTE2	BYTE3
0x1n	Variable	Variable	Variable

Figure 41 – BLE_SOURCE_ADDRESS

5.6.7 BLE Radio Channel Registers CH1, CH2, CH3

The BLE channel selection registers CH1, CH2 and CH3 define the radio channels used for custom radio transmission sequences as described in Chapter 3.1.2 if use of custom radio transmission sequences is enabled as described in Chapter 5.6.3.

If custom radio transmission sequences are enabled, then the radio channels specified in CH1, CH2 and CH3 will be used. Figure 42 shows the structure of these registers.

CH1, CH2 and CH3 (Default Value 0x25, 0x26, 0x27)			
BIT 7	BIT 6	...	BIT 0
RFU	BLE_RADIO_CHANNEL		

Figure 42 – BLE Radio Channel Registers CH1, CH2 and CH3

5.6.8 BLE_INPUT_STATUS_x

PTM 535BZ can transmit user defined-values in the INPUT_STATUS field of BLE data telegrams instead of the standard values as described in Chapter 3.4.1.2. This feature can be enabled by setting the BLE_INPUT_STATUS field of the BLE_TX_CONFIG register to `0b1`.

If this feature is enabled, then PTM 535BZ will select the value of the INPUT_STATUS field within the BLE data telegram from one of the eight registers BLE_INPUT_STATUS_0 … BLE_INPUT_STATUS_7 depending on the input status. The Index field provided in Table 3 of Chapter 3.4.1.2 for the applicable input status is used to select the register.

For instance, if INPUT1 is not connected, INPUT2 is not connected and the ECO 200 harvester is pressed then Index = 0. If user defined-values for the INPUT_STATUS field of BLE data telegrams are enabled, then the value of BLE_INPUT_STATUS_0 register would be transmitted in the INPUT_STATUS field of the BLE data telegram.

Setting the value of a BLE_INPUT_STATUS_x register to `0xFF` will cause PTM 535BZ to not transmit a data telegram. This can for instance be useful is PTM 535BZ should only transmit a data telegram upon button press.

5.6.9 ZGP_TX_CONFIG

The ZGP_TX_CONFIG register determines the radio channel used by PTM 535BZ for the transmission of ZGP telegrams.

The ZGP_TX_CHANNEL field defines the currently used radio channel (channel 11 … 26) as described in Table 4 of Chapter 4.1. By default, PTM 535BZ uses channel 11 for the transmission of ZGP data telegrams.

The ZGP_TX_CHANNEL_SELECTION field defines if the radio channel can be changed using the LRN button as described in Chapter 4.5.2. Channel selection with the LRN button might be enabled among all radio channels (this is the default setting), only among the primary radio channels or disabled (the radio channel cannot be changed with the LRN button).

Figure 43 below shows the structure of the ZGP_TX_CONFIG register.

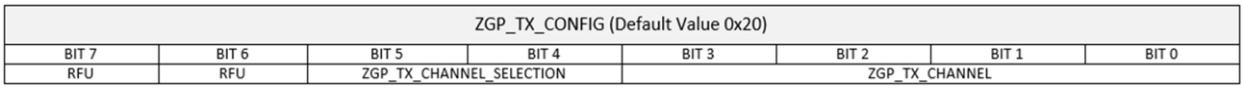

Figure 43 – ZGP_TX_CONFIG register structure

The bit fields within the ZGP_TX_CONFIG register are shown in Table 18 below. The default settings are shown in bold.

Bit	Configuration Option	Supported Settings
3:0	ZGP_TX_CHANNEL Defines the radio channel used for ZGP telegram transmissions	**0b0000: IEEE 802.15.4 Radio Channel 11** 0b0001: IEEE 802.15.4 Radio Channel 12 0b0010: IEEE 802.15.4 Radio Channel 13 … 0b1110: IEEE 802.15.4 Radio Channel 25 0b1111: IEEE 802.15.4 Radio Channel 26
5:4	ZGP_TX_CHANNEL_SELECTION Defines how the radio channel can be selected via the LRN button	**0b00: No radio channel selection with LRN button (Always use ZGP_TX_CHANNEL)** 0b01: Selection with LRN button amongst the primary radio channels (Selection starts with ZGP_TX_CHANNEL) 0b10: Selection with LRN button amongst all radio channels (Selection starts with ZGP_TX_CHANNEL) 0b11: RFU
7:5	RFU	0b000: RFU (Always set to 0b0)

Table 18 – ZGP_TX_CONFIG settings

5.6.10 ZGP_SEC_CONFIG

The ZGP_SEC_CONFIG register determines the security mode used by PTM 535BZ for the transmission of ZGP telegrams.

The ZGP_SEC_MODE field defines how the security key is encrypted in the commissioning telegram. By default, PTM 535BZ uses the ZA09 key to encrypt the security key. Optionally, PTM 535BZ can use an Install Code (IC) to encrypt the security key. Refer to the Zigbee Green Power specification [3] for a description of this feature.

The ZGP_KEY_SELECTION field determines which of the two security keys (SECURITY_KEY1 or SECURITY_KEY2) is used to authenticate ZGP data telegrams. SECURITY_KEY1 is used by default.

The ZGP_BUTTON_COMMISSIONG determines if the LRN button can trigger the transmission of a commissioning telegram. By default, this is possible. If transmission of commissioning telegrams is disabled, then the ZGP radio channel has to be selected using ZGP_TX_CHANNEL.

Figure 44 below shows the structure of the ZGP_SEC_CONFIG register.

ZGP_SEC_CONFIG (Default Value 0x20)							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU	RFU	RFU	ZGP_BUTTON_COMMISSIONING	ZGP_KEY_SELECTION		ZGP_COMMISSIONING_FORMAT	

Figure 44 – ZGP_SEC_CONFIG register structure

The bit fields within the ZGP_SEC_CONFIG register are shown in Table 19 below. The default settings are shown in bold.

Bit	Configuration Option	Supported Settings
1:0	ZGP_COMMISSIONING_MODE Defines the TLK used for transmission of the ZGP commissioning telegram	**0b00: Commissioning telegram uses ZA09 as TLK** 0b01: Commissioning telegram uses ZA09 as TLK, IC is supported 0b10: Commissioning telegram uses IC 0b11: RFU
3:1	ZGP_KEY_SELECTION Selects the security key used for authentication of ZGP telegrams	**0b00: Use SECURITY_KEY1** 0b01: Use SECURITY_KEY2 0b10, 0b11: RFU
4	ZGP_BUTTON_COMMISSIONING Selects if commissioning telegram can be triggered by the LRN button	**0b0: LRN button press with ECO 200 action will trigger transmission of a commissioning telegram** 0b1: LRN button press with ECO 200 action will not trigger transmission of a commissioning telegram
7:5	RFU	0b000: RFU (Always set to 0b0)

Table 19 – ZGP_TX_CONFIG settings

5.6.11 ZGP_PROTOCOL_CONFIG

The ZGP_PROTOCOL_CONFIG register defines the ZGP-specific parameters of PTM 535BZ.

The ZGP_DEVICE_ID field selects the DEVICE ID used by PTM 535BZ when transmitting ZGP telegrams. By default, Device ID 0x07 (Generic 8-contact switch) is used as described in Chapter 4.4.1.2.

The ZGP_COMMANDLIST_DISABLE field allows disabling the command list that is transmitted within the Application Info field of commissioning telegrams for Device ID other than 0x07 as described in Chapter 4.4.2.3. Disabling the transmission of the command list (and thereby disabling transmission of the Application Info field) can sometimes be required to ensure compatibility with legacy devices. This field has no effect when Device ID 0x07 is used.

The ZGP_CONTACT_NUMBER field determines the number of contacts reported within the Application Info field of commissioning telegrams when Device ID 0x07 is used. This field has no effect when a Device ID other than 0x07 is used.

Figure 45 below shows the structure of the ZGP_PROTOCOL_CONFIG register.

ZGP_PROTOCOL_CONFIG (Default Value 0x07)							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RFU	RFU	APPINFO_NUMBEROFBUTTONS		APPINFO_COMMANDLIST	ZGP_DEVICE_ID		

Figure 45 – ZGP_PROTOCOL_CONFIG register structure

The bit fields within the ZGP_PROTOCOL_CONFIG register are shown in Table 20 below. The default settings are shown in bold.

Bit	Configuration Option	Supported Settings
2:0	ZGP_DEVICE_ID Selects the ZGP Device ID	0b000: Device ID 0x00 (GP Simple Generic 1-state Switch) 0b001: Device ID 0x01 (GP Simple Generic 2-state Switch) 0b010: Device ID 0x02 (GP On/Off Switch) 0b011: Device ID 0x03 (GP Level Control Switch) 0b100: Device ID 0x10 (GP Color Dimmer Switch) 0b101: Device ID 0x05 (GP Advanced Generic 1-state Switch) 0b110: Device ID 0x06 (GP Advanced Generic 2-state Switch) **0b111: DeviceID 0x07 (GP Generic 8-contact switch)**
3	APPINFO_COMMANDLIST Selects if application information with the command list is provided for Device ID other than 0x07	**0b0: Enabled** 0b1: Disabled (only valid for backwards compatibility)
5:4	APPINFO_NUMBEROFBUTTONS Selects the number of buttons specified within the application information provided for Device ID 0x07	**0b00: 1 button** 0b01: 2 buttons 0b10: 3 buttons 0b11: 4 buttons
7:6	RFU	0b000: RFU (Always set to 0b0)

Table 20 – ZGP_PROTOCOL_CONFIG settings

5.6.12 ZGP_SOURCE_ID

Each PTM 535BZ module uses a unique four-byte address (ZGP Source ID) to identify itself as the originator of ZGP radio telegrams as described in Chapter 4.4.

The ZGP Source ID is assigned by Zigbee Alliance and cannot be changed by the user. For PTM 535BZ devices, the ZGP Source ID has the format `0x015n:nnnn`. The structure of the ZGP_SOURCE_ID register is shown in Figure 46 below.

ZGP_SOURCE_ID (Variable)			
BYTE0	BYTE1	BYTE2	BYTE3
0x01	0x5n	nn	nn

Figure 46 – ZGP_SOURCE_ID

5.6.13 ZGP_COMMAND_0 … ZGP_COMMAND_7

As described in Chapter 4.4.1.3, PTM 535BZ will transmit user-defined ZGP commands if a Device ID other than 0x07 is configured. In this case, PTM 535BZ will select the appropriate command to be sent from the command table shown in Table 8 according to the status of the INPUT1 and INPUT2 signals and the ECO 200 action.

Setting the value of a ZGP_COMMAND_x (x = 0…7) register to `0xFF` will cause PTM 535BZ no to transmit a data telegram. This can for instance be useful is PTM 535BZ should only transmit a data telegram upon button press but not on release.

5.6.14 SECURITY_KEY1, SECURITY_KEY2 and SECURITY_KEY3

As described in Chapter 2.5, PTM 535BZ by default uses SECURITY_KEY1 for the authentication of transmitted data telegrams and for the generation of Resolvable Private Addresses. SECURITY_KEY1 is NFC-readable and can be read via the SECURITY_KEY1 register.

It is possible to select using SECURITY_KEY2 instead of SECURITY_KEY1 via the ZGP_SEC_CONFIG register (if PTM 535BZ transmits ZGP telegrams) or via the BLE_SEC_CONFIG register (if PTM 535BZ transmits BLE telegrams). SECURITY_KEY2 is not NFC readable; it can only be written by the user.

SECURITY_KEY3 is an optional key for the encryption of the security key within the ZGP commissioning telegram. SECURITY_KEY3 is not NFC readable; it can only be written by the user.

5.6.15 USER*n*_CONFIGURATION_OPTIONS

As described in Chapter 5.1.1, the available configuration options for USER1 are defined in the register USER1_CONFIGURATION_OPTIONS while the available configuration options for USER2 are defined in the register USER2_CONFIGURATION_OPTIONS.

The USER1_CONFIGURATION_OPTIONS and the USER2_CONFIGURATION_OPTIONS registers have the same structure and are organized as four groups (BYTE0 = SECURITY OPTIONS, BYTE1 = ZGP OPTIONS, BYTE2 = BLE OPTIONS and BYTE3 = SYSTEM OPTIONS). This is shown in Figure 47 below.

USER1_CONFIGURATION_OPTIONS / USER2_CONFIGURATION_OPTIONS			
BYTE0	BYTE1	BYTE2	BYTE3
SECURITY CONFIGURATION	ZGP CONFIGURATION	BLE CONFIGURATION	SYSTEM CONFIGURATION

Figure 47 – USERn_CONFIGURATION_OPTIONS

Available configuration options for USER1 are marked by the corresponding bit in the USER1_CONFIGURATION_OPTIONS register set to `0b1`. Likewise, available configuration options for USER2 are marked by the corresponding bit in the USER2_CONFIGURATION_OPTIONS register set to `0b1`.

USER1 can restrict the available configuration options for USER2 by setting the corresponding bits in the USER2_CONFIGURATION_OPTIONS register to `0b0`.

5.6.15.1 SECURITY OPTIONS option group

Table 21 below shows the configuration options belonging to the SECURITY OPTIONS group. Two fields in this configuration group are reserved for future use and can therefore not be used by either USER1 or USER2.

Note that the configuration options available for USER2 can be changed only by USER1 and not by USER2. Note also that the PIN for USER2 can be changed both by USER1 and USER2.

Bit Position	Configuration Option	USER1	USER2
0 (0x01)	SECURITY_KEY1	0b1 (Allowed)	0b1 (Allowed)
1 (0x02)	SECURITY _KEY2	0b1 (Allowed)	0b1 (Allowed)
2 (0x04)	SECURITY _KEY3	0b1 (Allowed)	0b1 (Allowed)
3 (0x08)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
4 (0x10)	USER2_CONFIGURATION_OPTIONS	0b1 (Allowed)	0b0 (Not Allowed)
5 (0x20)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
6 (0x40)	USER1_PIN	0b1 (Allowed)	0b0 (Not Allowed)
7 (0x80)	USER2_PIN	0b1 (Allowed)	0b1 (Allowed)

Table 21 - SECURITY OPTIONS group

5.6.15.2 ZGP OPTIONS group

Table 22 below shows the configuration options belonging to the ZGP OPTIONS configuration group.

Four fields in this configuration group are reserved for future use and can therefore not be used by either USER1 or USER2. All other fields in this group can by default be changed both by USER1 and by USER2.

Bit Position	Configuration Option	USER1	USER2
0 (0x01)	ZGP_TX_CONFIG	0b1 (Allowed)	0b1 (Allowed)
1 (0x02)	ZGP_SEC_CONFIG	0b1 (Allowed)	0b1 (Allowed)
2 (0x04)	ZGP_BUTTON_MAP	0b1 (Allowed)	0b1 (Allowed)
3 (0x08)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
4 (0x10)	ZGP_PROTOCOL_CONFIG	0b1 (Allowed)	0b1 (Allowed)
5 (0x20)	ZGP_DECOMMISSIONING_REQUEST	0b1 (Allowed)	0b1 (Allowed)
6 (0x40)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
7 (0x80)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)

Table 22 - ZGP OPTIONS group

5.6.15.3 BLE OPTIONS group

Table 23 below shows the configuration options belonging to the BLE OPTIONS configuration group.

Two fields in this configuration group are reserved for future use and can therefore not be used by either USER1 or USER2. All other fields in this group can by default be changed both by USER1 and by USER2.

Bit Position	Configuration Option	USER1	USER2
0 (0x01)	BLE_TX_CONFIG	0b1 (Allowed)	0b1 (Allowed)
1 (0x02)	BLE_SEC_CONFIG	0b1 (Allowed)	0b1 (Allowed)
2 (0x04)	BLE_BUTTON_MAP	0b1 (Allowed)	0b1 (Allowed)
3 (0x08)	BLE_SOURCE_ADDRESS	0b1 (Allowed)	0b1 (Allowed)
4 (0x10)	BLE_MANUFACTURER_ID	0b1 (Allowed)	0b1 (Allowed)
5 (0x20)	BLE_CH_REGx	0b1 (Allowed)	0b1 (Allowed)
6 (0x40)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
7 (0x80)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)

Table 23 - BLE OPTIONS group

5.6.15.4 SYSTEM OPTIONS group

Table 24 below shows the configuration options belonging to the SYSTEM OPTIONS configuration group.

Four fields in this configuration group are reserved for future use and can therefore not be used by either USER1 or USER2. All other fields in this group can by default be changed both by USER1 and by USER2.

Bit Position	Configuration Option	USER1	USER2
0 (0x01)	COMMISSIONING_REQUEST	0b1 (Allowed)	0b1 (Allowed)
1 (0x02)	FACTORY_RESET_REQUEST	0b1 (Allowed)	0b1 (Allowed)
2 (0x04)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
3 (0x08)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
4 (0x10)	BUTTON_CONFIG	0b1 (Allowed)	0b1 (Allowed)
5 (0x20)	RADIO_CONFIG	0b1 (Allowed)	0b1 (Allowed)
6 (0x40)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)
7 (0x80)	RFU	0b0 (Not Allowed)	0b0 (Not Allowed)

Table 24 - SYSTEM OPTIONS group

5.6.16 SEQUENCE_COUNTER

PTM 535BZ maintains a 4 byte BLE Sequence Counter for the authentication of BLE data telegrams as described in Chapter 3.4.1 and a 4 byte ZGP Sequence Counter for the authentication of ZGP data telegrams as described in Chapter 4.4.1.

The four-byte SEQUENCE_COUNTER register contains the current value of the currently active sequence counter (BLE Sequence Counter if PTM 535BZ transmits BLE telegrams or ZGP Sequence Counter if PTM 535BZ transmits ZGP telegrams).

5.6.17 REQUEST_STATUS

The REQUEST_STATUS field informs the user about the status of the most recent NFC configuration update or NFC functional request. Table 25 shows the encoding of the supported status reports.

Request (REQUEST_TYPE _SELECTION)		Response (REQUEST_STATUS)				
Request Type		SUCCESS	IN PROGRESS	PIN ERROR	PERMISSION ERROR	PARAMETER ERROR
USER1	Configuration Update Request	0x31	0x51	0x71	0x91	0xB1
	Commissioning Request	0x32	0x52	0x72	0x92	0xB2
	Factory Reset Request	0x33	0x53	0x73	0x93	0xB3
	ZGP Decommissioning Request	0x34	0x54	0x74	0x94	0xB4
USER2	Configuration Update Request	0x39	0x59	0x79	0x99	0xB9
	Commissioning Request	0x3A	0x5A	0x7A	0x9A	0xBA
	Factory Reset Request	0x3B	0x5B	0x7B	0x9B	0xBB
	ZGP Decommissioning Request	0x3C	0x5C	0x7C	0x9C	0xBC

Table 25 – CONFIGURATION STATUS encoding

The following status can be reported:

- SUCCESS
  The configuration update or the function request were successfully executed
- IN PROGRESS
  The configuration update or the functional request is in progress (until the ECO 200 harvester has been actuated sufficiently often so that the requested action can be completed)
- PIN ERROR
  The PIN code provided to authenticate the configuration update or the functional request does not match the expected PIN code
- PERMISSION ERROR
  The specified request is not allowed (for instance if USER2 tries to change the PIN code of USER1)
- PARAMETER ERROR
  The request contains invalid parameters (for instance an undefined configuration register value)

5.6.18 DEVICE_STATUS

The DEVICE_STATUS register is an internal register that tracks the PTM 535BZ status during commissioning actions.

5.7 NEW CONFIGURATION

The NEW CONFIGURATION area is used to update the configuration parameters used by PTM 535BZ which are stored in the ACTIVE CONFIGURATION area. The structure of NEW CONFIGURATION area is shown in Table 26 below.

NFC Page	Content			
	Byte 0	Byte 1	Byte 2	Byte 3
0x50	REQUEST_TYPE			
0x51	CONFIGURATION_SELECTION			
0x52	REQUEST_AUTHENTICATION			
0x53				
0x54	INPUT_CONFIG	RADIO_CONFIG		
…				
0x58	BLE_TX_CONFIG	BLE_SEC_CONFIG	BLE_MANUFACTURER_ID	
0x59	BLE_SOURCE_ADDRESS			
0x5A	CH_REG1	CH_REG2	CH_REG3	
0x5B				
0x5C	BLE_INPUT_STATUS_0	BLE_INPUT_STATUS_1	BLE_INPUT_STATUS_2	BLE_INPUT_STATUS_3
0x5D	BLE_INPUT_STATUS_4	BLE_INPUT_STATUS_5	BLE_INPUT_STATUS_6	BLE_INPUT_STATUS_7
…				
0x60	ZGP_TX_CONFIG	ZGP_SEC_CONFIG	ZGP_PROTOCOL_CONFIG	
0x61	ZGP_SOURCE_ID			
…				
0x64	ZGP_COMMAND_0	ZGP_COMMAND_1	ZGP_COMMAND_2	ZGP_COMMAND_3
0x65	ZGP_COMMAND_4	ZGP_COMMAND_5	ZGP_COMMAND_6	ZGP_COMMAND_7
..				
0x68 … 0x6B	SECURITY_KEY1 (128 Bit)			
0x6C … 0x6F	SECURITY_KEY2 (128 Bit)			
0x70 … 0x73	SECURITY_KEY3 (128 Bit)			
..				
0x76	USER2_CONFIGURATION_OPTIONS			
…				
0x79	USER1_PIN			
0x4D	USER2_PIN			

Table 26 – NEW CONFIGURATION area structure

5.7.1 NFC configuration process

Updates to the active configuration or the execution of functional request are triggered by user requests using the following procedure:

- Specify request type and requesting user (USER1 or USER2) in the REQUEST_TYPE register
- Provide the required authentication (USER1_PIN or USER2_PIN) in the REQUEST_AUTHENTICATION register
- If the request is a configuration update, then specify the configuration items that should be updated in the CONFIGURATION_SELECTION register
- If the request is a configuration update, then specify the new configuration values for the registers that should be updated in the corresponding shadow registers in the NEW CONFIGURATION NFC area

After setting up all required data, the user has to provide the required energy for the update processing by pressing and releasing the ECO 200 harvester 5 times in each direction.

PTM 535BZ will then read the REQUEST_TYPE register and check if the correct PIN corresponding to the requesting user is provided in the REQUEST_AUTHENTICATION register. If an incorrect PIN is provided, then PTM 535BZ will abort the update process and set the REQUEST_STATUS register to "PIN ERROR".

If the request has been correctly authenticated, then PTM 535BZ will check if the user is permitted to execute the request and – for the case of a configuration update – check if the user is permitted to change the configuration registers specified in the CONFIGURATION_SELECTION register.

If one, several or all registers cannot be changed by the user, then PTM 535BZ will abort the update process and set the REQUEST_STATUS register to "PERMISSION ERROR".

If the request has been correctly authenticated and the registers can be configured by the user, then PTM 535BZ will check if the provided update values for the configuration registers are supported. If an incorrect value is specified, then PTM 535BZ will abort the update process and set the REQUEST_STATUS register to "PARAMETER ERROR".

Any of the three error conditions listed above will cause PTM 535BZ to abort the update process. Users should not rely on PTM 535BZ to detect potential error conditions and ensure that all provided parameters are correct to avoid cases of partial configuration updates leading to unexpected system behaviour. If the request has been correctly authenticated, the register(s) can be updated and the update value(s) are supported, then the update process will start.

For each action of the ECO 200 harvester, a subset of the registers will be updated. No telegrams will be sent while the update is in progress. The REQUEST_STATUS register will be set to "IN PROGRESS" while the update is executed. Once the update has completed, the REQUEST_STATUS register will be set to "SUCCESS" and PTM 535BZ will restart operation based on the new parameters.

5.7.2 REQUEST_TYPE

PTM 535BZ supports three different requests (NFC Parameter Update, Commissioning Telegram Transmission and Factory Reset) which can be issued by two users (USER1 and USER2).

Request origin (USER1 or USER2) and request type are identified by the REQUEST_ID which is written by the user into the REQUEST_TYPE register. Table 27 below shows the assigned REQUEST_ID values.

Request Origin and Request Type		REQUEST_ID
USER1	Configuration Update Request	0x11
	Commissioning Telegram Request	0x12
	Factory Reset Request	0x13
	ZGP Decommissioning Request	0x14
USER2	Configuration Update Request	0x19
	Commissioning Telegram Transmission	0x1A
	Factory Reset Request	0x1B
	ZGP Decommissioning Request	0x1C

Table 27 – REQUEST_ID encoding

The execution of the request can be verified using the REQUEST_STATUS register as described in Chapter 5.6.17.

5.7.3 CONFIGURATION_SELECTION

The CONFIGURATION_SELECTION register is used to specify the configuration registers that should be updated. The structure of the CONFIGURATION_SELECTION register follows the structure of the USERn_CONFIGURATION_OPTIONS register described in Chapter 5.6.15.

The CONFIGURATION_SELECTION register is byte-wise structured into four configuration groups:

- BYTE0 = Security configuration
- BYTE1 = ZGP configuration
- BYTE2 = BLE configuration
- BYTE3 = SYSTEM configuration

The structure of the CONFIGURATION_SELECTION register is shown in Figure 48 below.

CONFIGURATION_SELECTION			
BYTE0	BYTE1	BYTE2	BYTE3
SECURITY CONFIGURATION	ZGP CONFIGURATION	BLE CONFIGURATION	SYSTEM CONFIGURATION

Figure 48 – CONFIGURATION_SELECTION

Before requesting an update of the configuration registers, the user should verify that the registers can be changed by him. This is done by checking that the corresponding bits in the USERn_CONFIGURATION_OPTIONS register are set to `0b1`.

If the user can request an update to the intended configuration registers, then setting the corresponding bits in the CONFIGURATION_SELECTION register will instruct PTM 535BZ to update the registers in the currently active configuration (stored in the ACTIVE_CONFIGURATION area) with the values from the shadow registers in the NEW_CONFIGURATION area.

If the user attempts to update configuration registers that cannot be changed by him, then PTM 535BZ will respond to the request with CONFIG_STATUS = PERMISSION_ERROR.

The user may change the settings for one or several configuration registers at the same time.

5.7.3.1 SECURITY configuration group

Table 28 below shows the configuration options belonging to the SECURITY CONFIGURATION group. Two fields in this configuration group are reserved for future use and can therefore not be used.

Note that the configuration options available for USER2 and the PIN for USER1 can only be changed by USER1 (and not by USER2).

Bit Position	Configuration Option	Supported Settings	
0 (0x01)	SECURITY_KEY1	0b0 (Do not update)	0b1 (Update)
1 (0x02)	SECURITY_KEY2	0b0 (Do not update)	0b1 (Update)
2 (0x04)	SECURITY_KEY3	0b0 (Do not update)	0b1 (Update)
3 (0x08)	RFU	0b0 (Do not update)	
4 (0x10)	USER2_CONFIGURATION_OPTIONS	0b1 (Allowed)	0b1 (Update, only for USER1)
5 (0x20)	RFU	0b0 (Do not update)	
6 (0x40)	USER1_PIN	0b0 (Do not update)	0b1 (Update, only for USER1)
7 (0x80)	USER2_PIN	0b0 (Do not update)	0b1 (Update)

Table 28 - SECURITY configuration group

5.7.3.2 ZGP configuration group

Table 29 below shows the configuration options belonging to the ZGP configuration group. Four fields in this configuration group are reserved for future use and can therefore not be used. Note that the ZGP_DECOMMISSIONING_REQUEST field corresponds to a request and cannot be updated.

Bit Position	Configuration Option	Possible Settings	
0 (0x01)	ZGP_TX_CONFIG	0b0 (Do not update)	0b1 (Update)
1 (0x02)	ZGP_SEC_CONFIG	0b0 (Do not update)	0b1 (Update)
2 (0x04)	ZGP_BUTTON_MAP	0b0 (Do not update)	0b1 (Update)
3 (0x08)	RFU	0b0 (Do not update)	
4 (0x10)	ZGP_PROTOCOL_CONFIG	0b0 (Do not update)	0b1 (Update)
5 (0x20)	ZGP_DECOMMISSIONING_REQUEST	0b0 (This is a request and not a register)	
6 (0x40)	RFU	0b0 (Do not update)	
7 (0x80)	RFU	0b0 (Do not update)	

Table 29 - SECURITY OPTIONS group

5.7.3.3 BLE configuration group

Table 30 below shows the configuration options belonging to the BLE configuration group. Two fields in this configuration group are reserved for future use and can therefore not be used.

Bit Position	Configuration Option	Possible Settings	
0 (0x01)	BLE_TX_CONFIG	0b0 (Do not update)	0b1 (Update)
1 (0x02)	BLE_SEC_CONFIG	0b0 (Do not update)	0b1 (Update)
2 (0x04)	BLE_BUTTON_MAP	0b0 (Do not update)	0b1 (Update)
3 (0x08)	BLE_SOURCE_ADDRESS	0b0 (Do not update)	0b1 (Update)
4 (0x10)	BLE_MANUFACTURER_ID	0b0 (Do not update)	0b1 (Update)
5 (0x20)	BLE_CH_REGx	0b0 (Do not update)	0b1 (Update)
6 (0x40)	RFU	0b0 (Do not update)	
7 (0x80)	RFU	0b0 (Do not update)	

Table 30 - BLE OPTIONS group

5.7.3.4 SYSTEM configuration group

Table 31 below shows the configuration options belonging to the SYSTEM configuration group. Four fields in this configuration group are reserved for future use and can therefore not be used. Note that the LRN_TELEGRAM_REQUEST and FACTORY_RESET_REQUEST fields correspond to requests and can therefore not be updated.

Bit Position	Configuration Option	Possible Settings	
0 (0x01)	LRN_TELEGRAM_REQUEST	0b0 (This is a request and not a register)	
1 (0x02)	FACTORY_RESET_REQUEST	0b0 (This is a request and not a register)	
2 (0x04)	RFU	0b0 (Do not update)	
3 (0x08)	RFU	0b0 (Do not update)	
4 (0x10)	BUTTON_CONFIG	0b0 (Do not update)	0b1 (Update)
5 (0x20)	RADIO_CONFIG	0b0 (Do not update)	0b1 (Update)
6 (0x40)	RFU	0b0 (Do not update)	
7 (0x80)	RFU	0b0 (Do not update)	

Table 31 - SYSTEM configuration group

5.7.4 REQUEST_AUTHENTICATION

NFC configuration updates are authenticated by the user (USER1 or USER2) by writing his authentication PIN code (USER1_PIN or USER2_PIN) into the REQUEST_AUTHENTICATION register.

Figure 49 below shows the structure and the byte order of this register based on the default PIN codes for USER1 and USER2.

REQUEST_AUTHENTICATION				
	BYTE0	BYTE1	BYTE2	BYTE3
USER1	0x02	0x00	0x35	0xE5
USER2	0x03	0x00	0x35	0xE5

Figure 49 – NFC device configuration architecture

5.7.5 USER1_PIN and USER2_PIN

The PIN CODES used to authenticate NFC configuration updates by USER1 or USER2 can – and should - be changed from their default settings.

To do so, follow these steps:

- Request an update of the PIN code by setting Bit 6 (USER1_PIN) or Bit 7 (USER2_PIN) in the SECURITY configuration group of the CONFIGURATION_SELECTION register
- Specify the currently active NFC PIN in the REQUEST_AUTHENTICATION register to authenticate the request
- Specify the new NFC PIN in the USER1_PIN register (if updating the PIN for USER1) or the USER2_PIN register (if updating the PIN for USER2).

After that, click the connected ECO 200 harvester 5 times in each direction to provide the required energy for the update.

Make sure that the new PIN code is properly noted especially when changing USER1_PIN. For security reasons, it is not possible to reset USER1_PIN after it has been changed.

Figure 50 below shows the structure and the byte order of the USER1_PIN and USER2_PIN registers together with the default PIN codes for USER1 and USER2.

USER1_PIN / USER2_PIN				
	BYTE0	BYTE1	BYTE2	BYTE3
USER1_PIN	0x02	0x00	0x35	0xE5
USER2_PIN	0x03	0x00	0x35	0xE5

Figure 50 – USER1_PIN and USER2_PIN register structure with default values

5.7.5.1 USER1_PIN_HASH and USER2_PIN_HASH

PTM 535BZ provides 16-bit hash representations of the 32-bit USER1_PIN (called USER1_PIN_HASH) and the 32-bit USER2_PIN (called USER2_PIN_HASH). These hash values allow NFC tools to verify if they possess the correct NFC pin code.

The length of the hash value has been chosen to minimize the likelihood of a false match while on the other hand making it not feasible to derive the actual NFC pin code from it.

On one hand, the likelihood that an incorrect 32 bit NFC PIN would generate a matching 16-bit hash is 1/(2^16) meaning 1 in 65536. If an NFC tool possesses an NFC pin code which generates a matching hash, then the likelihood is 99,999% that this PIN code is correct.

On the other hand, each 16-bit hash corresponds to 2^16 different 32-bit NFC pin codes meaning that 65536 NFC pin codes would have to be tried. Each try requires a sequence of NFC write – ECO actuation – NFC read meaning that 65536 individual ECO actuations would be required to determine the correct NFC pin.

The hash values are generated using a simple algorithm such that they are dependent on the user (USER1_PIN will create a different has than USER2_PIN) and the individual PTM 535BZ device (using the globally unique NFC ID of the NFC tag in PTM 535BZ). This prevents attackers from determining if different devices use the same NFC pin code.

The implementation of the hash function is shown in Figure 51 below. Appendix F provides step by step instructions for this process.

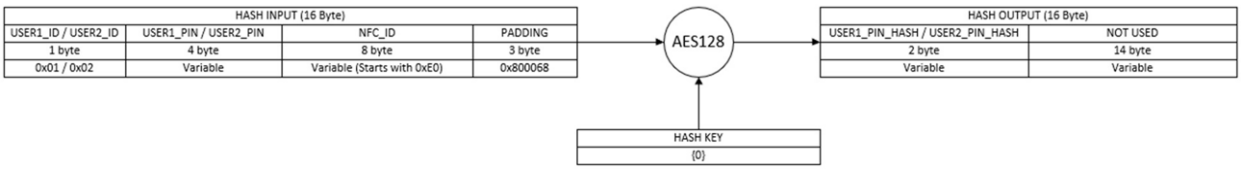

Figure 51 – USER1_PIN_HASH and USER2_PIN_HASH generation

5.8 Using the NFC interface

Using the NFC interface requires a suitable NFC reader. This can be either a USB NFC reader connected to a PC or a suitable smartphone with NFC functionality. The selected reader has to support NFC read and write operations according to the ISO15693 standard.

For PC-based applications, EnOcean recommends the TWN4 Multitech 2 HF NFC Reader (order code T4BT-FB2BEL2-SIMPL) from Elatec RFID Systems (sales-rfid@elatec.com). This reader is shown in Figure 52 below.

Figure 52 – Elatec TWN4 MultiTech Desktop NFC Reader

Many modern smart phones include NFC functionality and can be used to configure PTM 535BZ based on a customer-defined configuration app.

6 Mechanical interface

PTM 535BZ uses the same mechanical outline as the existing PTM 535 / PTM 535Z / PTM 535J and PTM 535Z products. Existing mechanical designs combining one of the existing variants with an ECO 200 harvester can therefore also be used with PTM 535BZ.

Note that PTM 535BZ does not provide meander contacts on board; those have been replaced with the NFC configuration interface.

Note also that PTM 535BZ provides five boundary contacts (AC1, AC2, INPUT1, INPUT2, GND) at different positions compared to previous designs.

6.1 Product dimensions

Figure 53 below provides a product drawing of PTM 535BZ. Refer to the PTM 535BZ product webpage for additional details.

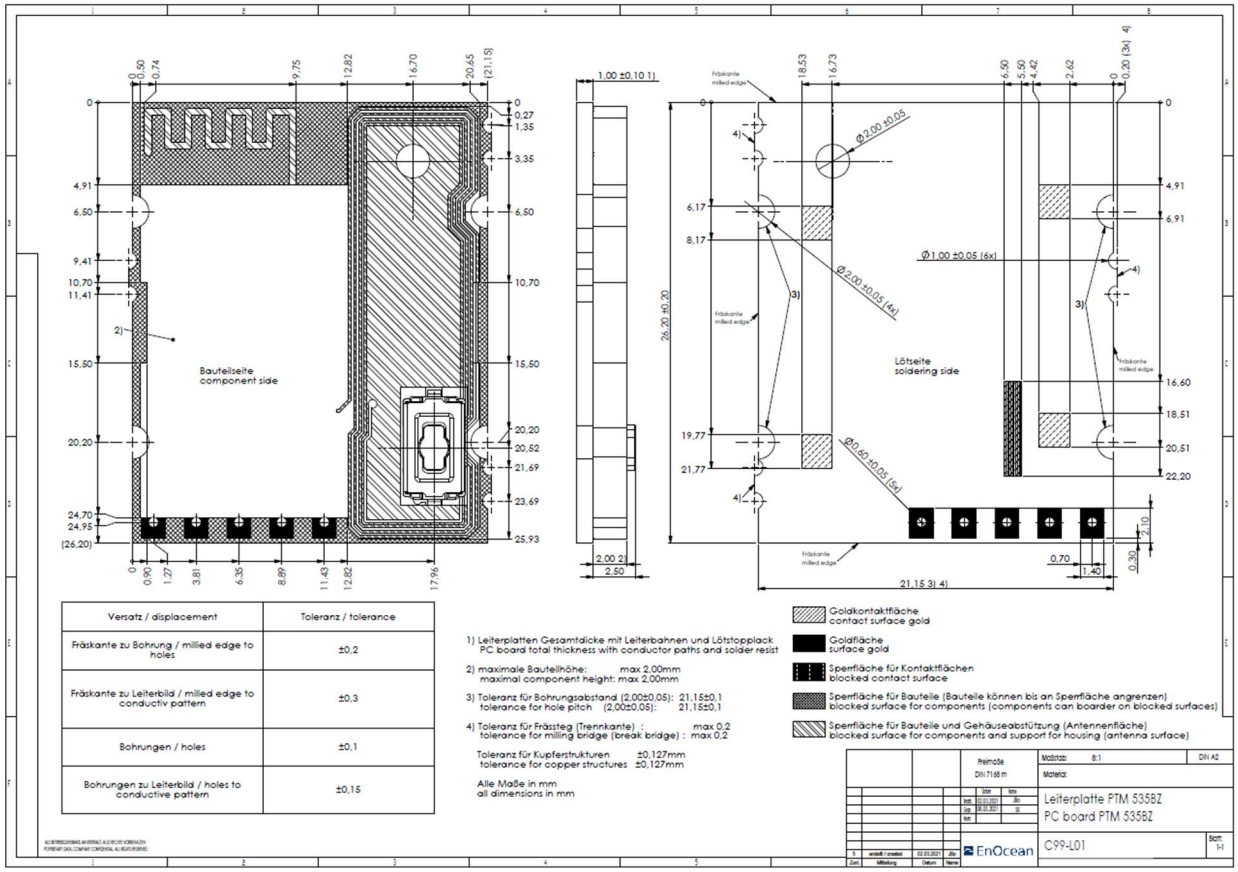

Figure 53 – PTM 535BZ product drawing

7 Application information

7.1 Transmission range

The main factors that influence the system transmission range are:
- Type and location of the antennas of receiver and transmitter
- Type of terrain and degree of obstruction of the link path
- Sources of interference affecting the receiver
- "Dead spots" caused by signal reflections from nearby conductive objects.

Since the expected transmission range strongly depends on this system conditions, range tests should always be performed to determine the reliably achievable range under the given conditions.

The following figures should be treated as a rough guide only:

- Line-of-sight connections
  Typically 10 m range in corridors, up to 30 m in halls

- Plasterboard walls / dry wood
  Typically 10 m range, through max. 2 walls

- Ferro concrete walls / ceilings
  Typically 5 m range, through max. 1 ceiling (depending on thickness)

- Fire-safety walls, elevator shafts, staircases and similar areas should be considered as shielded

The angle at which the transmitted signal hits the wall is very important. The effective wall thickness – and with it the signal attenuation – varies according to this angle. Signals should be transmitted as directly as possible through the wall. Wall niches should be avoided.

Other factors restricting transmission range include:
- Switch mounting on metal surfaces (up to 30% loss of transmission range)
- Hollow lightweight walls filled with insulating wool on metal foil
- False ceilings with panels of metal or carbon fibre
- Lead glass or glass with metal coating, steel furniture

The distance between the receiver and other transmitting devices such as computers, audio and video equipment that also emit high-frequency signals should be at least 0.5 m.

Note that interference from other radio equipment operating in the 2.4 GHz band (WiFi routers, smartphones, wireless audio and video systems, etc.) can have major impact on radio performance.

8 Regulatory approvals

8.1 European Union

8.1.1 Declaration of conformity

Hereby, EnOcean GmbH, declares that this radio equipment is in compliance with the essential requirements and other relevant provisions of Directive 2014/53/EU. A copy of the Declaration of Conformity can be obtained from the product webpage at www.enocean.com

8.1.2 Waste treatment

WEEE Directive Statement of the European Union
The marking below indicates that this product should not be disposed with other household wastes throughout the EU. To prevent possible harm to the environment or human health from uncontrolled waste disposal, recycle it responsibly to promote the sustainable reuse of material resources.

8.2 FCC (United States)

8.2.1 Certificate

<- To Be Inserted ->

8.2.2 Regulatory Statement

This device complies with Part 15 of the FCC rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and

2. This device must accept any interference received, including interference that may cause undesired operation.

To comply with FCC/IC RF exposure limits for general population / uncontrolled exposure, the antenna(s) used for this transmitter must be installed to provide a separation distance of at least 20 cm from all persons and must not be co-located or operating in conjunction with any other antenna or transmitter

Warning
Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

Interference
This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation.

This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

8.3 ISED (former Industry Canada)

8.3.1 Certificate

<- To Be Inserted ->

8.3.2 Regulatory Statement

8.3.2.1 English version

WARNING: Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This device complies with Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions:

1. This device may not cause interference, and

2. This device must accept any interference, including interference that may cause undesired operation of the device.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to ICES-003. These limits are designed to provide reasonable protection against harmful interference in a residential installation.

This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help

8.3.2.2 French version

PRUDENCE: Changements ou modifications pourraient annuler le droit de l'utilisateur à utiliser l'équipement non autorisées.

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes:

1. L'appareil ne doit pas produire de brouillage, et

2. L'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

Cet équipement a été testé et déclaré conforme aux limites d'un appareil numérique de classe B, conformément à la norme ICES-003. Ces limites sont conçues pour fournir une protection raisonnable contre les interférences nuisibles dans une installation résidentielle.

Cet équipement génère, utilise et peut émettre une énergie de radiofréquence et, s'il n'est pas installé et utilisé conformément a ux instructions, il peut causer des interférences nuisibles aux communications radio. Cependant, il n'existe aucune garantie que des interférences no se produiront pas dans une installation particulière.

Si cet équipement provoque des interférences nuisibles à la réception radio ou télévision, ce qui peut être déterminé en mettant l'équipement hors et sous tension, l'utilisateur est encouragé à essayer de corriger l'interférence par une ou plusieurs des mesures suivantes:

- Réorienter ou déplacer l'antenne de réception.

- Augmentez la distance entre l'équipement et le récepteur.

- Connecter l'équipement à une sortie sur un circuit différent de celui sur lequel le récepteur est branché.

- Consulter le revendeur ou un technicien radio / télévision expérimenté pour de l'aide

PTM 535BZ – BLUETOOTH AND ZIGBEE GREEN POWER PUSHBUTTON TRANSMITTER

8.4 ARIB (Japan)

8.4.1 ARIB certificate

EMCC DR. RAŠEK

Notified Body EMC Directive 2014/30/EU
Notified Body Directive 2014/53/EU
RF CAB under the Japan-EC MRA
FCB under the Canada-EC MRA
TCB under the USA-EC MRA

RF CAB ID No. 206

Designated by the German Regulator Bundesnetzagentur to act as a
Recognised Foreign Conformity Assessment Body in accordance with the Japan-EC MRA

CONSTRUCTION TYPE CONFORMITY CERTIFICATE
for
Specified Radio Equipment

Registration No.	JU000610N
Certificate Holder	EnOcean GmbH Kolpingring 18a 82041 Oberhaching Germany
Product Category	Article 2, Paragraph 1, Item 19 (Bluetooth Low Energy 1Mbit/2Mbit) Article 2, Paragraph 1, Item 19 (Zigbee)
Product Designation	Model PTM 535BZ
Product Description	BLE and ZGP Pushbutton Transmitter Module
Software Release No.	0.0.1.35
Manufacturer	Katek GmbH Körtingstraße 1 83224 Grassau Germany

When the product is placed on the Japanese market, it must carry the Specified Radio Equipment marking as shown on the right

R 206-000610

The scope of evaluation relates to the submitted documents only.

This Certificate confirms that the listed product has demonstrated conformity with the relevant technical regulations defined in the attached Annex. It is only valid in conjunction with the Annex.

Unterleinleiter,
2021-03-04

Karlheinz Kraft
Recognised Foreign Conformity Assessment Body

EMCCons DR. RAŠEK GmbH & Co. KG • Stoernhofer Berg 15, 91364 Unterleinleiter, Germany
Tel.: +49 9194 7263-888 • Fax: +49 9194 7263-889 • E-mail: emc.cert@emcc.de • Web: www.emcc.de

9 Product history

Table 32 below lists the product history of PTM 535BZ.

Revision	Release date	Key changes versus previous revision
CA-05	January 2021	Product preview (lead customers only)
DA-06	May 2021	Market release for all customers

Table 32 – Product History

10 References

[1] ECO 200 Website

[2] Bluetooth Core Specification

[3] Zigbee Green Power Specification

[4] RFC3610

[5] IEEE 802.15.4

[6] Bluetooth Assigned Numbers – Company Identifiers

[7] Elatec SW Development Pack

[8] Online AES Calculator

[9] Online XOR Calculator

A. NFC configuration

PTM 535BZ can be configured using the integrated NFC interface as described Chapter 5.

This appendix provides a set of examples for common configuration tasks using the recommended Elatec NFC reader described in Chapter 5.8. These examples can be adapted for use by any suitable NFC writer supporting the NFC ISO15693 interface, for instance a smartphone with built-in NFC support or a PC with a suitable NFC writer.

A.1 Elatec NFC configuration tool

Elatec RFID Systems provides a PC NFC configuration tool called "Director" as part of their software support package [7] which will be used as reference for the examples given here. Figure 54 below shows the user interface of this software.

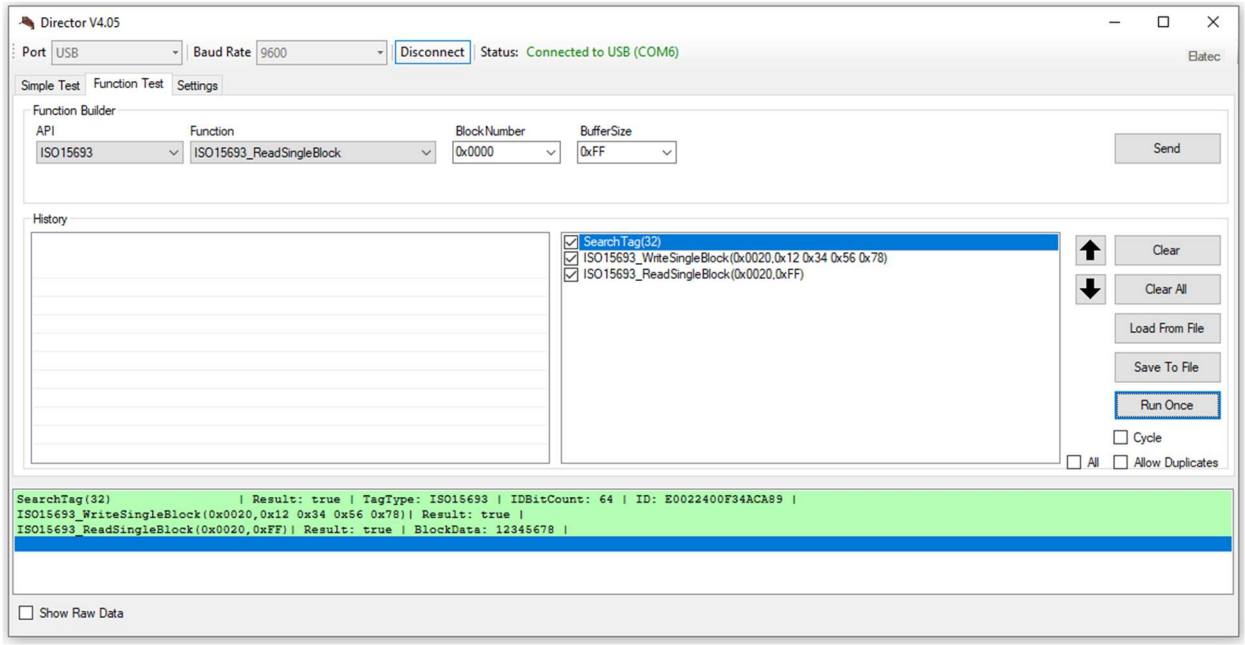

Figure 54 – User interface of TWN4 Director

This software can, it is easily possible to generate the required serial commands that have to be sent via CDC / Virtual COM port to TWN4 and understand the structure of the response that will be received back.

A.1.1 Useful commands

The following commands are used when configuring PTM 535BZ using the Elatec NFC reader:

- `SearchTag(maximum ID length)`
  This command is used to search and identify (tag type, tag ID) an NFC tag.
  This command is used to establish connection to the NFC tag. It has to be issued before any read / write actions.
  Example: `SearchTag(32)`

- `ISO15693_WriteSingleBlock(page_address, page_data)`
  This command is used to write a four-byte NFC data page
  Data is provided in the order `data_byte0 data_byte1 data_byte2 data_byte3`
  Example: `ISO15693_WriteSingleBlock (0x20, 0x12 0x34 0x56 0x78)`

- `ISO15693_ReadSingleBlock (page_address, read_buffer_size)`
  This command is used to read a four-byte NFC data page
  Data is returned in the order `data_byte0 data_byte1 data_byte2 data_byte3`
  Read buffer size must be large enough to hold all returned data (i.e. minimum 4)
  Example: `ISO15693_ReadSingleBlock (0x20, FF)`

A.1.2 Translation into binary data

If the user intends to use these commands within an own application (for instance in a factory configuration application), then they have to be translated into raw data that can be issued to the connected NFC reader via its serial COM interface.

The raw data corresponding to each command can be determined by enabling the "Show Raw Data" feature in the command log of the Director software as shown in Figure 55 below. This data can then be used in user applications to directly communicate with the NFC reader via its serial COM interface.

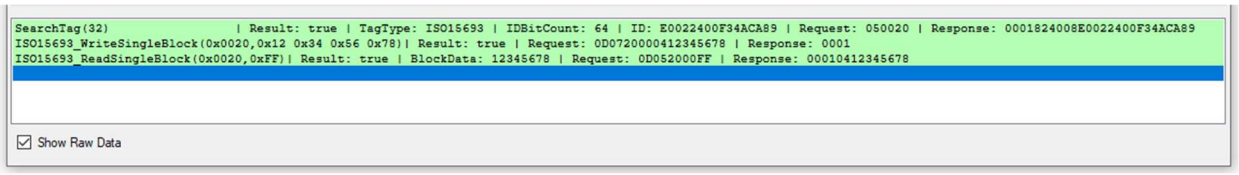

Figure 55 – Enabling raw data display

A.1.3 Direct communication with the NFC reader

User applications can directly communicate with the Elatec NFC reader via its virtual COM port by issuing request data and parsing the corresponding response data.

The Elatec NFC reader uses 9600 baud as baud rate (this is normally detected automatically by the virtual COM port driver). If the user application sends a binary command with the required data, then the Elatec NFC reader will respond accordingly.

Figure 56 below shows this using the example of a write and a read operation to NFC page address 0x20 from the previous chapter.

Figure 56 – Direct communication with the NFC reader

A.2 Configuration examples

This chapter provides examples of common NFC configuration tasks. These examples all follow the same basic configuration sequence as discussed in Chapter 5.7.

A.2.1 Configuration sequence

This configuration sequence consists of the following steps:

1. Specify the request type and the request originator as defined in Table 27
   Register update or functional request, USER1 or USER2

2. Authenticate the configuration request
   Provide the corresponding NFC PIN code as discussed in Chapter 5.1.2

3. Specify registers to be updated
   Applies if configuration request type is "register update". Register specification is done by the CONFIGURATION_SELECTION register described in Chapter 5.7.3

4. Specify the content of the registers to be updated
   Applies if configuration request type is "register update".
   Update values are provided in the corresponding register of the NEW CONFIGURATION area as described in Chapter 5.7. Registers for which no update has been requested can be written as 0x00; their content will not be changed.

5. Actuate the connected ECO 200 harvester five times in each direction to provide the required energy for the configuration sequence

A.2.2 Request status

PTM 535BZ uses the REQUEST_STATUS register to inform the user about the status of a configuration request in described in Chapter 5.6.17. This register should be consulted to check if a configuration request was successful and to determine the root cause of a configuration request failure.

The REQUEST_STATUS register can be read as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_ReadSingleBlock(0x004D,0xFF)`	Read REQUEST_STATUS (Status is allocated in Byte 0 / MSB)

The first (most significant) byte that is returned by the NFC reader contains the configuration request status.

A.3 Functional Requests

PTM 535BZ enables the user to trigger the following functional requests via the NFC interface:

- Commissioning telegram request
- Decommissioning telegram request (only when transmitting ZGP data telegrams)
- Factory reset request

These requests are describe in detail in the next chapters.

A.3.1 Commissioning telegram request

Transmission of a commissioning telegram can be triggered by pressing the LRN button both when transmitting BLE radio telegrams and when transmitting ZGP radio telegrams. In addition to that, transmission of a commissioning telegram can also be requested via the NFC interface.

A.3.1.1 Commissioning telegram request by USER1

Transmission of a commission telegram can be requested by USER1 as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x12 0x00 0x00 0x00)`	Identify request and originator Commissioning Telegram Request Issued by USER1
`ISO15693_WriteSingleBlock(0x0052,0x02 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER1)

PTM 535BZ will evaluate and execute this request upon the next actuation of the ECO 200 harvester as described in Chapter 5.1. PTM 535BZ will then return one of the following responses:

Response Code	Response Type	Description
0x32	SUCCESS	The requested operation was successfully executed
0x52	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
0x72	PIN ERROR	The provided PIN code (USER1_PIN) is incorrect

A.3.1.2 Commissioning telegram request by USER2

Transmission of a commission telegram can be requested by USER2 as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x1A 0x00 0x00 0x00)`	Identify request and originator Commissioning Telegram Request Issued by USER2
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)

PTM 535BZ will evaluate and execute this request upon the next actuation of the ECO 200 harvester as described in Chapter 5.1. PTM 535BZ will then return one of the following responses:

Response Code	Response Type	Description
0x3A	SUCCESS	The requested operation was successfully executed
0x5A	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
0x7A	PIN ERROR	The provided PIN code (USER2_PIN) is incorrect
0x9A	PERMISSION ERROR	The requested operation is not permitted (USER1 has disabled this request for USER2)

A.3.2 ZGP decommissioning telegram request

Transmission of a ZGP decommissioning telegram can be requested via the NFC interface if PTM 535BZ is configured to transmit ZGP data telegrams.

A.3.2.1 ZGP decommissioning telegram request by USER1

Transmission of a ZGP decommission telegram can be requested by USER1 as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x14 0x00 0x00 0x00)`	Identify request and originator Commissioning Telegram Request Issued by USER1
`ISO15693_WriteSingleBlock(0x0052,0x02 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER1)

PTM 535BZ will evaluate and execute this request upon the next actuation of the ECO 200 harvester as described in Chapter 5.1.

PTM 535BZ will return one of the following responses:

Response Code	Response Type	Description
0x34	SUCCESS	The requested operation was successfully executed
0x54	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
0x74	PIN ERROR	The provided PIN code (USER1_PIN) is incorrect
0xB4	PARAMETER ERROR	The requested operation is not possible (PTM 535BZ is configured to transmit BLE data telegrams)

A.3.2.2 ZGP decommissioning telegram request by USER2

Transmission of a commission telegram can be requested by USER2 as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x1C 0x00 0x00 0x00)`	Identify request and originator Commissioning Telegram Request Issued by USER2
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)

PTM 535BZ will evaluate and execute this request upon the next actuation of the ECO 200 harvester as described in Chapter 5.1.

PTM 535BZ will return one of the following responses:

Response Code	Response Type	Description
0x3C	SUCCESS	The requested operation was successfully executed
0x5C	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
0x7C	PIN ERROR	The provided PIN code (USER2_PIN) is incorrect
0x9C	PERMISSION ERROR	The requested operation is not permitted (USER1 has disabled this request for USER2)
0xBC	PARAMETER ERROR	The requested operation is not possible (PTM 535BZ is configured to transmit BLE data telegrams)

A.3.3 Factory reset request

The NFC configuration parameters of PTM 535BZ can be reset to its factory defaults by factory reset which can be requested via the NFC interface.

A.3.3.1 Factory reset request by USER1

Factory reset can be requested by USER1 as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x13 0x00 0x00 0x00)`	Identify request and originator Commissioning Telegram Request by USER1
`ISO15693_WriteSingleBlock(0x0052,0x02 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER1)

PTM 535BZ will evaluate and execute this request upon the next actuation of the ECO 200 harvester as described in Chapter 5.1 and return one of the following responses:

Response Code	Response Type	Description
`0x33`	SUCCESS	The requested operation was successfully executed
`0x53`	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
`0x73`	PIN ERROR	The provided PIN code (USER1_PIN) is incorrect

A.3.3.2 Factory reset request by USER2

Factory reset can be requested by USER2 as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x1B 0x00 0x00 0x00)`	Identify request and originator Commissioning Telegram Request by USER2
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)

PTM 535BZ will evaluate and execute this request upon the next actuation of the ECO 200 harvester as described in Chapter 5.1 and return one of the following responses:

Response Code	Response Type	Description
`0x3B`	SUCCESS	The requested operation was successfully executed
`0x5B`	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
`0x7B`	PIN ERROR	The provided PIN code (USER2_PIN) is incorrect
`0x9B`	PERMISSION ERROR	The requested operation is not permitted (USER1 has disabled this request for USER2)

A.4 Configuration requests

Configuration requests are used to update the value of one or several NFC configuration registers.

A.4.1 Configuration request structure

NFC configuration requests have to follow these steps:

1. Connect to NFC tag
2. Identify the request (Configuration request by USER1 or USER2)
3. Authenticate the request (Provide USER1_PIN or USER2_PIN)
4. Identify the configuration items (which registers to update)
5. Provide the new register values for the update
6. Actuate the connected ECO 200 harvester five times in each direction to provide the required energy for the configuration update
7. Check the REQUEST_STATUS register for the status of the configuration request (optional)

A.4.1.1 Configuration request for USER1

Configuration requests for USER1 use the following sequence:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x11 0x00 0x00 0x00)`	Identify request (Configuration request by USER1)
`ISO15693_WriteSingleBlock(0x0052,0x02 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER1)
`ISO15693_WriteSingleBlock(0x0051,0xNN 0xNN 0xNN 0xNN)`	Identify configuration register(s) to update (CONFIGURATION_SELECTION register)
`ISO15693_WriteSingleBlock(0x00XX,0xYY 0xYY 0xYY 0xYY)`	Provide new value (YY) for first register (XX)
…	
`ISO15693_WriteSingleBlock(0x00ZZ,0xYY 0xYY 0xYY 0xYY)`	Provide new value (YY) for last register (ZZ)

Each register to be updated is identified by the corresponding bit (marked as 0xNN) being set in the CONFIGURATION_SELECTION register. The new value (marked as 0xYY) for each register to be updated has to be provided for the register in the corresponding location (marked 0x00XX) in the NEW CONFIGURATION area.

A.4.1.2 Configuration status for USER1

PTM 535BZ will evaluate and execute the configuration request provided by USER1 and provide one of the following responses in the REQUEST_STATUS register:

Response Code	Response Type	Description
0x31	SUCCESS	The requested operation was successfully executed
0x51	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
0x71	PIN ERROR	The provided PIN code (USER1_PIN) is incorrect
0x91	PERMISSION ERROR	The requested operation is not permitted (Modification of the selected register(s) is not possible)
0xB1	PARAMETER ERROR	Incorrect register value(s) provided

A.4.1.3 Configuration request for USER2

Configuration update requests for USER2 use the same structure as configuration requests for USER1. The only differences are in the request encoding, the request authentication and the request status reporting as shown below. Subsequent configuration examples will all be given for USER2; they can be adjusted to USER1 accordingly.

Command	Description
SearchTag(32)	Connect to tag (Search for up to 32 byte ID)
ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)	Identify request (Configuration request by USER2)
ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)	Authenticate request (PIN Code of USER2)
ISO15693_WriteSingleBlock(0x0051,0xNN 0xNN 0xNN 0xNN)	Identify configuration register(s) to update (CONFIGURATION_SELECTION register)
ISO15693_WriteSingleBlock(0x00XX,0xYY 0xYY 0xYY 0xYY)	Provide new value (YY) for first register (XX)
…	
ISO15693_WriteSingleBlock(0x00ZZ,0xYY 0xYY 0xYY 0xYY)	Provide new value (YY) for last register (ZZ)

A.4.1.4 Configuration status for USER2

PTM 535BZ will evaluate and execute the configuration request provided by USER2 and provide one of the following responses in the REQUEST_STATUS register:

Response Code	Response Type	Description
0x39	SUCCESS	The requested operation was successfully executed
0x59	IN PROGRESS	The requested operation is in progress (Additional ECO actuations are required to complete the operation)
0x79	PIN ERROR	The provided PIN code (USER2_PIN) is incorrect
0x99	PERMISSION ERROR	The requested operation is not permitted (Modification of the selected register(s) not possible or disabled by USER1)
0xB9	PARAMETER ERROR	Incorrect register value(s) provided

A.4.2 Security configuration

The security configuration group allows changing the security keys and the authentication PIN codes.

A.4.2.1 Changing USER1_PIN

USER1_PIN is used to authenticate requests by USER1; it can only be changed by USER1. USER1_PIN can be changed as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x11 0x00 0x00 0x00)`	Identify request (Configuration request by USER1)
`ISO15693_WriteSingleBlock(0x0052,0x02 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER1)
`ISO15693_WriteSingleBlock(0x0051,0x40 0x00 0x00 0x00)`	Identify register(s) to update (SECURITY -> USER1_PIN)
`ISO15693_WriteSingleBlock(0x0079,0x12 0x34 0x56 0x78)`	Provide new value (0x12345678) for USER1_PIN

Make sure that the new PIN code is properly noted when changing USER1_PIN. For security reasons, it is not possible to modify or reset USER1_PIN unless the current USER1_PIN is known.

A.4.2.2 Changing USER2_PIN

USER2_PIN is used to authenticate requests by USER2; it can be changed either by USER2 or by USER1.

USER2_PIN can be changed by USER2 as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x80 0x00 0x00 0x00)`	Identify register(s) to update (SECURITY -> USER2_PIN)
`ISO15693_WriteSingleBlock(0x007A,0x12 0x34 0x56 0x78)`	Provide new value (0x12345678) for USER2_PIN

USER2_PIN can be changed by USER1 as follows:

Command	Description
SearchTag(32)	Connect to tag (Search for up to 32 byte ID)
ISO15693_WriteSingleBlock(0x0050,0x11 0x00 0x00 0x00)	Identify request (Configuration request by USER1)
ISO15693_WriteSingleBlock(0x0052,0x02 0x00 0x35 0xE5)	Authenticate request (PIN Code of USER1)
ISO15693_WriteSingleBlock(0x0051,0x80 0x00 0x00 0x00)	Identify register(s) to update (SECURITY -> USER2_PIN)
ISO15693_WriteSingleBlock(0x007A,0x12 0x34 0x56 0x78)	Provide new value (0x12345678) for USER2_PIN

Being able to change the PIN code used by USER2 allows USER1 (e.g. an OEM) to pre-assign a PIN code for use by USER2 (e.g. an installer or the end customer) to PTM 535BZ devices. It also allows USER1 to reset USER2_PIN in case this is lost or forgotten.

A.4.2.3 Reading USER1_CONFIGURATION_OPTIONS

USER1 can determine the configuration options that are available by reading the USER1_CONFIGURATION_OPTIONS register. This allows handling different product revisions where some features are present in one revision only. The USER1_CONFIGURATION_OPTIONS register can be read as follows:

Command	Description
SearchTag(32)	Connect to tag (Search for up to 32 byte ID)
ISO15693_ReadSingleBlock(0x0045,0xFF)	Read USER1_CONFIGURATION_OPTIONS

This command will return the four-byte USER1_CONFIGURATION_OPTIONS as described in Chapter 5.6.15 and shown below for reference.

USER1_CONFIGURATION_OPTIONS (ADDRESS 0x45, VALUE = 0xD7373F33)								
Position	Byte 0 (SECURITY)		Byte 1 (ZGP)		Byte 2 (BLE)		Byte 3 (SYSTEM)	
0x01	SECURITY_KEY1	0b1	ZGP_TX_CONFIG	0b1	BLE_TX_CONFIG	0b1	LRN_TELEGRAM_ REQUEST	0b1
0x02	SECURITY_KEY2	0b1	ZGP_SEC_CONFIG	0b1	BLE_SEC_CONFIG	0b1	FACTORY_RESET_ REQUEST	0b1
0x04	SECURITY_KEY3	0b1	ZGP_BUTTON_MAP	0b1	BLE_BUTTON_MAP	0b1	RFU	0b0
0x08	RFU	0b0	RFU	0b0	BLE_SOURCE_ADDRESS	0b1	RFU	0b0
0x10	USER2_CONFIG_ OPTIONS	0b1	ZGP_PROTOCOL_ CONFIG	0b1	BLE_ MANUFACTURER_ID	0b1	BUTTON_CONFIG	0b1
0x20	RFU	0b0	ZGP_DECOMMISSIONING_ REQUEST	0b1	BLE_CH_REGx	0b1	RADIO_CONFIG	0b1
0x40	USER1_PIN	0b1	RFU	0b0	RFU	0b0	RFU	0b0
0x80	USER2_PIN	0b1	RFU	0b0	RFU	0b0	RFU	0b0
Value	**0xD7**		**0x37**		**0x3F**		**0x33**	

A.4.2.4 Reading USER2_CONFIGURATION_OPTIONS

USER2 can determine the configuration options that are available by reading the USER2_CONFIGURATION_OPTIONS register. This allows identifying configuration options that are not available due to a restriction imposed by USER1 or to handle the case different product revisions where some features are present in one revision only.

The USER2_CONFIGURATION_OPTIONS register can be read as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_ReadSingleBlock(0x0046,0xFF)`	Read USER2_CONFIGURATION_OPTIONS

This command will return the four-byte USER2_CONFIGURATION_OPTIONS as described in Chapter 5.6.15.This is shown below for reference.

USER2_CONFIGURATION_OPTIONS (ADDRESS 0x46, VALUE = 0x87:37:3F:33)								
Position	Byte 0 (SECURITY)		Byte 1 (ZGP)		Byte 2 (BLE)		Byte 3 (SYSTEM)	
0x01	SECURITY_KEY1	0b1	ZGP_TX_CONFIG	0b1	BLE_TX_CONFIG	0b1	LRN_TELEGRAM_ REQUEST	0b1
0x02	SECURITY_KEY2	0b1	ZGP_SEC_CONFIG	0b1	BLE_SEC_CONFIG	0b1	FACTORY_RESET_ REQUEST	0b1
0x04	SECURITY_KEY3	0b1	ZGP_BUTTON_MAP	0b1	BLE_BUTTON_MAP	0b1	RFU	0b0
0x08	RFU	0b0	RFU	0b0	BLE_SOURCE_ADDRESS	0b1	RFU	0b0
0x10	USER2_CONFIG_ OPTIONS	0b0	ZGP_PROTOCOL_ CONFIG	0b1	BLE_ MANUFACTURER_ID	0b1	BUTTON_CONFIG	0b1
0x20	RFU	0b0	ZGP_DECOMMISSIONING_ REQUEST	0b1	BLE_CH_REGx	0b1	RADIO_CONFIG	0b1
0x40	USER1_PIN	0b0	RFU	0b0	RFU	0b0	RFU	0b0
0x80	USER2_PIN	0b1	RFU	0b0	RFU	0b0	RFU	0b0
Value	**0x87**		**0x37**		**0x3F**		**0x33**	

Note that by default, USER2 has access to the same configuration options as USER1 except USER1_PIN and USER2_CONFIGURATION_OPTIONS (which both can only be changed by USER1).

A.4.2.5 Restricting USER2_CONFIGURATION_OPTIONS

USER1 can restrict the configuration options that are available to USER2. This allows USER1 (for instance an OEM) to pre-configure certain parameters (for instance the radio protocol) and prevent USER2 (for instance an installer) from changing those. To do so, USER1 has to update USER2_CONFIGURATION_OPTIONS with the desired restrictions.

Consider the example of USER1 wanting to prevent USER2 from changing the radio configuration register RADIO_CONFIG so that USER2 cannot change the radio protocol from BLE to ZGP or vice versa.

To do so, USER1 has to modify the USER2_CONFIGURATION_OPTIONS register described in Chapter 5.6.15 such that RADIO_CONFIG cannot be changed by USER2. The required change (via the shadow register in the NEW CONFIGURATION area) is shown below for reference (marked in red).

USER2_CONFIGURATION_OPTIONS (ADDRESS 0x76, VALUE = 0x87:37:3F:13)								
Position	Byte 0 (SECURITY)		Byte 1 (ZGP)		Byte 2 (BLE)		Byte 3 (SYSTEM)	
0x01	SECURITY_KEY1	0b1	ZGP_TX_CONFIG	0b1	BLE_TX_CONFIG	0b1	LRN_TELEGRAM_ REQUEST	0b1
0x02	SECURITY_KEY2	0b1	ZGP_SEC_CONFIG	0b1	BLE_SEC_CONFIG	0b1	FACTORY_RESET_ REQUEST	0b1
0x04	SECURITY_KEY3	0b1	ZGP_BUTTON_MAP	0b1	BLE_BUTTON_MAP	0b1	RFU	0b0
0x08	RFU	0b0	RFU	0b0	BLE_SOURCE_ADDRESS	0b1	RFU	0b0
0x10	USER2_CONFIG_ OPTIONS	0b0	ZGP_PROTOCOL_ CONFIG	0b1	BLE_ MANUFACTURER_ID	0b1	BUTTON_CONFIG	0b1
0x20	RFU	0b0	ZGP_DECOMMISSIONING_ REQUEST	0b1	BLE_CH_REGx	0b1	RADIO_CONFIG	0b0
0x40	USER1_PIN	0b0	RFU	0b0	RFU	0b0	RFU	0b0
0x80	USER2_PIN	0b1	RFU	0b0	RFU	0b0	RFU	0b0
Value	**0x87**		**0x37**		**0x3F**		**0x13**	

USER1 can then request a configuration update with this USER2_CONFIGURATION_OPTIONS register value as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x11 0x00 0x00 0x00)`	Identify request (Configuration request by USER1)
`ISO15693_WriteSingleBlock(0x0052,0x02 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER1)
`ISO15693_WriteSingleBlock(0x0051,0x10 0x00 0x00 0x00)`	Identify configuration register(s) to update (SECURITY -> USER2_CONFIGURATION_OPTIONS)
`ISO15693_WriteSingleBlock(0x0076,0x87 0x37 0x3F 0x13)`	Provide new register value (0x87373F13) for USER2_CONFIGURATION_OPTIONS

A.4.2.6 Reading SECURITY_KEY1

PTM 535BZ uses different security keys as discussed in Chapter 2.5. SECURITY_KEY1 is used by default; this key can be read and written via the NFC interface. To read SECURITY_KEY1, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_ReadSingleBlock(0x0040,0xFF)`	Read SECURITY_KEY1 (16 byte)
`ISO15693_ReadSingleBlock(0x0041,0xFF)`	
`ISO15693_ReadSingleBlock(0x0042,0xFF)`	
`ISO15693_ReadSingleBlock(0x0043,0xFF)`	

Note that SECURITY_KEY2 cannot be read via NFC.

A.4.2.7 Writing SECURITY_KEY1

SECURITY_KEY1 is initialized to a random value during production; this value can be changed by the user. Factory reset will restore the value set at production.

To write SECURITY_KEY1, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x01 0x00 0x00 0x00)`	Identify configuration register(s) to update (SECURITY -> SECURITY_KEY1)
`ISO15693_WriteSingleBlock(0x0068,0x00 0x01 0x02 0x03)`	Provide new value for SECURITY_KEY1: `000102030405060708090A0B0C0D0E0F`
`ISO15693_WriteSingleBlock(0x0069,0x04 0x05 0x06 0x07)`	
`ISO15693_WriteSingleBlock(0x006A,0x08 0x09 0x0A 0x0B)`	
`ISO15693_WriteSingleBlock(0x006B,0x0C 0x0D 0x0E 0x0F)`	

A.4.2.8 Writing SECURITY_KEY2

SECURITY_KEY2 is initialized to a random value during production; this value can be changed by the user. Factory reset will update SECURITY_KEY2 to a different random value to prevent unauthorized users from determining a security key that was previously used.

SECURITY_KEY2 can only be written; it is not possible to read SECURITY_KEY2 via the NFC interface. To write SECURITY_KEY2, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x02 0x00 0x00 0x00)`	Identify configuration register(s) to update (SECURITY -> SECURITY_KEY2)
`ISO15693_WriteSingleBlock(0x006C,0x00 0x01 0x02 0x03)`	Provide new value for SECURITY_KEY2: 000102030405060708090A0B0C0D0E0F
`ISO15693_WriteSingleBlock(0x006D,0x04 0x05 0x06 0x07)`	
`ISO15693_WriteSingleBlock(0x006E,0x08 0x09 0x0A 0x0B)`	
`ISO15693_WriteSingleBlock(0x006F,0x0C 0x0D 0x0E 0x0F)`	

A.4.3 ZGP configuration

The ZGP configuration group allows defining the radio and protocol parameters related to ZGP data telegrams.

A.4.3.1 ZGP radio channel selection

PTM 535BZ allows the user to select the ZGP radio channel during the commissioning process using the LRN button as described in Chapter 4.5.2. Alternatively, it is possible to directly select the ZGP radio channel via the NFC interface. In this case it is also possible to disable the channel selection via the LRN button to prevent unintended change of the radio channel.

In this example, we assume that we want to configure PTM 535BZ to transmit ZGP telegrams using ZGP radio channel 15 with radio channel selection via LRN button disabled:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x01 0x00 0x20)`	Identify register(s) to update SYSTEM -> RADIO_CONFIG ZGP -> ZGP_TX_CONFIG
`ISO15693_WriteSingleBlock(0x0054,0x00 0x01 0x00 0x00)`	Provide new register value for RADIO_CFG 0x01: Transmit ZGP Telegrams
`ISO15693_WriteSingleBlock(0x0060,0x04 0x00 0x00 0x00)`	Provide new value for ZGP_TX_CONFIG 0x04: Radio channel 15, Radio channel selection with button disabled

In this example, we assume that we want to configure PTM 535BZ to transmit ZGP telegrams using ZGP radio channel 20 with radio channel selection via LRN button enabled amongst the primary radio channels (11, 15, 20, 25):

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x01 0x00 0x20)`	Identify register(s) to update SYSTEM -> RADIO_CONFIG ZGP -> ZGP_TX_CONFIG
`ISO15693_WriteSingleBlock(0x0054,0x00 0x01 0x00 0x00)`	Provide new register value for RADIO_CFG 0x01: Transmit ZGP Telegrams
`ISO15693_WriteSingleBlock(0x0060,0x29 0x00 0x00 0x00)`	Provide new value for ZGP_TX_CONFIG 0x29: Radio channel 20, Primary selection enabled

A.4.3.2 ZGP Device ID selection

PTM 535BZ allows the user to select the ZGP Device ID used within ZGP data telegrams as discussed in Chapter 4.4.1. By default, PTM 535BZ uses Device ID 0x07 (Generic Switch). To use Device ID 0x02 (On / Off switch) instead, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x10 0x00 0x20)`	Identify register(s) to update SYSTEM -> RADIO_CONFIG ZGP -> ZGP_PROTOCOL_CONFIG
`ISO15693_WriteSingleBlock(0x0054,0x00 0x01 0x00 0x00)`	Provide new register value for RADIO_CFG 0x01: Transmit ZGP Telegrams
`ISO15693_WriteSingleBlock(0x0060,0x00 0x00 0x02 0x00)`	Provide new value for ZGP_PROTOCOL_CONFIG 0x02: Command List enabled, Device ID 0x02

A.4.3.3 ZGP input status encoding

As described in Chapter 4.4.1.3, the user can change the input status encoding for Device ID other than 0x07 from the default encoding (listed in Table 8) to a user-defined encoding. For this example, we assume that we want to change the commands send when ECO 200 is actuated from 0x22 / 0x23 (Toggle / Release) to 0x20 / 0x21 (On / Off) and leave the remaining commands as is. To do so, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x14 0x00 0x20)`	Identify register(s) to update SYSTEM -> RADIO_CONFIG ZGP -> ZGP_TX_CONFIG ZGP -> ZGP_BUTTON_MAP
`ISO15693_WriteSingleBlock(0x0054,0x00 0x01 0x00 0x00)`	Provide new register value for RADIO_CFG 0x01: Transmit ZGP Telegrams
`ISO15693_WriteSingleBlock(0x0060,0x00 0x00 0x02 0x00)`	Provide new value for ZGP_PROTOCOL_CONFIG (Command List enabled, Device ID 0x02) ZGP_PROTOCOL_CONFIG = 0x02
`ISO15693_WriteSingleBlock(0x0064,0x20 0x21 0x12 0x13)`	Provide new values for ZGP_COMMAND[0:3]
`ISO15693_WriteSingleBlock(0x0065,0x14 0x15 0x16 0x17)`	Provide new values for ZGP_COMMAND[4:7]

A.4.4 BLE configuration

The BLE configuration group allows changing protocol configuration, the input status encoding and the security key used for the transmission of BLE data telegrams.

A.4.4.1 BLE protocol configuration

By default, PTM 535BZ transmits BLE data telegrams using Static Source Addresses, 1 Mbit data rate and 20 ms advertising interval on the primary advertising channels (37, 38 and 39) with the input status encoded according to Table 2. Different parameters can be selected by the user via the BLE_TX_CONFIG register described in Chapter 5.6.3.

For this example, we assume the user wants to reduce the advertising interval to 10 ms and use custom input status encoding where the ECO press will be encoded like a B0 press on a PTM 215B module (INPUT_STATUS = 0x09) and no telegram will be sent when the ECO is released (INPUT_STATUS = 0xFF meaning that no telegram will be sent). Such input status encoding could be helpful when using a push button based on PTM 535BZ in toggle mode where each press of the button will toggle the status of the receiver.

This can be achieved by following these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x00 0x05 0x00)`	Identify register(s) to update BLE -> BLE_TX_CONFIG BLE -> BLE_BUTTON_MAP
`ISO15693_WriteSingleBlock(0x0058,0x50 0x00 0x00 0x00)`	Provide new value (0x50) for BLE_TX_CONFIG
`ISO15693_WriteSingleBlock(0x005C,0x09 0xFF 0x03 0x02)`	Provide new values for BLE_INPUT_STATUS [0:3]
`ISO15693_WriteSingleBlock(0x005D,0x05 0x04 0x07 0x06)`	Provide new values for BLE_INPUT_STATUS[4:7]

A.4.4.2 Security key selection for BLE

It is user-selectable which security key (SECURITY_KEY1 or SECURITY_KEY2) should be used to authenticate BLE radio telegrams as described in Chapter 3.4.2 and to generate re-solvable private addresses as described in Chapter 3.3.5.2. By default, SECURITY_KEY1 is used.

To select SECURITY_KEY2, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x00 0x02 0x00)`	Identify register(s) to update (BLE -> BLE_SEC_CONFIG)
`ISO15693_WriteSingleBlock(0x0058,0x00 0x04 0x00 0x00)`	Provide new value: BLE_SEC_CONFIG = 0x04

To select SECURITY_KEY1 again, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration Update by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x00 0x02 0x00)`	Identify register(s) to update (BLE -> BLE_SEC_CONFIG)
`ISO15693_WriteSingleBlock(0x0058,0x00 0x00 0x00 0x00)`	Provide new value: BLE_SEC_CONFIG = 0x00

A.4.5 System configuration

The System configuration area allows selecting the radio protocol (BLE or ZGP) used for the telegram transmission.

A.4.5.1 Selecting the radio protocol

PTM 535BZ uses BLE radio protocol by default. PTM 535BZ can be configured by USER2 to use ZGP as radio protocol as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x00 0x00 0x20)`	Identify register(s) to update (SYSTEM -> RADIO_CONFIG)
`ISO15693_WriteSingleBlock(0x0054,0x00 0x01 0x00 0x00)`	Provide new value for RADIO_CFG 0x01: Transmit BLE telegrams

PTM 535BZ can be configured by USER2 to use BLE as radio protocol as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x00 0x00 0x20)`	Identify register(s) to update (SYSTEM -> RADIO_CONFIG)
`ISO15693_WriteSingleBlock(0x0054,0x00 0x00 0x00 0x00)`	Provide value for RADIO_CFG 0x00: Transmit BLE Telegrams

To determine which radio protocol is currently used by PTM 535BZ, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_ReadSingleBlock(0x002C,0xFF)`	Read RADIO_CFG (Byte 1 is the RADIO_CFG register)

The NFC read operation will return the RADIO_CFG register value in Byte 1 (meaning in the second byte of the four-byte return value).

A.4.5.2 Changing the input configuration

PTM 535BZ allows the user to configure which direction of ECO 200 actuation is considered as "press" and which is consider as "release" for the status encoding in BLE or ZGP data telegrams.

As discussed in Chapter 2.4.1, a move of the ECO 200 spring away from the connected PCB is by default considered to be a "press" while a move of the ECO 200 spring towards the connected PCB is by default considered to be a "release".

PTM 535BZ can be configured by USER2 to reverse this encoding (so that a move of the ECO 200 spring towards the connected PCB is considered to be a "press" while a move of the ECO 200 spring away from the connected PCB is considered to be a "release") as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x00 0x00 0x10)`	Identify register(s) to update (SYSTEM -> BUTTON_CONFIG)
`ISO15693_WriteSingleBlock(0x0054,0x01 0x00 0x00 0x00)`	Provide new value for BUTTON_CFG 0x01: ECO_DIRECTION inverted

PTM 535BZ can be configured by USER2 to use the standard encoding as follows:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_WriteSingleBlock(0x0050,0x19 0x00 0x00 0x00)`	Identify request (Configuration request by USER2)
`ISO15693_WriteSingleBlock(0x0052,0x03 0x00 0x35 0xE5)`	Authenticate request (PIN Code of USER2)
`ISO15693_WriteSingleBlock(0x0051,0x00 0x00 0x00 0x10)`	Identify register(s) to update (SYSTEM -> BUTTON_CONFIG)
`ISO15693_WriteSingleBlock(0x0054,0x00 0x00 0x00 0x00)`	Provide new value for BUTTON_CFG 0x00: ECO_DIRECTION default

To determine which encoding is currently used by PTM 535BZ, follow these steps:

Command	Description
`SearchTag(32)`	Connect to tag (Search for up to 32 byte ID)
`ISO15693_ReadSingleBlock(0x002C,0xFF)`	Read BUTTON_CFG (Byte 0 is the BUTTON_CFG register)

The NFC read operation will return the BUTTON_CFG register value in Byte 0 (meaning in the first byte of the four-byte return value).

B. Receiver configuration for BLE

PTM 535BZ transmits sensor information as a set of advertising events either on the BLE advertising channels or on user-defined radio channels as described in Chapter 3.2.

To maximize the likelihood of reception of these telegrams, it is necessary that the receiver is either permanently in receive mode on one of the radio channels used by PTM 535BZ or – if this is not possible – periodically in receive mode for a sufficiently long duration.

B.1 Scanning parameters

Three key timing parameters have to be considered when configuring a receiver (scanner) for periodical reception of advertising events sent by a transmitter (advertiser). These three parameters are:

- Advertising interval
  Time between two advertising events sent by the transmitter
- Scan interval
  Time between the start of two consecutive scanning cycles of the receiver
- Scan window
  Duration for which the receiver will scan within each scanning cycle

Figure 57 below illustrates these three parameters.

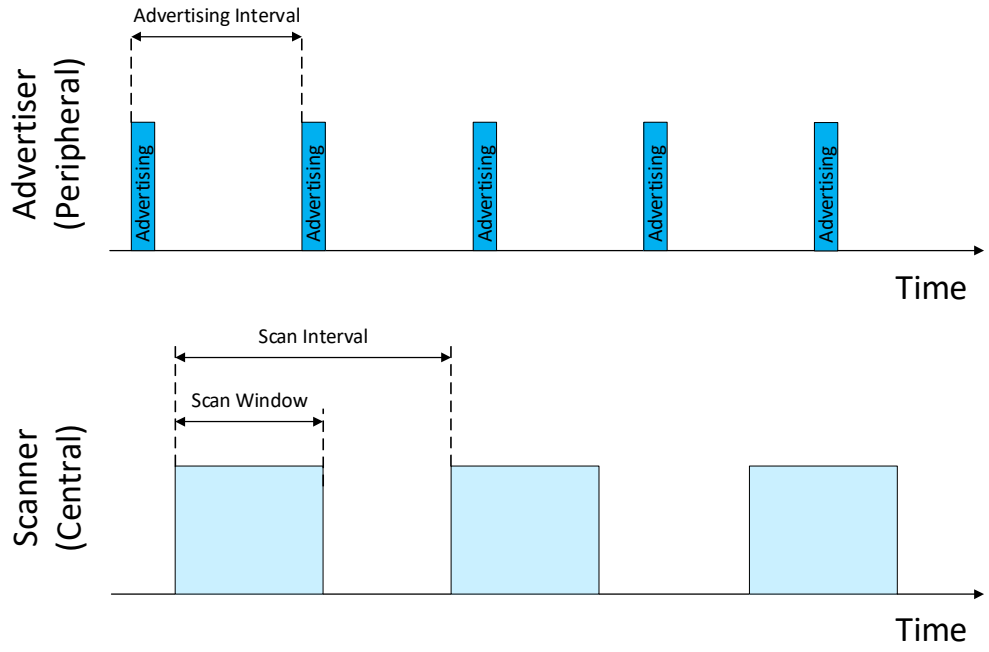

Figure 57 – Scanning parameters

B.1.1 Advertising interval

PTM 535BZ transmits advertising events with an advertising interval of either 20 ms (default setting) or 10 ms (NFC configurable setting).

The time required to transmit each advertising telegram within the advertising event is approximately 0.5 ms and the time required to transmit the entire advertising event (transmission of three advertising telegrams on three different radio channels including radio channel change) is approximately 2.5 ms.

B.1.2 Scan window

The scan window has to be selected such that the receiver will under all conditions receive at least one full advertising telegram.

To ensure this requirement, we consider the worst-case condition where the receiver starts scanning directly after the start of one transmission and therefore misses a part of it. Under these conditions, it is necessary that the receiver remains active until the next advertising telegram has been fully transmitted. This is illustrated in Figure 58 below.

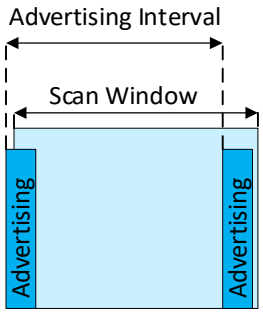

Figure 58 – Scan window setting

Figure 58 above shows that the minimum duration of the scan window is dependent on the advertising interval:

- If PTM 535BZ uses 20 ms advertising intervals, then the scan window has to be at least 20 ms (advertising interval) plus 0.5 ms (telegram duration) plus a timing margin to account for the random time offset at the transmitter.
  Using a scan window of at least 23 ms is recommended for this case.

- If PTM 535BZ uses 10 ms advertising intervals, then the scan window has to be at least 10 ms (advertising interval) plus 0.5 ms (telegram duration) plus a timing margin to account for the random time offset at the transmitter.
  Using a scan window of at least 13 ms is recommended for this case.

B.1.3 Scan interval

The scan interval has to be selected such that the receiver will not be inactive so long that it misses all three advertising events.

The longest period for which the receiver can be inactive is given by the time between the end of the first advertising events (assuming that the receiver exactly misses the last bit of it) and the beginning of the third advertising event (so that this will certainly be received). Figure 59 illustrates this.

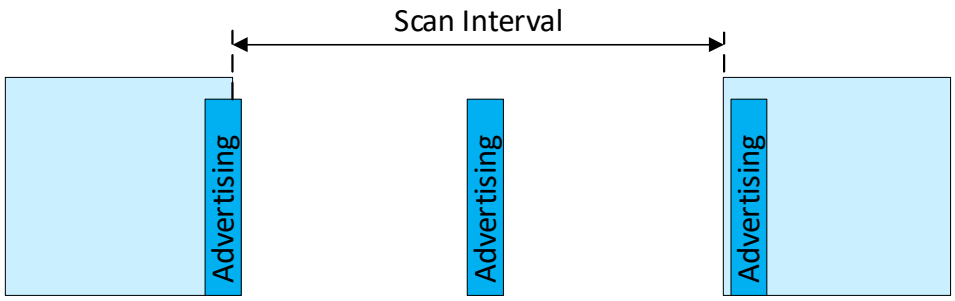

Figure 59 – Scan interval setting

From Figure 59 above, it can be seen that the maximum duration of the scan interval is dependent on the advertising interval:

- If PTM 535BZ uses 20 ms advertising intervals, then the scan interval has to be less than the time between the end of the first advertising event and the begin of the third advertising event (2 * 20 ms = 40 ms) minus 0.5 ms (telegram duration) minus a timing margin to account for the random time offset at the transmitter. Using a scan interval of no more than 37 ms is recommended for this case.
- If PTM 535BZ uses 10 ms advertising intervals, then the scan interval has to be less than the time between the end of the first advertising event and the begin of the third advertising event (2 * 10 ms = 20 ms) minus 0.5 ms (telegram duration) minus a timing margin to account for the random time offset at the transmitter. Using a scan interval of no more than 17 ms is recommended for this case.

B.1.4 Summary

Table 33 below summarizes the recommended receiver scan settings.

PTM 535BZ Advertising Interval	Receiver Scan Window (Minimum)	Receiver Scan Interval (Maximum)
20 ms	23 ms	37 ms
10 ms	13 ms	17 ms

Table 33 – Recommended receiver scan settings

C. Parsing of PTM 535BZ BLE radio telegrams

This appendix is intended as an example of how start to parse received PTM 535BZ radio telegrams. Please refer to Chapter 3 for a description of the BLE frame structure.

C.1 Data telegram example

We consider the following raw data telegram data captured from a PTM 535BZ device:

```
D6 BE 89 8E 42 13 06 00 00 10 15 E2 0C FF DA 03 40 00 00 00 01 B0 56 1C 03 89 F4 6E
```

C.1.1 BLE frame structure

The message shown above can be parsed into the following components (keep in mind the little-endian byte order):

BLE Access Address (4 byte): `0x8E89BED6 (Advertising)`

BLE Frame Control (2 byte):
```
0x1342
Size of source address + payload: 0x13 (19 byte)
Telegram type: Non-connectable Advertising
```

BLE Source Address (6 byte): `0xE21510000006`

Length of payload (1 byte): `0x0C (12 byte)`

Type of payload (1 byte): `0xFF (manufacturer-specific data)`

Manufacturer ID (2 byte): `0x03DA (EnOcean GmbH)`

EnOcean Payload (9 byte): `40 00 00 00 01 B0 56 1C 03`

CRC (3 byte): `89 F4 6E`

C.1.2 EnOcean data telegram payload structure

The EnOcean data telegram payload can now be parsed as follows:

Sequence Counter (4 byte): `0x00000040`

Input Status (1 byte): `01 (ECO 200 Press, INPUT1 and INPUT2 disconnected)`

Telegram Signature (4 byte): `B0 56 1C 03`

C.2 Commissioning telegram example

We consider the following raw commissioning telegram data captured from a PTM 535BZ device:

```
D6 BE 89 8E 42 24 06 00 00 10 15 E2 1D FF DA 03 41 00 00 00 1D 76 A7 A0 DE 93 E7 F5
53 13 2D 58 94 CF F9 9B 06 00 00 10 15 E2 97 B2 B6
```

C.2.1 BLE frame structure

The message shown above can be parsed into the following components (keep in mind the little endian byte order):

BLE Access Address (4 byte):	`0x8E89BED6`

BLE Frame Control (2 byte):	`0x2442`
	`Size of source address + payload: 0x24 (36 byte)`
	`Telegram type: Non-connectable Advertising`

BLE Source Address (6 byte):	`0xE21510000006`
Length of payload (1 byte):	`0x1D (29 byte)`
Type of payload (1 byte):	`0xFF (manufacturer-specific data)`
Manufacturer ID (2 byte):	`0x03DA (EnOcean GmbH)`
EnOcean Payload (27 byte):	`41 00 00 00 1D 76 A7 A0 DE 93 E7 F5 53 13 2D 58 94 CF F9 9B 06 00 00 10 15 E2`
CRC (3 byte):	`0xB6B297`

C.2.2 EnOcean commissioning telegram payload structure

The EnOcean commissioning telegram payload can now be parsed as follows:

Sequence Counter (4 byte):	`0x00000041`
Security Key:	`1D76A7A0DE93E7F553132D5894CFF99B`
Static Source Address:	`0xE21510000006`

D. Authentication of PTM 535BZ BLE data telegrams

PTM 535BZ provides the option to authenticate BLE data telegrams as described in Chapter 3.4.2. The authentication mechanism used by PTM 535BZ is standardized as RFC3610 [1].

The following description aims to summarize the security processing steps for users not deeply familiar with cryptography in general or RFC3610 in particular.

D.1 Algorithm input parameters

The purpose of the security processing in PTM 535BZ is to calculate a unique signature that can be used to verify authenticity (telegram has not been modified) and originality (telegram comes from the assumed sender) of a telegram.

To do so, two types of algorithm parameters are required:

- Constant algorithm input parameters
  These parameters identify high level algorithm and telegram properties and are the same for any PTM 535BZ telegram

- Variable algorithm input parameters
  These parameters identify telegram-specific parameters and therefore depend on the specifics of the transmitted telegram

D.1.1 Constant input parameters

The RFC3610 implementation in PTM 535BZ requires two constant input parameters:

- Length field size
  This is the size (in byte) of the field used to encode the length of the input data (which is the payload to be authenticated).
  The size of the authenticated PTM 535BZ payload is 9 byte; therefore one byte would be easily sufficient to encode the payload size. The minimum value permitted by the standard is however 2 bytes which is therefore chosen.

- Signature size
  This is the desired size of the generated signature which is 4 byte for PTM 535BZ

Table 34 below summarizes these constant algorithm parameters.

Parameter	Comment / Description	Example
Length Field Size	Size (in bytes) of the field used to encode the input length	2 (always, minimum permissible size)
Signature Size	Desired size (in byte) of the signature generated by the algorithm	4 (always)

Table 34 – Constant algorithm input parameters

D.1.2 Variable input parameters

The RFC3610 implementation in PTM 535BZ requires four variable input parameters:

- Source address
  The 6 byte source address used to identify the sender of an authenticated message. The source address is required in little endian (least significant byte first) format.

- Input data (Payload to be authenticated)
  The authenticated payload consists of the Length, Telegram Type, Manufacturer ID, Sequence Counter and Input Status fields of the BLE data telegram (9 byte in total).

- Length of input data (Size of the payload to be authenticated)
  The length of the payload to be authenticated is 9 byte for PTM 535BZ data telegrams.

- Security key
  Each PTM 535BZ is programmed with a random 16 byte security key during manufacturing.

Table 35 below summarizes these parameters and provides the corresponding values from the data telegram example in Appendix C.1.

Parameter	Comment / Description	Example
Source Address	Unique source address of the PTM 535BZ module (little endian)	0600001015E2 (little endian representation of E21510000006)
Input Data	Telegram data to be authenticated	0CFFDA034000000001
Input Length	Length of input data (in bytes, encoded using 2 bytes)	0x0009
Sequence Counter	Incrementing counter to avoid replay Part of the input data (byte 4 … 7)	40000000 (little endian representation of the counter value 00000040)
Security Key	128 bit random key that is known both to sender and receiver	1D76A7A0DE93E7F553132D5894CFF99B

Table 35 – Variable input parameters

D.1.3 Obtaining the security key

All required parameters except the security key can be directly extracted from the received message that shall be authenticated.

The security key – the common secret shared between sender and receiver – has to be obtained via specific mechanisms.

There are three different ways to obtain the security key used by a PTM 535BZ module when it is transmitting BLE telegrams:

- Via NFC (by reading SECURITY_KEY1 or by writing SECURITY_KEY2)
- Via the product label (which might provide information about SECURITY_KEY1)
- Via a commissioning telegram as described in Chapter 3.4.3

D.1.4 Internal parameters

The RFC3610 implementation in PTM 535BZ derives a set of internal parameters for further processing from the provided input parameters.

Again, there are two types of internal parameters:

- Constant internal parameters
  These parameters are based on the high-level algorithm and telegram properties and are the same for any PTM 535BZ telegram
- Variable input parameters
  These parameters are based on the telegram-specific parameters and therefore depend on the specifics of the transmitted telegram

D.1.5 Constant internal parameters

The RFC3610 implementation in PTM 535BZ derives two internal parameters – M' and L' – based on the input data and uses them to construct A0_Flag and B_0_Flag which – together with the iteration counter i – are required for subsequent processing.

The value of these internal parameters - listed in Table 36 below - is the same for all PTM 535BZ telegrams.

Parameter	Comment / Description	Example
M'	Binary encoded output length M' = (Output length / 2) - 1	0b001 (always)
L'	Binary encoded length field size L' = length field size - 1	0b001 (always)
A0_Flag	L'	0x01 (always)
B0_Flag	(0b01<<6) + (M'<<3) + L'	0x49 (always)
i	Iteration counter	0x0000 (always)

Table 36 – Constant internal parameters

D.1.6 Variable internal parameters

The RFC3610 implementation in PTM 535BZ derives four internal parameters – Nonce, A0, B0 and B1 – based on the telegram specific input data and the constant internal parameters.

These variable internal parameters - listed in Table 37 below - are then used together with the security key to calculate the actual signature.

Parameter	Comment / Description	Example
Nonce	13 byte initialization vector formed as concatenation of source address, sequence counter and 0x00 padding	0600001015E240000000000000
A0	A0_Flag followed by Nonce followed by 2 byte 0x00	0106000010 15E2400000000000000000
B0	B0_Flag followed by Nonce followed by 2 byte 0x00	4906000010 15E2400000000000000000
B1	Input Length followed by Input Data followed by 5 byte of 0x00 padding	00090CFFDA0340000000010000000000

Table 37 – Variable internal parameters

D.2 Algorithm execution sequence

The algorithm uses the variable internal parameters `A_0`, `B_0`, `B_1` together with the private key to generate the authentication vector `T_0` using three AES-128 and two XOR operations. The algorithm execution sequence is shown in Figure 60 below. The first four bytes of T_0 are then used to authenticate PTM 535BZ telegrams.

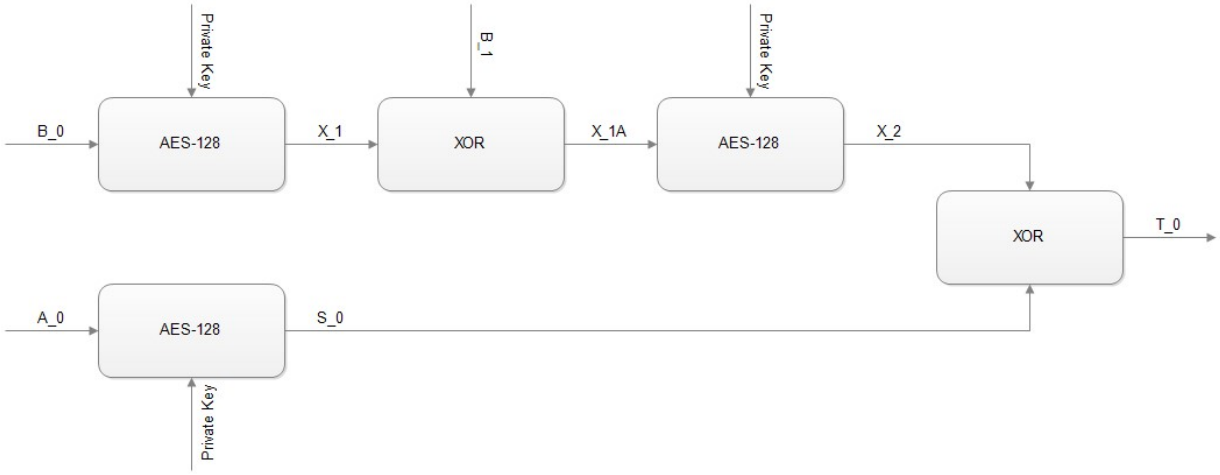

Figure 60 – Authentication algorithm sequence

D.3 Example

For this example, we consider the BLE data telegram payload `0CFFDA034000000001B0561C03` received from a PTM 535BZ with source address `E21510000006` which is authenticated with security key `1D76A7A0DE93E7F553132D5894CFF99B`.

The last four bytes of this payload (`B0561C03`) are the sender-provided signature which has to be authenticated (compared against the signature the receiver calculates based on its own security key).

The variable input parameters are therefore the following:

Parameter	**In this example**
Source Address	`0600001015E2 (little endian representation of E215000019B8)`
Input Data	`0CFFDA034000000001B0561C03`
Input Length	`0x0009`
Sequence Counter	`40000000`
Security Key	`1D76A7A0DE93E7F553132D5894CFF99B`

The constant internal parameters are always the same:

Parameter	**In this example**
A0_Flag	`0x01` (always)
B0_Flag	`0x49` (always)
i	`0x0000` (always)

Based on variable input data and constant internal algorithm parameters, we can now derive the following variable internal parameters:

Parameter	**In this example**
Nonce	`0600001015E240000000000000`
A0	`0106000001015E2400000000000000000000`
B0	`4906000001015E2400000000000000000000`
B1	`00090CFFDA0340000000010000000000`

We can now calculate the signature using AES128 and XOR operations.

At the time of writing, a suitable online AES calculator could be found in [8]. Likewise, a suitable XOR calculator could be found in [9].

The execution sequence would then be as follows:

```
X_1 = AES128(B0, Key)
X_1 = AES128(490600001015E2400000000000000000, 1D76A7A0DE93E7F553132D5894CFF99B)
X_1 = B3B9A7D1B3A1F898D7C8A984AC7A2771

X_1A = XOR(X_1, B_1)
X_1A = XOR(B3B9A7D1B3A1F898D7C8A984AC7A2771, 00090CFFDA0340000000010000000000)
X_1A = B3B0AB2E69A2B898D7C8A884AC7A2771

X_2 = AES128(X1A, Key)
X_2 = AES128(B3B0AB2E69A2B898D7C8A884AC7A2771, 1D76A7A0DE93E7F553132D5894CFF99B)
X_2 = B974A8873BE64E9EB0171A81D8FD53FB

S_0 = AES128(A0, Key)
S_0 = AES128(010600001015E2400000000000000000, 1D76A7A0DE93E7F553132D5894CFF99B)
S_0 = 0922B484107EEA202AAB404DB08A0F87

T_0 = XOR(X_2, S_0)
T_0 = XOR(B974A8873BE64E9EB0171A81D8FD53FB, 0922B484107EEA202AAB404DB08A0F87)
T_0 = B0561C032B98A4BE9ABC5ACC68775C7C
```

The calculated signature is formed by the first four bytes of `T_0`, i.e. it is `B0 56 1C 03`.

The calculated signature matches the signature that was transmitted as part of the payload. This proves that the telegram originates from a sender that possesses the same security key and the telegram content has not been modified.

E. Address resolution for resolvable private addresses (RPA)

PTM 535BZ provides the option to obfuscate its identity by means of using resolvable private addresses (RPA that are generated using an Identity Resolution Key as described in Chapter 3.3.5.2. This appendix provides an example how to generate and resolve such address.

E.1 Address resolution example

We consider a PTM 535BZ module with the following 128 bit IRK:
`IRK = BE759A027A4870FD242794F4C45220FB`

We further consider a telegram with the following 48 bit resolvable private address:
`RPA = 493970E51944`

We will now test if this resolvable private address was generated using the IRK above.

Referring to the resolvable private address structure shown in Figure 12, we split the resolvable private address into `prand` and `hash` as follows:

```
prand  = (RPA & 0xFFFFFF000000) >> 24
prand  = 0x493970

hash   = RPA & 0x000000FFFFFF
hash   = 0xE51944
```

Next, we verify the address mode by checking the two most significant bit of `prand`:

```
mode   = (prand & 0xC00000) >> 22
mode   = 0b01
```

Referring to Chapter 3.3.5.2, the setting of `0b01` indicates resolvable private address mode.

To generate the hash, we add 104 bit of padding (all zeros) to `prand`:
`padded prand = 0x000000000000000000000000493970`

We can now generate the hash as AES128 operation between IRK and `padded prand`:

```
hash = AES128(IRK; padded prand)
hash = AES128(0xBE759A027A4870FD242794F4C45220FB;0x000000000000000000000000493970)
```

At the time of writing, a suitable online AES calculator could be found in [8]. With this, we can calculate the result as:
`hash   = 0x286ACB1F9C8A80EE21B3F02225E51944`

With that, we can verify that the lowest 24 bit of the calculated `hash` (`0xE51944`) match the hash that was received as part of the resolvable private address. Therefore, the transmitter of this telegram used this specific IRK to generate this resolvable private address.

F. Calulcating the NFC PIN hash

As described in Chapter 5.7.5.1, PTM 535BZ provides hash representations of USER1_PIN and USER2_PIN to allow an NFC tool to determine with high likelihood if it possesses the required pin code. This appendix provides two examples how a tool would generate such hash representation and use it to verify if it possesses the correct NFC pin code.

F.1 USER1_PIN_HASH example

We consider a PTM 535BZ module with the following 64 bit globally unique NFC ID:

```
NFC_ID              = E0022400F340E0D4
```

PTM 535BZ provides the following USER1_PIN_HASH:

```
USER1_PIN_HASH      = DEAE
```

We further consider that a connected NFC tool wants to test if PTM 535BZ uses the default USER1_PIN (as described in Chapter 5.1.2):

```
USER1_PIN           = 020035E5
```

To do so, we first construct the hash input as described in Chapter 5.7.5.1:

Byte 0	Byte 1…4	Byte 5..12	Byte 13…15
USER_ID	PIN_CODE	NFC_ID	PADDING
01 (USER1)	020035E5	E0022400F340E0D4	800068

With these settings, we obtain HASH_INPUT:

```
HASH_INPUT          = 01020035E5E0022400F340E0D4800068
```

We now encrypt HASH_INPUT using AES128 and a key of {0}, meaning the zero vector. At the time of writing, an online AES128 calculator could be found in [8]. With this, we can calculate H1 as follows:

```
H1 = AES128(HASH_INPUT, {0})
H1 = AES128(01020035E5E0022400F340E0D4800068,00000000000000000000000000000000)
H1 = DEAE1F602E8343F680EA1F2D606669AC
```

USER1_PIN_HASH equals the most significant 16 bit of H1, meaning:

```
USER1_PIN_HASH = DEAE
```

The calculated USER1_PIN_HASH matches the one provided by PTM 535BZ; it is therefore likely that PTM 535BZ uses the tested USER1_PIN (020035E5).

F.2 USER2_PIN_HASH example

We consider a PTM 535BZ module with the following 64 bit globally unique NFC ID:

`NFC_ID = E0022400F340E0D4`

PTM 535BZ provides the following USER2_PIN_HASH:

`USER2_PIN_HASH = 1234`

We further consider that a connected NFC tool wants to test if PTM 535BZ uses the default USER1_PIN as described in Chapter 5.1.2:

`USER1_PIN = 030035E5`

To do so, we first construct the hash input as described in Chapter 5.7.5.1:

Byte 0	Byte 1…4	Byte 5..12	Byte 13…15
USER_ID	PIN_CODE	NFC_ID	PADDING
02 (USER2)	030035E5	E0022400F340E0D4	800068

With these settings, we obtain HASH_INPUT:

`HASH_INPUT = 02030035E5E0022400F340E0D4800068`

We now encrypt HASH_INPUT using AES128 and a key of {0}, meaning the zero vector. At the time of writing, an online AES128 calculator could be found in [8]. With this, we can calculate H1 as follows:

```
H1 = AES128(HASH_INPUT, {0})
H1 = AES128(02030035E5E0022400F340E0D4800068,00000000000000000000000000000000)
H1 = 463B6C6FD67AEA16F21E51B9EE4AB229
```

USER2_PIN_HASH equals the most significant 16 bit of H1, meaning:

`USER2_PIN_HASH = 463B`

The calculated USER2_PIN_HASH (9E1A) does not match the one provided by PTM 535BZ (1234). PTM 535BZ therefore does not use the tested USER2_PIN (030035E5).