TCM 615 / TCM 615U / TCM 615J

EnOcean Transceiver Gateway Module

Observe precautions! Electrostatic sensitive devices!

REVISION HISTORY

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

Version	Author	Reviewer	Date	Major Changes
1.0	MKA	RS, JB, PF, DL, EO, MH	14 Apr 2025	First public release for TCM 615

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

Disclaimer

This user manual 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.
Recycling and disposal according to local regulations.

TABLE OF CONTENT

1 General description

This document is a preview of the functionality of the upcoming TCM 615, TCM 615U and TCM 615J devices. These devices will be commonly referred to as "TCM 615" throughout the remainder of this document unless functionality that is specific to one of these devices is described.

This document also provides an introduction to EnOcean radio protocol in Appendix 0 and an introduction to EnOcean security protocol in Appendix 0. This information is intended to aid the understanding of TCM 615 device functionality.

1.1 Basic functionality

TCM 615 is a transceiver gateway for EnOcean radio providing a radio link between EnOcean radio devices, and an external host connected via a serial interface. The serial interface uses the well-established EnOcean Serial Protocol, version 3 (ESP3).

TCM 615 receives and transmits EnOcean radio telegrams based on an external antenna which is connected via the host PCB.

TCM 615 will process received telegrams and forward them to an external host processor or host PC via the ESP3 interface. Conversely, messages received by TCM 615 from an external host via the ESP3 interface will be processed by TCM 615 and then transmitted as EnOcean radio telegrams.

TCM 615, TCM 615U and TCM 615J differ by the supported operating frequency as required by the different target markets:

- TCM 615
  Transceiver Gateway for 868.3 MHz ASK (EnOcean Radio Protocol version 1)
  Target market is Europe

- TCM 615U (Roadmap Product)
  Transceiver Gateway for 902.875 MHz FSK (EnOcean Radio Protocol version 2)
  Target markets are US and Canada

- TCM 615J (Roadmap Product)
  Transceiver Gateway for 928.350 MHz FSK (EnOcean Radio Protocol version 2)
  Target markets are Japan and Australia

1.2 Target applications

TCM 615 devices provide EnOcean radio gateway functionality for applications requiring small size, high processing performance, the latest security features and sufficient memory to allow for firmware updates.

Typical applications for TCM 615 devices include line-powered actuators such as dimmers, relays or shutter controllers as well as radio gateways or controllers. TCM 615 devices are not optimized for ultra-low power designs such as sensor or switch applications.

TCM 615 products are limited to OEM installation ONLY.

1.3 Device interface

TCM 615 is implemented as 31 pin reflow-solderable module with optimized form factor for size constrained applications. TCM 615 is backwards pin-to-pin compatible with previous TCM 515 products. Figure 1 below shows TCM 615.

Figure 1 – TCM 615

1.4 TCM 615 versus TCM 515

TCM 615 significantly enhances key performance parameters of TCM 515:

- Computation performance and memory size
- Security functionality (including secure firmware update)
- Power consumption during transmission and reception

TCM 615 provides a secure environment for software execution and secure (encrypted and authenticated) firmware update designed to meet the latest security requirements. It is therefore the ideal solution for permanently active, line-powered solutions such as actuators and gateways.

This secure environment (based on a secure bootloader) results in a significantly longer start-up time of TCM 615 compared to TCM 515. Applications with intermittent power supply where fast start-up time is critical may continue to use TCM 515.

Note that TCM 615 does not provide support for SmartAck functionality.

1.5 Technical data

Antenna		50 Ohm antenna (connected at host board)
Supported Radio Frequencies	TCM 615: TCM 615U: TCM 615J:	868.300 MHz ASK 902.875 MHz FSK 928.350 MHz FSK
Data Rate		125 kbps
Receiver Sensitivity (1)	TCM 615: TCM 615U: TCM 615J:	-95 dBm -95 dBm -95 dBm
Maximum Input Power (1)		-10 dBm
Receiver Blocking Performance (TCM 615)		Class 2 according to EN 300 220-1
Radiated RF Immunity (TCM 615)		10 V / m according to EN 301 489-3
Transmit Power	TCM 615: TCM 615U: TCM 615J:	+10 dBm +1 dBm 0 dBm
Supply Voltage (min / max / typ) (2)		1.8 V / 3.6 V / 3.3 V
Supply Current Receive State (at 2.0 V)		15 mA
Supply Current Transmit State (at 2.0 V) (3)		20 mA
Supply Current Idle State (at 2.0V) (4)		5 mA
Supply Current Sleep State (at 2.0V)		< 10 µA
Power-up to Receive State Timing (5)		230 ms
Supply Current between Power-up and Receive State		12 mA
Transmit to Receive State switching time (6)		< 2 ms
Sleep State to Receive State switching time		< 5 ms
Serial Interface To Host		UART (ESP3 Standard with TURBO mode)

General Note: All figures are typical values at 25°C unless otherwise specified.

Note 1: Sensitivity and Maximum Input Power figures are based on 0.1% telegram error rate for the combination of 3 received sub-telegrams

Note 2: TCM 615 automatically switches between two power modes (DCDC and LDO) depending on the supply voltage

Note 3: ASK modulation encodes the bit status (0 or 1) using different radio power levels where 0 is encoded with a high-power level and 1 with a low power level. The Transmitter current therefore depends on the ratio between bits with the value 0 and bits with the value 1 in the bit stream.
The figure given here is for a PN9 sequence.

Note 4: Idle Mode is used when TCM 615 operates in transmit-only mode while no telegram is transmitted.

Note 5: During start-up, TCM 615 can wait for a configurable additional delay before transitioning to Receive state to allow for power supply stabilization and start-up of the external host. See chapter 3.

Note 6: Transmit to Receive State switch over time is measured from the transmission of the last bit (end of frame) of a radio frame until the receiver is ready to receive the first bit (preamble) of a radio frame

1.6 Physical dimensions

Module Dimensions	19.0 mm x 14.7 mm x 3.0 mm (all +- 0.3 mm)
Module Weight	1 g

1.7 Environmental conditions

Operating Temperature	-40°C ... +85°C
Storage Temperature	-40°C ... +85°C
Humidity	0% to 95% r.h. (non-condensing)

1.8 Packaging information

Packaging Unit / Method	250 units / Tape and reel

1.9 Ordering information

Type	Current Revision	Ordering Code
TCM 615	DA	S3003-K615:DA
TCM 615U	DA	S3053-K615:DA
TCM 615J	DA	S3063-K615:DA

2 Functional information

2.1 High-level functionality

TCM 615 is a fully integrated radio transceiver family which enables communication with other devices implementing the EnOcean Radio Protocol (ERP) as specified in [2].

TCM 615 is used to exchange (send and / or receive) radio telegrams with external sensors, switches or actuators.

TCM 615 is connected to an external host which for instance could be a microprocessor, a controller or a gateway via the EnOcean Serial Protocol v3 (ESP3) interface. ESP3 commands are listed within this document for information purposes only; for details about ESP3 commands refer to the ESP3 specification [1].

Figure 2 below shows the integration of TCM 615 into a typical system environment.

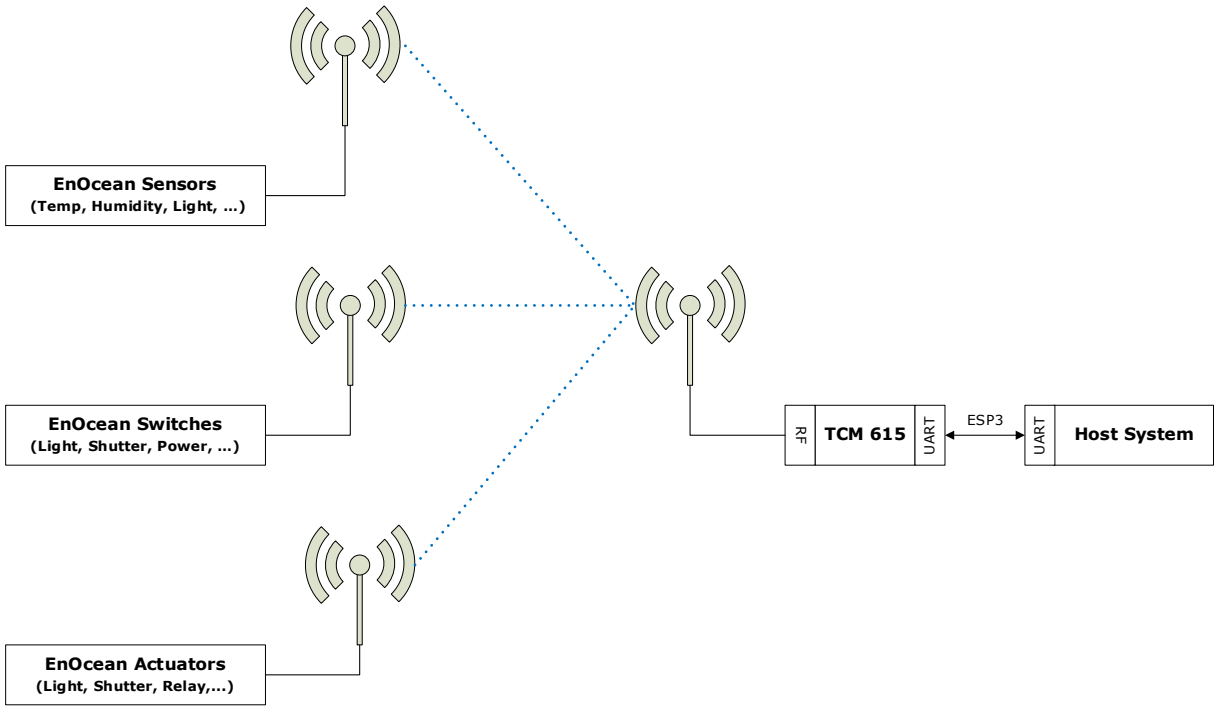

Figure 2 – TCM 615 system environment

2.2 Functional states

TCM 615 implements the following functional states:

- Power-up and system initialization (with user-configurable delay)
  This state is described in chapter 3
- Receive (RX) state (telegram reception with security processing, filtering, repeating as required) This state is described in chapter 4
- Transmit (TX) state (telegram transmission with security processing as required)
  This state is described in chapter 5
- Sleep state (low power state to conserve energy)
  This state is described in chapter 8

The transition between these functional states is shown in Figure 3 below.

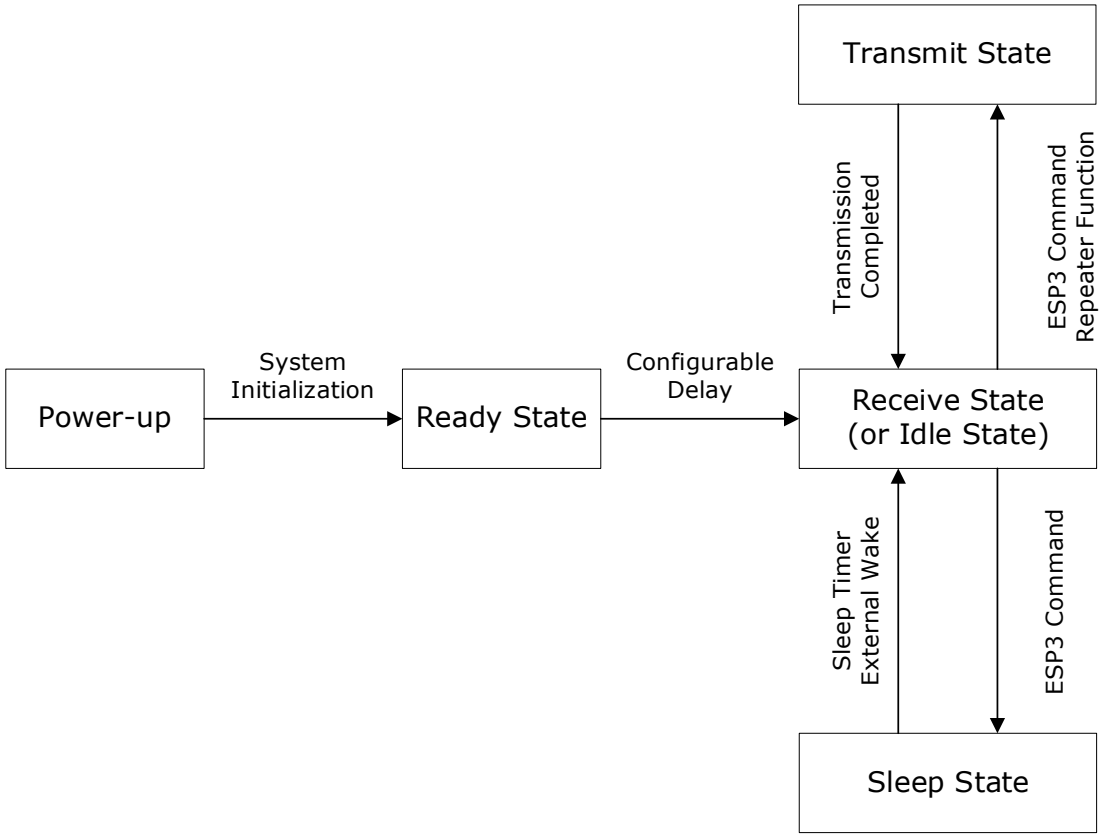

Figure 3 – TCM 615 functional states

Note that it is possible to configure TCM 615 to operate as transmit-only device which disables receive functionality. If TCM 615 is configured to operate as transmit-only device, then Receive state is replaced by Idle state where TCM 615 will wait for ESP3 commands. Transmit-only functionality is described in chapter 5.6.

2.3 Device interface

TCM 615 implements a 31 pin reflow-solderable interface. Solder mask data is available on request from EnOcean.

2.3.1 Pin-out

The pin assignment (as seen from the top of the TCM 615 device) is shown in Figure 4 below. Solder mask and mechanical data is available from EnOcean.

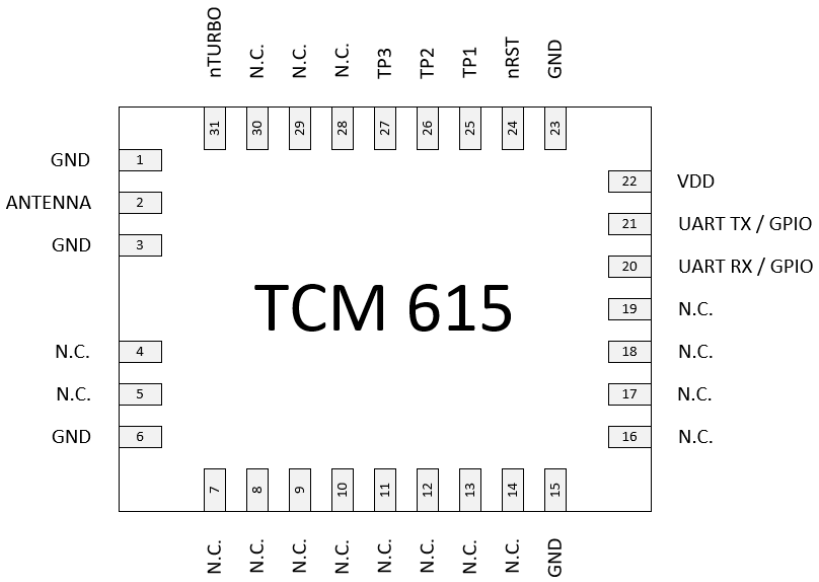

Figure 4 – TCM 615 device interface

Table 1 below summarizes the signal assignment. Signals marked with "NC" are reserved for production test and future device variants and must not be connected in the design.

PIN	NAME	PIN	NAME	PIN	NAME
1	GND	12	NC	23	GND
2	RF_50 (50Ω antenna)	13	NC	24	nRESET (Reset input, active low)
3	GND	14	NC	25	TP1 (Test Interface)
4	NC	15	GND	26	TP2 (Test Interface)
5	NC	16	NC	27	TP3 (Test Interface)
6	GND	17	NC	28	NC
7	NC	18	NC	29	NC
8	NC	19	NC	30	NC
9	NC	20	UART_RX (Input to TCM 615)	31	nTURBO (UART speed, active low)
10	NC	21	UART_TX (Output from TCM 615)		
11	NC	22	VDD		

Table 1 - TCM 615 pin assignment

2.4 Power supply

TCM 615 is supplied by the VDD and GND pins and supports a supply voltage range between 1.8 V and 3.6 V. For best radio performance it is very important to minimize noise on the supply voltage lines. Please see chapter 11.5 and 11.6.

The TCM 615 supply voltage must not drop below the minimum supply voltage of 1.8 V during operation; otherwise, correct operation cannot be guaranteed.
Power supply design must account for load transients (e.g. at start-up or wake-up from Sleep state) and possible voltage drops to provide the required supply voltage.

TCM 615 automatically switches between two different operation modes (DCDC mode and LDO mode) depending on the supply voltage. Wide fluctuations of the supply voltage should be avoided to minimize the impact.

2.5 Serial interface (UART)

TCM 615 communicates with the external host using the standard ESP3 serial (UART) interface based on the signals UART_TX (Pin 21, direction from TCM 615 to external host) and UART_RX (Pin 20, direction from external host to TCM 615).

The UART interface – in conjunction with the VDD and GND pins – can also be used for the secure device firmware update functionality provided by TCM 615.
It is therefore strongly recommended that the PCB design provides the ability to connect UART_RX, UART_TX, VDD and GND to an external interface for the purpose of secure device firmware update.

The default interface speed of the ESP3 interface is 57600 bit per second and data is transmitted using 8 data bits, 1 STOP bit and no parity (8N1).

It is possible to select a faster communication speed of 460800 bit per second using the ESP3 CO_SET_BAUDRATE command (see chapter 9.1).

Additionally, it is possible to change the default ESP3 interface speed at power up from 57.600 bit per second to 460.800 bit per second by connecting the nTURBO input (Pin 31, active low) to Ground.

Subsequent selection of the interface speed (57.600 or 460.800 bit per second) is always possible during operation using the CO_SET_BAUDRATE command irrespective of the state of the TURBO input pin.

Do not select a UART interface speed which cannot be supported by the connected host processor as this would prevent subsequent communication.

2.6 Antenna

TCM 615 receives and transmits data using a 50Ω antenna connected to its RF_50 input (Pin 2). Please see chapter 12 for recommendations regarding the antenna design.

2.7 Reset

TCM 615 can be reset by pulling the nRESET pin (Pin 24, active low) to Ground. Please see Chapter 11.8 for reset circuit recommendations.

It is strongly recommended that the PCB design provides the ability to connect to the nRESET signal – e.g. by means of providing a suitable test point pad on the PCB - for the purpose of analysis and debug.

2.8 Test interface (TP1, TP2, TP3)

TCM 615 provides a test interface (TP1, TP2 and TP3). The intended use of this interface is for analysis and debugging of customer products by EnOcean.

It is strongly recommended that customer PCB design provides the ability to connect external devices to the TP1, TP2 and TP3 signals – e.g. by providing suitable test point pads on the PCB - for the purpose of analysis and debug.

2.9 Product label

Each TCM 615 contains a product label with a structure similar to the one shown in Figure 5 below.

Figure 5 – TCM 615 product label structure

The label shown above identifies the following parameters in writing:

- Product name (TCM 615)
- Order number (S3003-K615)
- Product revision (DA-04)
- Manufacturing date (week 40, 2024)
- Manufacturer traceability code (592001002206)

2.9.1 QR code

The TCM 615 product label contains an automatically readable QR code in the lower right corner which encodes certain product parameters according to the ANSI/MH10.8.2-2013 standard as listed in Table 2 below.

Data Identifier	Data Length (excluding identifier)	Data Content
30S	12 characters (hexadecimal)	EnOcean Universal Radio ID (EURID)
30P	10 characters (alphanumeric)	Ordering Code
2P	4 characters (alphanumeric)	Step Code and Revision
S	14 characters (decimal)	Serial Number (starts with 01)

Table 2 – TCM 615 product QR code structure

3 Power-up, initialization and system operation

After power-up, TCM 615 first initializes the security subsystem before executing the system and radio setup. The total time required for this process is 230 milliseconds.

TCM 615 can be configured to wait for an additional pre-configured delay after the security subsystem initialization if required (for instance, to stabilize the power supply). The user can configure the total initialization time as persistent parameter (maintained after power down) using the ESP3 command CO_WR_STARTUP_DELAY. Setting this parameter to less than 230 milliseconds will not have any effect as this is the minimum required start-up time.

After that, TCM 615 is ready for operation and will either enter Receive mode (if TCM 615 has been configured for transmit and receive mode, this is the default case) or Sleep mode (if TCM 615 has been configured for transmit-only mode).

3.1 Typical operation sequence for transmit and receive mode

The default configuration of TCM 615 is transmit and receive mode. In this mode, TCM 615 is continuously scanning for EnOcean radio telegrams in Receive state unless it receives a request from the host to transmit a telegram.

When TCM 615 receives a valid EnOcean radio telegram, it will process it as described in chapter 4 and forward it to the host via ESP3.

When TCM 615 receives a request from the host to transmit a telegram, it will transition to Transmit state and transmit the telegram as described in chapter 5. After that, it will automatically transition back to Receive state and continue to scan for EnOcean radio telegrams.

Figure 6 below shows a typical operation sequence for transmit and receive mode with manual sleep entry and exit.

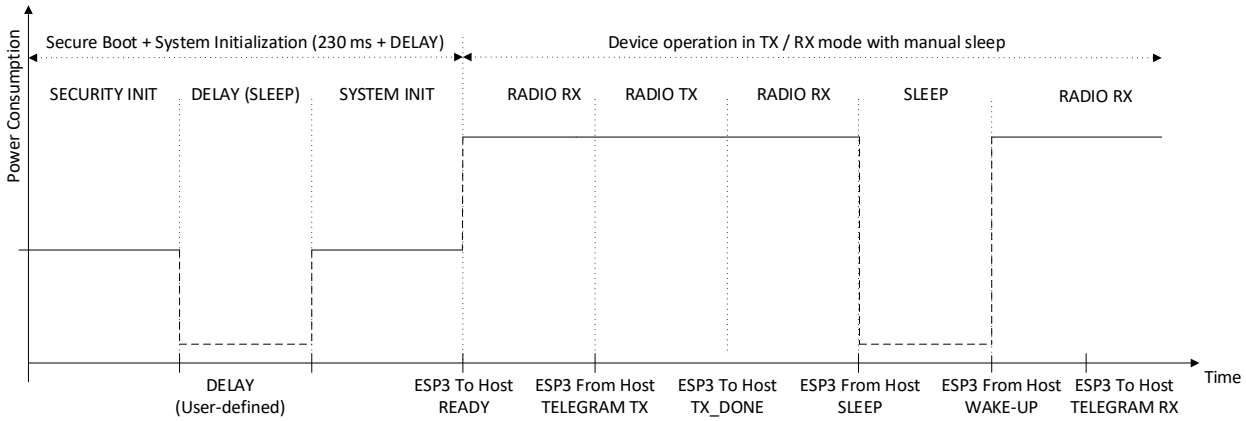

Figure 6 – Operation sequence for transmit and receive mode with manual sleep

3.2 Typical operation sequence for transmit-only mode

If TCM 615 is configured to operate in transmit-only mode, then TCM 615 will execute the same start-up sequence as for transmit and receive mode. Upon completion of the start-up sequence, TCM 615 will not enter receive mode, but wait in Idle state until an ESP3 command from the host requesting the transmission of a telegram has been received.

TCM 615 will then transmit the telegram as described in chapter 5 and inform the host once the transmission of a telegram has been completed.

After completion of the telegram transmission, TCM 615 will either transition back to Idle state waiting for the next command from the host (default configuration) or automatically enter Sleep state waiting for a wake-up via ESP3 command (Auto Sleep configuration).

Figure 7 below shows a typical operation sequence for transmit-only mode with automatic sleep entry (Auto Sleep). See chapter 5.6 for a detailed description of transmit-only mode.

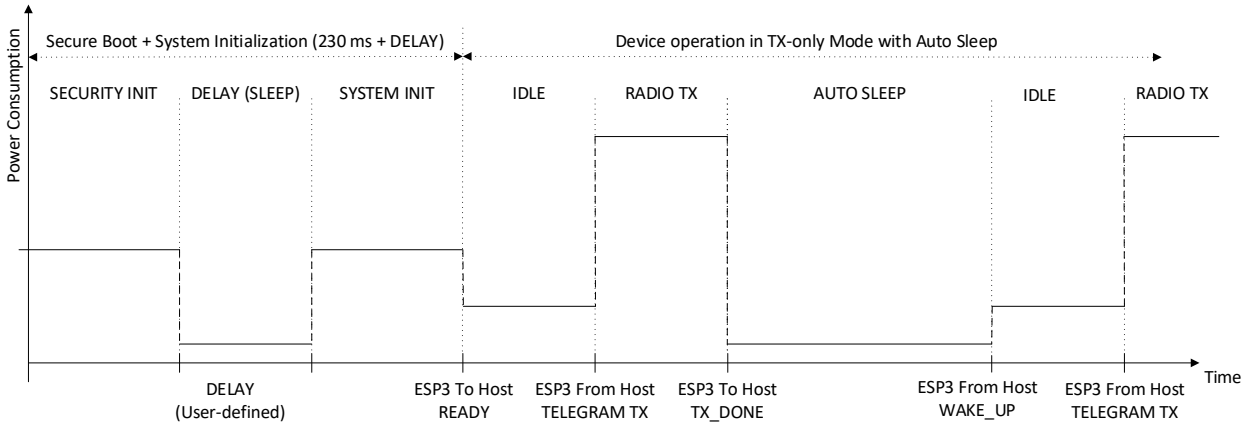

Figure 7 – Operation sequence for transmit-only mode with Auto Sleep

4 Telegram reception

After start-up, TCM 615 will enter Receive (RX) state unless Transmit-only mode is active as discussed in chapter 5.6.

4.1 Telegram reception flow

While in receive state, TCM 615 will wait for valid EnOcean radio telegrams and then perform the following functions:

- Received telegram processing
  The received data bitstream is processed and control fields such as preamble, start of frame, end of frame, inverse bits, Hash / CRC check are checked for correctness. Sub-telegrams containing incorrect data in these fields are discarded. Redundant sub-telegrams might be discarded depending on the receiver maturity time configuration as described in chapter 4.4.

- Telegram filtering
  Received telegrams can be classified according to user-defined characteristics so that only telegrams matching these characteristics will be processed and forwarded to the external host via the ESP3 interface. See chapter 4.2 for details.

- Repeater handling
  Received telegrams are checked if they should be repeated based on the repeater mode configured at TCM 615 (Level1 Repeater, Level2 Repeater, Selective Repeater) and the repeater information reported as part of the radio telegram. If the received telegram should be repeated, it will be inserted into the transmission queue.
  See chapter 5.6 for details on the repeater functionality.

- Security processing
  Telegrams from senders using high security mode can be automatically decrypted and authenticated according to their security parameters stored in the secure link table. See chapter 7 for details.

- ESP3 formatting and telegram forwarding
  Processed telegrams will be formatted as ESP3 packet (RADIO_ERP1 or RADIO_ERP2) and forwarded to the external host via the ESP3 interface. See chapter 9 for details regarding the ESP3 interface.

Figure 8 below shows the processing flow for received telegrams.

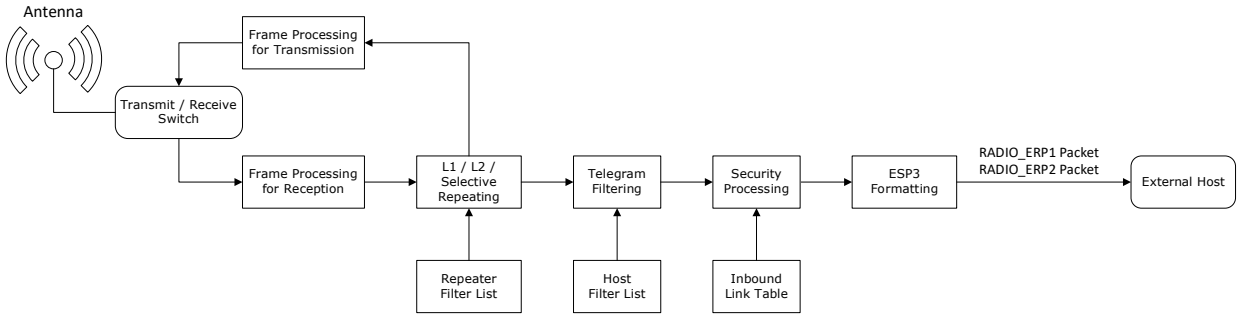

Figure 8 – Telegram Reception Flow

4.2 Telegram filtering

By default, TCM 615 will forward all valid telegrams received by it (including such that are addressed to a different receiver) to the host via its ESP3 interface. Additionally, TCM 615 will repeat these telegrams if repeating is enabled.

Filtering allows the host to configure via the ESP3 interface conditions based on which telegrams are forwarded to the host or repeated. Telegram filtering is based on the following parameters:

- Filter type
  The filter type defines based on what property TCM 615 should evaluate in received telegrams, e.g. if it should check the source address, the destination address, the telegram type or the signal strength

- Filter value
  The filter value defines the reference value against which TCM 615 will compare the property of the received telegram

- Filter condition
  The filter condition defines the desired relation between the defined filter value and the corresponding property of the received telegram.
  For the case of source address, destination address and R-ORG, the filter condition can be *Equal* (e.g. the source address of received telegram is the same as the defined filter value) or *Not Equal* (e.g. the R-ORG of the received telegram is not the same as the defined filter value).
  For the case of signal strength, the filter condition can be *Lower Than Or Equal* (the received signal strength is lower than the defined value or equal to it) or *Higher Than* (the received signal strength is higher than the defined value).

- Filter action
  The filter action defines what TCM 615 should do if the filter condition is true, e.g. if it should forward the telegram to the host or if it should forward the telegram to the host and repeat the telegram or drop the telegram completely.

- Filter combination
  The filter combination defines what happens if more than one filter condition is defined for a specific defined filter action, e.g. if the filters controlling telegram forwarding to the host should be combined in a logic OR fashion or a logic AND fashion.

The following chapters describe these parameters in more detail.

4.2.1 Filter type

TCM 615 supports the following filter types:

- Source EURID Filter
  The source EURID (EnOcean Universal Radio ID = EURID of the sender of the telegram) is evaluated.
  This filter type can for instance be used in actuators which only accept input from certain devices (e.g. switches) identified by their EURID

- Destination EURID Filter
  The destination EURID (EnOcean Universal Radio ID = EURID of the intended receiver of the telegram) is evaluated.
  This filter type can for instance be used by a receiver to not repeat radio telegrams that are directly addressed to it (and therefore do not need to be received by other devices).

- Telegram Type (R-ORG) Filter
  The telegram type of the received telegram is evaluated.
  This filter type can be used for instance be used in actuators which should react only to switch telegrams (RPS Telegram Type).

- Received signal strength (RSSI) Filter
  The received signal strength (RSSI) of the received telegram is evaluated.
  This filter type can for instance be used during learn-in if an actuator should only accept teach-in telegrams from devices close to the receiver.
  Alternatively, this filter type could also be used in repeaters so that only telegrams with weak signal strength (low RSSI value) would be repeated to limit radio congestion.

4.2.2 Filter value

The filter value field contains the value against which the corresponding property of the received telegram is compared. The filter value field is 4 byte long and – depending on the configured filter type - contains the following:

- 32-bit Source EURID (EnOcean Universal Radio ID of the sender)

- 32-bit Destination EURID (EnOcean Universal Radio ID of the intended receiver)

- 8-bit R-ORG
  The R-ORG value must be allocated in the least significant byte and the remaining 3 byte of the value field must be set to 0x000000

- 8-bit RSSI
  The RSSI value must be allocated in the least significant byte and the remaining 3 byte of the value field must be set to 0x000000. The absolute value of the desired

RSSI shall be entered, i.e. if an RSSI threshold of -80 dBm is desired, then the value 80 shall be entered

4.2.3 Filter condition

TCM 615 supports the following filter conditions for Source ID, Destination ID and R-ORG:

- Is Equal
  The value in the received telegram is the same as the defined filter value

- Is Not Equal
  The value in the received telegram is different from the defined filter value

TCM 615 supports the following filter conditions for signal strength (RSSI):

- Is Less Than Or Equal (used instead of the Is Equal condition for RSSI)
  If the defined signal strength (RSSI) value is -50 dBm then received telegrams with signal strength – 50 dBm, -51 dBm, …, -98 dBm will all match this condition. Note that TCM 615 cannot receive signals with a signal strength below the specified receiver sensitivity.

- Is Greater Than (used instead of the Is Not Equal condition for RSSI)
  If the defined signal strength (RSSI) value is -50 dBm then received telegrams with signal strength -49 dBm, -48 dBm, … -17 dBm will all match this condition. Note that TCM 615 cannot receive signals with a signal strength above the specified maximum input power.

4.2.4 Filter action

TCM 615 supports two types of filter actions:

- Filter for ESP3 Forwarding
  This filter is evaluated for the forwarding of the received telegram to the host via ESP3. If the filter condition is true, then the telegram is forwarded.

- Filter for Selective Repeating
  This filter is evaluated for the for the selective repeating functionality.
  If the filter condition is true and selective repeating is enabled as described in chapter 6.1, then the telegram is repeated. If selective repeating is not enabled, then this filter condition is not evaluated.

4.2.5 Filter combination

For each of the two actions (telegram forwarding to the host, telegram repeating) it is possible to define one or several filters.

The combination between the defined filters for the same filter action can either be a logical AND (all filter conditions must be true to execute the filter action) or a logical OR (one of the filter conditions must be true to execute the filter action). This behaviour is chosen as a parameter when enabling the filtering with the command CO_WR_FILTER_ENABLE.

For the case of selective repeating, filters with condition / action codes 0x00 and 0x40 will be ignored when evaluating the defined filters.

TCM 615 support the definition of up to 30 individual filters in total. Attempting to define more than 30 filters will result in the response 01: RET_ERROR (memory space full).

4.2.6 Filter definition

Telegram filters are defined using the CO_WR_FILTER_ADD command as shown in Table 3 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0007	7 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x0B	0x0B: CO_WR_FILTER_ADD
	7	1	Filter type	0x00…0x03	Telegram property that will be evaluated 0x00: Source EURID 0x01: Telegram type (R-ORG) 0x02: Received signal strength (RSSI, in dBm) 0x03: Destination EURID
	8	4	Filter value	0xnnnnnnnn	Value to compare against - Source EURID (4 byte) - R-ORG (1 byte) - Signal strength (1 byte, interpreted as negative of this value, e.g. 85 means -85 dBm) - Destination EURID (4 byte)
	12	1	Filter condition and action	0x00 0x80 0x40 0xC0	0x00: Forward to host if condition is false Ignore this filter for selective repeating 0x80: Forward to host if condition is true Ignore this filter for selective repeating 0x40: Repeat telegram if condition is false Ignore this filter for host forwarding 0xC0: Repeat telegram if condition is true Ignore this filter for host forwarding
-	13	1	CRC8D	0xnn	

Table 3 – Syntax for CO_WR_FILTER_ADD

Note that if the filter value is only 8 bit long (for R-ORG or RSSI filters), then the remaining three bytes of the filter value field must be set to 0x00:00:00.

4.2.7 Filter enabling

Once all filters have been defined, the CO_WR_FILTER_ENABLE command shown in Table 4 below must be used to select the logical relation between the defined filters (logical AND versus logical OR) and to enable the filtering mechanism for telegram forwarding via ESP3.

Note that the combination between the defined filters can be set independently for the host filters determining if a received telegram will be forwarded to the host via the ESP3 interface and the repeater filters determining if a received telegram will be repeated.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0003	3 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x0E	0x0E: CO_WR_FILTER_ENABLE = 14
	7	1	Forward Filter ON/OFF	0x00 0x01	0x00: Forwarding filter disabled 0x01: Forwarding filter enabled
	8	1	Filter Operator	0x00 0x01 0x08 0x09	0x00: OR connection between all filters 0x01: AND connection between all filters 0x08: OR connection between host filters AND connection between repeater filters 0x09: AND connection between host filters OR connection between repeater filters
-	9	1	CRC8D	0xnn	

Table 4 – Syntax for CO_WR_FILTER_ENABLE command

The use of the defined filters for the repeater is enabled separately by means of the CO_WR_REPEATER command shown in Table 29 in chapter 6.2. There, REP_ENABLE must be set to 0x02 to enable selective repeating based on the defined filters.

Note that if a filter is set to be ignored for the cases of repeating (filter condition / action 0x00 or 0x80), then this filter will not be evaluated and the result of the evaluation of the other filters (not set to be ignored) will not be influenced by it.

4.2.8 Filter reading

It is possible to read the currently defined filters using the CO_RD_FILTER command using the syntax shown in Table 5 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0001	1 byte
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x0F	0x0F: CO_RD_FILTER
-	7	1	CRC8D	0xnn	

Table 5 – Syntax for CO_RD_FILTER

TCM 615 will reply to the CO_RD_FILTER command with a response containing all defined filters using the syntax shown in Table 6 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0xnnnn	1 + 5*f bytes (f = number of filters)
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x02	0x02: RESPONSE
-	5	1	CRC8H	0xnn	
Data	6	1	Return Code	0x00	0x00: RET_OK
	7+5*f	1	Filter type	0xnn	Telegram property that will be evaluated 0x00: Source EURID 0x01: Telegram type (R-ORG) 0x02: Received signal strength (RSSI, in dBm) 0x03: Destination EURID
	8+5*f	4	Filter value	0xnnnnnnnn	Value to compare against - Source EURID (4 byte) - R-ORG (1 byte) - Signal strength (1 byte, interpreted as negative of this value, e.g. 85 means -85 dBm) - Destination EURID (4 byte)
-	12+5*f	1	CRC8D	0xnn	

Table 6 – Syntax of the response to CO_RD_FILTER command

4.2.9 Filter deletion

Filters can be deleted individually using the CO_WR_FILTER_DEL command with the syntax shown in Table 7 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0007	7 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x0C	0x0C: CO_WR_FILTER_DEL
	7	1	Filter type	0x00…0x03	Telegram property that will be evaluated 0x00: Source EURID 0x01: Telegram type (R-ORG) 0x02: Received signal strength (RSSI, in dBm) 0x03: Destination EURID
	8	4	Filter value	0xnnnnnnnn	Value to compare against - Source EURID (4 byte) - R-ORG (1 byte) - Signal strength (1 byte, interpreted as negative of this value, e.g. 85 means -85 dBm) - Destination EURID (4 byte)
	12	1	Filter action and condition	0x00 0x80 0x40 0xC0	0x00: Forward to host if condition is false Ignore this filter for selective repeating 0x80: Forward to host if condition is true Ignore this filter for selective repeating 0x40: Repeat telegram if condition is false Ignore this filter for host forwarding 0xC0: Repeat telegram if condition is true Ignore this filter for host forwarding
-	13	1	CRC8D	0xnn	

Table 7 – Syntax for CO_WR_FILTER_DEL command

It is possible to delete all configured filters using the CO_WR_FILTER_DEL_ALL command with the syntax shown in Table 8 below. It is strongly recommended to use this command to clear the filter table from existing entries before starting the filter table configuration.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0001	1 byte
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x0D	0x0D: CO_WR_FILTER_DEL_ALL
-	13	1	CRC8D	0xnn	

Table 8 – Syntax for CO_WR_FILTER_DEL_ALL command

4.2.10 Filter examples

4.2.10.1 Forwarding (ESP3 to host) filter examples

The examples below show common filter conditions for the telegram forwarding of received telegrams to the external host via the ESP3 interface.

// Do not forward telegrams sent from the specified ID
// All telegrams will be forwarded except those from the specified ID
Filter_type = 0x00 (Sender EURID matches specified value)
Filter_value = 0x12345678 (device source ID)
Filter_action = 0x00 (Forward to host via ESP3 if condition is false)

// Forward telegrams sent from the specified ID
// Only telegrams from the specified ID will be forwarded
Filter_type = 0x00 (Sender EURID matches specified value)
Filter_value = 0x12345678 (device source ID)
Filter_action = 0x80 (Forward to host via ESP3 if condition is true)

// Do not forward telegrams having the specified R-ORG
// All telegrams will be forwarded except those having the specified R-ORG
Filter_type = 0x01 (R-ORG matches specified value)
Filter_value = 0x000000A5 (4BS)
Filter_ action = 0x00 (Forward to host via ESP3 if condition is true)

// Forward telegrams with the specified R-ORG
// Only telegrams with the specified R-ORG will be forwarded
Filter_type = 0x01 (R-ORG matches specified value)
Filter_value = 0x00000A5 (4BS)
Filter_ action = 0x80 (Forward to host via ESP3 if condition is true)

// Do not forward telegrams with a signal strength below -70dBm (ignore weak telegrams)
// Only telegrams with a signal strength greater than -70dBm will be forwarded
Filter_type = 0x02 (RSSI is less than or equal the specified value)
Filter_value = 0x00000046 (decimal: 70)
Filter_ action = 0x00 (Forward to host via ESP3 if condition is false)

4.2.10.2 Repeater filter examples

The examples below show possible filter conditions for the telegram repeating of received telegrams (selective repeating). Note that repeating always works in conjunction with forwarding of a telegram to the host, i.e. you can not specify an individual filter to repeat a telegram but not forward it to the host.

// Repeat telegrams sent from the specified EURID (requires REP_ENABLE = 0x02)
// Telegrams sent from other senders (with different EURID) will not be repeated
Filter_type = 0x00 (Sender EURID matches specified value)
Filter_value = 0x12345678 (sender EURID)
Filter_action = 0xC0 (Forward to host via ESP3 and repeat telegram if condition is true)

// Repeat telegrams with an R-ORG other than 0xA5 (requires REP_ENABLE = 0x02)
// Telegrams with R-ORG 0xA5 will not be repeated
Filter_type = 0x01 (R-ORG matches specified value)
Filter_value = 0x000000A5 (4BS)
Filter_action = 0x40 (Forward to host via ESP3 and repeat telegram if condition is false)

// Repeat telegrams with a signal strength <= -70dBm (requires REP_ENABLE = 0x02)
// Telegrams with a signal strength above -70dBm will not be repeated
Filter_type = 0x02 (RSSI is less than or equal the specified value)
Filter_value = 0x00000046 (decimal: 70)
Filter_action = 0xC0 (Forward to host via ESP3 and repeat telegram if condition is true)

4.3 Forwarding of received telegrams to the host

TCM 615 will forward received and processed telegrams to the host using its ESP3 interface. TCM 615 provides the following options for the format in which the received telegram is provided to the external host:

- RADIO_ERP1 packet
  RADIO_ERP1 is the format used by EnOcean radio telegrams in EU (868.3 MHz).

- RADIO_ERP2 packet
  RADIO_ERP2 is the format used by EnOcean radio telegrams in US / Canada (902.875 MHz) and Japan / Australia (928.350 MHz).

4.3.1 Selection of the packet format

The packet format used to forward received radio messages to the host is selected using the ESP3 command CO_WR_MODE with the syntax shown below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	COMMON_COMMAND = 5
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x1C	CO_WR_MODE = 28
	6	1	Mode	0xnn	0x00: RADIO_ERP1 (TCM 615/U only) 0x01: RADIO_ERP2 (TCM 615U/J only)
-	7	1	CRC8D	0xnn	

Table 9 – Syntax of the CO_WR_MODE command

TCM 615 will respond to this ESP3 command a RESPONSE message using one of the following return codes:
00 RET_OK
01 RET_ERROR
02 RET_NOT_SUPPORTED (Selected mode not supported on this device)

4.3.2 RADIO_ERP1 packet format

If the telegram payload of received telegrams is forwarded to the external host using the RADIO_ERP1 packet, then the structure shown in Table 10 below is used.

For TCM 615 and TCM 615U, this is the packet format that is used by default. For TCM 615J, this packet format is not supported.

The Data field of the RADIO_ERP1 packet contains the ERP1 telegram (excluding the Hash field used for data verification).

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0xnnnn	Variable length of radio telegram
	3	1	Optional Length	0x07	7 fields fixed
	4	1	Packet Type	0x01	RADIO_ERP1 = 1
-	5	1	CRC8H	0xnn	
Data	6	x	... ...	... ...	Radio telegram without checksum/CRC x = variable length / size
Optional Data	6+x	1	SubTelNum	0xnn	Number of received sub-telegrams If "wait for maturity time" is disabled, then this field will be set to 1 (because the first sub-telegram is forwarded)
	7+x	4	Destination ID	0xnnnnnnnn	Broadcast: Broadcast ID (FF FF FF FF) ADT: Destination EURID
	11+x	1	dBm	0xnn	Highest (best) RSSI value of all received sub-telegrams. Value is expressed as positive decimal number (therefore 60 means –60 dBm)
	12+x		Security Level	0x0n	0x00: Telegram not processed by TCM 615 0x01: Obsolete (old security concept) 0x02: Telegram decrypted by TCM 615 0x03: Telegram authenticated by TCM 615 0x04: Telegram decrypted + authenticated
-	13+x	1	CRC8D	0xnn	CRC8 Data byte; calculated checksum for DATA and OPTIONAL_DATA fields

Table 10 – Syntax of the RADIO_ERP1 packet for received messages

4.3.3 RADIO_ERP2 packet format (TCM 615U and TCM 615J only)

TCM 615U and TCM 615J use EnOcean Radio Protocol 2 (ERP2) for radio communication as described in [3]. Both devices can use RADIO_ERP2 packet to provide the telegram payload of received telegrams to the external host with the packet format shown in Table 11 below.

For TCM 615J, this is the packet format that is used by default. For TCM 615U, this is an optional packet format. For TCM 615, this packet format is not supported.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0xnnnn	Variable length of radio telegram
	3	1	Optional Length	0x02	2 fields fixed
	4	1	Packet Type	0x0A	RADIO_ERP2 = 10
-	5	1	CRC8H	0xnn	
Data	6	x	Raw data	... ...	ERP2 telegram without the first Length byte
Optional Data	6+x	1	SubTelNum	0xnn	Number of received sub-telegrams If "wait for maturity time" is disabled, then this field will be set to 0 (not applicable)
	7+x	1	dBm	0xnn	Highest (best) RSSI value of all received sub-telegrams. Value is expressed as positive decimal number (60 means – 60 dBm)
	8+x	1	Security Level	0x0n	0x00: Telegram not processed by TCM 615 0x01: Obsolete (old security concept) 0x02: Telegram decrypted by TCM 615 0x03: Telegram authenticated by TCM 615 0x04: Telegram decrypted + authenticated
-	8+x	1	CRC8D	0xnn	CRC8 checksum

Table 11 – ESP3 structure for RADIO_ERP2 packet used for reception

4.4 Wait for receive maturity time

As discussed in appendix A.3.3, the receive maturity time defines the longest possible interval between the reception of the first sub-telegram and the reception of the last sub-telegram belonging to the same telegram.

All sub-telegrams from the same sender and with the same content that are received within receiver telegram maturity time are considered to represent the same original telegram. TCM 615 provides the following options to handle sub-telegrams belonging to the same original telegram:

- TCM 615 can be configured to immediately forward a received sub-telegram to the host and discard subsequent identical sub-telegrams. This provides the lowest latency and is the default operation mode for TCM 615.
- TCM 615 can be configured to wait for the receive telegram maturity time (100 ms) after reception of a sub-telegram to determine the number of received sub-telegrams. TCM 615 will in that case report the number of received sub-telegrams to the external host together with the highest signal strength of all received sub-telegrams and the repeater level of the first received sub-telegram.
- TCM 615 can be configured to forward all received sub-telegrams to the external host without doing any sub-telegram processing. In this mode (0x02), the host will receive redundant sub-telegrams and can use those to determine advanced metrics such as Repeater Level, RSSI, number of sub-telegrams and sub-telegram timing.

The selection between these three options is done using the CO_WR_WAIT_MATURITY command as shown in Table 12 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x10	0x10: CO_WR_WAIT_MATURITY
	7	1	Wait End Maturity	0xnn	0x00: The first received telegram is forwarded to the external host immediately, subsequent identical telegrams are discarded 0x01: The received telegram, the number of sub-telegrams, the highest RSSI of all telegrams and the repeater level of the first received sub-telegram are forwarded to the external host after the maturity time elapsed 0x02: Received telegrams are forwarded as they arrive. No sub-telegram merging.
-	8	1	CRC8D	0xnn	

Table 12 – CO_WR_MATURITY

4.5 Transparent Mode

Certain applications - such as gateways – implement the processing of advanced protocol features on the host.

TCM 615 therefore allows disabling certain features by selecting "Transparent Mode" so that it can be used as simple transmitter / receiver. The following features will be disabled while Transparent Mode is active:

- Telegram chaining
- Reman chaining
- Secure teach-in telegram processing
- Encryption / decryption

The following basic receive and transmit functionality will remain active while Transparent Mode is active:

- Sub-telegram merging
- Telegram repeating
- Telegram filtering (for repeating and for forwarding to the host)

Transparent Mode can be enabled and disabled using the CO_WR_TRANSPARENT_MODE command as shown in Table 13 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x3E	0x3E: CO_WR_TRANSPARENT_MODE
	7	1	Transparent Mode	0xnn	0x00: Disable Transparent Mode 0x01: Enable Transparent Mode
-	8	1	CRC8D	0xnn	

Table 13 – Syntax for CO_WR_TRANSPARENT_MODE command

4.6 RSSI test mode

TCM 615 can report the signal strength of received radio telegrams using SIGNAL telegram type 0x0A. This allows evaluation of the radio conditions without the need to physically connect to the ESP3 interface and is intended to support product qualification.

Use of RSSI Test Mode functionality is only permitted during product development and qualification. RSSI Test Mode shall not be used in production devices since it significantly increases the radio traffic. Do not permanently enable this mode.

RSSI Test Mode is enabled using an ESP3 command as shown in Table 14 below. It is strongly recommended to specify a timeout when using this command to ensure that the retransmission of all received telegrams will not be permanently active.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0004	4 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x3A	0x3A: CO_WR_RSSITESTMODE
	7	1	Enable	0x00 0x01	0x00: RSSI Test Mode Disabled 0x01: RSSI Test Mode Enabled
	8	2	Timeout (s)	0xnnnnn	0x0000: No timeout (Stop using this command) 0x0001 … 0xFFFF: Timeout (in seconds)
-	12	1	CRC8D	0xnn	

Table 14 – Syntax for CO_WR_RSSITESTMODE command

4.6.1 Response in RSSI test mode

If RSSI test mode is active, then TCM 615 will evaluate for each received telegram if it matches the filter criteria (if filter criteria have been configured).

If this is the case (or if no filter criteria have been configured), then TCM 615 will report the signal strength and the repeater level for each received telegram using a SIGNAL telegram with MID (type) 0x0A. The payload format of that telegram type is shown in Table 15 below.

Offset	Size	Content	Description
0	8	Message index	Enumeration: 0x0A: Receiver (RX) channel quality
8	32	ID	Least significant 32 bit of the EURID of the sender of the telegram for which the quality is reported
40	8	Lowest RSSI	0x00: Lowest RSSI was +127 dBm … 0x7F: Lowest RSSI was 0 dBm … 0xFE: Lowest RSSI was -127 dBm 0xFF: Lowest RSSI is unknown
48	8	Highest RSSI	0x00: Highest RSSI was +127 dBm … 0x7F: Highest RSSI was 0 dBm … 0xFE: Highest RSSI was -127 dBm 0xFF: Highest RSSI is unknown
56	4	Sub-telegram count	0b0000: Sub-telegram count unknown 0b0001: 1 sub telegram received … 0b1111: 15 or more sub telegrams received
60	4	Maximum repeater level	0b0000: No repeated telegrams received 0b0001: One-time repeated telegrams received 0b0010: Two-time repeated telegrams received 0b0011 … 0b1110: Reserved 0b1111: Maximum repeater level unknown

Table 15 – Syntax for SIGNAL 0x0A

4.7 CRC length selection (TCM 615J devices only)

EnOcean devices using ERP2 radio protocol use an 8-bit CRC value to ensure the integrity of the data payload within the radio telegram as described in Appendix A.2. For some legacy devices it has been observed that the last bit of the CRC is sensitive to disturbance and might sometimes not be correctly received.

For such cases, TCM 615J provides the option to restrict the CRC calculation to 7 bit such that the last – sometimes incorrectly received – bit is not considered. This option can be selected using the ESP3 command CO_SET_CRC_SIZE shown in Table 16 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x34	0x34: CO_SET_CRC_SIZE
	7	1	CRC Size	0xnn	0x00: 8-bit CRC (Default) 0x01: 7-bit CRC (For legacy devices)
-	8	1	CRC8D	0xnn	

Table 16 – CO_SET_CRC_SIZE command syntax

The currently active setting for the CRC size can be determined using the command CO_GET_CRC_SIZE shown in Table 17.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0x0001	1 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x35	0x35: CO_GET_CRC_SIZE
-	7	1	CRC8D	0xnn	

Table 17 – CO_GET_CRC_SIZE command syntax

The structure of the RESPONSE message from TCM 615J is shown in Table 18.Table 17

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x02	0x02: RESPONSE
-	5	1	CRC8H	0xnn	
Data	6	1	Return Code	0x00	0x00: RET_OK
	7	1	CRC Size	0xnn	0x00: 8-bit CRC (Default) 0x01: 7-bit CRC (For legacy devices)
-	8	1	CRC8D	0xnn	

Table 18 – Syntax of the RESPONSE to the CO_GET_CRC_SIZE command

5 Telegram transmission

TCM 615 will enter transmit state if it receives radio telegrams for transmission from the external host via the ESP3 interface or if repeating is enabled and a telegram is received that must be repeated based on the defined conditions.

5.1 Telegram transmission flow

TCM 615 performs the following functions to transmit radio telegrams:

- Frame decoding
  TCM 615 either receives the radio telegram data from the external host via the ESP3 interface as described in chapter 9 or from its own receiver in case repeating is enabled and a telegram is received that has to be repeated as described in chapter 5.6

- Security processing
  If TCM 615 is transmitting (not repeating) a telegram to a receiver that supports high security mode, then this telegram can be automatically encrypted and authenticated according to the parameters specified in the corresponding entry of the secure link table of TCM 615 as described in chapter 7.

- Message chaining
  EnOcean radio protocol uses very short radio messages to maximize the likelihood of successful communication. If transmission of larger data messages is required, then one such large message will be split (chained or segmented) into smaller messages for transmission.

- Telegram transmission
  Processed telegrams will be transmitted as a set of redundant sub-telegrams as described in Appendix A.3

Figure 9 below shows the process for the transmission of EnOcean radio telegrams.

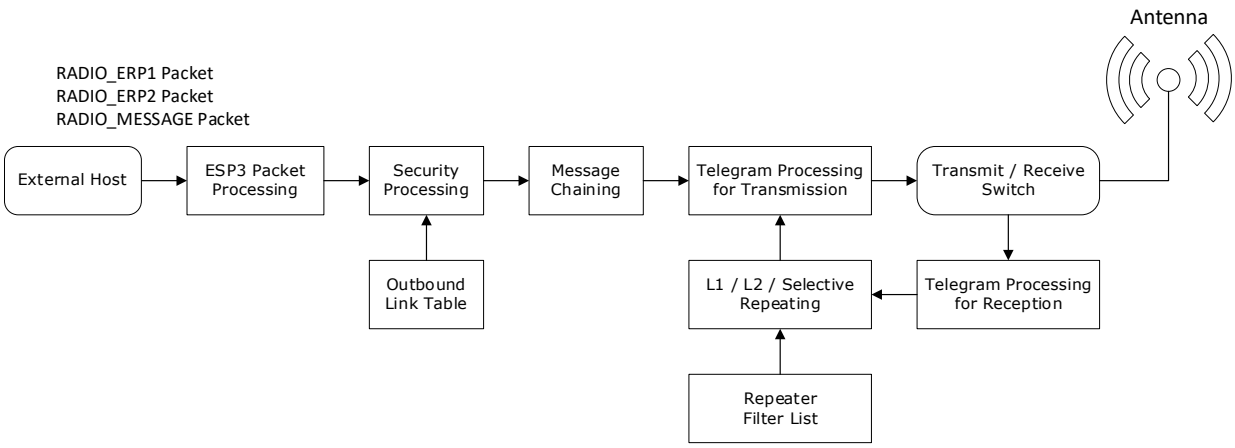

Figure 9 – Telegram Transmission Flow

5.2 ESP3 packet processing

Transmission of data telegrams can be initiated by the external host via the ESP3 interface using one of the following ESP3 packet types:

- RADIO_ERP1 packet
- RADIO_ERP2 packet (for TCM 615U or TCM 615J only)
- RADIO_MESSAGE packet

Additionally, transmission of a secure teach-in telegram can be initiated by the external host via the ESP3 interface using the CO_WR_SECUREDEVICE_SENDTEACHIN command.

5.2.1 RADIO_ERP1 packet

Telegram transmission can be initiated by the external host by sending the ESP3 packet RADIO_ERP1 to TCM 615 using the structure shown in Table 19 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0xnnnn	Length *x* of radio telegram (variable)
	3	1	Optional Length	0x07	Length of Optional Data (always 7 bytes)
	4	1	Packet Type	0x01	0x01: RADIO_ERP1
-	5	1	CRC8H	0xnn	CRC8 checksum for Header
Data	6	x	... ...	... ...	Radio telegram content (variable length *x*) Maximum length for broadcast: 14 byte Maximum length for addressed: 9 byte
Optional Data	6+x	1	SubTelNum	0x03	Number of sub-telegrams to send (3)
	7+x	4	Destination ID	0xnnnnnnnn	Broadcast: FF FF FF FF Addressed (ADT): Destination EURID
	11+x	1	dBm	0xFF	Send case: FF (not used)
	12+x	1	Security Level	0x00	Will be ignored (Security level is defined by the corresponding link table entry)
-	13+x	1	CRC8D	0xnn	CRC8 checksum for Data and Optional Data

Table 19 – ESP3 structure for RADIO_ERP1 packet used for transmission

TCM 615 will respond to the RADIO_ERP1 command immediately with the RESPONSE message 00: RET_OK if TCM 615 can transmit the message (correct format used in the command and duty cycle limit not active).

Note that the maximum payload length for RADIO_ERP1 is 14 byte for the case of a broadcast and 9 byte for the case of an addressed transmission (ADT). Attempting to send longer messages will result in the RESPONSE 0x03 (RET_WRONG_PARAM). Use RADIO_MESSAGE for the transmission of larger radio telegrams or create chained messages in the host.

If duty cycle lock is active (permissible duty cycle has been exceeded) and no transmission is possible, TCM 615 will respond with the RESPONSE message 05: RET_LOCK_SET.
See chapter 5.5 for a description of the duty cycle limit functionality.

TCM 615 will send an EVENT with code 0x08: CO_TX_DONE to the host as soon as the requested telegram transmission has been completed.

Note that the transmission of the three sub-telegrams will last for up to 40 ms after receiving the RET_OK message. Do not shut-down TCM 615 before this period has elapsed or the CO_TX_DONE event has been received.

5.2.2 RADIO_ERP2 packet for telegram transmission (TCM 615U and TCM 615J only)

TCM 615U and TCM 615J communicate using EnOcean Radio Protocol 2 (ERP2 as described in [3] which implements a different telegram format compared to EnOcean Radio Protocol 1.

Radio transmission can be initiated on TCM 615U or TCM 615J using the RADIO_ERP2 packet with the structure shown in Table 20 below. Attempting to use the RADIO_ERP2 packet with TCM 615 will result in RESPONSE 02: RET_NOT_SUPPORTED.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0xnnnn	Variable length of radio telegram
	3	1	Optional Length	0x03	3 bytes of Optional Data
	4	1	Packet Type	0x0A	RADIO_ERP2 = 10
-	5	1	CRC8H	0xnn	
Data	6	x	Raw data	... ...	ERP2 radio protocol telegram without the first Length byte. The ERP2 CRC8 byte can be set to any value.
Optional Data	6+x	1	SubTelNum	0xnn	Number of sub telegrams Set to 0x03 (3 sub-telegrams)
	7+x	1	dBm	0xnn	Set to 0xFF
	8+x	1	Security Level	0x0n	Will be ignored (Security is selected by link table entries)
-	8+x	1	CRC8D	0xnn	CRC8 Data byte; calculated checksum for DATA and OPTIONAL_DATA

Table 20 – ESP3 structure for RADIO_ERP2 packet used for transmission

TCM 615U and TCM 615J will respond to the RADIO_ERP2 packet immediately with the RESPONSE message 00: RET_OK if TCM 615U or TCM 615J can transmit the message (correct format used in the command).

TCM 615 will additionally send an event with code 0x08: CO_TX_DONE to the host as soon as the transmission of the telegram has been completed.

Note that the transmission of the three sub-telegrams will last for up to 40 ms after receiving the RET_OK RESPONSE.
Do not shut-down TCM 615 before this period has elapsed or the CO_TX_DONE EVENT has been received.

To maximize ESP3 compatibility between the different variants, TCM 615U and TCM 615J accept also RADIO_ERP1 packets from the remote host for the transmission of radio telegrams.

5.2.3 RADIO_MESSAGE packet for telegram transmission

TCM 615, TCM 615U and TCM 615J all support RADIO_MESSAGE packets which provides a unified way to transmit telegrams with the same format for all TCM 615 variants and all supported payload lengths.

The structure of the RADIO_MESSAGE packet is shown in Table 21 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0xnnnn	Variable length of message
	3	1	Optional Length	0x0A	Optional Data = 10 bytes
	4	1	Packet Type	0x09	RADIO_MESSAGE = 9
-	5	1	CRC8H	0xnn	
Data	6	1	Message R-ORG	0xnn	R-ORG
Data	7	x	Message Data	...	Message Data Content
Optional Data	7+x	4	Destination ID	0xnnnnnnnn	Destination ID Broadcast ID: FF FF FF FF
	11+x	4	Source ID	0xnnnnnnnn	Set to 0x00000000 for transmission
	15+x	1	dBm	0xnn	Set to 0xFF for transmission
	16+x	1	Security Level	0x0n	Ignored for transmission (Security is selected by link table entries)
-	17+x	1	CRC8D	0xnn	CRC8 Data byte; calculated checksum for DATA and OPTIONAL_DATA

Table 21 – ESP3 structure for RADIO_MESSAGE packet used for transmission

TCM 615 will respond to the RADIO_MESSAGE packet immediately with the RESPONSE 00: RET_OK if it can transmit the message (correct format used in the command).

If duty cycle lock is active (permissible duty cycle has been exceeded) and no transmission is possible then TCM 615 will respond with the RESPONSE 05: RET_LOCK_SET.
See chapter 5.5 for a description of the duty cycle limit functionality.

TCM 615 will send an EVENT 0x08: CO_TX_DONE to the host as soon as the requested telegram transmission has been completed.

Note that the transmission of the three sub-telegrams will last for up to 40 ms after receiving the RET_OK EVENT as described in appendix 0.
Do not shut-down TCM 615 before this period has elapsed or the CO_TX_DONE EVENT has been received.

5.3 Message Chaining

EnOcean Radio Protocol 1 (ERP1) and EnOcean Radio Protocol 2 (ERP2) communicate using very short telegrams to maximize communication robustness and reliability.

The maximum payload size of individual telegram is limited as follows:

- TCM 615
  The payload size (excluding R-ORG) is limited to 14 bytes
- TCM 615U
  The payload size (excluding HEADER, EXTENDED HEADER and EXTENDED TYPE) is limited to 14 bytes
- TCM 615J
  The payload size (excluding HEADER, EXTENDED HEADER and EXTENDED TYPE) is limited to 34 bytes. The higher maximum payload size addresses the specific radio duty cycle regulation in Japan.

If a telegram shall be transmitted where the telegram payload exceeds the maximum payload size, then the telegram payload will be segmented (distributed) over several telegrams. This mechanism is called telegram chaining.

If telegram chaining is used, then the receiver needs to reassemble the payload from the different telegrams that were used for transmission. This reassembly process is called de-chaining.

TCM 615, TCM 615U and TCM 615J support telegram chaining for ESP3 transmission using the RADIO_MESSAGE packet.

TCM 615U and TCM 615J additionally support telegram chaining using the RADIO_ERP2 packet.

TCM 615 / TCM 615U / TCM 615J will send one telegram chain at a time, i.e. the transmission of the second telegram chain can only start once transmission of the first telegram chain has been completed. It is not possible to transmit two or more telegram chains at the same time (interleaved).

For reliability reasons, it is not recommended to transmit very large telegrams. Payload sizes of more than 128 byte shall not be used.

Refer to Appendix A.6 for a description of telegram chaining.

5.4 Using Base ID for transmission

As described in Appendix A.4.4, the use of Base ID allows TCM 615 modules to transmit messages using an ID different from its own EURID. Base ID is a legacy feature supported by TCM 615 for backwards compatibility and should not be used in new designs.

The use of Base ID is not supported for secure transmission, remote management (Reman) and RSSI Test Mode.

The Base ID Range (128 addresses) of a device can be allocated anywhere in between `0xFF80:0000` and `0xFFFF:FFFE` (which represents a total range of approximately 8 million addresses).

The location of the Base ID Range is defined by the start (lowest) address of the range which must be aligned on a 7-bit (128) boundary, i.e. the last byte of the start address must be either `0x00` or `0x80`. T
The start address of the Base ID range can be set using the ESP3 command CO_WR_IDBASE shown in Table 22 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0005	5 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x07	0x07: CO_WR_IDBASE
	7	4	Base ID	0xFFnnnnnn	Start address of Base ID range (Between 0xFF800000 and 0xFFFFFF80)
-	11	1	CRC8D	0xnn	

Table 22 – CO_WR_IDBASE

Alignment of the start address to a 7-bit (128) boundary is automatically enforced within TCM 615, i.e. if a non-aligned address is provided in the ESP3 command, then TCM 615 will use the next lower, correctly aligned address as start address of the Base ID range.

The currently configured start address can be determined using the ESP3 command CO_RD_IDBASE command as shown in Table 23 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0x0001	1 byte
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	COMMON_COMMAND = 5
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x08	CO_RD_IDBASE = 8
-	7	1	CRC8D	0xnn	

Table 23 – CO_RD_IDBASE

TCM 615 will respond using the syntax shown in Table 24**Fehler! Verweisquelle konnte nicht gefunden werden.** below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0x0005	5 bytes
	3	1	Optional Length	0x01	1 byte
	4	1	Packet Type	0x02	RESPONSE = 2
-	5	1	CRC8H	0xnn	
Data	6	1	Return Code	0x00	RET_OK = 0
	7	4	Base ID	0xFFnnnnnn	Start address of Base ID range (Between 0xFF800000 and 0xFFFFFF80)
Optional Data	8	1	Remaining write cycles for Base ID	0xnn	FF: No limit for changing Base ID
-	9	1	CRC8D	0xnn	

Table 24 – Response to CO_RD_IDBASE

5.4.1 Source address selection

As describe above, TCM 615 can transmit radio telegrams using a source address that is either its own EnOcean Universal Radio ID (EURID) or a customer-assigned address from within the Base ID range setup as described above.

The selection between the source address options is made based on the source address specified within the ESP3 telegram transmission command (RADIO_ERP1, RADIO_ERP2 or RADIO_MESSAGE) as follows:

- If the provided source address is between 00:00:00:00 and FF:F7:FF:FF (below the Base ID range) then TCM 615 will transmit the telegram using its own EURID as source address.
  Any address within this range can be specified and will be replaced by TCM 615 with its own EURID as source address for transmissions. Using 00:00:00:00 as source address is recommended for better SW readability.

- If the provided source address is within the defined Base ID range (i.e. within the 128 addresses from the start of the Base ID range as defined by CO_WR_IDBASE), then TCM 615 will transmit the telegram using the Base ID as source address

- Otherwise, TCM 615 will not transmit the telegram and return RET_ERROR

5.4.2 Usage recommendation

Base ID are user-selected from within a limited address range. Base ID therefore cannot be guaranteed to be unique; especially in larger installations there is a high likelihood that two devices might use the same Base ID.

5.5 Duty cycle limit (TCM 615 / 868.300 MHz variant only)

European radio regulation mandates that the duty cycle limits of radio transmitters must be enforced by technical means. To comply with this requirement, TCM 615 (868.3 MHz for EU) implements a duty cycle monitor which enforces the regulatory duty cycle limit of 1% per hour.

This duty cycle limit applies to any type of transmission from a device; for TCM 615, the duty cycle limit therefore applies to the combination of transmissions requested by the host and to transmissions due to the repeater functionality as described in chapter 6. The duty cycle limit therefore must be considered when configuring repeater functionality. Use of selective repeating is strongly recommended.

If transmission of a telegram is requested by the host and the duty cycle limit has been reached, then TCM 615 will respond with the event CO_DUTYCYCLE_LIMIT using the format shown in Table 25 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x04	EVENT = 4
-	5	1	CRC8H	0xnn	
Data	6	1	Event Code	0x06	CO_DUTYCYCLE_LIMIT = 6
	7	1	Event Cause	0xnn	00: Duty cycle limit not yet reached It is possible to send telegrams 01: Duty cycle limit reached It is not possible to send telegrams 02…FF Reserved
-	8	1	CRC8D	0xnn	

Table 25 – CO_RD_DUTYCYCLE_LIMIT

5.5.1 Duty cycle monitor functionality

The duty cycle monitor is implemented as follows:

- Each 1-hour (3600 second) period is sub-divided into 10 time slots of 360 seconds each and during each time slot, the used transmission time is accumulated.
- The total used transmission time during the last hour is calculated as the sum of the transmission time of the last 10 time slots
- The total available transmission time within a one 1-hour period is 36 seconds (1% of 3600 s) and the remaining available transmission time is calculated as difference between 36 seconds and the total used transmission time during the last 10 time slots. This difference is the available transmission time in the current time slot.
- If the available transmission time reaches zero (no more transmission time available) then TCM 615 will not transmit any additional messages during this time interval. TCM 615 will respond with RET_LOCK_SET to the host if this requests transmission of additional telegrams in this case.
- After the current time slot elapses, the used transmission time of this time slot is added as first entry to the list, the last entry (the oldest time slot) is deleted from the list and the available transmission time for the next time slot is calculated.

Figure 10 below illustrates the duty cycle mechanism used in TCM 615.

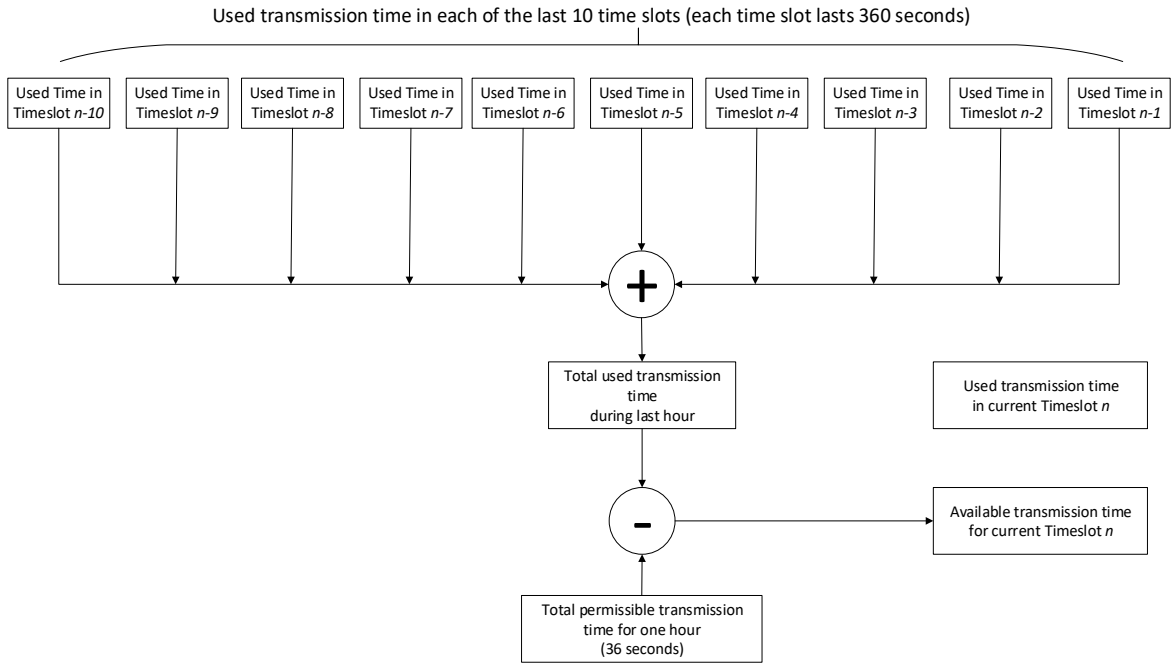

Figure 10 – Duty cycle monitor implementation

5.5.2 Determining available transmission time

The host can query the duty cycle status (available transmission time) using the ESP3 command CO_RD_DUTYCYCLE_LIMIT as shown in Table 26 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0001	1 byte
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x23	0x23: CO_RD_DUTYCYCLE_LIMIT
-	-	1	CRC8D	0xnn	

Table 26 – CO_RD_DUTYCYCLE_LIMIT

The response from TCM 615 will specify the following information:

- The percentage of currently available transmission time
  This is expressed in percent (0% … 100%) of the total available transmission time within 1 hour (which in EU is 1% duty cycle per hour = 36 seconds per hour).
  - 100% means that the entire permitted transmission time (36 s) is available
  - 0% means that no transmission time is currently available

- The number of time slots and the duration of one time slot
  TCM 615 uses 10 time slots and a duration of 360 seconds per time slot.

- The time (in seconds) until the next time slot starts

- The percentage of available transmission time once the next time slot starts
  This is expressed in percent (0% … 100%) as described above

Table 27 shows the structure of the response.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0008	8 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x02	0x02: RESPONSE
-	5	1	CRC8H	0xnn	
Data	6	1	Return Code	0x00	0x00: RET_OK
	7	1	Available duty cycle	0..0x64	Currently available transmission time (0% … 100% of permitted transmission time)
	8	1	Slots	0xnn	Total number of duty cycle slots
	9	2	Slot period	0xnnnn	Period of one slot (in seconds)
	11	2	Current slot left	0xnnnn	Remaining time until the beginning of the next duty cycle slot (in seconds)
	13	1	Load after current	0..0x64	Available transmission time (0 ...100% of permitted transmission time) once the next duty cycle slot starts
-	14	1	CRC8D	0xnn	

Table 27 – Response to CO_RD_DUTYCYCLE_LIMIT

5.6 Transmit-only mode

As described in chapter 2.2, TCM 615 is in receive state whenever it is not transmitting a telegram or has not been put into Sleep state. In some applications such as simple button or sensor transmitters, TCM 615 is used only for transmission. In these cases, the additional power consumption in receive state or the added complexity of putting TCM 615 into Sleep state after each telegram transmission might not be desired.

Reception functionality can be disabled using the ESP3 command CO_WR_TX_ONLY_MODE so that TCM 615 operates as transmit-only device. Table 28 below shows the syntax of the CO_WR_TX_ONLY_MODE command.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x40	0x40: CO_WR_TX_ONLY_MODE
	7	1	Enable	0x00 0x01 0x02	0x00: Receive / Transmit Mode Default setting 0x01: Transmit-only Mode Auto Sleep disabled 0x02: Transmit -only Mode Auto Sleep enabled
-	8	1	CRC8D	0xnn	

Table 28 – Syntax for CO_WR_TX_ONLY_MODE command

If Transmit-only mode is active and Auto Sleep is disabled, then TCM 615 will transition into Idle state after completion of a transmission where it will be waiting for reception of the next ESP3 command requesting the transmission of a telegram. Once such command is received, TCM 615 will transmit the telegram, report the successful completion of a telegram transmission using the CO_TX_DONE EVENT and then transition back to Idle state waiting for the next ESP3 command.

If Transmit-only mode is active and Auto Sleep is enabled, then TCM 615 will transition into indefinite Sleep state after completion of a transmission. In this configuration, TCM 615 will enter Sleep state in the same way as if it had received a CO_WR_SLEEP command with parameter 0x0000. TCM 615 will remain in Sleep state until it is woken up again via an ESP3 command.

Please refer to chapter 8 for a detailed description of Sleep state.

6 Telegram repeating

TCM 615 can act as repeater for all or selected radio telegrams. The repeating functionality is configured via ESP3 interface. Note that repeating functionality is not available if TCM 615 is configured to operate in Transmit-only mode as described in chapter 5.6.

If TCM 615 is configured to act as repeater and it receives a radio telegram that it is configured to repeat, then TCM 615 will automatically transition from receive to transmit state to re-transmit (repeat) this telegram. After successful transmission, it will automatically transition back to receive mode.

TCM 615 and TCM 615U provide the option to activate a one or two-level repeater for received EnOcean radio telegrams. TCM 615J supports only one-level repeater due to Japanese radio regulations.

- One-level repeater: If a received telegram is a valid and original (not yet repeated), then the telegram is repeated after a random delay.
- Two-level repeater: If a received telegram is valid and original or repeated once, then the telegram is repeated after a random delay.

Repeated telegrams are marked as "repeated" by an increased repeater counter. Configuration of the repeater functionality is done via serial interface commands.

When using TCM 615 (EU version) as repeater, the regulatory transmitter duty cycle limit (1% per hour) needs to be considered.
The 1% duty cycle limit applies to the total amount of telegrams transmitted by a device and therefore includes telegrams that are repeated by a device.
If TCM 615 is configured as repeater and many telegrams are repeated, then the available duty cycle might be consumed and TCM 615 might therefore have to temporarily stop transmission and repeating.
Use of selective repeating is strongly suggested.

Two-level repeating function should only be activated after careful study of the radio conditions! Otherwise, the system function can be compromised due to high radio traffic and the resulting telegram collisions.

Note that TCM 615J supports only one-level repeating due to Japanese radio regulations.

For detailed recommendations regarding the usage of repeaters, please refer to our application note: Range Planning Guide for Systems using EnOcean Radio Standard.

6.1 Selective repeating

TCM 615 provides the option of selective repeating, i.e. the option to only repeat certain telegrams which match pre-defined filter criteria. The filter criteria that can be applied for repeating are the same as the ones for telegram reception, see chapter 4.2.

The repeater configuration (no repeating, one-level repeating, two-level repeating, selective repeating) is stored persistently in non-volatile memory and will therefore not be affected by a power cycle. This mechanism enables the option of configuring USB stick repeaters on a PC via the ESP3 interface and then transferring them to a USB power supply for subsequent operation.

6.2 Configuration of telegram repeating

The telegram repeating functionality of TCM 615 is configured using the ESP3 command CO_WR_REPEATER as shown in Table 29 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0003	3 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x09	0x09: CO_WR_REPEATER
	7	1	REP_ENABLE	0x00…0x02	0x00: No repeating 0x01: Repeating of all telegrams 0x02: Selective repeating
	8	1	REP_LEVEL	0x00…0x02	0x00: No repeating 0x01: One-level repeating 0x02: Two-level repeating
-	9	1	CRC8D	0xnn	

Table 29 – CO_WR_REPEATER

7 Security processing

TCM 615 implements the security handling functions as specified in the EnOcean Alliance Security Specification: https://www.enocean-alliance.org/sec/.

TCM 615 can process secure messages from the following EnOcean products (note that the sender must use the same radio frequency as the TCM 615 receiver):

- PTM 210 (from revision DC)
- PTM 215 / PTM 215U / PTM 215J
- PTM 535 / PTM 535U / PTM 535J
- STM 320 / STM 329 / STM 320U / STM 429J / EMCSA / EMCSU / EMCSJ
- STM 330 / STM 331 / STM 332U / STM 333U / STM 431J
- STM 350 / STM 350U / ETHS / ETHSU
- STM 550 / STM 550U / STM 550J / EMSIA / EMSIU / EMSIJ
- EMDCA / EMDCU / EMDCJ
- TCM 515 / TCM 515U / TCM 515J
- TCM 615 / TCM 615U / TCM 615J

7.1 Security architecture

TCM 615 supports the security mechanisms defined in the EnOcean Alliance Security Specification and can manage secure, bi-directional communication with remote devices using its secure link table.

For each remote device, TCM 615 maintains separate security keys and rolling codes to transmit telegrams to the remote device (using its own LOCAL_KEY and LOCAL_RLC) and reception of telegrams from the remote device (using REMOTE_KEY and REMOTE_RLC from the remote device). Figure 11 below illustrates the two different directions of secure communication from the perspective of TCM 615.

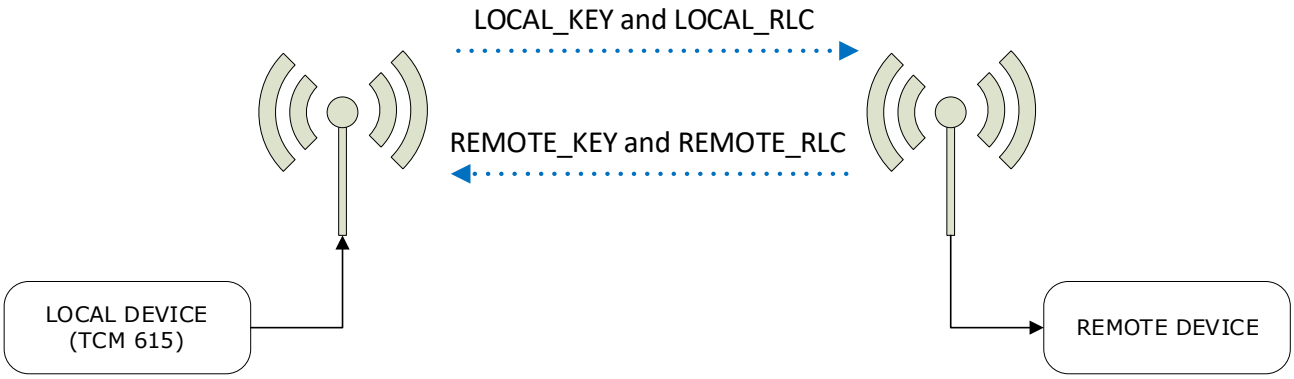

Figure 11 – Secure communication flow

7.2 Security functionality

7.2.1 Telegram encryption and decryption

TCM 615 uses the AES-128 algorithm together with a 16-byte security key and an RLC to encrypt and decrypt radio telegrams as described in chapter B.2. using the VAES mode of the AES-128 algorithm.

Refer to the EnOcean Alliance Security Specification for details about the VAES and AES-CBC modes.

7.2.2 Telegram authentication

TCM 615 can authenticate the content of received telegrams based on the telegram signature (CMAC) in conjunction with the security key and the rolling code of the remote device using the process described in chapter B.3.

For transmitted telegrams, TCM 615 can calculate the signature and add it to the telegrams according to the same mechanism.

TCM 615 supports signature lengths of 3 byte and 4 byte. The signature length is defined by the CMAC_SIZE field in the Security Level Format (SLF) as described in chapter 7.4.3.

7.3 Secure telegram processing flow

TCM 615 can automatically decrypt and authenticate received telegrams or encrypt and sign (authenticate) transmitted telegrams according to the EnOcean Alliance Security Specification.

Security processing requires TCM 615 to know the security parameters (security key, expected rolling code counter value, security configuration) that shall be used. Security processing is therefore only possible for devices that have previously been teached-in as discussed in chapter 7.7.

7.3.1 Security processing of received telegrams

Figure 12 below illustrates the high-level processing flow for received EnOcean secure radio telegrams.

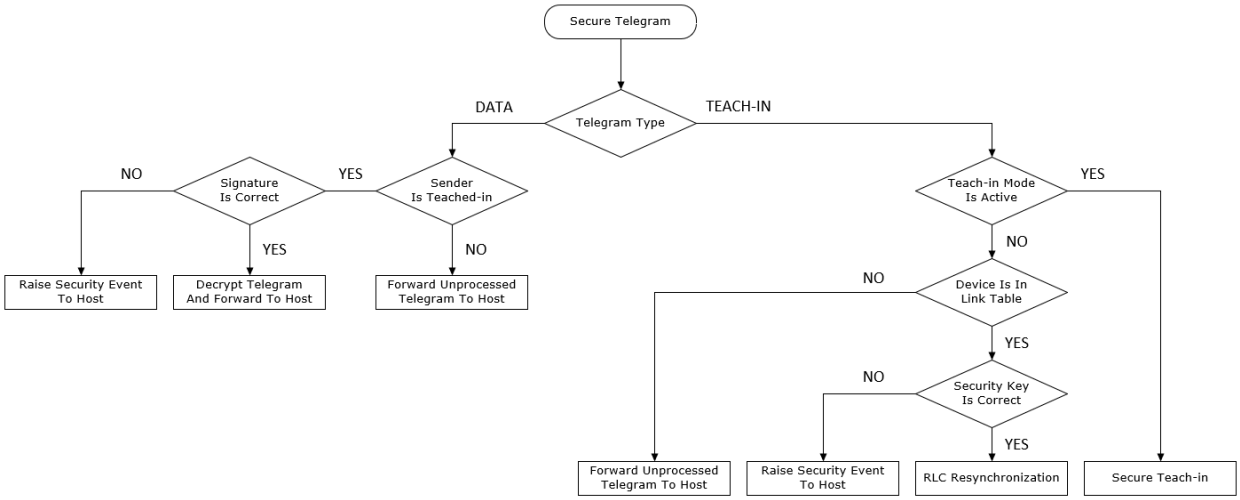

Figure 12 – TCM 615 security processing flow

Note that if a secure radio telegram is received from a device that has not been teached-in, then TCM 615 will forward this secure radio telegram to the host without security processing.

7.3.2 Security processing of transmitted telegrams

If TCM 615 is transmitting a telegram, then it will check if its secure link table contains and entry corresponding to the destination address of the telegram (which is either the Broadcast address or a device-specific address).

If such entry exists, then TCM 615 will automatically encrypt and authenticate (sign) the telegram. Otherwise, TCM 615 will not execute any security processing on this telegram.

7.3.3 Processing of secure chained messages

If security processing for a received or transmitted telegram is required, then TCM 615 must process the entire telegram for authentication and encryption / decryption. This is especially important for the case of reception of chained messages (which are transmitted using several telegrams to enable larger payload).

For the case of reception of chained messages, TCM 615 must receive and de-chain (re-assemble) all telegrams of the chained message. Once all telegrams have been received and the message has been re-assembled, TCM 615 will decrypt and authenticate the message payload before forwarding it to the host.

Due to the potentially large size of such chained messages (which can contain up to 128 byte of payload), TCM 615 can only de-chain (re-assemble) one such chain at a time. Should telegrams of a second chained message be received while TCM 615 is already receiving another chained message, then the parts of the second chained message will be forwarded to the host for processing.

The use of secure chained messages is therefore not recommended.

7.4 Security parameters

The following security parameters are used to define secure communication between a sender and a receiver:

- Security key
- RLC size and current value
- Signature (CMAC) size
- Security algorithm

Those parameters are described in the subsequent chapters and have to be setup by means of a secure teach-in procedure as described in chapter 7.7.1.

7.4.1 Security key

The security key is a random 128-bit (16 byte) value that is known only to the sender and the receiver(s). It is used to encrypt, decrypt and authenticate telegrams.

For the case of transmission, TCM 615 defines the security key that will be used to secure communication. It must be generated by the external host using a suitable random number generation algorithm.

For the case of reception, the external sender defines the security key that will be used to secure communication.

7.4.2 Rolling code (RLC)

The RLC is a monotonously incrementing counter used to modify the content of secure telegrams as described in chapter B.4. The RLC is generated by the sender and monitored by the receiver. For bi-directional communication, TCM 615 monitors two RLC values:

- Local RLC (RLC_L)
  The local RLC is used by TCM 615 for the transmission of telegrams to a specific remote device (addressed telegram) or to all devices (broadcast telegram). TCM 615 will increment RLC_L for each telegram that it transmits to the remote device.

- Remote RLC (RLC_R)
  The remote RLC is used by the remote device for telegram transmission to TCM 615. TCM 615 maintains the most recently received RLC value for each remote device and will only accept telegrams from that remote device with a higher RLC values to avoid the retransmission of a previously transmitted telegram.

7.4.3 Security level format (SLF)

The security level format (SLF) defines the security parameters used for communication between two devices.

If the communication is bi-directional (send telegrams from the local device to a remote device, receive telegrams from a remote device by the local device), then the same SLF setting must be used in both directions.

Figure 13 below shows the supported security parameter options of the SLF field.

7	6	5	4	3	2	1	0
RLC_MODE			**CMAC_SIZE**		**ENCRYPTION_ALGO**		
0b000: No RLC used 0b001: RFU 0b010: 16 bit RLC (not transmitted) 0b011: Deprecated 0b100: Deprecated 0b101: 24 bit RLC (24 bit transmitted) 0b110: 32 bit RLC (24 bit transmitted) 0b111: 32 bit RLC (32 bit transmitted)			0b00: No MAC provided 0b01: 24 bit CMAC 0b10: 32 bit CMAC 0b11: RFU		0b000: Data not encrypted 0b011: VAES using AES128 0b100: CCM using AES128 Others: RFU		

Figure 13 – SLF structure

Table 30 below provides a list of common security settings together with the resulting SLF values and examples of products using those.

Description	SLF Value	Product Example
16 bit implicit RLC with 24 bit CMAC	0x4B	PTM 535
24 bit RLC with 24 bit CMAC	0xAB	PTM 215, STM 32x, STM 33x, STM 35x
32 bit RLC with 32 bit CMAC	0xF3	STM 550, EMDC

Table 30 – Common SLF values

7.4.4 Teach-in Info (TI)

The Teach-in Info field is transmitted as part of the secure teach-in telegram. It specifies the following parameters of the secure teach-in process:

- Segmentation information
  The length of a secure teach-in telegram exceeds the maximum size of an ERP1 or ERP2 telegram (except for the special case TCM 615J / Japan). It is therefore required to segment (chain) this telegram. The IDX and CNT fields provide the required information for that.

- PSK usage
  The secure teach-in telegram contains the security key used by the device for securing the telegrams that it transmits. This information is typically sent only once during the initial setup, but it might still be required to encrypt this using a pre-shared key (PSK).

- Device type
  For the case of switches, it is possible to specify if the left or right rocker of a dual-rocker device was used to trigger the transmission of the secure teach-in telegram. This allows learning in the different (left and right) sides of a switch to different remote device such that for instance two separate light can be controlled by the same switch (the first one with the left rocker and the other one using the right rocker).

- Teach-in type
  The teach-in telegram indicates if a response is expected (bi-directional teach-in) or not (uni-directional teach-in).

Figure 14 below shows the structure of the Teach-in Info field.

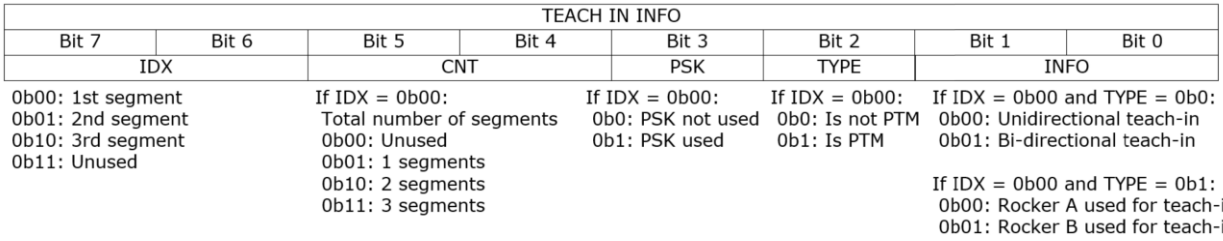

Figure 14 – Teach-in Info structure

7.5 Secure link table

TCM 615 stores all required information for secure communication with remote devices in the integrated secure link table; the use of a dedicated external memory is not required.

The secure link table can store up to 32 entries to manage secure communication with to up to 32 remote devices. For communication with more than 32 devices, it is required to execute the security processing in the external host system.

Each entry in the table specifies the key and the rolling code used by the local device (KEY_L, RLC_L) and by the remote device (KEY_R, RLC_R) for the transmission of secure telegrams. TCM 615 will therefore use KEY_L and RLC_L for transmission of telegrams to the remote device and KEY_R and RLC_R for reception of telegrams from the remote device.

Note that TCM 615 requires approximately 100 milliseconds to process a secure link table update request (addition, removal or modification of a link table entry) which has been received via ESP3. Host SW must provide a sufficient interval between ESP3 update request and any ESP3 command using the updated link table entry.

Figure 15 below shows the structure of the secure link table.

Secure Link Table Structure						
Index	**Remote Device EURID**	**Originator (TCM 615 Direction)**	**Security Key**	**RLC**	**Teach-In Info**	**Security Format**
0	EURID0	From Local (Transmit)	KEY0_L	RLC0_L	TI0	SLF0
		From Remote (Receive)	KEY0_R	RLC0_R		
1	EURID1	From Local (Transmit)	KEY1_L	RLC1_L	TI1	SLF1
		From Remote (Receive)	KEY1_R	RLC1_R		
2	EURID2	From Local (Transmit)	KEY2_L	RLC2_L	TI2	SLF2
		From Remote (Receive)	KEY2_R	RLC2_R		
…						
n	EURID*n*	From Local (Transmit)	KEY*n*_L	RLC*n*_L	TI*n*	SLF*n*
		From Remote (Receive)	KEY*n*_R	RLC*n*_R		

Figure 15 – Secure link table structure

7.5.1 Secure link table parameters

Each entry in the secure link table contains the following parameters:

- Index
  The index indicates the location of the entry in the secure link table. The table will be filled starting with Index = 0 and is full once Index = 31

- Remote Device EURID
  This field contains the EURID (radio address) of the remote device with which TCM 615 can communicate based on the parameters for this entry

- Security Key
  This field contains the security key used by TCM 615 to transmit telegrams to the remote device (KEY_O) and the security key used by the remote device to transmit telegrams to TCM 615 (KEY_I)

- RLC
  This field contains the RLC used by the local device (TCM 615) to transmit telegrams to the remote device (RLC_L) and the RLC used by the remote device to transmit telegrams to TCM 615 (RLC_R)

- Teach-in Info
  This field contains information about the type of the remote device (specifically if this is a rocker switch or not and if A or B side of the rocker switch were used for teach-in)

- Security Level (SLF)
  This field contains the security level (SLF) which specifies the encryption, authentication and RLC parameters used for the communication with the remote device as described below. For bi-directional communication, the same SLF must be used by the remote device (telegrams transmitted by the remote device to TCM 615) and by the local device (telegrams transmitted by TCM 615 to the remote device).

The security processing in TCM 615 supports both secure messages that specify the original telegram type (R-ORG) and those who don't. Table 31 below summarizes the different R-ORG values and types that are supported by TCM 615 security processing.

R-ORG	Description
0x30 (SEC)	Secure RPS message that does not identify the R-ORG of the encrypted telegram Used by switches only.
0x31 (SEC_R)	Secure message that identifies the type R-ORG of the encrypted telegram Used by all devices except switches.
0x32 (SEC_D)	Message that results from the decryption of a secure RPS message (R-ORG 0x30) Used by switches only.
0x33 (SEC_CDM)	Secure Chained Messages (Sequence of several messages to encode longer data)
0x35 (SEC_TI)	Secure Teach-in telegram (Used to setup secure communication)

Table 31 – Secure R-ORG supported by TCM 615

7.6 RLC support

TCM 615 supports the use of RLC generated by a monotonously incrementing sequence counter as described in chapter B.4. TCM 615 supports both explicit RLC mode and implicit RLC mode as described in chapter B.4.2.

TCM 615 supports RLC sizes of 16 bit (implicit) 24-bit and 32-bit according to the setting of the RLC_MODE field in the SLF as described in chapter 7.4.3; use of 32-bit RLC is suggested. Use of 16-bit RLC is only permitted for the reception of switch telegrams.

7.6.1 Explicit and implicit rolling code support

The maximum number of RLC values that will be tested in implicit RLC mode (the RLC Window size) is 128 in TCM 615. If the RLC window has been exhausted without successfully decrypting and authenticating the telegram, then the telegram will be discarded.

The RLC window size can be temporarily changed (increased to attempt resynchronization) using ESP3 Command Code 33: CO_WR_TEMPORARY_RLC_WINDOW. After the reception of this command, the increased RLC window will be applied to the first telegram received from a remote device that is setup in the secure link table and uses implicit RLC. Refer to the ESP3 documentation for reference.

If resynchronization of the sequence counter between transmitter and receiver fails, then the transmitter must send a teach-in telegram. The receiver – upon receiving a valid teach-in telegram from a previously taught-in transmitter – will adjust its own sequence counter to the one specified in the teach-in telegram.

Successful resynchronization of the RLC by means of a secure teach-in telegram will be indicated to the host by a CO_EVENT_SECUREDEVICES EVENT with code 0x0A (successful RLC resynchronization).

7.6.2 RLC roll-over

For the case of 16-bit or 24-bit RLC sizes, it is possible that the number of transmitted telegrams during the product lifetime exceeds the amount of possible RLC values. In this case, the sequence counter that generates the RLC will be reset to zero after reaching the maximum value (65535 for 16-bit RLC, 16.777.216 for 24-bit RLC) and start counting up again. This means that previously used RLC values will be used again and is called *RLC roll-over*. The case of RLC roll-over can be addressed in two ways:

1. Roll-over is not allowed and the only restriction for consecutive RLC values is that the most recently received one is higher than previously received ones.
   This mode is always used for the 32-bit explicit RLC modes and is the default setting for the 24 bit explicit RLC mode.

2. Roll-over is allowed but two consecutively received RLC values have to be no more than a certain value - called *RLC Window* - apart. The value of RLC Window is 128 in EnOcean devices.
   This mode is used for the 16-bit RLC modes and the 24-bit implicit RLC mode. It is an option for the 24-bit explicit RLC mode configurable via ESP3 command as described below.

It is possible to select which strategy is applied for the case of 24-bit explicit RLC mode with the first option (no roll-over allowed) being the default setting. It is possible to select the second option (roll-over allowed if within RLC window) using the ESP3 command CO_WR_RLC_LEGACY_MODE as shown in Table 32 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x37	0x37: CO_WR_RLC_LEGACY_MODE
Data	7	1	Legacy Mode	0x00 0x01	0x00: Default setting No roll-over allowed in 24 bit explicit RLC mode (SLF = 0b101), no restriction on distance between consecutive RLC 0x01: Legacy mode Roll-over allowed in 24 bit explicit RLC mode (SLF = 0b101), consecutive RLC must be within RLC WINDOW
-	8	1	CRC8D	0xnn	

Table 32 – CO_WR_RLC_LEGACY_MODE

7.6.3 RLC backup

The constant part of the secure link table entries (device addresses, security keys, security level format, teach-in info) is stored in non-volatile memory to preserve the content in case of a temporary power loss.

In contrast to that, the RLC values – which change for each transmitted or received telegram – are stored in internal volatile memory to optimize encryption and decryption performance since the storage to non-volatile memory requires a significant amount of time.

To account for the option of a power loss, it is necessary to periodically backup the RLC from volatile to non-volatile memory. The RLC value of a link table entry is by default backed up to non-volatile memory once for every 64 telegrams that have been sent by the local to the remote device (local RLC – RLC_L) or sent by the remote to the local device (remote RLC – RLC_R) for that entry.

Should TCM 615 encounter a power loss, then the local RLC value for each entry in the secure link table will be incremented by 64 to account for the possibility that the last backup of the RLC used by TCM 615 for transmission might have occurred 63 telegrams ago (if power loss occurred directly before the next RLC backup).

If TCM 615 is continuously power-cycled such that it is only active during a brief period for the transmission of one or several telegrams, then the RLC used by TCM 615 for transmission will "jump" by up to 64 every time the device is powered up and transmits a telegram.

It is possible to change the rate at which the RLC is backed up to non-volatile memory from its default setting of 64 to a user-defined setting using the command CO_WR_RLC_SAVE_PERIOD as shown in Table 33.

Note that lowering the backup interval will increase the time spent for backing up the RLC values and thereby reduce the device performance. This function should therefore only be used if necessary.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x36	0x36: CO_WR_RLC_SAVE_PERIOD
Data	7	1	Save Period	0xnn	0x00: All RLC in the secure link table will be saved immediately 0x01..0xFF: RLC are saved every n times
-	8	1	CRC8D	0xnn	

Table 33 – CO_WR_RLC_SAVE_PERIOD

Using a Save Period of 0 in this command will result in TCM 615 backing up all RLC values in its link table to non-volatile memory leaving the RLC backup interval otherwise unchanged. This is intended for cases of expected power down where volatile data should be stored before power loss.

7.7 Teach-in of secure devices

When establishing secure communication, the sender and the receiver have to agree on the parameters to be used and exchange the security credentials (security key, current RLC value). This process is called *Secure Teach-in* or teach-in in short.

7.7.1 Secure teach-in procedure

Secure teach-in can be performed in two different ways:

- Using a secure teach-in telegram if TCM 615 is in teach-in mode (see chapter 7.7.2)
  TCM 615 can automatically derive the required parameters for telegram encryption, decryption and authentication from such secure teach-in telegram.
  Conversely, TCM 615 can also be instructed via its ESP3 interface to transmit such secure teach-in telegram to a remote device.

- Using an ESP3 command (see Chapter 7.7.3)
  The required parameters for telegram encryption, decryption and authentication can also be configured TCM 615 can be configured via an ESP3 command

In both cases, the configured parameters must be the same for both the sender and the receiver.

Until secure communication has been established, TCM 615 will forward received telegrams to the external host and transmit telegrams from the external host without security processing. If secure communication between a remote device and TCM 615 has been established, then TCM 615 will handle all security-related functionality such as encryption, decryption, authentication and RLC management. This greatly facilitates the implementation of secure communication in resource-constrained applications such as simple actuators.

7.7.2 Teach-in of secure devices with secure teach-in telegram

Teach-in is the process by which a remote device communicates to TCM 615 all parameters required to establish secure communication using a special radio telegram described below.

7.7.2.1 Format of the secure teach-in telegram

The secure teach-in telegram uses the structure shown in Figure 16 below.

RORG (0x35: SEC_TI)	TEACH-IN INFO	SECURITY FORMAT (SLF)	CURRENT RLC VALUE	SECURITY KEY
1 byte	1 byte	1 byte	2 / 3 / 4 byte	16 byte

Figure 16 – Secure teach-in telegram structure

As shown above, the secure teach-in telegram contains the following parameters:

- R-ORG 0x35 (SEC_TI)
  Secure teach-in telegrams are identified by the R-ORG 0x35 (SEC_TI)

- Teach-in Info
  This field contains information about the secure teach-in telegram as described in chapter 7.4.4.

- SLF
  The SLF specifies the type of encryption and authentication used by for the communication with the remote device as described in chapter 7.4.3

- Current RLC Value
  This field contains the current value of the RLC used by the sender as described in chapter 7.4.2.

- Security Key
  The 128-bit security key is used by the sender to encrypt and authenticate the transmitted telegram and by the receiver to decrypt and authenticate the received telegram as described in chapter 7.4.1

7.7.2.2 Transmission of a secure teach-in telegram

If the parameters for secure communication with a remote device have been setup in the secure link table, then a secure teach-in telegram can be transmitted to that device using the CO_WR_SENDTEACHIN command as shown in Table 34 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0005	5 bytes
	3	1	Optional Length	0x00...0x01	1 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x20	0x20: CO_WR_SECUREDEVICE_SENDTEACHIN
	8	4	ID	0xnnnnnnnn	Device ID
Optional Data	8	1	TeachInInfo	0xnn	Teach-In Info
-	-	1	CRC8D	0xnn	

Table 34 – CO_WR_SECUREDEVICE_SENDTEACHIN

7.7.2.3 Reception of a secure teach-in telegram (Teach-in mode)

TCM 615 can be configured to automatically accept secure teach-in telegrams and store their parameters in the secure link table by enabling the so-called *Teach-in Mode*. Teach-in mode can be enabled for a specific time (the default setting is 60 seconds) using the CO_WR_LEARNMODE command shown in Table 35 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0006	6 bytes
	3	1	Optional Length	0x01	1 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x17	0x17: CO_WR_LEARNMODE
	7	1	Enable	0x0n	0x00: Stop Teach-in Mode 0x01: Start Teach-in mode
	8	4	Timeout	0xnnnnnnnn	Time-Out for Teach-in Mode in ms. When time is set to 0x00000000 then the default period of 60’000 ms is used
Optional Data	12	1	Channel	0xnn	0x00 ... 0xFD: Channel number (absolute) 0xFE Previous channel (relative) 0xFF Next channel (relative)
-	-	1	CRC8D	0xnn	

Table 35 – CO_WR_LEARNMODE

If a valid teach-in telegram is received while teach-in mode is active, then an entry with the corresponding parameters of the remote device is added to the secure link table.

TCM 615 will indicate successful teach-in with a CO_EVENT_SECUREDEVICES event message as described in chapter 7.8. Additionally, TCM 615 will indicate that the teach-in mode has

ended by sending the Event CO_LRN_MODE_DISABLED shown in Table 36 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0001	1 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x04	0x04: EVENT
-	5	1	CRC8H	0xnn	
Data	6	1	Event Code	0x09	0x09: CO_LRN_MODE_DISABLED
-	7	1	CRC8D	0xnn	

Table 36 – CO_LRN_MODE_DISABLED

The maximum number of remote devices that can be teached-in is 32. Attempting to teach in additional devices will result in a CO_EVENT_SECUREDEVICES EVENT with code 00 (Teach-in failed, because no more space available).

If TCM 615 is not in teach-in mode and it receives a valid (same key, same SLF, same Teach-in Info) secure teach-in telegram then it will adjust the RLC of the remote device to the RLC specified within this secure teach-in telegram as described in chapter 7.7.2.4.

7.7.2.4 Handling of secure teach-in telegrams if teach-in mode is not active

If TCM 615 is not in teach-in mode, then secure teach-in telegrams from unknown senders are ignored.

If TCM 615 receives a secure teach-in telegram from a known (previously teached-in) sender containing the correct security key, then the sequence counter information in the TCM 615 secure link table is updated to the value specified in the telegram. This approach is used in case sequence counters of receiver and sender become desynchronized.

TCM 615 will indicate a successful sequence counter resynchronization using this mechanism by sending a CO_EVENT_SECUREDEVICES event message as described in chapter 7.8.

7.7.3 Teach-in of secure devices using ESP3

The security parameters required for secure communication with a remote device can be setup by the external host via the ESP3 interface using the CO_WR_SECUREDEVICEV2_ADD command. This approach is always used for setting up the security parameters (local key, local RLC, SLF and TII) for the transmission of telegrams from TCM 615 to a remote device.

This approach might also be used (instead of relying on secure teach-in telegrams) for setting up the security parameters (remote key, remote RLC) for the reception of telegrams transmitted by the remote device to TCM 615 (TCM 615 is receiver). This could for instance be the case if the security information of the remote device has been read by the host from a QR code on the remote device.

The provided information will either be added to the remote (for telegrams transmitted by the remote device – TCM 615 is receiver) or to the local (for transmission of telegrams by TCM 615 – TCM 615 is transmitter) parameters of the link table entry depending on the value of the Direction field.

For the case of addition to the local link table entry parameters, setting the ID field to the own EURID (or 0x00000000) will cause the provided information to be used for secure broadcast transmissions. Otherwise, it will be used for secure addressed transmissions to the specified ID.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x001B	27 bytes
	3	1	Optional Length	0x01	1 bytes
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x38	0x38: CO_WR_SECUREDEVICE2_ADD
	7	1	SLF	0xnn	Security Level Format
	8	4	ID	0xnnnnnnnn	Device ID
	12	16	Private key	0xnnnnnnnn 0xnnnnnnnn 0xnnnnnnnn 0xnnnnnnnn	16 bytes private key of the device
	28	4	Rolling code	0xnnnnnnnn	If a 24/16 bit rolling code is defined in SLF, then the MSBs are undefined
	32	1	Teach-Info	0xnn	Full SEC_TEACH_INFO, like defined in the security SPEC
Optional Data	31	1	Direction	0xnn	Device security information for: 0x00: Remote device (for reception, default) ID = Remote device EURID 0x01: Local device (for transmission) ID = Destination EURID Used for secure addressed telegrams ID = Own EURID or 0x00000000 Used for secure broadcast telegrams 0x02 ... 0xFF: Not used
-	48	1	CRC8D	0xnn	

Table 37 – CO_WR_SECUREDEVICE2_ADD

7.8 Reporting of security-related events

TCM 615 can report to the host the following security-related events by means of a CO_EVENT_SECUREDEVICES event using the structure shown below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0006	6 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x04	0x04: EVENT
-	5	1	CRC8H	0xnn	
Data	6	1	Event Code	0x05	0x05: CO_EVENT_SECUREDEVICES
	7	1	Event Type	0xnn	0x00: Teach in failed because no more space is available in the secure link table 0x02: Resynchronization attempt with wrong private key 0x03: Configured count of telegrams with wrong CMAC received 0x04: Teach-in failed due to incorrect teach-in telegram content or format 0x07: CMAC or RLC not correct 0x08: Standard telegram received from device in secure link table 0x09: Teach-In successful 0x0A: Received valid RLC sync via Teach-In Others: Reserved or not supported
	8	4	Device ID	0xnnnnnnnn	Device ID
-	12	1	CRC8D	0xnn	

Table 38 – Secure event reporting

7.8.1 Security event description

TCM 615 reports to the host the following security events using the Event Code 0x05: CO_EVENT_SECUREDEVICES with the following Event Data:

- 0x00: Teach in failed
  No more space is available in the secure link table and the device can therefore not be added to the secure link table

- 0x02: RLC resynchronization attempt with wrong private key
  When using implicit RLC, a secure teach-in telegram might be used to update the RLC at the receiver as described in chapter 7.7.2.4.
  If a secure teach-in telegram is received from a device that is already in the secure link table, but the security key provided in the secure teach-in telegram is different from the security key stored in the secure link table, then the secure teach-in telegram is ignored.

- 0x03: Configured count of telegrams with wrong CMAC received
  128 messages with wrong CMAC have been received from the same sender which might indicate a brute force attack

- 0x04: Teach-in failed due to unexpected structure and content
  The content or the structure of the secure teach-in telegram is incorrect and therefore the secure teach-in telegram is ignored.

- 0x07: CMAC or RLC not correct
  The authentication signature received does not match the authentication signature that is expected. For the case of using implicit RLC, all RLC within the RLC window have been tried, but none of them resulted in an authentication signature match.

- 0x08: Standard telegram received from device in secure link table
  The sender of the telegram is setup in the secure link table, but an unencrypted and unauthenticated message was received from the sender. The received telegram will be provided to the host together with this error message.

- 0x09: Teach-in successful
  The setup of a new device in the secure link table was successful.

- 0x0A: Successful RLC resync via secure teach-in telegram
  The expected RLC value of a sender using implicit RLC has been changed due to the reception of secure teach-in telegram with the correct security key as described in chapter 7.7.2.4.

8 Low power Sleep state

TCM 615 can be set into Sleep state with much lower power consumption for a defined period by means of the CO_WR_SLEEP command shown in Table 39 below. After expiry of the requested sleep period, TCM 615 will automatically wake-up and transition back to Receive state.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0005	5 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x01	0x01: CO_WR_SLEEP
	7	4	Deep sleep period	0x00nnnnnn	0x00000000: Sleep indefinitely, wake by UART event 0x00000001 … 0x00FFFFFF: Duration of sleep in 10 ms units (maximum value ~ 46h). After waking up, the module generates an internal hardware reset
-	11	1	CRC8D	0xnn	

Table 39 – CO_WR_SLEEP

8.1 HW wake-up from Sleep state

It is possible to put TCM 615 into Sleep state indefinitely by using 0x00000000 as sleep period. TCM 615 will in this case remain in Sleep state until it is woken up by the external host via activity on the ESP3 interface.

Any activity on the ESP3 interface will wake-up TCM 615 in this case but the command used for wake-up will not be processed. Any ESP3 command can be used for the purpose of wake-up. It is suggested however to use a command without possible side effects such as CO_RD_VERSION.

TCM 615 will then confirm successful wake-up using the Event Code 0x04: CO_READY with Event Data: 0x0A: UART Wake-up.

9 ESP3 interface

TCM 615 provides an external interface according to the EnOcean Serial Protocol, version 3 (ESP3).

This interface is used both to exchange telegrams and command / status messages with an external host system (e.g. microcontroller or PC) and EnOcean gateway transceiver modules.

The information in the subsequent chapters as well as any previous references to specific ESP3 commands are provided for information purposes only. For detailed information, please refer to the ESP3 specification [1].

9.1 ESP3 physical interface

The physical interface used by ESP3 for communication between host system and an EnOcean Gateway Controller is a 3-wire full-duplex UART connection (UART RX, UART TX, GND).

9.2 ESP3 packet structure

ESP3 is a point-to-point (one to one) protocol based on a packet data structure.
Figure 17 below illustrates the ESP3 packet structure.

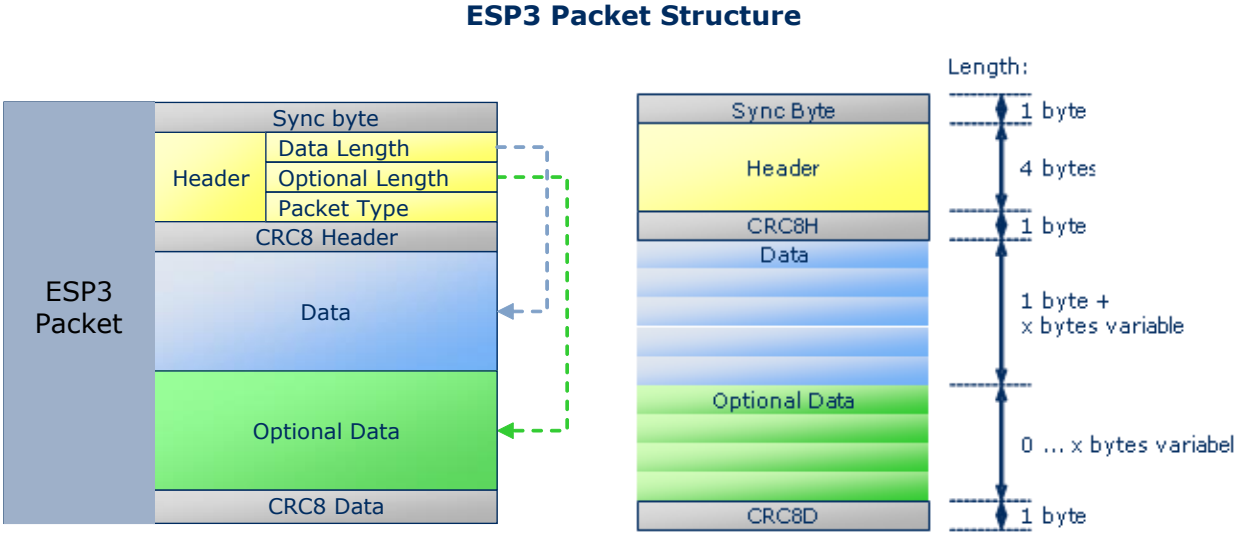

Figure 17 – ESP3 Packet Structure

9.2.1 ESP3 packet fields

Each ESP3 packet contains the following fields:

- Header
- Data
- Optional Data

In addition to those fields, the Sync byte (`0x55`) identifies the start of the packet while separate CRC8 for Header and Data (incl. Optional Data) are used to verify data integrity.

The Header consists of the following fields:

- Data Length (number of bytes of the group Data)
- Optional Length (number of bytes of the group Optional Data)
- Packet Type (RADIO, RESPONSE, EVENT, COMMAND ...)

The Data field encodes the ESP3 command together with the required parameter data. For some commands, the Optional Data field is used to provide additional parameter data.

9.3 Supported ESP3 commands

The following ESP3 commands are supported by TCM 615:

- Type 1: ERP1 Radio Telegram
  - Transmit or receive ERP1 telegrams with up to 14 bytes of payload

- Type 2: Responses
  - RET_OK
  - RET_ERROR
  - RET_NOT_SUPPORTED
  - RET_WRONG_PARAM
  - RET_OPERATION_DENIED
  - RET_LOCK_SET
  - RET_BUFFER_TO_SMALL
  - RET_NO_FREE_BUFFER

- Type 4: Events
  - CO_READY to indicate wake up from deep sleep initiated by CO_WR_SLEEP
  - CO_EVENT_SECUREDEVICES to inform about security processing issues
  - CO_DUTYCYCLE_LIMIT to inform about a current limitation due to duty cycle
  - CO_TX_DONE to inform that the transmission of a telegram has completed
  - CO_LRN_MODE_DISABLED to inform that the learn mode has timed-out

- Type 5: Common commands
  - CO_WR_RESET to reset the device
  - CO_RD_VERSION to read SW/HW versions, chip ID etc.
  - CO_GET_FREQUENCY_INFO to read the operating frequency of the device
  - CO_WR_STARTUP_DELAY to increase the start-up time
  - CO_WR_SLEEP to put the device into low power sleep mode
  - CO_WR_IDBASE to set the Base ID range
  - CO_RD_IDBASE to read the Base ID range
  - CO_WR_REPEATER to set repeater functionality
  - CO_RD_REPEATER to read repeater functionality
  - CO_WR_FILTER_ADD to add filter to filter list or to selective repeating
  - CO_WR_FILTER_DEL and CO_WR_FILTER_DEL_ALL to delete filters
  - CO_RD_FILTER to read the configured filters
  - CO_WR_FILTER_ENABLE to enable/disable the configured filters
  - CO_WR_LEARNMODE to set teach-in mode
  - CO_RD_LEARNMODE to read teach-in mode status
  - CO_WR_WAIT_MATURITY to wait until the end of the maturity time
  - CO_RD_DUTYCYCLE_LIMIT to read the duty cycle (for 868 MHz EU version)
  - CO_SET_BAUDRATE to set the baud rate of the ESP3 interface
  - CO_WR_SECUREDEVICE_DEL to delete a device from a link table
  - CO_RD_SECUREDEVICE_COUNT to read the number of devices in a link table
  - CO_RD_SECUREDEVICE_BY_ID to read a link table entry using its EURID
  - CO_WR_SECUREDEVICE_SENDTEACHIN to send a secure teach-in telegram
  - CO_WR_RLC_SAVE_PERIOD to set the interval for the backup of RLC values
  - CO_WR_RLC_LEGACY_MODE to set the legacy RLC mode (window-based)
  - CO_WR_SECUREDEVICEV2_ADD to add a device to a link table

- CO_RD_SECUREDEVICEV2_BY_INDEX to read a link table entry using its index
- CO_WR_RSSITEST_MODE to enable RSSI test mode
- CO_RD_RSSITEST_MODE to read the status of RSSI test mode
- CO_WR_SECUREDEVICE_MAINTENANCEKEY to set the Reman security key
- CO_RD_SECUREDEVICE_MAINTENANCEKEY to read the Reman security key
- CO_WR_TRANSPARENT_MODE to enable transparent mode
- CO_RD_TRANSPARENT_MODE to check if transparent mode is active
- CO_WR_TX_ONLY_MODE to enable Transmit-only mode
- CO_RD_TX_ONLY_MODE to check if Transmit-only mode is active
- CO_WR_MODE to select RADIO_ERP1 or RADIO_ERP2 packets for RX
- CO_SET_CRC_SIZE to select the CRC size (TCM 615J only)
- CO_GET_CRC_SIZE to determine the CRC size (TCM 615J only)

- Type 7 Remote Management
  - Transmit or receive remote management messages with up to 128 bytes of payload. TCM 615 will automatically chain (segment) messages as needed.
  - Note: TCM 615 will not interpret or execute any received remote management message; it will only forward these messages to the connected host via the ESP3 interface.
  - Note: TCM 615 will not forward Remote Management commands that are addressed to a different device.

- Type 9 Radio Message
  - Transmit or receive EnOcean messages with up to 128 bytes of payload. TCM 615 will automatically chain (segment) messages as needed.

- Type 10 ERP2 Radio Telegram (TCM 615U and TCM 615J only)
  - Transmit or receive ERP2 telegrams with up to 128 byte of payload. TCM 615 will automatically chain (segment) messages as needed.

9.4 Persistent versus not persistent configuration settings

TCM 615 will store certain configuration settings in persistent memory, i.e. those settings will be maintained even after a power cycle. The CO_WR_RESET command can be used to reset the persistent settings.

There are three classes of persistent settings:

1. Repeater and filter configuration
   The repeater and filter configuration defined via the following commands will be maintained after power failure:
   - CO_WR_REPEATER
   - CO_WR_FILTER_ADD
   - CO_WR_FILTER_DEL
   - CO_WR_FILTER_DEL_ALL
   - CO_WR_FILTER_ENABLE

2. List of secure devices as defined by the following commands:
   - CO_WR_SECUREDEVICE_ADD or CO_WR_SECUREDEVICEV2_ADD
   - CO_WR_SECUREDEVICE_DEL

3. System parameters as defined by the following commands:
   - CO_WR_STARTUP_DELAY
   - CO_WR_IDBASE
   - CO_WR_RLC_SAVE_PERIOD
   - CO_WR_TX_ONLY_MODE

All other settings need to be reinitialized at power up or after a reset.

9.5 Factory reset

TCM 615 can be reset to the default configuration using the CO_WR_RESET command as shown in Table 40.

If RESET TYPE is set to 0x01, then TCM 615 will reset the persistent parameters listed above. Note that the Base Address will be set to 00:00:00:00 in this case and needs to be re-initialized before using the Base Addresses for transmission.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0001	1 byte
	3	1	Optional Length	0x01	1 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x02	CO_WR_RESET
Optional Data	7	1	RESET TYPE	0xnn	0x00: Restart application only 0x01: Restart application and reset parameters
-	-	1	CRC8D	0xnn	

Table 40 – CO_WR_RESET

TCM 615 will indicate completion of the reset procedure by sending a CO_READY event with indicating Wakeup Cause 0x0B (SW reset using CO_WR_RESET) as shown in Table 41.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0x0002	2 bytes
	3	1	Optional Length	0x00	1 byte
	4	1	Packet Type	0x04	EVENT = 4
-	5	1	CRC8H	0xnn	
Data	6	1	Event Code	0x04	CO_READY = 4
	7	1	Wakeup Cause	0xnn	0x00: Voltage supply drop or VDD > VON 0x01: HW reset (reset pin) 0x02: Watchdog timer time-out 0x03: Flywheel timer time-out 0x04: Parity error 0x05: Memory error 0x06: Invalid memory address 0x07: HW wake-up via Pin 0 0x08: HW wake-up via Pin 1 0x09: Unknown reset source 0x0A: UART wake-up 0x0B: SW reset using CO_WR_RESET
Optional Data	8	1	Mode	0xnn	0x00: Standard Security 0x01: Extended Security
-	8	1	CRC8D	0xnn	

Table 41 – CO_READ

10 Remote management telegrams

TCM 615 provides a transparent radio channel for remote management messages with a message length of up to 128 bytes. This enables an external host connected to TCM 615 to handle remote management request from external devices or to control other devices via remote management.

TCM 615 will not interpret or execute any received remote management message; it will only forward such messages to the connected host via the ESP3 interface. Note that TCM 615 will not forward Remote Management commands that are addressed to a device address that is not the EURID used by TCM 615.

For more information on remote management please refer to the EnOcean Equipment Profiles (EEP) specification [5] and the Remote Management specification [7].

10.1 Sending or receiving remote management telegrams

Remote management telegrams can be sent or received by using ESP3 Packet Type 0x07: REMOTE_MAN_COMMAND with the syntax shown in Table 42 below.

Note that using very large messages affect the overall performance of EnOcean radio networks. The maximum size of transmitted or received remote management telegrams shall not exceed 128 bytes.

Note that TCM 615 does not provide any remote management functionality itself; this functionality including the timing of remote management messages and the addition of random delay therefore must be controlled by the host (external MCU). The option to add a random delay is not supported by TCM 615 and shall not be used.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. Byte	0x55	
Header	1	2	Data Length	0xnnnn	4 + x bytes
	3	1	Optional Length	0x0A	10 bytes
	4	1	Packet Type	0x07	REMOTE_MAN_COMMAND = 7
-	5	1	CRC8H	0xnn	
Data	6	2	Function No.	0x0nnn	Range: 0x0000 ... 0x0FFF
	8	2	Manufacturer ID	0x0nnn	Range: 0x0000 ... 0x07FF
	10	x	Message data	...	N bytes
Optional Data	10+x	4	Destination ID	0xnnnnnnnn	Destination ID Broadcast ID: FF FF FF FF
	14+x	4	Source ID	0xnnnnnnnn	Receive case: Source ID of the sender Send case: 0x00000000
	18+x	1	dBm	0xnn	Send case: 0xFF Receive case: Best RSSI value of all received sub telegrams (only if wait for maturity is set to true!) (value decimal without minus)
	19+x	1	Send With Delay	0x0n	0x00: No random delay (Default) 0x01: Not supported (Do not use)
-	20+x	1	CRC8D	0xnn	CRC8 Data byte; calculated checksum for whole byte groups: DATA and OPTIONAL_DATA

Table 42 – REMOTE_MAN_COMMAND

11 Device integration

TCM 615 is designed for integration onto a host PCB. Detailed Gerber data of the device footprint is available from EnOcean.

11.1 Recommended PCB Footprint

Figure 18 below shows the recommended PCB footprint for TCM 615.

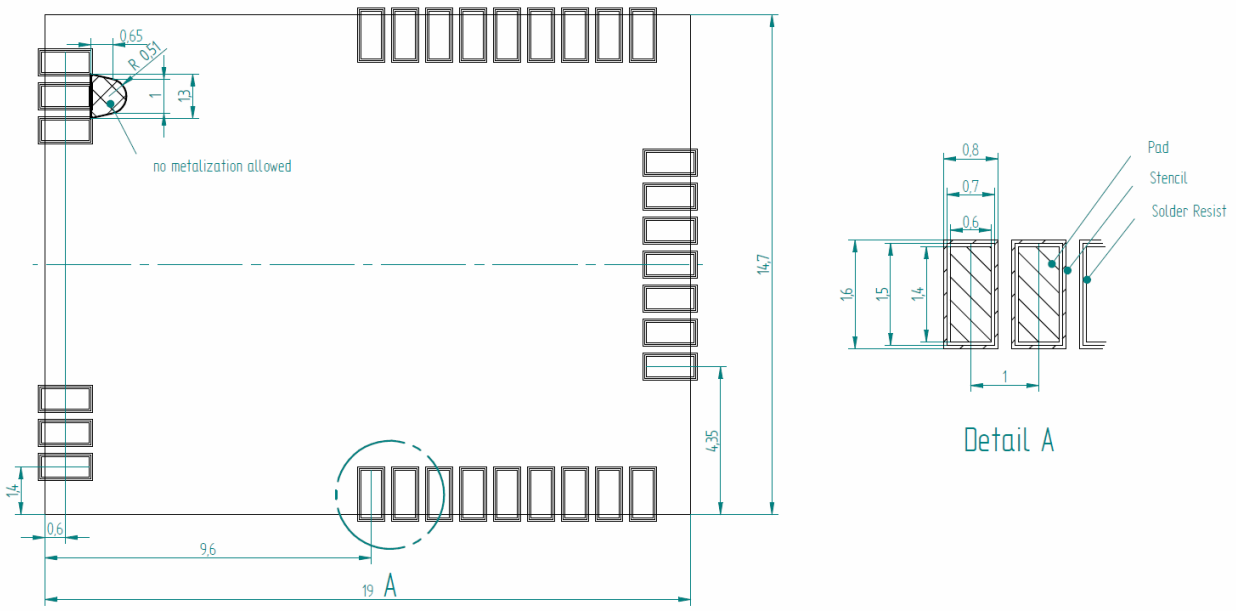

Figure 18 – Recommended PCB footprint

11.2 Device outline

Figure 19 below shows the device outline of TCM 615. In addition, EnOcean can provide upon request a 3D model of TCM 615.

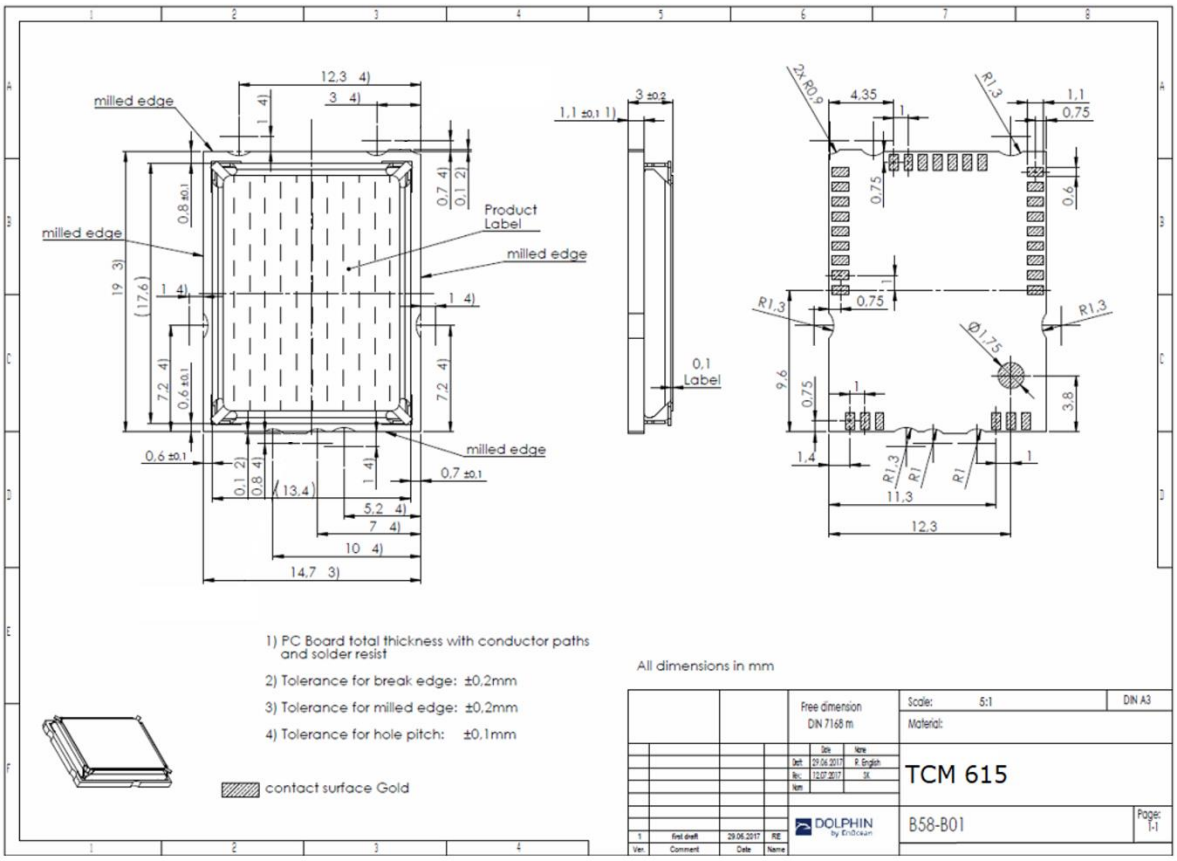

Figure 19 – Device outline

11.3 Soldering information

TCM 615 shall be soldered according to IPC/JEDEC J-STD-020C standard.

Profile Feature	Pb-Free Assembly
Average Ramp-Up Rate (Ts_{max} to Tp)	3° C/second max.
Preheat – Temperature Min (Ts_{min}) – Temperature Max (Ts_{max}) – Time (ts_{min} to ts_{max})	150 °C 200 °C 60-180 seconds
Time maintained above: – Temperature (T_L) – Time (t_L)	217 °C 60-150 seconds
Peak/Classification Temperature (Tp)	260 °C
Time within 5 °C of actual Peak Temperature (tp)	20-40 seconds
Ramp-Down Rate	6 °C/second max.
Time 25 °C to Peak Temperature	8 minutes max.

Note 1: All temperatures refer to topside of the package, measured on the package body surface.

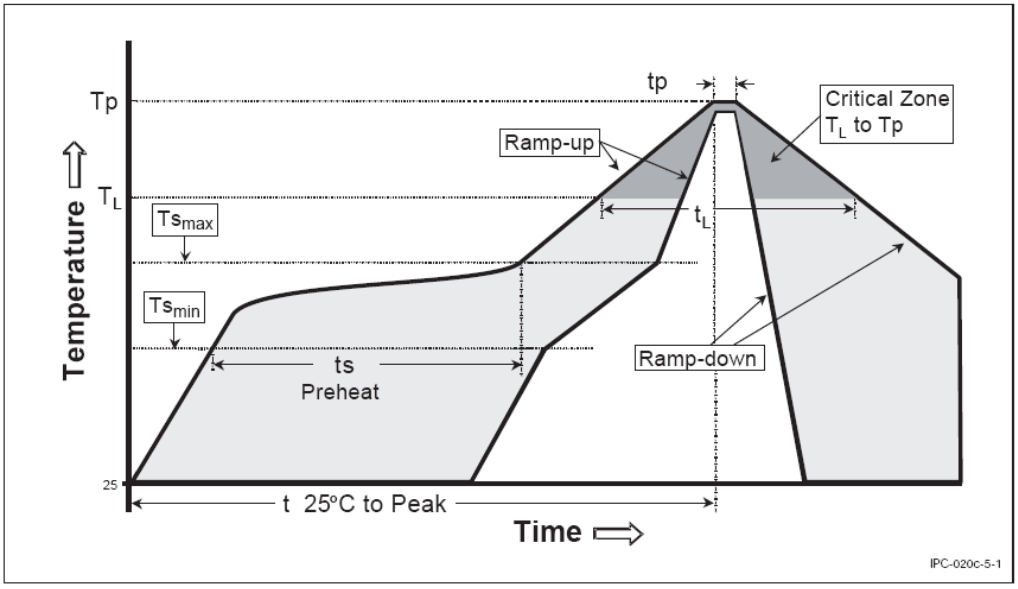

Figure 20 – Recommended soldering profile

TCM 615 shall be handled according to Moisture Sensitivity Level MSL3 which means a floor time of 168 h. TCM 615 may be soldered only once, since one time is already consumed at production of the module itself.

Once the dry pack bag is opened, the desired quantity of units should be removed, and the bag resealed within two hours. If the bag is left open longer than 30 minutes the desiccant should be replaced with dry desiccant. If devices have exceeded the specified floor lifetime of 72 h, they may be baked according to IPC/JEDEC J-STD-033B at max. 90°C for less than 60 h.

Devices packaged in moisture-proof packaging should be stored in ambient conditions not exceeding temperatures of 40 °C or humidity levels of 90% r.H.

TCM 615 modules shall be soldered within 6 months after delivery!

11.4 Packaging information

TCM 615 is delivered in Tape & Reel packaging with 250 units per reel. Figure 21 below illustrates the dimensions.

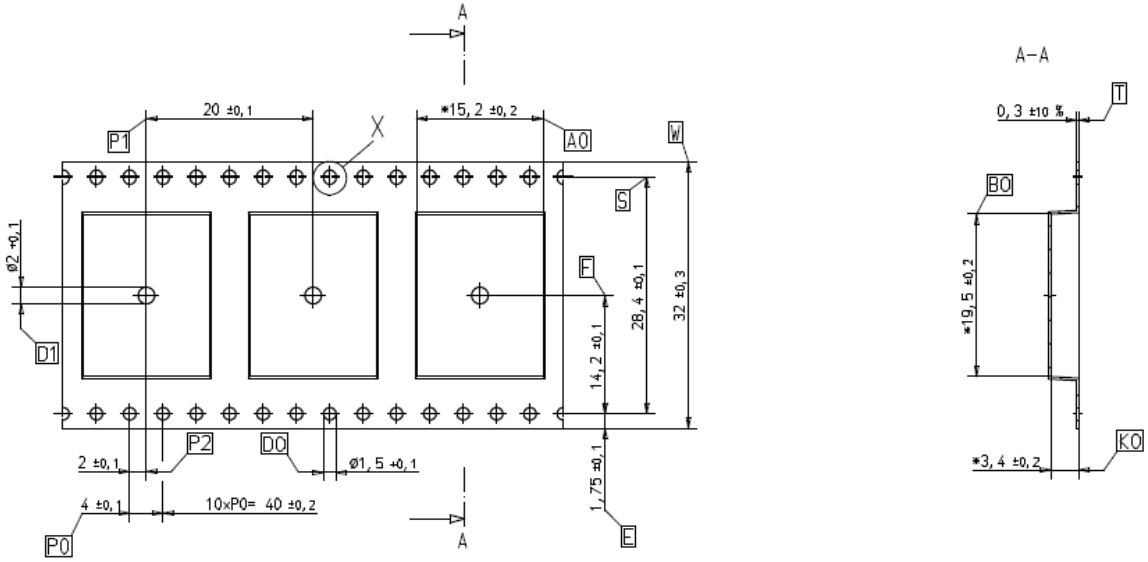

Figure 21 – Tape & Reel dimensions of TCM 615

Figure 22 below shows the positioning of TCM 615 in the Tape & Reel packaging.

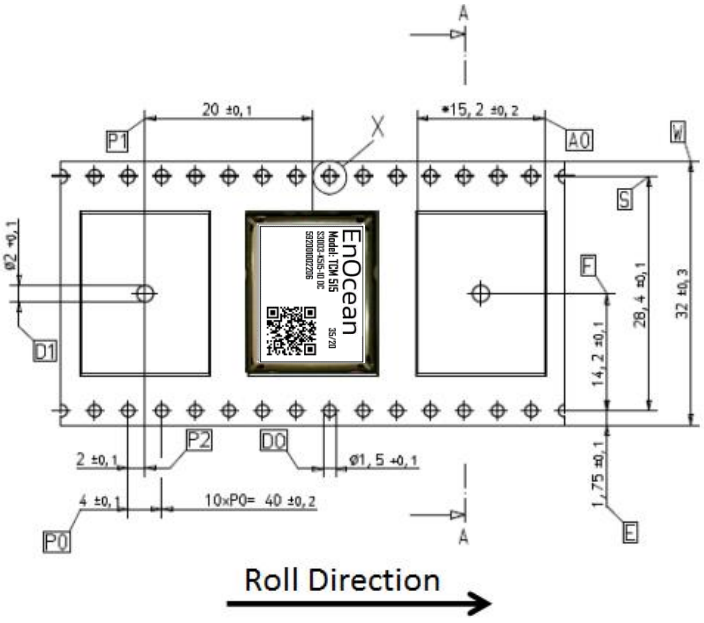

Figure 22 – Position of TCM 615 in the reel

11.5 Layout recommendations

The length of lines connected to I/O signals should not exceed 5 cm.

It is recommended to have a complete GND layer (for instance the mid-layer of your application PCB) at least in the area below the module and the directly connected components.

Due to non-isolated test points, there are live signals accessible on the bottom side of the module. We suggest avoiding any copper structure in the area directly underneath the module (top-layer layout of your application PCB). If this is not possible in your design, please provide coating on top of your PCB to prevent short circuits to the module. All bare metal surfaces including vias must be covered (for instance by using an adequate layout of solder resist).

Distortive signals (such as input or output signals, signals from other radio transmitters or signals from switched power supplies) should not be routed underneath the module. If such signals are present in your design, we suggest separating them from the TCM 615 module as much as possible and to provide a ground plane between the TCM 615 module and such signal lines.

11.6 Power supply requirements

Suitable power supply design, layout and shielding is essential to optimize the radio performance of TCM 615. It is recommended to place a 22 µF ceramic capacitor between VDD and GND close to the module (material: X5R, X7R, min 6.3 V to avoid derating effects).

In addition, an HF SMD EMI Suppression Ferrite Bead such as the Würth WE-CBF HF SMD EMI Suppression Ferrite Bead (Würth order number 742863160) shall be inserted in the power supply line.

For best performance it is recommended to keep the ripple on the power supply rail below 10 mVpp.

Radiated emissions from power supplies (especially DCDC designs) towards the TCM 615 RF input must be minimized as they can significantly impact RF performance. Place such power supplies as much as possible away from the radio path between antenna and TCM 615 or consider using designs with low RF emissions.

TCM 615 integrates approximately 10 uF of capacitance for filtering the internal supply voltage bus. The power supply architecture needs to be designed to supply sufficient current to charge this capacitance during power up.

11.7 Low noise design considerations

For best performance, the HW design of TCM 615 systems must minimize radiated or conducted noise that interferes with the correct reception of RF signals. Strong emphasis should be placed onto good RF and power supply design to eliminate or minimize the level of noise introduced into the RF path.

In addition, special consideration should be used to minimize periodic noise sources (such as radiated noise from DCDC inductors or from high data rate input or output signals) in TCM 615 based systems. TCM 615 (868.300 MHz ASK) transmits and receives signals using amplitude shift keying where the amplitude of a carrier frequency (868.300 MHz) is changed according to the encoded bit value (0 = high amplitude or 1 = low amplitude).

Periodic noise signals where the period between high and low signal states is close to the symbol duration of 8 us can be erroneously interpreted as the preamble of an ASK telegram (10101010 sequence) and therefore prevent correct reception of other ASK telegrams that are received at the same time.

Suitable RF design techniques such as a good separation between signal lines and the RF path, a ground plane in the PCB layout as well as decoupling and filtering on the power supply should be used to minimize the radio performance degradation due to noise.

11.8 Suggested Reset circuit

TCM 615 can be reset by pulling the nRESET pin (active low) to Ground. TCM 615 integrated a weak (50kΩ) pull-up resistor that will maintain the internal nRESET input active high (not active).

In order to avoid spurious reset events, it is recommended to filter the input signals by means of a small capacitor which is placed as close as possible to the TCM 615 nRESET pin as shown in Figure 23 below.

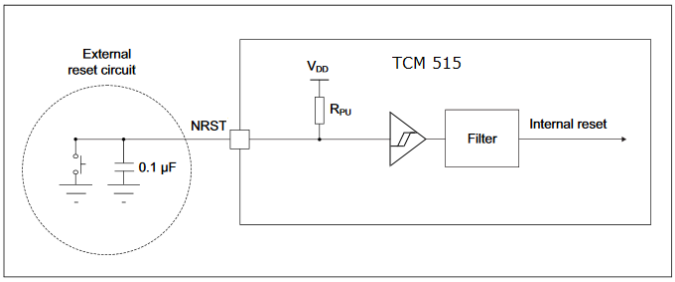

Figure 23 – Recommended reset circuit

The reset pulse should have a duration of at least 1 ms to guarantee reliable reset operation.

11.9 Test interface

TCM 615 provides 3 test points (TP1, TP2, TP3) which together with the RESET, UART_TX and UART_RX signals can be used for product test and debug.

It is strongly recommended to make the pins TP1, TP2, TP3, nRESET, UART_TX and UART_RX together with VDD and GND accessible to external devices - e.g. by means of providing suitable test point pads on the PCB – for the purpose of debug and analysis.

11.10 Identifying the TCM 615 product revision

The connected host can determine the TCM 615 product revision using the CO_GET_STEP-CODE command as shown in Table 43 below. This is especially helpful if the firmware used by TCM 615 has been upgraded by the user.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0001	1 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x05	0x05: COMMON_COMMAND
-	5	1	CRC8H	0xnn	
Data	6	1	COMMAND Code	0x27	0x27: CO_GET_STEPCODE
-	7	1	CRC8D	0xnn	

Table 43 – CO_GET_STEPCODE

TCM 615 will respond to this command with a response as shown in Table 44 below.

Group	Offset	Size	Field	Value hex	Description
-	0	1	Sync. byte	0x55	
Header	1	2	Data Length	0x0003	3 bytes
	3	1	Optional Length	0x00	0 byte
	4	1	Packet Type	0x02	0x02: RESPONSE
-	5	1	CRC8H	0xnn	
Data	6	1	Return Code	0x00	0x00: RET_OK
	7	1	Step code	0xnn	e.g. 0xDA ,0xCA …
	8	1	Status code	0xnn	e.g. 0x01, 0x02 …
-	9	1	CRC8D	0xnn	

Table 44 – Response to CO_GET_STEPCODE

12 Antenna options

This chapter outlines options for antenna that can be used with TCM 615. Note that this chapter is for guidance purposes only, please consult with an authorized certification body for specific information.

12.1 Antenna options for 868 MHz (European Union)

In order to be compliant with the Radio Equipment Directive (RED) of the European Union, an antenna needs to fulfil at least following requirements to be usable with TCM 615:

Frequency band	868.300 MHz ISM	Antenna must be suited for this band
Antenna type	Passive	Mandatory for radio approval
Impedance	~50 Ohm	Mandatory for radio approval
Maximum	≤ 0 dBd	Mandatory for radio approval

In addition, it is important to fulfill the following requirements in order to achieve compatibility with other EnOcean products and to ensure EMI robustness:

VSWR	≤ 3:1	Important for compatibility with EnOcean protocol
Return Loss	> 6 dB	Important for compatibility with EnOcean protocol
Bandwidth	≤ 20 MHz	Important if 10 V/m EMI robustness required for device

See chapter 14.1 for additional important remarks regarding RED certification.

12.1.1 Whip antenna

TCM 615 modules have been certified for use with a whip antenna under EU (RED) regulations. Figure 24 below shows key whip antenna parameters.

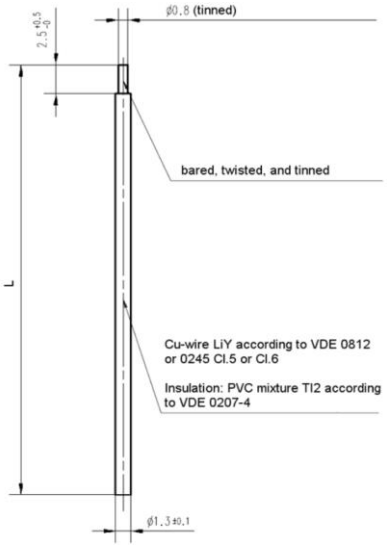

Figure 24 – Whip antenna parameters

The whip antenna be implemented with the following parameters in order to be compliant to the regulations mentioned above:

- Antenna length (L): 86 mm wire, connect to RF_50
- Minimum size of GND plane: 38 mm x 18 mm
- Minimum distance between antenna and ground plane (d): 10 mm

The whip antenna should ideally be mounted vertically as shown on the left side of Figure 25. If this is not possible then the whip antenna should be placed such that a minimum distance *d* between GND plane and antenna is provided.

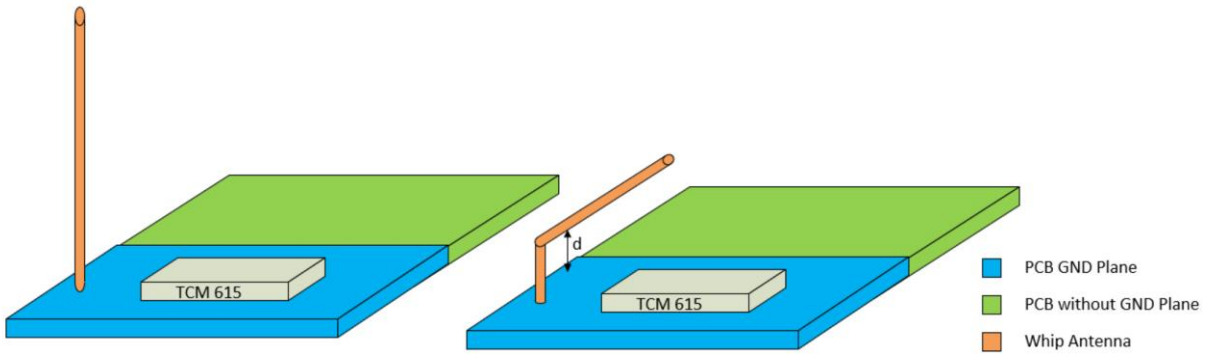

Figure 25 – Whip antenna positioning

12.2 Antenna options for 902 MHz (US / Canada)

TCM 615U has been tested and certified with a number of antennas as described below. A separate approval is required for all other operating configurations, including portable configurations with respect to Part 2.1093 and different antennas.

12.2.1 Whip antenna

TCM 615U has been certified for use with a whip antenna which meets the following parameters (see Figure 24 and Figure 25):

- Antenna length (L): 64 mm wire, connect to RF_50
- Minimum size of GND plane: 50 mm x 50 mm
- Minimum distance between antenna and ground plane (d): 10 mm

12.2.2 Helical antenna

TCM 615U has been certified for use with the ANT 300 helix antenna from EnOcean which uses the following parameters (see Figure 26):

- Shape according to drawing below
- Minimum GND plane: 35 mm x 30 mm
- Minimum distance space: 10 mm

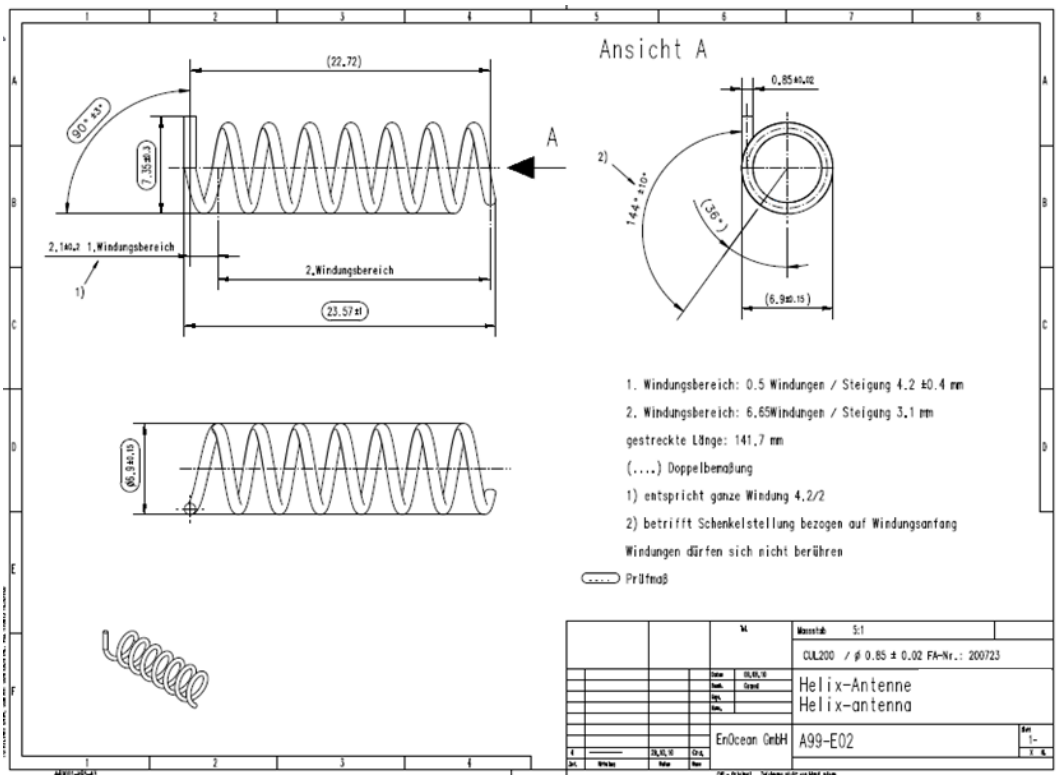

Figure 26 – Helix antenna parameters

12.2.3 Dipole antenna (ANT-916-CW-HWR-RPS)

TCM 615U has been certified for use with the dipole antenna ANT-916-CW-HWR-RPS from Linx provided that a non-standard connector such as RP-SMA-Female from Linx is used.

Figure 27 below shows ANT-916-CW-HWR-RPS from Linx.

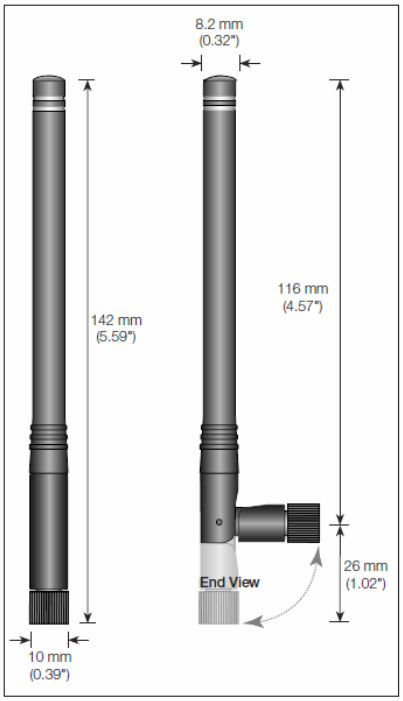

Figure 27 – ANT-916-CW-HWR-RPS

Figure 28 below shows RP-SMA- Female from Linx.

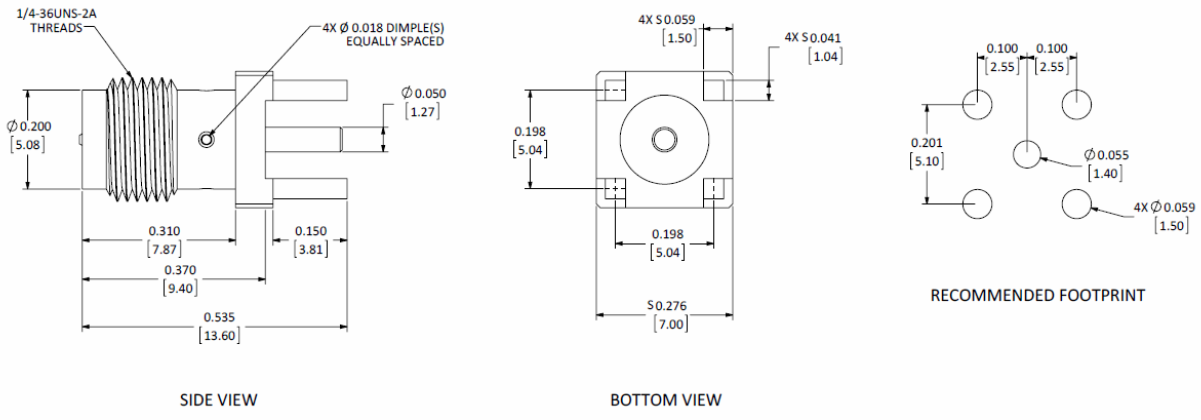

Figure 28 – RP-SMA-Female

12.3 Antenna options for 928 MHz (Japan)

TCM 615J has been tested and certified with the following antennas:

- Whip antenna, 64mm length, -3.89dBi gain
- ANT 300 helical antenna, 2.15dBi gain
- USB 400J top-loaded PCB spiral antenna, 1.14 dBi gain
- USB 400J top-loaded PCB spiral antenna (mirrored orientation), 1.17dBi gain
- ANT-GXM602, "758-0965", monopole antenna, 2.14dBi gain
- 2J520, "758-0961", monopole antenna, 2.14dBi gain
- MC0114033,"758-0910", monopole antenna, 1dBi gain
- 1019-010A, λ/2 monopole, L type, 3dBi gain
- 1019-008A, λ /2 monopole, straight type, 3dBi gain
- ME467XSAXX, λ/2 monopole, straight type, 2dBi gain

12.3.1 Whip antenna

TCM 615J has been certified for use with a whip antenna which meets the following parameters (see Figure 24 and Figure 25):

- Antenna length (L): 64 mm wire, connect to RF_50
- Minimum size of GND plane: 50 mm x 50 mm
- Minimum distance between antenna and ground plane (d): 10 mm

12.3.2 Helical antenna

TCM 615J has been certified for use with the ANT 300 helix antenna from EnOcean which uses the following parameters (see Figure 26):

- Shape according to drawing
- Minimum GND plane: 35 mm x 30 mm
- Minimum distance space: 10 mm

12.3.3 Top-loaded PCB spiral antenna

TCM 615J has been certified for use with a top loaded PCB spiral antenna with the dimensions provided in Figure 29 below. The large, hatched area to the left is the ground area for the antenna; components can be placed onto the reverse PCB side as long as the routing does not require long cuts of the ground area which act as radiators itself.

The top-loaded PCB spiral antenna that can be used either in the orientation shown in Figure 29 below or with a mirrored orientation.

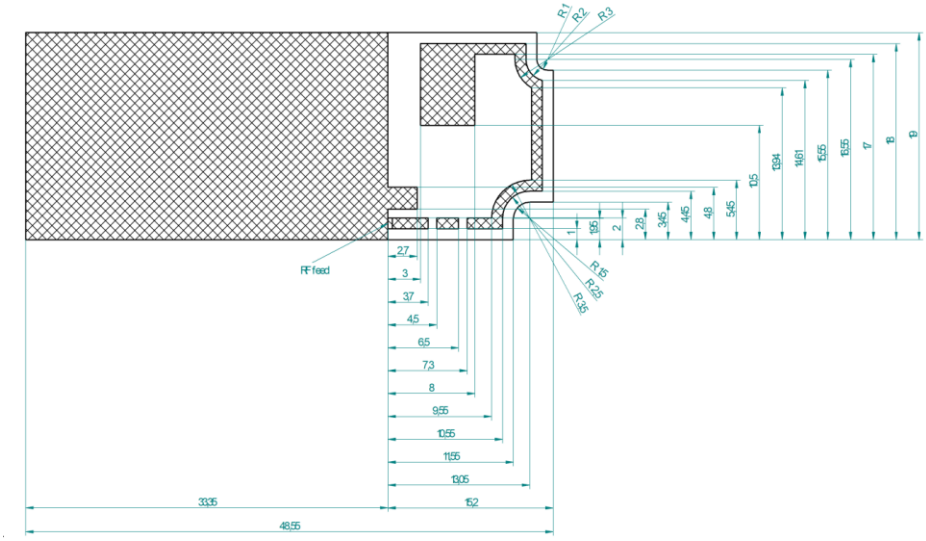

Figure 29 – Top-loaded PCB spiral antenna

This antenna should be implemented on a 1mm thick, two-layer PCB using FR4 material with the following parameters:

Parameter of PCB	VALUE
PCB material	FR4, 2 layer
Thickness (total)	1,27mm
Shape	Rectangular with millings
Dimension	19*48,55 mm

The PCB stack up should be as follows:

Layer	Thickness in µm	Exact description
Solder Mask		Solder resist
Top Layer	35	Cu, >35um after electroplating
Core	1200	
Bottom Layer	35	Cu, >35um after electroplating
Solder Mask		Solder resist
Total	1270	

The PCB antenna design uses three discrete matching components. The position of these components can be seen in Figure 30 below.

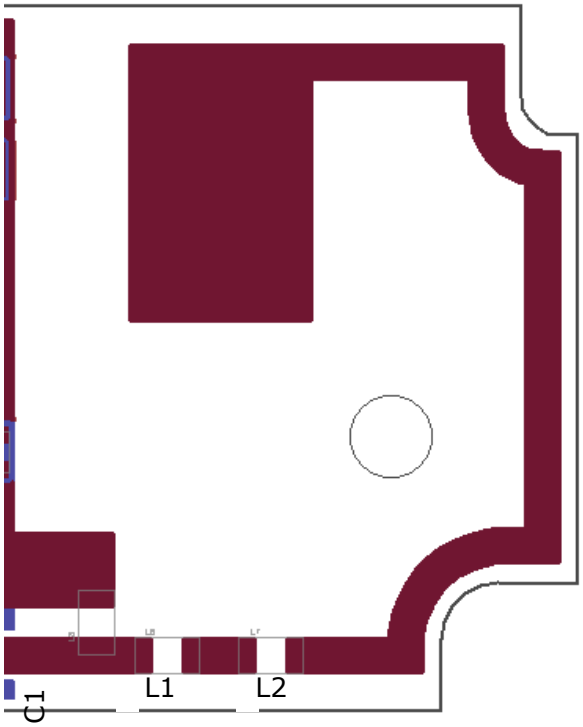

Figure 30 – Matching components for the top-loaded PCB spiral antenna

The antenna was matched to 50Ω input impedance at the feed point using the following component parameters:

Component	Value	Type
C1	3.3pF	0603 form factor 5% tolerance
L1	12nH	Wire-wound inductor Würth WE-KI or Murata LQW series
L2	12nH	

13 Application information

13.1 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 30 m range in corridors, up to 100 m in halls
- Plasterboard walls / dry wood
  Typically 30 m range, through max. 5 walls
- Ferro concrete walls / ceilings
  Typically 10 m range, through max. 1 ceiling
- Fire-safety walls, elevator shafts, staircases and supply areas
  Such areas should be considered as screening.

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, WiFi routers, audio and video equipment that also emit high-frequency signals should be at least 0.5 m.

13.2 RSSI reporting

TCM 615 will report the signal strength (RSSI) for received telegrams as part of the ERP1 or ERP2 radio packet. This information can be treated as an indicator for the quality of the radio link keeping in mind that this is affected by several factors such as temporary fading or obstructions.

The RSSI reporting of TCM 615 (868.300 MHz ASK radio) works within a range from -95 dBm up to -40 dBm with a typical accuracy of +- 2dBm.

Due to limitations of the RSSI detection functionality for ASK radio signals, the RSSI level reported by TCM 615 might under rare conditions be that of the low power state - and therefore significantly too low - as the signal strength of the low power state can sometimes be strong enough to trigger the RSSI detection mechanism.

14 Regulatory information

TCM 615 has been tested according to standards for RED (European Union) certification, FCC (US) and ISED (Canada) regulations.

14.1 CE / RED (European Union)

The Radio Equipment Directive (2014/53/EU, typically referred to as RED) is the regulatory framework for radio products in the European Union. All products sold to final customers after 12th of June, 2017 have to be compliant to RED.

At the time of writing, the text of the RED legislation was available from this link:
http://eur-lex.europa.eu/eli/dir/2014/53/oj

Radio modules such as TCM 615 are components which are delivered to OEM manufacturers for their use in final or combined products.

It is the responsibility of the OEM manufacturer to demonstrate compliance to all applicable EU directives and standards. The attestation of conformity for TCM 615 serves as input to the declaration of conformity for the full product.

At the time of writing, guidance on the implementation of EU product rules – the so called "RED Guide" – was available from this link: http://ec.europa.eu/docsroom/documents/23321

Specifically within the RED framework, all OEM manufacturers have for instance to fulfill the following additional requirements:

- Provide product branding (on the product) clearly identifying company name or brand and product name as well as type, charge or serial number for market surveillance
- Include (with the product) documentation containing full postal address of the manufacturer as well as radio frequency band and max. transmitting power
- Include (with the product) user manual, safety information and a declaration of conformity for the final product in local language
- Provide product development and test documentation upon request

Please contact an accredited test house for detailed guidance.

The maximum transmitting power of TCM 615 using a whip antenna is +10.8 dBm.

15 References

Please use below references for an in-depth description of features supported by TCM 615.

[1] EnOcean Serial Protocol 3

[2] EnOcean Radio Protocol 1 (ERP1)

[3] EnOcean Radio Protocol 2 (ERP2)

[4] Security of EnOcean Radio Networks

[5] EnOcean Equipment Profiles

[6] Signal Telegram

[7] Remote Management

[8] Range Planning Guide for Systems using EnOcean Radio Standard

16 License information

This product contains TinyCrypt Cryptographic Library

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

17 Product history

Table 45 below outlines the product history of TCM 615 and key changes made between different revisions.

Product	Revision	Release	Key features
TCM 615	DA-08	April 2025	Initial product release

Table 45 – TCM 615 product history

A. Introduction to EnOcean radio protocol

This appendix gives a high-level introduction to key aspects of the EnOcean radio protocol to help the understanding of TCM 615 features. The content provided here is for information only; it not part of the TCM 615 product specification and shall not be considered as assured product characteristics.

For detailed description, refer to the EnOcean Radio Protocol 1 (ERP1) specification [2] and the EnOcean Radio Protocol 2 (ERP2) specification [3].

Devices within the EnOcean ecosystem communicate using the EnOcean Radio Protocol (ERP). Two versions of this radio protocol are in use today – ERP version 1 (ERP1 in short) is used for 868.300 MHz radio systems in Europe while ERP version 2 (ERP2 in short) is used for 902.875 MHz radio systems in the US / Canada and 928.350 MHz radio systems in Japan.

A.1 ERP1 telegram format

The ERP1 telegram format is shown in Figure 31 below for the case of a broadcast telegram.

R-ORG	DATA	SENDER EURID	STATUS	HASH
1 byte	1 … 14 byte	4 byte	1 byte	1 byte

Figure 31 – ERP1 telegram format for broadcast telegrams

An ERP1 telegram contains the following fields:

- R-ORG specifies the EEP or SIGNAL type used by this telegram
- DATA contains the telegram payload
- SENDER EURID specifies the address of the sender
- STATUS specifies transmission properties such as the repeater hop count
- HASH is used to verify the integrity of the telegram

It is possible to specify the intended receiver (the destination) of a telegram by prefixing the telegram content with the R-ORG 0xA6 (ADT = Addressed Data Telegram) to indicate that a destination address is present and including the DESTINATION EURID before the SENDER EURID as shown in Figure 32 below.

ADT	R-ORG	DATA	DESTINATION EURID	SENDER EURID	STATUS	HASH
0xA6	1 Byte	1 … 9 Byte	4 Byte	4 Byte	1 Byte	1 Byte

Figure 32 – ERP1 telegram format for addressed telegrams

A.1.1 ERP1 STATUS field format

The format of the STATUS field in ERP1 telegrams is shown in Figure 33 below.

STATUS							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CHECKSUM / CRC	RFU	PTM: GENERATION	PTM: UNIDENTIFIED / IDENTIFIED	REPEATER INFORMATION			
0b0: Checksum 0b1: CRC	0b0: RFU	0b0: PTM 1xx or no PTM 0b1: PTM 2xx	0b0: Button unidentified or no PTM (No button or more than 2 buttons) 0b1: Button identified (1 or 2 buttons)	0b0000: Original Telegram 0b0001: One-Hop Repeated Telegram 0b0010: Two-Hop Repeated Telegram 0b1111: Do Not Repeat Others: RFU			

Figure 33 – ERP1 STATUS field format

A.2 ERP2 telegram format

The ERP2 radio telegram format is shown in Figure 34 below. Fields marked in dark grey are optional and might not be present.

LENGTH	HEADER	EXT_HEADER	EXT_TYPE	SENDER EURID	DESTINATION EURID	DATA	OPTIONAL_DATA	CRC
1 Byte	1 Byte	0 / 1 Byte	0 / 1 Byte	4 / 6 Byte	0 / 4 Byte	Variable	0 / Variable	1 Byte

Figure 34 – ERP2 Telegram Format

The ERP2 telegram contains the following fields:

- LENGTH specifies the total length of the ERP2 radio telegram
- HEADER specifies the EURID types and sizes, the R-ORG that is used (based on a selection of the most common EEP) and specifies if EXT_HEADER is present
- EXTENDED_HEADER is an optional field that specifies the repeater count. It might be omitted by energy-constrained devices.
- EXTENDED_TYPE is an optional field that is used to specify less common R-ORG which are not available within the HEADER field
- SENDER EURID specifies the device address of the sender
- DESTINATION EURID is an optional field that can be used to specify the device address of the intended recipient of a data telegram
- DATA contains the telegram data
- OPTIONAL_DATA is an optional field that can be used to transmit additional data that should be treated separately from the main telegram data (optional)
- CRC is used to verify the integrity of the telegram

A.2.1 ERP2 HEADER field format

The ERP2 HEADER field specifies the source and destination addressing mode and the R-ORG used for communication. The format of the HEADER field in ERP2 telegrams is shown in Figure 35 below.

Should the R-ORG not be one of the common types, then the R-ORG is specified in the EXTENDED_TYPE field (which otherwise is not present). If the telegram is repeated, then the repeater information is provided in the EXTENDED_HEADER field (which otherwise is not present).

HEADER							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ADDRESS CONTROL			EXTENDED HEADER	R-ORG			
0b000: 24-bit Source Address, Destination Address not present 0b001: 32-bit Source Address, Destination Address not present 0b010: 32-bit Source Address, 32-bit Destination Address 0b011: 48-bit Source Address, Destination Address not present Others: RFU			0b0: Not present 0b1: Present	0b0000: 0xF6 (RPS) 0b0001: 0xD5 (1BS) 0b0010: 0xA5 (4BS) 0b0011: 0xD0 (SIGNAL) 0b0100: 0xD2 (VLD) 0b0101: 0xD4 (UTE) 0b0110: 0xD1 (MSC) 0b0111: 0x30 (SEC) 0b1000: 0x31 (SEC_ENC) 0b1001: 0x35 (SEC_TI) 0b1010: 0xB3 (GP) 0b1111: R-ORG specified in EXTENDED TYPE Others: RFU			

Figure 35 – ERP2 HEADER field format

A.2.2 ERP2 EXTENDED_HEADER field format

The EXTENDED_HEADER field contains information regarding the repeater status of the ERP2 telegram. This field may be omitted by an energy-constrained the sender of a telegram. If such telegram is repeated, then the repeater will add that field. If this field is not present, then the received ERP2 telegram is the original (not repeated) telegram.

The format of the EXTENDED_HEADER field in ERP2 telegrams is shown in Figure 36 below.

EXTENDED HEADER							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
REPEATER STATUS				RFU			

0b0000: Original Telegram (Not Repeated)
0b0001: One-Hop Repeated Telegram
0b0010: Two-Hop Repeated Telegram
0b1111: Do not repeat
Others: RFU

RFU: (Do not use)

Figure 36 – ERP2 EXTENDED_HEADER field format

A.2.3 ERP2 EXTENDED_TYPE field format

The EXTENDED_TYPE field is used to specify less commonly used R-ORG (the most common ones are encoded in the 4-bit R-ORG field of HEADER). The format of the EXT_HEADER field in ERP2 telegrams is shown in Figure 37 below.

EXTENDED TYPE							
BIT 3	BIT 2	BIT 1	BIT 0	BIT 3	BIT 2	BIT 1	BIT 0
R-ORG							

0x00: 0xC5 (SYS_EX)
0x01: 0xC6 (SmartAck Learn Request)
0x02: 0xC7 (SmartAck Learn Response)
0x03: 0x40 (CDM)
0x04: 0x32 (Decrypted Secure Telegram without R-ORG)
0x05: 0xB0 (GP Teach-in Request)
0x06: 0xB1 (GP Teach-in Response)
0x07: 0xB2 (GP Complete Data)
Others: R-ORG

Figure 37 – ERP2 EXTENDED_TYPE field format

A.3 Sub-telegrams

EnOcean radio systems use the concept of redundant sub-telegrams to increase the communication reliability. In addition to using redundant transmissions, first and second level repeaters can be used to increase communication distance and reliability as described in chapter 6.

Within this scheme, telegrams are transmitted redundantly with random (but small) delays between them. The total number of redundant sub-telegrams can be either two or three. Certain telegram types (e.g. those used in very limited energy scenarios such as SMART_ACK) do not support redundant transmission, i.e. they are transmitted only once.

If a telegram is transmitted redundantly as set of two or three sub-telegrams then the first sub-telegram is sent immediately upon receiving and processing the ESP3 command for telegram transmission. The timing offset between this first sub-telegram and the remaining (second or third) sub-telegrams is random within pre-defined time intervals.

A.3.1 Sub-telegram timing

EnOcean Radio Protocol 1 (ERP1) and EnOcean Radio Protocol 2 (ERP2) uses a repeater-level dependent time slot mechanism for the sub-telegram timing during transmission.

The sender of a radio telegram will transmit the first telegram immediately upon receiving the request for transmission. After that, the time offset (interval) between the first sub-telegram and the second sub-telegram is a random value between 1 ms and 9 ms. Likewise, the time offset (interval) between the first sub-telegram and the third sub-telegram is a random value between 20 ms and 39 ms.

First-level repeaters repeat (re-transmit) sub-telegrams that they received directly from the sender. For first-level repeaters, the time offset (interval) between the reception of an original sub-telegram and its own re-transmission of this sub-telegram is as follows:

- The first sub-telegram is transmitted after a random delay between 10 … 19 ms
- The second sub-telegram is transmitted after a random delay between 20 … 29 ms
- The third sub-telegram is transmitted after a random delay between 20 … 29 ms
  The transmission of the third sub-telegram shall start no earlier than 1 ms after the completion of the transmission of the second sub-telegram.

Second-level repeater repeat (re-transmit) sub-telegrams that they received from a first-level repeater. For second-level repeaters, the time offset (interval) between the reception of a repeated sub-telegram and its own re-transmission of this sub-telegram is as follows:

- The first sub-telegram is transmitted after a random delay between 0 … 9 ms
- The second sub-telegram is transmitted after a random delay between 20 … 29 ms
- The third sub-telegram is transmitted after a random delay between 20 … 29 ms
  The transmission of the third sub-telegram shall start no earlier than 1 ms after the completion of the transmission of the second sub-telegram.

The standard sub-telegram timing is summarized in Table 46 below. It is used both by TCM 615 and TCM 615U.

Repeater Level	Time Offset [ms] First Sub-telegram	Time Offset [ms] Second Sub-telegram	Time Offset [ms] Third Sub-telegram
0 (Original Telegram)	0	1 … 9	20 … 39
1 (Repeated for the first time)	10 … 19	20 … 29	No 3rd Sub-telegram
2 (Repeated for the second time)	0 … 9	20 … 29	No 3rd Sub-telegram

Table 46 – Standard sub-telegram timing

A.3.1.1 Reduced sub-telegram timing

Certain countries – such as Japan - have regulatory limitations for the total duration of a radio transmission in certain frequency bands including those used by EnOcean products. For these cases, a compressed sub-telegram timing has been defined. This timing is used by TCM 615J to ensure that all transmissions related to one event are finished after 50 milliseconds.

Table 47 below summarizes the compressed sub-telegram timing.

Repeater Level	Time Offset [ms] First Subtelegram	Time Offset [ms] Second Subtelegram	Time Offset [ms] Third Subtelegram
0 (Original Telegram)	0 … 1	4 … 12	14 … 22
1 (Repeated for the first time)	0 … 1	4 … 12	14 … 22
2 (Repeated for the second time)	Not Permitted in Japan due to Radio Regulation		

Table 47 – Compressed sub-telegram timing

A.3.2 Transmit (TX) maturity time

The maximum time between the request for transmission and the end of transmission of all sub-telegrams is called the Transmit (TX) Maturity Time.

In radio systems using standard sub-telegram timing, the transmit maturity time is 40 ms measured from the end of the first received sub-telegram to the end of the last received sub-telegram.

In radio systems using compressed sub-telegram timing, the transmit maturity time is 25 ms measured from the end of the first received sub-telegram to the end of the last received sub-telegram.

After the TX maturity time has elapsed, the host can be sure that all sub-telegrams corresponding to the telegram have been transmitted. In practical applications, this means for instance that an external controller can power down the transmitter after the transmit maturity time has elapsed.

A.3.3 Receive (RX) maturity time

The maximum time allowed for reception of a radio telegram is called the Receive (RX) Maturity Time. Identical sub-telegrams from the same sender are considered to belong to the same telegram if they are received within the receiver (RX) maturity time starting from the end of the first received sub-telegram.

In EnOcean radio systems using standard sub-telegram timing (as shown in Table 46), the receive maturity time is 100 milliseconds measured from the end of the first received sub-telegram to the end of the last received sub-telegram. In EnOcean radio systems using compressed sub-telegram timing (as shown in Table 47), the receive maturity time is 50 milliseconds.

A.4 Addressing

Each radio transmission within an EnOcean radio network will contain information about the originator (sender) of the transmitted radio telegram.

In addition, the intended receiver of a transmitted telegram can optionally be specified as well. Telegrams where the intended receiver is designated are called Addressed Data Telegram or ADT in short. Telegrams where the intended receiver is not designated are called Broadcast Telegrams.

Different types of addresses can be used to designate sender and receiver of an EnOcean radio telegram.

A.4.1 Address types

EnOcean radio systems support three different types of addresses:

- EnOcean Unique Radio ID (EURID)
- Base ID
- Broadcast ID

Each of these three address types corresponds to a specific address or address range as shown Figure 38 below.

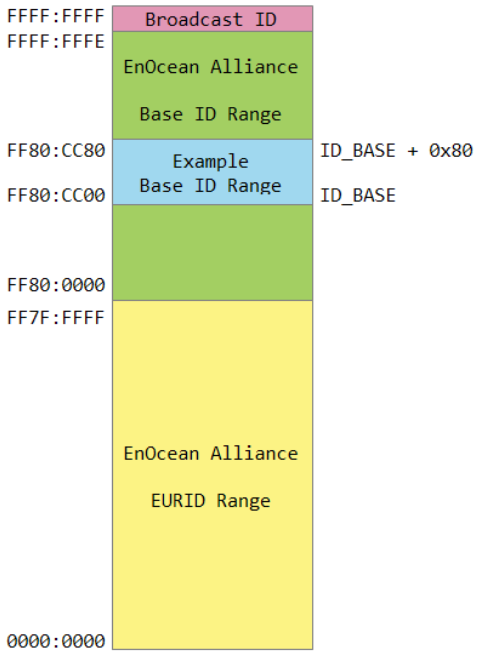

Figure 38 – Address map of EnOcean radio systems

A.4.2 EURID (Radio ID)

Each device communicating within an EnOcean radio network contains its own EnOcean Unique Radio ID (EURID) which is assigned by EnOcean Alliance. The EURID uniquely identifies each EnOcean device; no two EnOcean devices can have the same EURID.

When transmitting a radio telegram, the sender might either use the EURID or a selected Base ID (as described below) to identify itself as the originator of the telegram.

In addition, the sender might use the EURID of the intended receiver to designate this as the intended recipient of the telegram. If no receiver is designated, then the radio telegram will be transmitted as a broadcast. In this case, the receivers of such broadcast telegram decide if they accept this telegram.

A.4.3 Broadcast ID

The Broadcast ID can be used as destination address instead of the EURID of the intended receiver if a telegram should be received by more than one receiver or if the EURID of the intended receiver is unknown.

Telegrams where the destination address is the Broadcast ID are called "Broadcast Telegrams" and are commonly used by sensors and switches. The Broadcast ID is `0xFFFF:FFFF`. Note that the broadcast ID is not transmitted as part of the radio telegram.

Receivers of broadcast telegrams can decide based on the EURID of the sender (originator) of the telegram if this telegram is relevant for them or not.

A.4.4 Base ID

Normally, EnOcean devices will use their own EURID to identify themselves as the originator of transmitted telegrams. For very specific use cases, they can instead choose to use an address (ID) from within a defined range of 128 addresses. These 128 addresses are called the Base ID Range of the device.

The Base ID Range (128 addresses) of a device can be allocated anywhere in between `0xFF80:0000` and `0xFFFF:FFFE` (which represents a total range of approximately 8 million addresses). The location of the Base ID Range is defined by the start (lowest) address of the range which will always be aligned on a 7 bit (128) boundary, i.e. the last byte of the start address can be either `0x00` or `0x80`.

Note that Base ID - unlike EURID - are not guaranteed to be globally unique. Many devices with the same Base ID might exist within the EnOcean ecosystem. Having several devices using the same Base ID within a system might lead to undefined system behaviour.

Note also that the use of Base ID is not defined within the scope of secure communication, remote management or smart acknowledge. TCM 615 applications shall not use the Base ID functionality for these applications. TCM 615 supports the Base ID feature only for the purpose of backwards compatibility; it is not recommended for new designs.

A.5 Data payload

A.5.1 EnOcean Equipment Profiles (EEP)

EnOcean radio systems encode the data using so called EEP (EnOcean Equipment Profile). Each transmitter might choose one (or sometimes several) EEP for data transmission depending on the type of transmitted data.

EnOcean Equipment Profiles (EEP) are identified using three fields:

- R-ORG
  R-ORG identifies the high-level telegram type, e.g. rocker switch telegram, four-byte sensor telegram, variable length telegram etc.

- FUNCTION
  FUNCTION identifies the function group to which this telegram belongs, e.g. the function group of temperature sensors within the four-byte sensor telegram type

- VARIANT
  VARIANT identifies the exact sensor variant within the function group, e.g. a 0 °C – 40 °C temperature sensor that is defined within the function group of temperature sensors

The full EEP identifier is only transmitted within teach-in telegrams for devices using VLD and 4BS EEP. Subsequent data telegrams from the same device using the same R-ORG (VLD or 4BS) are assumed to use the FUNCTION and VARIANT values that were communicated in the teach-in telegram.

Devices transmitting telegrams with VLD R-ORG use UTE (Universal Teach-in for EnOcean devices) format for teach-in telegrams as described in chapter A.5.2.4. R-ORG, FUNCTION and VARIANT are encoded as 8-bit fields within the UTE telegram as shown in Figure 39.

EEP Identification in UTE Telegrams		
R-ORG	FUNCTION	VARIANT
0x00 … 0xFF	0x00 … 0xFF	0x00 … 0xFF
8 bit	8 bit	8 bit

Figure 39 – EEP identifier structure for UTE telegrams

Devices transmitting telegrams with 4BS R-ORG use 4BS format for teach-in telegrams. This implies that the entire teach-in information will be provided within a 4-byte telegram. For doing so, R-ORG, FUNCTION and VARIANT must be shortened to the format shown in Figure 40.

EEP Identification in 4BS Teach-in Telegrams		
R-ORG	FUNCTION	VARIANT
0x00 … 0xFF	0x00 … 0x3F	0x00 … 0x7F
8 bit	6 bit	7 bit

Figure 40 – EEP identifier structure for 4BS teach-in telegrams

A.5.2 Common R-ORG types

Within EnOcean radio telegrams, the R-ORG field identifies the telegram type as described in the previous chapter. Table 48 below lists common R-ORG types used for communication in EnOcean systems.

R-ORG	Description	Typical Use
0x30 (SEC)	Encrypted telegram without R-ORG of the original telegram	Encrypted switch telegrams
0x31 (SEC-R)	Secure message that does identify the type (R-ORG) of the encrypted telegram	Encrypted sensor telegrams
0x32 (SEC_D)	Decrypted telegram without R-ORG	Decrypted switch telegrams
0x33 (SEC_CDM)	Secure chained messages	Encrypted sensor telegrams requiring chaining due to length
0x35 (SEC_TI)	Secure teach-in telegram	Setup of a secure communication channel
0xA5 (4BS)	4-byte sensor telegram	Sensor telegrams using 4 byte payload
0xA6 (ADT)	Addressed data telegram	Telegrams that specify the address of the intended receiver
0xC5 (SYS_EX)	Remote management telegram	Configuration of functional parameters in the receiver
0xD0 (SIGNAL)	Signal telegram	Reporting of system parameters
0xD1 (MSC)	Manufacturer-specific content	Manufacturer-defined telegrams
0xD2 (VLD)	Variable length telegram	Variable length telegrams requiring more than 4 byte of payload
0xD4 (UTE)	Universal Teach-in for EnOcean devices	Teach-in telegram for devices not using 4BS telegram type (e.g. VLD)
0xD5 (1BS)	1-byte sensor telegram	Simple sensors with 1 byte payload such as contact sensors
0xF6 (RPS)	Rocker and pushbutton switches	Rocker switches or push buttons

Table 48 – Common R-ORG used in EnOcean radio systems

For full details about EnOcean Equipment Profiles (EEP) please refer to the EnOcean Equipment Profiles specification [5].

A.5.2.1 1BS telegram

1 Byte Sensor (1BS) telegrams are identified by the R-ORG field being set to 0xD5 which is followed by one byte of payload (Bit0 … Bit7. 1BS telegrams are used exclusively to encode the status (open / closed) of a binary contact (typically a magnet contact)

The payload of 1BS telegrams encodes either the contact status (1BS Data Telegram) during normal operation or identifies a teach-in telegram (1BS Teach-in Telegram).

The distinction between data and teach-in telegrams is made based on the status of Bit4. If this bit is set to 0 then the telegram is a 1BS Teach-in Telegram; if this bit is set to 1 then the telegram is a 1BS Data Telegram.

A.5.2.2 4BS telegram

4 Byte Sensor (4BS) telegrams are identified by the R-ORG field being set to 0xA5 which is followed by four bytes of payload (Bit0 … Bit31).

The payload of 4BS telegrams encodes either the sensor status (4BS Data Telegram) during normal operation or identifies EEP and manufacturer of the device during teach-in (4BS Teach-in Telegram).

The distinction between data and teach-in telegrams is made based on the status of Bit28. If this bit is set to 0 then the telegram is a 4BS Teach-in Telegram; if this bit is set to 1 then the telegram is a 4BS Data Telegram.

A.5.2.3 VLD telegram

Variable Length Data (VLD) telegrams are identified by the R-ORG field being set to 0xD2. They carry a variable length payload which can be between 1 and 14 byte long.

A.5.2.4 UTE (Universal Teach-in with EEP) telegram

Variable Length Data (VLD) telegrams carry a variable length payload, therefore it is not possible to use one bit at a pre-defined location to distinguish between data and teach-in telegrams.

Devices communicating using VLD data telegrams therefore use the generic Universal Teach-in with EEP (UTE) format when transmitting a teach-in telegram. The format of such UTE telegram is shown in Figure 41 below.

UTE R-ORG	UTE DATA						
BYTE0	BYTE6	BYTE5	BYTE4	BYTE3	BYTE2	BYTE1	BYTE0
0xD4	CONTROL	CHANNEL	MANUFACTURER_ID		VARIANT	FUNCTION	R-ORG

Figure 41 – UTE telegram structure

UTE telegrams are identified by their R-ORG 0xD4 and contain the following data:

- CONTROL
- CHANNEL
- MANUFACTURER_ID
- EEP (R-ORG, FUNCTION, VARIANT)

A.5.2.4.1 CONTROL

The CONTROL field identifies the type of the UTE request using the structure shown in Figure 42 below.

CONTROL							
BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
DIRECTION	RESPONSE	REQUEST_TYPE		COMMAND_ID			

Figure 42 – Structure of the CONTROL field of the UTE telegram

The different sub-fields within the CONTROL field are described in Table 49 below. The most common setting for the CONTROL field is 0x40 which identifies a uni-directional teach-in request with no expected response.

Field	Description	Supported Values
DIRECTION	Specifies if uni-directional or bi-directional teach-in is used	0b0: Uni-directional teach-in 0b1: Bi-directional teach-in
RESPONSE	Specifies if a response is expected to this UTE telegram	0b0: Response expected 0b1: No response expected
REQUEST_TYPE	Specifies if this UTE telegram is a teach-in request (to add a device) or a teach-out request (to delete a device)	0b00: Teach-in request (add device) 0b01: Teach-out request (delete device) 0b10: Unspecified (teach-in or teach-out) 0b11: Reserved (do not use)
COMMAND_ID	Specifies if this UTE telegram is a request or a response	0b0000: Request 0b0001: Response Others: Reserved (do not use)

Table 49 – Data fields in the CONTROL field of the UTE telegram

A.5.2.4.2 CHANNEL

The CHANNEL field of the UTE telegram allows specifying if a device should be added to a specific functional domain of the target device (for instance to a specific group of devices if more than one group is supported).

In most cases, this field is set to 0xFF indicating that the device should be added to all functional domains of the target device.

A.5.2.4.3 MANUFACTURER_ID

The MANUFACTURER_ID field contains the Manufacturer ID assigned by EnOcean Alliance in little endian byte order. The Manufacturer ID of EnOcean GmbH is 0x000B.

A.5.2.4.4 EEP (R-ORG, FUNCTION, VARIANT)

The R-ORG, FUNCTION and VARIANT fields of the UTE telegram identify the EEP used by the device as described in chapter A.5.1.

A.5.2.5 SIGNAL telegram

SIGNAL telegrams are used to encode generic system conditions independent of specific sensor functionality of the device. Examples of such system conditions are internal energy level, available ambient energy and backup battery status.

SIGNAL telegrams are identified by having the R-ORG field of the data telegram set to 0xD0. After that, the SIGNAL type (what is reported) is identified by the 1 byte long MID field which is followed by the data corresponding to this SIGNAL type. Figure 43 below shows the structure of a SIGNAL telegram.

SIGNAL RORG	SIGNAL Type (MID)	SIGNAL Data
0xD0	0x00 ... 0xFF	Depending on SIGNAL Type

Figure 43 – SIGNAL Telegram Structure

Table 50 below lists common SIGNAL types with their reported data.

MID	Content	Data
0x06	Energy status (remaining energy)	1 byte integer value (expressing %) Valid values: 0 … 100
0x0D	Energy delivery of the harvester	1 byte Enumeration Valid values: 0x00 (best) ... 0x04 (worst)
0x0E	Radio disabled	Transmitted upon disabling the radio (For instance when entering standby mode) No additional data
0x0F	Radio enabled	Transmitted upon enabling the radio (For instance when exiting standby mode) No additional data
0x10	Backup battery status	1 byte integer value (expressing %) Valid values: 0 … 100

Table 50 – Common SIGNAL Types

A.5.3 Data payload size

For ERP1 radio telegrams, the maximum permitted data payload size (excluding R-ORG) is 14 bytes for the case of standard broadcast telegrams.

For ERP2 radio telegrams, TCM 615U internally limits the maximum payload size (excluding R-ORG) to 14 byte (TCM 615U) and 34 bytes (TCM 615J) to reduce the collision risk as result of longer payload sizes.

If the radio telegram contains security information such as the RLC value, authentication signature or the intended destination address, then the maximum data payload of one EnOcean radio telegram is reduced by the size of the additional data.

If the telegram data payload exceeds the maximum available data payload, then it must be transmitted as a chain of radio telegrams which together transfer the message payload.

The type of chaining that is used depends on the type of telegram that is transmitted. Standard telegrams are transmitted as Chained Data Messages (CDM) while secure telegrams are transmitted as Secure Chained Data Messages (SEC_CDM).

A.6 Telegram chaining

Telegram chaining is a feature that allows transmission of a data payload that is larger than the maximum supported payload.

For the transmission of a telegram with a data payload larger than 14 byte (TCM 615 and TCM 615U) or larger than 34 byte (TCM 615J), the payload is distributed (segmented) across several telegrams using the telegram structure shown below.

Upon reception, the payload of the received telegrams is combined (reassembled) into the original telegram and forwarded to the host via the ESP3 interface once the last telegram in the chain has been received.

A.6.1 Telegram chaining for broadcast telegrams

Chained broadcast telegrams can be identified by the R-ORG 0x40 (CDM). The first telegram in a chain (with IDX = 0b000000) uses the CHAIN_LEN field to specify the total length of the DATA payload that is transported by this chain. Figure 44 below shows the structure of the first telegram in a chain of broadcast telegrams.

0x40 (CDM)	CHAIN_CTRL		CHAIN_LEN	RORG	DATA	SENDER EURID	STATUS	CRC / HASH
	ID	IDX						
1 Byte	1 Byte		2 Byte	1 Byte	10 Byte	4 Byte	1 Byte	1 Byte

Figure 44 – Structure of the first telegram in a chain of broadcast telegrams

Subsequent telegrams in the chain (with IDX > 0b000000) omit the CHAIN_LEN field as shown in Figure 45 below.

0x40 (CDM)	CHAIN_CTRL		RORG	DATA	SENDER EURID	STATUS	CRC / HASH
	ID	IDX					
1 Byte	1 Byte		1 Byte	1 … 12 Byte	4 Byte	1 Byte	1 Byte

Figure 45 – Structure of subsequent telegrams in a chain of broadcast telegrams

Up to 4 telegram chains from the same sender can be in progress at any time. The individual chains are identified by the 2 bit wide ID field. Telegrams having the same ID field setting are considered to be part of the same chain.

The order of the telegrams within each chain are identified by the 6 bit IDX field with the first telegram using IDX = 0b000000, the second telegram IDX = 0b000001 and so on. The maximum length of a telegram chain is therefore 64 telegrams.

The theoretical maximum DATA length within a chain of telegrams is 766 byte (63 * 12 byte + 1 * 10 byte). Note that in TCM 615 the maximum length is limited by the maximum size of an ESP3 command accepted by TCM 615 which is 255 byte.

A.6.2 Telegram chaining for addressed telegrams (ADT)

Chained addressed telegrams extend the format of chained broadcast telegrams by adding the R-ORG 0xA6 (Addressed Data Telegram) at the begin of the message and EURID of the intended receiver of the message before the EURID of the sender.

Figure 46 below shows the structure for the first telegram in a chain of addressed telegrams.

0xA6 (ADT)	0x40 (CDM)	CHAIN_CTRL		CHAIN_LEN	RORG	DATA	DESTINATION EURID	SENDER EURID	STATUS	CRC / HASH
		ID	IDX							
1 Byte	1 Byte	1 Byte		2 Byte	1 Byte	5 Byte	4 Byte	4 Byte	1 Byte	1 Byte

Figure 46 – Structure of the first telegram in a chain of addressed telegrams

Subsequent telegrams in a chain of addressed telegrams omit both the CHAIN_LEN and the R-ORG field as shown in Figure 47 below.

0xA6 (ADT)	0x40 (CDM)	CHAIN_CTRL		DATA	DESTINATION EURID	SENDER EURID	STATUS	CRC / HASH
		ID	IDX					
1 Byte	1 Byte	1 Byte		1 … 8 Byte	4 Byte	4 Byte	1 Byte	1 Byte

Figure 47 – Structure of subsequent telegrams in a chain of addressed telegrams

A.6.3 Telegram chaining for secure telegram (SEC_CDM)

Chained secure telegrams – identified by R-ORG 0x33 (SEC_CDM) - extend the format of chained broadcast telegrams by defining three different telegram structures – one for the first telegram in a chain, one for the last telegram in a chain and one for all telegrams in between the first and the last.

Figure 48 shows the structure for the first telegram in a chain of secure telegrams.

0x33 (SEC_CDM)	CHAIN_CTRL		CHAIN_LEN	RORG	DATA	SENDER EURID	STATUS	CRC / HASH
	ID	IDX						
1 Byte	1 Byte		2 Byte	1 Byte	10 Byte	4 Byte	1 Byte	1 Byte

Figure 48 – Structure of the first telegram in a chain of secure telegrams

Intermediary telegrams in a chain of secure telegrams omit both the CHAIN_LEN and the R-ORG field as shown in Figure 49 below.

0x33 (SEC_CDM)	CHAIN_CTRL		DATA	SENDER EURID	STATUS	CRC / HASH
	ID	IDX				
1 Byte	1 Byte		1 … 13 Byte	4 Byte	1 Byte	1 Byte

Figure 49 – Structure of intermediary telegrams in a chain of secure telegrams

The last telegram of the chain contains the rolling code (RLC) value and the message signature (CMAC) as shown in Figure 50 below. Note that the last telegram in a chain of secure telegrams might have no data payload (if the data exactly fits into the previous telegram in the chain).

0x33 (SEC_CDM)	CHAIN_CTRL		DATA	CMAC	RLC	SENDER EURID	STATUS	CRC / HASH
	ID	IDX						
1 Byte	1 Byte		0 … 5 / 7 Byte	3 / 4 Byte	3 / 4 Byte	4 Byte	1 Byte	1 Byte

Figure 50 – Structure of the last telegram in a chain of secure telegrams

A.6.4 Telegram chaining for addressed secure telegram (ADT SEC_CDM)

Chained secure telegrams may also be transmitted as addressed telegram (ADT) to identify the intended receiver of this telegram chain.

Chained addressed secure telegrams extend the format of chained secure telegrams by adding the R-ORG 0xA6 (Addressed Data Telegram) at the begin of the message and EURID of the intended receiver of the message before the EURID of the sender. Figure 51 below shows the structure for the first telegram in a chain of secure telegrams.

0xA6 (ADT)	0x33 (SEC_CDM)	CHAIN_CTRL		CHAIN_LEN	RORG	DATA	DESTINATION EURID	SENDER EURID	STATUS	CRC / HASH
		ID	IDX							
1 Byte	1 Byte	1 Byte		2 Byte	1 Byte	5 Byte	4 Byte	4 Byte	1 Byte	1 Byte

Figure 51 – First telegram in a chain of addressed secure telegrams

Intermediary telegrams in a chain of secure telegrams omit both the CHAIN_LEN and the R-ORG field as shown in Figure 52 below.

0xA6 (ADT)	0x33 (SEC_CDM)	CHAIN_CTRL		DATA	DESITNATION EURID	SENDER EURID	STATUS	CRC / HASH
		ID	IDX					
1 Byte	1 Byte	1 Byte		1 … 8 Byte	4 Byte	4 Byte	1 Byte	1 Byte

Figure 52 – Intermediary telegrams in a chain of addressed secure telegrams

The last telegram of the chain contains the rolling code (RLC) value and the message signature (CMAC) as shown in Figure 53 below.

0xA6 (ADT)	0x33 (SEC_CDM)	CHAIN_CTRL		DATA	CMAC	RLC	DESTINATION EURID	SENDER EURID	STATUS	CRC / HASH
		ID	IDX							
1 Byte	1 Byte	1 Byte		0 … 5 / 7 Byte	3 / 4 Byte	3 / 4 Byte	4 Byte	4 Byte	1 Byte	1 Byte

Figure 53 – Last telegram in a chain of addressed secure telegrams

Note that the encapsulation as addressed (ADT) telegram is applied after the SEC_CDM telegram has been formed. The last SEC_CDM telegram might therefore be split into two addressed SEC_CDM telegrams due to the addition of the R-ORG and DESTINATION EURID addressing fields resulting in a telegram size larger than the maximum size of EnOcean radio telegrams.

B. Introduction to EnOcean security protocol

This chapter gives a high-level introduction to key aspects of the security protocol used in EnOcean radio networks to help understanding the security features provided by TCM 615.

The content of this chapter is provided for information only; it is not part of the TCM 615 product specification and shall not be considered as assured characteristics of TCM 615.

Refer to the EnOcean Alliance Security Specification for a detailed up to date description of all features.

B.1 Goals of secure radio communication

Secure radio communication aims to address two main issues:

- Unauthorized interception (reception and correct interpretation) of transmitted data
  In doing so, a third (unauthorized) party can understand the content of a received content.

- Unauthorized transmission of radio telegrams
  In doing so, a third (unauthorized) party can transmit a radio telegram that is treated by a receiver as valid request.

Somewhat loosely speaking, the goal of security is to prevent an unauthorized person (often referred to as an *Attacker*) both from learning about the current state of system parameters and from actively changing such parameters.

These goals can be achieved via techniques such as telegram encryption, telegram authorization and dynamic modification. All three techniques will be reviewed in the subsequent chapters for reference.

B.2 Telegram encryption

The goal of telegram encryption is to prevent unauthorized receivers from correctly interpreting the content of a telegram.

To do so, the original (plain text) data is *encrypted* with a *security key* thus transforming it into encrypted, unreadable data. Only when the correct key is known it is possible to transform – *decrypt* - the encrypted data into readable data again. Figure 54 below shows the concept.

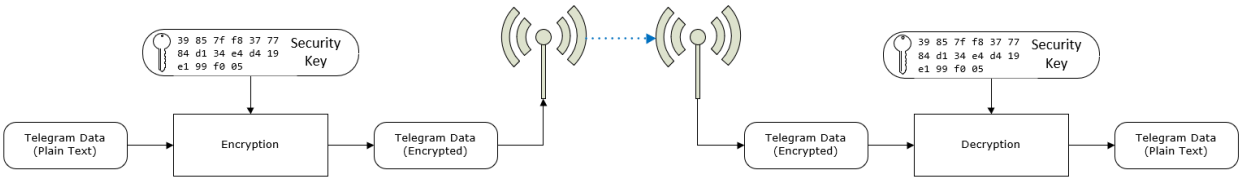

Figure 54 – Telegram encryption

If the same security key is used for encryption at the sender and decryption at the receiver, then this is called a *symmetric key* algorithm. AES (AES128 / AES256) and DES / 3DES algorithms are typical examples of this category. TCM 615 uses this approach.

If different security keys are used for encryption at the sender and decryption at the receiver, then this is called an *asymmetric key* algorithm or a *public key* algorithm. Public / private key algorithms such as PGP, GPG or TLS fall into this category. TCM 615 does not support asymmetric key algorithms.

B.3 Telegram authentication

The goal of telegram authentication is to prevent unauthorized senders to transmit apparently valid commands causing the receiver to perform unauthorized actions. Telegram authentication is typically used in conjunction with telegram encryption.

Telegram authentication works by creating a *signature* (often called *Cipher-based Message Authentication Code* or C*MAC* in short) based on the content of the telegram and the security key.

Essentially, the telegram data is transformed via a defined algorithm using the security key into a unique, fixed size signature (where typical signature lengths include 24-bit, 32-bit, 64-bit, 512-bit and 1024-bit) which identifies this specific message.

For an optimal signature algorithm, the likelihood of two different telegrams creating the same telegram signature should be inversely proportional to the signature size, so for instance for 24-bit signatures the likelihood should be one in 16 million and for 32-bit signatures it should be one in 4 billion.

Conceptually the correspondence between telegram content and telegram signature is like the one between a person and a fingerprint:

- Each person has a unique fingerprint. Based on a given person one can determine her or his fingerprint
- Based on a given fingerprint one can check if it originated from a given person
- Based on the fingerprint one cannot determine any other properties of the person

For telegram authentication purposes, the telegram signature (CMAC) is usually appended to the telegram content so that the telegram content and the telegram signature are transmitted together.

When the receiver receives such a telegram, it will itself calculate the expected telegram signature (CMAC) based on the security key and the telegram content. The receiver then compares the signature that it calculated with the signature it received as part of the telegram.

If both signatures are the same, then the receiver can establish two important facts:

1. The telegram originates from a sender knowing the security key
2. The content of the telegram has not been modified after the sender added the signature to it

Figure 55 below illustrates the concept of telegram authorization via a telegram signature.

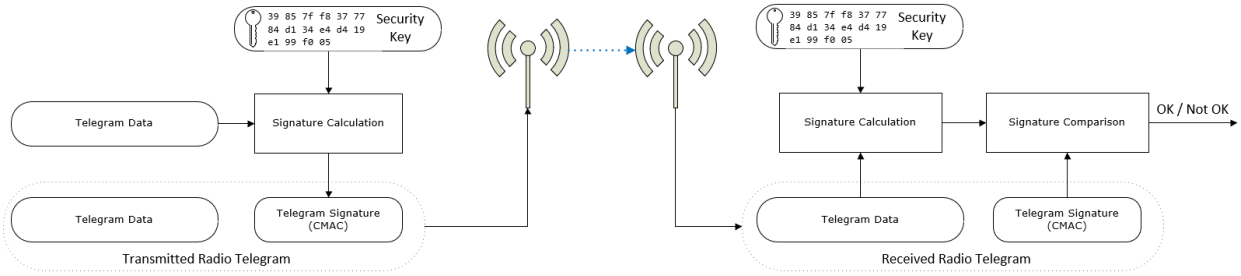

Figure 55 – Telegram authentication via telegram signature

B.4 Replay protection

One fundamental problem with both telegram encryption and telegram authorization is that using the same input data (plain text) with the same security key will always result in the same encrypted data and the same signature. This enables attacks based on monitoring previous system behaviour. If an attacker has observed that a certain data telegram results in a certain light being turned on, then he could use this information to identify - or even actively send - similar telegrams in the future. This type of attack is often called *Replay Attack* since it works by reusing (replaying) previously transmitted (valid) data telegrams.

To prevent this type of attack, either the telegram data or the security material (e.g. the security key or the initialization vector / nonce) must change to ensure that identical input data does not create identical encrypted radio telegrams.

Such change is commonly done based on a sequence of values that are guaranteed to be unique so that the same value will not be used twice. This sequence of changing values is often referred to as *Rolling Code* or *RLC* in short.

To prevent replay of an already received message, the receiver will keep track of the latest received RLC value and will only accept telegrams with a previously unused (higher than the last) RLC value.

Both sender and receiver must know the mechanism how to generate the next RLC (the next value in the sequence) based on the current RLC (the current value of the sequence). The simplest - and most common - approach for that is to use the value of a monotonously incrementing counter that is incremented with each transmitted telegram.

The value of the RLC can either be provided as part of the data telegram (explicit RLC) or be determined by the receiver based on the initial RLC value and the number of received telegrams (implicit RLC).

Figure 56 below shows the concept of providing the RLC value as part of the telegram data (explicit RLC).

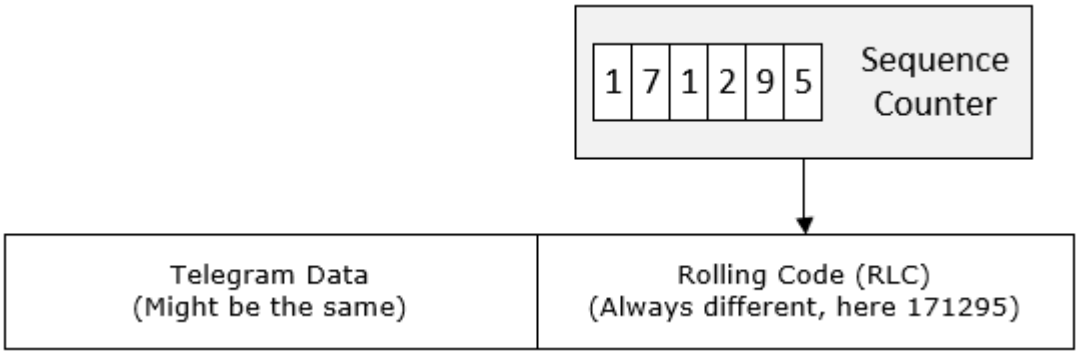

Figure 56 – Transmission of the current RLC value as part of the telegram data

The combination of security key and RLC value are used both for encryption / decryption and for authentication / verification. Figure 57 below illustrates this.

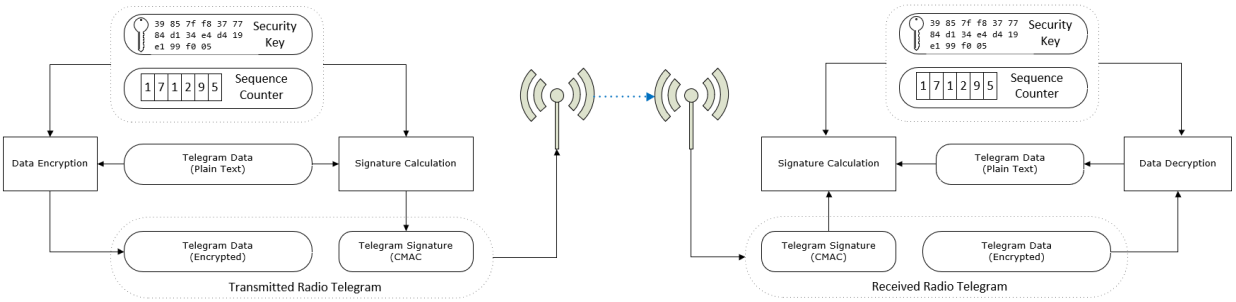

Figure 57 – Encryption and authentication in TCM 615

B.4.1 RLC and security key in bi-directional communication

If the communication between two devices (*Device1* and *Device2*) is bi-directional, i.e. each device can either transmit or receive telegrams, then two independent RLC must be used (since the number of telegrams one direction might be different from the number of telegrams in the other direction) and two different security Keys may be used (using the same key in both directions would also be possible).

From the perspective of Device1, the first pair (RLC_L, Key_L) is the local security material used by the local device (Device1) to transmit telegrams to the remote device (Device2). The second pair (RLC_R, KEY_R) is the remote security material which is used by Device2 to transmit telegrams to Device1.

Figure 58 below illustrates that.

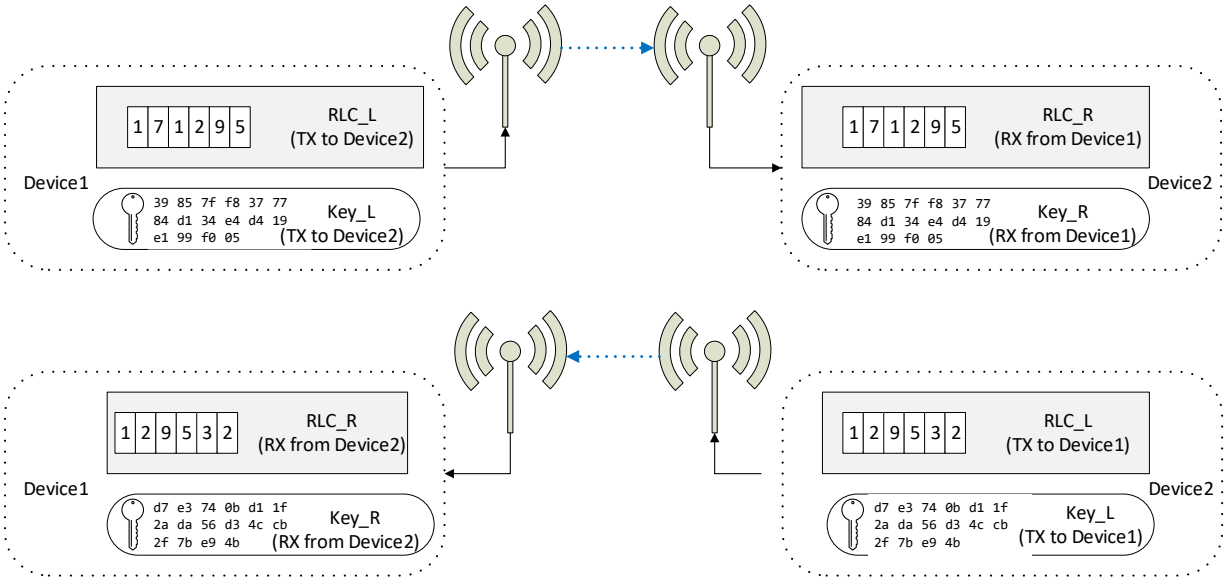

Figure 58 – Security key and RLC usage in bi-directional communication

B.4.2 RLC synchronization between sender and receiver

For encryption and authentication using RLC, it is important that the RLC on the transmitter side and the RLC on the receiver side remain synchronized, i.e. that they always have the same value.

This can be ensured either by transmitting the RLC as part of the telegram (this is called *explicit RLC mode*) or by tracking the expected RLC when it is not transmitted as part of the telegram (this is called *implicit RLC mode*).

Explicit RLC mode is the recommended procedure since it ensures that the receiver always knows the current RLC used by the sender; it requires however to increase the size of the telegram to transmit this RLC.

Implicit RLC mode might be used in energy-constrained systems where there might not be enough energy to additionally transmit the current RLC as part of the telegram.

For implicit RLC mode, the initial value of the RLC at the sender and at the receiver will be aligned during the establishment of the secure communication so that the receiver knows the current RLC used by the sender. For systems using TCM 615, this can be done either via a dedicated secure teach-in telegram as described in chapter 7.7.2 or via the ESP3 interface as described in Chapter 7.7.3.

After that, both sender and receiver will adjust (increment for the case of using a sequence counter to generate the RLC) the RLC for each telegram that is transmitted to this specific receiver (RLC adjustment in the sender) or received from this specific sender (RLC adjustment in the receiver).

To guard against the case of telegrams being lost (not received by the receiver), the receiver will check if the RLC it assumes is used in the received telegram will result in a matching message signature (CMAC) when executing telegram authentication using this RLC together with the security key.

If this is the case, then the receiver will decrypt the telegram content using this RLC together with the security key. If this is not the case, then the receiver can retry using the next RLC in the sequence and so on. Typically, a maximum number of future RLC values to be tried will be defined. This parameter is often referred to as the *Rolling Code Window Size*.

If message decryption based on a future RLC is successful then the RLC used by the receiver will be updated to this value, thereby re-synchronizing the transmitter and receiver RLC. If no matching RLC is found within the rolling code window, then the message cannot be decrypted and authenticated and might be forwarded to the host for further analysis.