System Specification

enocean alliance
Building Smarter Connectivity

Generic Profiles

V 1.4

San Ramon, CA, USA, June 20, 2013

Executive Summary

This document provides the specification of Generic Profiles. The full specification includes the Generic Profiles appendix document.

Generic Profiles are the successor of EnOcean Equipment Profiles and targets the shortcomings of it. Both EnOcean Equipment Profiles and Generic Profiles describe the data communication of products utilizing The EnOcean Radio Protocol and enables manufacturers to develop interoperable products. The strength of Generic Profiles is to enable devices to have self-described dynamic communication.

With this capability, new products can be developed without submission of its profile to the EnOcean Alliance, allowing an unlimited variety of possibilities.

Ver.	Editor	Change	Date
1.0	MH	First release	20.6.2013
1.1	MH	Final release, remove chapter of development, no change in content	23.07.2018
1.2	MH	Editiorial corrections – Teach In Info Signal corrected to 8 bits.	10.12.2019
1.3	AP	Typo's in example and telegram chaining corrected, added and updated links	28.10.2020
1.4	AP	Added CDM example for ERP2	21.05.2021

The EnOcean Generic Profiles Specification is available free of charge to companies, individuals and institutions for all non-commercial purposes (including educational research, technical evaluation and development of non-commercial tools or documentation.)

This specification includes intellectual property („IPR“) of the EnOcean Alliance and joint intellectual properties („joint IPR“) with contributing member companies. No part of this specification may be used in development of a product or service for sale without being a participant or promoter member of the EnOcean Alliance and/or joint owner of the appropriate joint IPR. EnOcean Alliance grants no rights to any third party IP, patents or trademarks.

These errata may not have been subjected to an Intellectual Property review, and as such, may contain undeclared Necessary Claims.

EnOcean Alliance Inc.
5000 Executive Parkway, Suite 302
San Ramon, CA 94583
USA
Graham Martin Chairman & CEO EnOcean Alliance

Table of content

1. Introduction

1.1. Introduction

EnOcean GmbH developed the structure of the EnOcean Equipment Profiles (EEP) to achieve a standardized communication between devices applying EnOcean's energy harvesting and wireless technology. Based on this structure the EnOcean Alliance created and maintains a system specification by its Technical Working Group (TWG). This specification summarizes all profiles for different application and implementation scenarios developed by the members of the EnOcean Alliance.

A growing number of EEPs and an even faster time-to-market requirement created the need for new communication architecture between the various devices within an EnOcean wireless infrastructure.

In November 2010 the EnOcean Alliance tasked a team within the TWG to draft a communication architecture which can overcome the challenges seen for the upcoming three to five years. Two objectives were followed up by the team – (1) a communication architecture able to handle the large variety of sensors and actuators without creating a complex system, and (2) a communication architecture, which requires an administrative effort much lower than today's EEP-scheme.

Contributions to this team were made by

- Ad Hoc Electronics LLC, USA
- alphaEOS GmbH, Germany
- EnOcean GmbH, Germany
- EnOcean Inc., USA
- Kieback&Peter GmbH & Co. KG, Germany
- Probare GmbH, Germany
- Servodan A/S, Denmark
- Thermokon Sensortechnik GmbH, Germany

1.2. "Generic" – definition

The ideal objective was a "generic specification" which could mean: a device of a manufacturer communicates with a device of another manufacturer and the ability to exchange data is possible.

The pre-requisite for such a "worry-free" communication is either a long "synchronization period" which requires a non-restricted energy source or a well-defined communication architecture which enables both devices to exchange information in a carefully structured way, without imposing unnecessary limits on the designers.

Thus, the degree to which a specification will allow for a "generic" implementation of devices depends largely on the intelligence invested in the definition of such a communication architecture and language.

The EnOcean Alliance decided to aim for an architecture, which minimizes the overhead for product designers and provides enough flexibility for the next three to five years.

This document specifies the communication architecture and the language, which can be applied by the members of the EnOcean Alliance for their future product implementations.

1.3. Terms & Abbreviations

4BS	**4 b**ytes **S**ensor telegram
ADC	**A**nalog-to-**d**igital **c**onverter
ADT	**A**ddressed **D**estination **T**elegram
API	**A**pplication **P**rogramming **I**nterface
CDM	**C**hained **D**ata **M**essage
EEP	**E**nOcean **E**quipment **P**rofiles
ERP	**E**nOcean **R**adio **P**rotocol
ESP	**E**nOcean **S**erial **P**rotocol
FCC	**F**ederal **C**ommunications **C**ommission
FW	**F**irm**w**are
GP	**G**eneric **P**rofiles
Message	Communication entity consisting of one or more telegrams.
OSI	**O**pen **S**ystems **I**nterconnection Reference Model
RMCC	**R**emote **M**anagement **C**ontrol **C**ommand
RPC	**R**emote **P**rocedure **C**all
R-ORG	**R**adio **Org**anization number for EnOcean radio telegram types
TWG	**T**echnical **W**orking **G**roup of the EnOcean Alliance
Inbound	Incoming - incoming communication from described device perspective
Outbound	Outgoing - outgoing communication from described device perspective

2. Communication layers

2.1. Introduction

Computer network protocols are using abstraction layers for hiding implementation details for a particular set of functionality. To confine the tasks, abstraction layers for Generic Profiles communication are applied.

Generic Profile communication defines the following layers, similar to the OSI layer model:

Layer	Services
Application	Product specific software application / Generic Profile message generation
Presentation	Radio telegram processing
Session	Not used for Generic Profiles
Transport	Not used for Generic Profiles
Network	Addressing telegrams / R-ORG / Status processing

FIGURE 2.1: LAYER MODEL OF GENERIC PROFILES

Adopting such a view of the tasks one will be independent from the radio, serial or any other communication type to exchange the Generic Profiles messages.

2.2. Message types

Generic Profiles define four different message types as described in the following table:

Message type	Properties	Restrictions
Teach-in request	Generic Profiles Teach-in request	512 bytes length
Teach-in response	Response to a Generic Profiles Teach-in request message (if bidirectional communication)	512 bytes length
Complete data	Data message containing complete measurement data	512 bytes length
Selective data	Data message containing selected parts of measurement data	512 bytes length

TABLE 2.1: TYPE OF MESSAGES DEFINED BY GENERIC PROFILES

Each message can be addressed to a destination ID (ADT).

2.3. Radio communication

For the radio communication of Generic Profiles the following layers are defined:

Layer	Services
Application	Generates Generic Profiles message as a bit stream and determines message type
Generic Profiles API	Selects R-ORG and translates message to one or more radio telegrams
Dolphin API	Sends radio telegram(s)
EnOcean Dolphin chip	Physical radio telegram transmission

FIGURE 2.2: LAYER MODEL OF RADIO COMMUNICATION

Telegram summary

In the layer "Generic Profiles API" the R-ORG of EnOcean radio telegrams will be selected depending on the message type to transmit. If the message exceeds the length of one telegram then the message will be split into the necessary number of telegrams by telegram chaining mechanisms described in chapter 2.3.1.

R-ORG	Telegram type	Properties
0xB0	GP_TI = Teach-in request	Teach-in message up to 512 bytes length. **Allowed telegram chaining:** yes **Broadcast:** yes **Unicast:** no
0xB1	GP_TR = Teach-in response	Response to a Teach-in message up to 512 bytes length. **Allowed telegram chaining:** yes **Broadcast:** no **Unicast:** yes
0xB2	GP_CD = Complete Data	Contains all channel data up to 512 bytes payload. **Allowed telegram chaining:** yes **Broadcast:** yes **Unicast:** yes
0xB3	GP_SD = Selective data	Data message containing parts of measurement data. **Allowed telegram chaining:** yes **Broadcast:** yes **Unicast:** yes

TABLE 2.2: R-ORG APPLIED WITHIN GENERIC PROFILES

2.3.1. Telegram chaining

Chained radio telegrams are required for a Generic Profiles message payload exceeding the payload of one telegram (on EnOcean Radio Protocol 1 maximum payload is 13 bytes or 9 bytes, if it is an ADT message).

Such a message will be split into the necessary number of telegrams by the EnOcean radio stack.

Example for a chained complete data message with 23 bytes payload with EnOcean Radio Protocol 1 telegram:

R-ORG CDM	SEQ	IDX	LEN	R-ORG GP_CD	data field 1st part of message	originator id	status	crc8
1 byte	2 bit	6 bit	2 bytes	1 byte	10 bytes	4 bytes	1 byte	1 byte
	1 byte							
0x40	0x40		23	0xB2	1 2 3 4 5 6 7 8 9 10	0xnnnnnnnn	0xnn	0xnn

FIGURE 2.3: RADIO TELEGRAM STRUCTURE OF FIRST CHAINED TELEGRAM – NOT ADDRESSED

R-ORG CDM	SEQ	IDX	data field 2nd part of message	originator id	status	crc8
1 byte	2 bit	6 bit	13 bytes	4 bytes	1 byte	1 byte
	1 byte					
0x40	0x41		11 12 13 14 15 16 17 18 19 20 21 22 23	0xnnnnnnnn	0xnn	0xnn

FIGURE 2.4: RADIO TELEGRAM STRUCTURE OF SECOND OR FURTHER CHAINED TELEGRAM – NOT ADRESSED

Field	Value	Description
SEQ	0bnn	SEQ code is the same on every telegram belonging to the same message. SEQ code changes from message to message. SEQ 0b00 is not allowed.
IDX	0bnnnnnn	The order of the telegrams. The first telegram receives IDX=0, the second IDX=1 and so on.
LEN	0xnnnn	The total amount of bytes contained in data fields

NOTE:

For detailed explanation of the fields and process please look up the:

Remote Management Specification (Chapter SYS_EX telegram structure):
https://www.enocean-alliance.org/reman/

EnOcean Radio Protocol 1 Specification:
http://www.enocean.com/fileadmin/redaktion/pdf/tec_docs/EnOceanRadioProtocol.pdf

Example for a chained Teach-in request message with 51 bytes payload with EnOcean Radio Protocol 2 telegram:

header	Ext. Telegram type	originator id	SEQ	IDX	LEN	R-ORG GP_CD	data field 1st part of message	crc8
1 byte	1 byte	4 bytes	2 bit	6 bit	2 bytes	1 byte	24 bytes	1 byte
			1 byte					
0x2F	0x03	0xnnnnnnnn	0x40		51	0xB0	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24	0xnn

FIGURE 2.5: RADIO TELEGRAM STRUCTURE OF FIRST CHAINED TELEGRAM – NOT ADDRESSED

header	Ext. Telegram type	originator id	SEQ	IDX	data field 2nd part of message	crc8
1 byte	1 byte	4 bytes	2 bit	6 bit	27 bytes	1 byte
			1 byte			
0x2F	0x03	0xnnnnnnnn	0x41		25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51	0xnn

FIGURE 2.6: RADIO TELEGRAM STRUCTURE OF SECOND OR FURTHER CHAINED TELEGRAM – NOT ADRESSED

Field	Value	Description
Header	0x2F	Originator-ID 32-bit, no Destination-ID No extended header Extended telegram type available
SEQ	0bnn	SEQ code is the same on every telegram belonging to the same message. SEQ code changes from message to message. SEQ 0b00 is not allowed.
IDX	0bnnnnnn	The order of the telegrams. The first telegram receives IDX=0, the second IDX=1 and so on.
LEN	0xnnnn	The total amount of bytes contained in data fields

NOTE:

For detailed explanation of the fields and process please look up the:

Remote Management Specification (Chapter SYS_EX telegram structure):
https://www.enocean-alliance.org/reman/

EnOcean Radio Protocol 2 Specification:
https://www.enocean.com/fileadmin/redaktion/pdf/tec_docs/EnOceanRadioProtocol2.pdf

2.4. Other communication types

The concept of using messages instead of defining telegrams provides the opportunity to use Generic Profiles with other communication types than radio.

The available layers may be expanded in the future as needed (e.g. for serial communication).

Layer	Services
Application	Generates Generic Profiles message as a bit stream and determines message type
Generic Profiles API (added serial support)	Creates serial message(s)
Dolphin API	Sends serial message(s), e.g. via ESP3
EnOcean Dolphin chip	Physical serial telegram transmission

FIGURE 2.7: LAYER MODEL OF SERIAL COMMUNICATION

3. Convention

This chapter describes Generic Profiles. It focuses on the data exchange between devices, which is the essential function of a wireless sensor network.

3.1. Introduction

The recent EnOcean Equipment Profiles consist of a set of tables to define each officially supported device and its transmitted data. The specific definition of a device is referenced by the EEP number (R-ORG, FUNC, TYPE). The Generic Profiles approach instead defines a language to communicate the transmitted data types and ranges. The devices become self-describing on their data structures in communication.

To handle the huge variety of possible data this language has to be versatile and compact.

3.2. Approach

The data sent over-the-air is generally the result of an analogue-to-digital conversion, the state of a counter in the transmitting device or etc. To conserve energy, these raw measurements are transmitted directly, using only as many bits as the native conversion produced. To determine the actual value, it is necessary to have a set of parameters to map the pure digital values into physical units. Declaring this set of parameters will enable the receiver to recalculate the originally measured value as a preparation for further processing.

The Generic Profiles include a language definition with a parameter selection that covers every possible measured value to be transmitted. Therefore, the approach does not only define parameters for the value recalculation algorithm but also includes specific signal definition. (e.g. physical units).

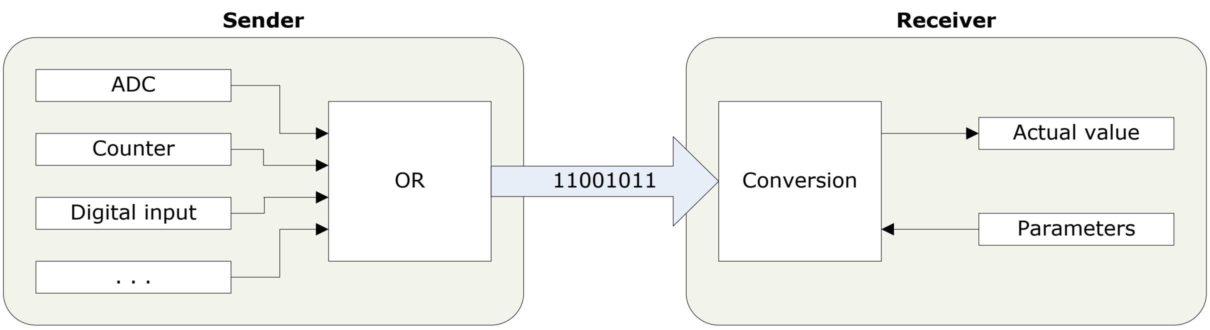

FIGURE 3.1: GENERIC DATA TRANSMISSION

For every measurement the set of parameters has to be transmitted before the first operational data exchange. This is done during the Teach-in process. Using this process the device describes its future communication self.

3.3. Parameters

The defined set of parameters describes every aspect of a digital value to enable the recalculation of the actual physical value.

To mathematically reclaim of a value conversion from digital to the actual physical value the resolution, the actual minimum and the actual maximum value are needed. For the interpretation of that value, the character (e.g. set point, relative or absolute measurement) of the original measurement has to be provided, too.

Note:

All signed numbers used in over-the-air transmissions are coded in "two‘s complement" also called "complement-2" format.

All frames and bytes are coded as big endian, meaning when sending or receiving a series of bytes, the most significant byte is transmitted and received first.

3.3.1. Channel characterization

Automated processing of digital data is only possible if all information about the acquisition type of the received data is available. Through this classification, a value can be combined with its physical unit and its proposed use. Therefore, three different parameters have to be communicated:

- channel type
- signal type
- value type

‘Channel Type’

The channel type divides all channels into different functional classes of measurements. With the three defined channel types ‘data’, ‘flag’ and ‘enumeration’ measurement results and complex counter values are separated from single bit logical channels and enumerated values.

Teach-in information is neither a measured value nor used during operational mode. This channel type is used only during the Teach-in process. For detailed explanation, please refer to chapter 4.

Channel Type			
2 bit value	Data		
00	=	Teach-in information	Teach-in signals / flags
01	=	Data	Complex bit values
10	=	Flag	Single bit value
11	=	Enumeration	Enumerated values

TABLE 3.1: CHANNEL TYPE

For detailed definition of the measurable channel types, refer to the appendix, please.

'Signal Type'

The signal type classifies the origin of the transmitted value itself and its character (e.g. physical unit or field of use). The signal types differ between the channel types.

For detailed definition of the signal types and list, please refer to the appendix, please.

'Value Type'

Value Type			
2 bit value	Data		
00	=	Reserved	
01	=	Current value	
10	=	Set point absolute	
11	=	Set point relative	

TABLE 3.2: VALUE TYPE

With the value type the context of a certain value shall be described.

3.3.2. ADC parameters

Beside the information about the origin and purpose of the channel, it is essential to transmit all necessary parameters for the data conversion.

'Resolution'

Resolution			
4 bit value	Data, Enumeration		
0000	=	Reserved	
0001	=	2 bit	
0010	=	3 bit	
0011	=	4 bit	
0100	=	5 bit	
0101	=	6 bit	
0110	=	8 bit	
0111	=	10 bit	
1000	=	12 bit	
1001	=	16 bit	
1010	=	20 bit	
1011	=	24 bit	
1100	=	32 bit	
1101	=	Reserved	
1110	=	Reserved	
1111	=	Reserved	

TABLE 3.3: RESOLUTION 'DATA' AND 'ENUMERATION'

For flag channels, the resolution is defined as 1 bit. This is an implicit definition and is valid for all flag channels.

Resolution		
	Flag	
	= 1 bit	

TABLE 3.4: RESOLUTION 'FLAG'

'Engineering minimum'

The engineering minimum represents the bottom of the measurement range. The transmitted parameter has to be multiplied with its scaling factor.

Engineering minimum		
8 bit	Data	
	= [-128 … 127]	

TABLE 3.5: ENGINEERING MINIMUM 'DATA'

For flag channels the engineering minimum is always zero. This is an implicit definition and is valid for all flag channels.

Engineering minimum		
	Flag	
	= 0	

TABLE 3.6: ENGINEERING MINIMUM 'FLAG'

'Scaling minimum'

To allow for a wide range of minimum values the engineering minimum can be scaled by one of the supported factors.

Scaling minimum			
4 bit value	Data		
0000	=	Reserved	N/A
0001	=	x 1	x 1
0010	=	x 10	x 1e+01
0011	=	x 100	x 1e+02
0100	=	x 1,000	x 1e+03
0101	=	x 10,000	x 1e+04
0110	=	x 100,000	x 1e+05
0111	=	x 1,000,000	x 1e+06
1000	=	x 10,000,000	x 1e+07
1001	=	x 0.1	x 1e-01
1010	=	x 0.01	x 1e-02
1011	=	x 0.001	x 1e-03
1100	=	x 0.000001	x 1e-06
1101	=	x 0.000000001	x 1e-09
1110	=	Reserved	N/A
1111	=	Reserved	N/A

TABLE 3.7: SCALING MINIMUM 'DATA'

For flag channels, there is no scaling option.

Scaling minimum			
	Flag		
	=	x1	

TABLE 3.8: SCALING MINIMUM 'DATA'

'Engineering maximum'

The engineering maximum works the same way as the engineering minimum.

Engineering maximum			
8 bit	Data		
	=	[-128 … 127]	

TABLE 3.9: ENGINEERING MAXIMUM 'DATA'

For flag channels, the engineering maximum is always one. This is an implicit definition and is valid for all flag channels.

Engineering maximum			
	Flag		
	=	1	

TABLE 3.10: ENGINEERING MAXIMUM 'FLAG'

'Scaling maximum'

To allow for a wide range of maximum values the engineering maximum can be scaled by one of the supported factors.

Scaling maximum			
4 bit value	Data		
0000	=	Reserved	N/A
0001	=	x 1	x 1
0010	=	x 10	x 1e+01
0011	=	x 100	x 1e+02
0100	=	x 1,000	x 1e+03
0101	=	x 10,000	x 1e+04
0110	=	x 100,000	x 1e+05
0111	=	x 1,000,000	x 1e+06
1000	=	x 10,000,000	x 1e+07
1001	=	x 0.1	x 1e-01
1010	=	x 0.01	x 1e-02
1011	=	x 0.001	x 1e-03
1100	=	x 0.000001	x 1e-06
1101	=	x 0.000000001	x 1e-09
1110	=	Reserved	N/A
1111	=	Reserved	N/A

TABLE 3.11: SCALING MAXIMUM 'DATA'

For flag channels there is no scaling option.

Scaling maximum			
	Flag		
	=	x1	

TABLE 3.12: SCALING MAXIMUM 'FLAG'

3.4. Measurement value quantization

The measurement value quantization should follow these equations:

x = actual value

X_{min} = actual engineering minimum

X_{max} = actual engineering maximum

Eng_{min} = scaled engineering minimum

Eng_{max} = scaled engineering maximum

F_{min} = scaling factor minimum

F_{max} = scaling factor maximum

n = quantized value

N = number of steps (bit range)

$$Eng_{min} = \frac{X_{min}}{F_{min}} \qquad Eng_{max} = \frac{X_{max}}{F_{max}}$$

$$n = N \times \frac{x - X_{min}}{X_{max} - X_{min}} \qquad x = \frac{n}{N} \times (X_{max} - X_{min}) + X_{min}$$

FIGURE 3.2: MEASUREMENT FORMULAS

3.5. Examples

3.5.1. Data channel definition

Measurement

Temperature sensor

Range: 0 – 40 °C
Resolution: 10 bit
Purpose: current value

Channel definition

Channel type:	Data	01
Signal type:	Temperature	00011000
Value type:	Current value	01
Resolution:	10 bit	0111
Scaled eng. minimum:	0°C	$[00000000]_2$
Scaling minimum:	x 1	0001
Scaled eng. maximum:	40 °C	$[00101000]_2$
Scaling maximum:	x 1	0001

FIGURE 3.3: EXAMPLE TEMPERATURE SENSOR DEFINITION

Measurement

Concentration sensor

Range: 1 – 1e + 06 ppm
Resolution: 32 bit
Purpose: current value

Channel definition

Channel type:	Data	01
Signal type:	Concentration	00000101
Value type:	Current value	01
Resolution:	32 bit	1100
Scaled eng. minimum:	1 ppm	$[00000001]_2$
Scaling minimum:	x 1	0001
Scaled eng. maximum:	1 ppm	$[00000001]_2$
Scaling maximum:	x 1e + 06	0111

FIGURE 3.4: EXAMPLE CONCENTRATION SENSOR DEFINITION

Measurement

Voltmeter

Range: -230 – 230 V
Resolution: 16 bit
Purpose: current value

Channel definition

Channel type:	Data	01
Signal type:	Voltage	00011100
Value type:	Current value	01
Resolution:	16 bit	1001
Scaled eng. minimum:	-23 V	$[11101001]_2$
Scaling minimum:	x 10	0010
Scaled eng. maximum:	23 V	$[00010111]_2$
Scaling maximum:	x 10	0010

FIGURE 3.5: EXAMPLE VOLTMETER DEFINITION

3.5.2. Flag channel definition

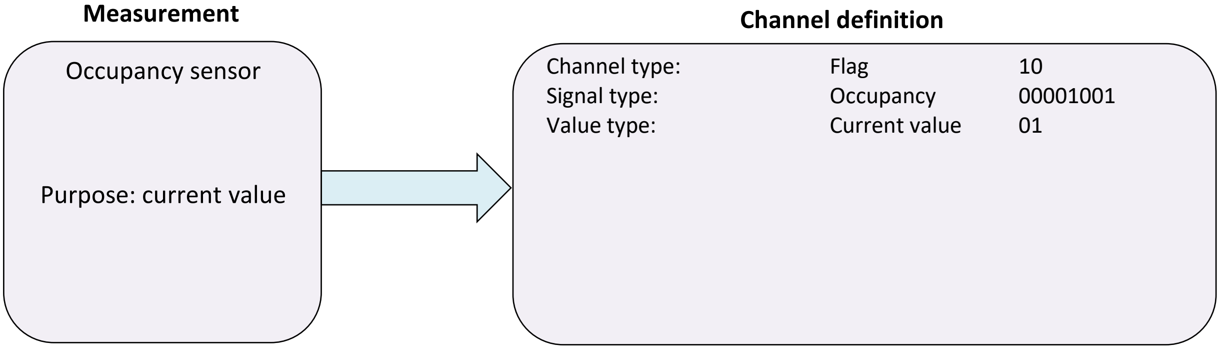

FIGURE 3.5: EXAMPLE OCCUPANCY SENSOR DEFINITION

3.5.3. ENUM channel definition

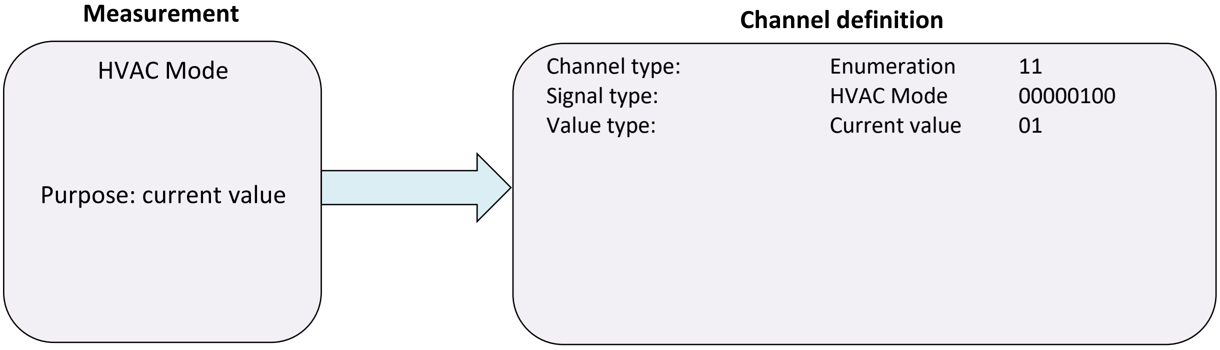

FIGURE 3.6: EXAMPLE HVAC STATE INFO DEFINITION

3.5.4. Quantization

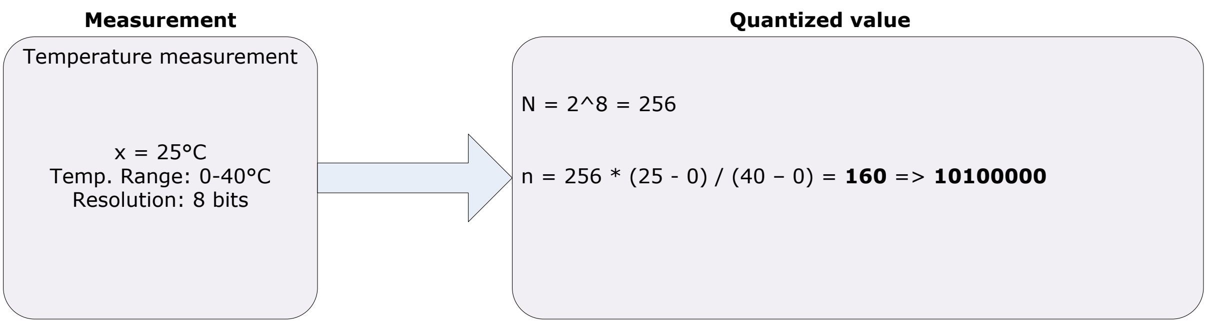

FIGURE 3.7: EXAMPLE QUANTIZATION

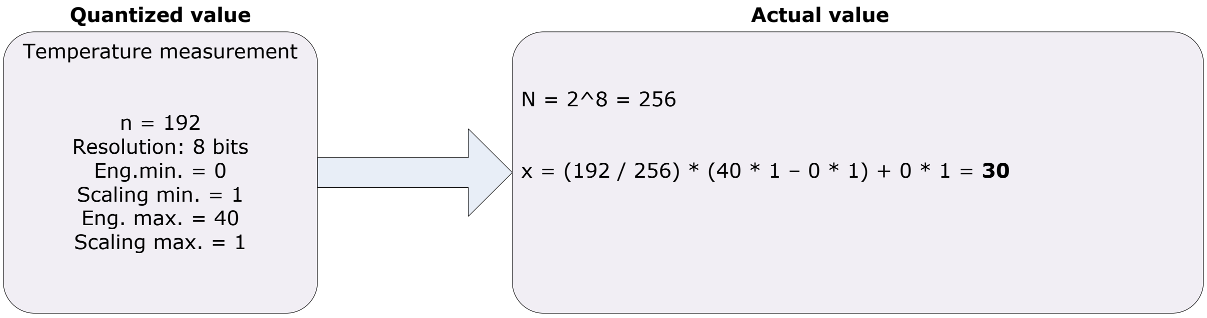

FIGURE 3.8: EXAMPLE RECALCULATION ACTUAL VALUE

4. Teach-in Process

Teach-in is the process where communication partners exchange information about how to interpret data which will be exchanged in the data communication. This chapter describes how to execute the Teach-in process to enable data exchange based on Generic Profiles.

4.1. Introduction

Following the guidelines of the defined communication layers and Generic Profiles, every generic EnOcean device can exchange data with compatible devices.

Therefore, the interpretation of received data messages is based on two conditions:

1. Generally, the message has to be accepted first.
   That means that it has to carry a valid EnOcean ID that is known by the receiver or it can address the receivers EnOcean ID.
2. The receiver has to be aware of the user data structure.
   As this structure is almost infinitely variable due to the generic approach, the transmitter has to transmit its channel characteristics too.

The process of connecting two EnOcean radio devices and exchanging initiating information is called ‘Teach-in’ and has to be passed before the first operational communication. An intentional disconnection of this binding, called ‘teach-out’, is also included in the following definition.

A generic Teach-in procedure allows a device to connect to different radio partners. It does <u>not</u> prevent the case of connecting to the wrong device.

4.2. Procedure

4.2.1. General procedure

The Teach-in process has a bidirectional character. Therefore, it consists of two consecutive messages:

- First, after the receiver has been switched into learn mode, the transmitter broadcasts a **Teach-in request** message.
- The receiver answers with a **Teach-in response** message, which should be addressed to the transmitter.

If the receiver has bidirectional communication capabilities, then it shall transmit a Teach-in response. This is required to enable commissioning devices to see and document the Teach-in result. Simple example is shown in Figure 4.1.

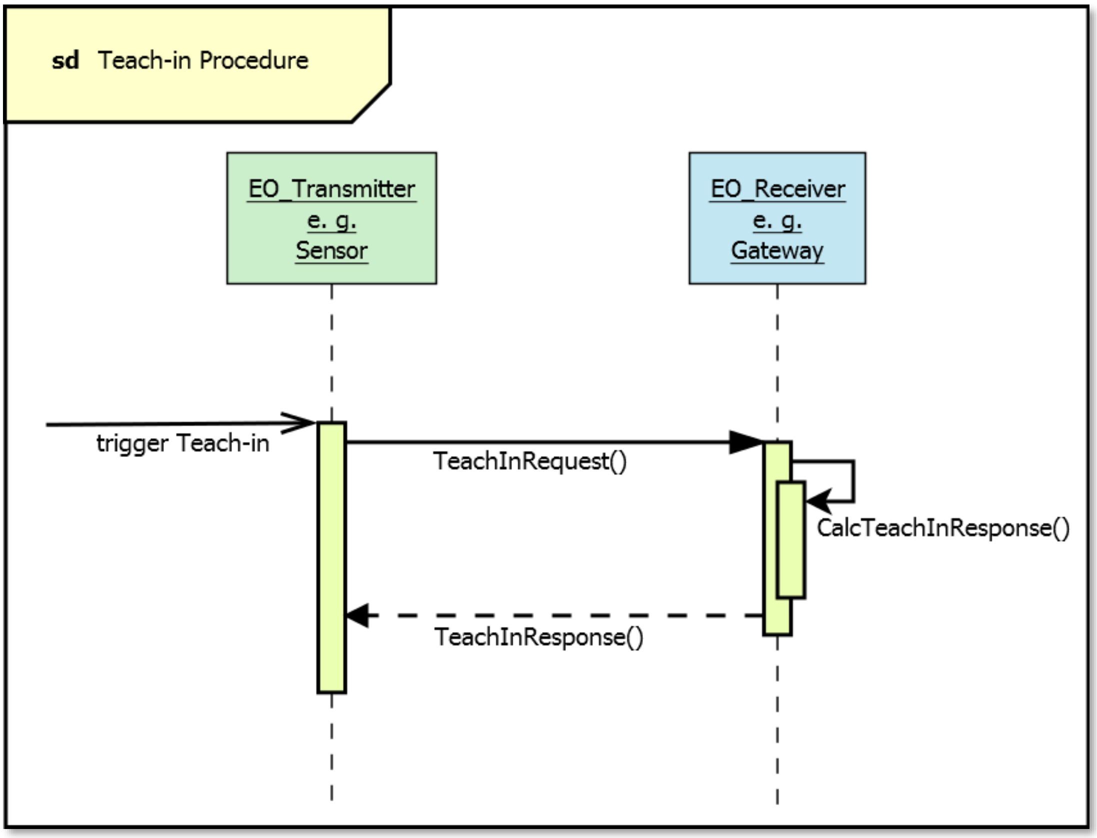

FIGURE 4.1: TEACH-IN PROCEDURE

4.3. Definition

The general structure of the Teach-in message is divided into a header and a definition area. The Teach-in request header does not contain the same information as the Teach-in response header and while the Teach-in request message includes the channel definitions, the Teach-in response gives information about possible rejected channels.

In the definition area of the Teach-in request, first the outbound channels are defined. Outbound/ Outgoing channels are the channels the device will send in data communication.

Teach-in request message			
Header	Channel definition 0	Channel definition 1	…

FIGURE 4.2: TEACH-IN REQUEST MESSAGE STRUCTURE

There is no padding or byte aligning between channel definitions. The first byte following after one definition is already used for the next definition.

Teach-in response message	
Header	Channel acknowledgement list

FIGURE 4.3: TEACH-IN RESPONSE MESSAGE STRUCTURE

When executing bidirectional Teach-in with inbound and outbound channel definitions the channel definitions of outbound and inbound are separated with the appropriate Teach-in information channel type. For details on Teach-in information channel type please see chapter 4.3.2.

Teach-in request message with bidirectional application			
Header	Outbound channel def	Teach-in information – Inbound definition follows (signal type = 0x01)	Inbound channel def.

FIGURE 4.4: TEACH-IN REQUEST MESSAGE WITH BIDIRECTIONAL DEFINITION

Inbound / incoming channels are channels the device expects to receive in the data communication.

4.3.1. Teach-in request

A Teach-in request message is always pre-described in the Teach-in request header. This header is followed by the channel definition area where every channel is defined separately.

Teach-in request header			
Manufacturer ID	Data direction	Purpose	Not used
11 bit	1 bit	2 bits	2 bits
	0 = unidirectional 1 = bidirectional	00 = teach-in 01 = teach-in deletion 10 = teach-in or deletion of teach-in 11 = not used	

FIGURE 4.5: TEACH-IN REQUEST HEADER

Field details and purpose:

- Manufacturer ID
  Is the EnOcean Alliance Manufacturer ID of the device, which transmits the Teach-in request.
- Data Direction
  Operational data transmission can be unidirectional or bidirectional. The 'data direction' bits define whether data exchange will be bilateral or not. It does not define the device hardware capabilities. If direction is bidirectional and no response is received then it is to assume the Teach-in process has failed.
- Purpose
  - 0b00 teach-in – explicit request to teach-in.
    Possible return codes:
    00 = rejected generally
    01 = teach-in successful

11 = rejected channels outbound or inbound

- 0b01 teach-in deletion – explicit request to teach-in deletion / teach-out.
  Possible return codes:
  10 = teach-out

- 0b10 teach-in or deletion of teach-in – toggle teach.
  Possible return codes:
  00 = rejected generally
  01 = teach-in successful
  10 = teach-out
  11 = rejected channels outbound or inbound

4.3.2. Channel definition

The goal of a channel definition is to provide all necessary information about how certain data is coded for transmission and how it should be processed at the receiver. However, it does not dictate the purpose of this data.

Due to the diversity of channel definitions and the transmitted information, the general channel definition is divided into different channel types. For detailed description of the character of the channels please refer to chapter 3.3.1.

Next is the channel definition of all channel types. The channel definition frame is different for each channel type. The only common characteristics are the first two bits, which define the channel type. Based on the channel type a receiver can interpret the remaining information.

The exact parameter lists and explanation of the values are shown in the chapter 3.3.2. The signal type definition depends on the channel type. The signal type list for every channel type is available in the Generic Profiles appendix.

Channel definition 'Data'							
Channel type	Signal type	Value type	Resolution	Engineering minimum	Scaling minimum	Engineering maximum	Scaling maximum
2 bits	8 bits	2 bits	4 bits	8 bits	4 bits	8 bits	4 bits

FIGURE 4.6: CHANNEL DEFINITION 'DATA'

The length of a channel definition 'Data' is 40 bits.

Channel definition 'Flag'		
Channel type	Signal type	Value type
2 bits	8 bits	2 bits

FIGURE 4.7 CHANNEL DEFINITION 'FLAG'

The length of a channel definition 'Flag' is 12 bits.

Channel definition 'Enumeration'			
Channel type	Signal type	Value type	Resolution
2 bits	8 bits	2 bits	4 bits

FIGURE 4.8: CHANNEL DEFINITION 'ENUMERATION'

The length of a channel definition 'Enumeration' is 16 sixteen bits. The field resolution of 'Enumeration' shall be used from Table 3.3: Resolution 'data' and 'enumeration' and NOT from the appendix.

Channel definition 'Teach-in information'			
Channel type	Signal type	Length indication for following data in bytes.	Data
2 bits	8 bits	8 bits	N

FIGURE 4.9 CHANNEL DEFINITION 'TEACH-IN INFORMATION'

The length of a channel definition 'Teach-in information' is 18 + N bits. This channel definition has a variable length indicator for the data content following. The length of the data content is defined for every signal type and can be found in the Generic Profiles appendix[1]. The length indication of the following data in is given in bytes (e.g. if field = 0x04, then 4 bytes of data will follow). The Teach-in information channel type neither has influence on the operational data communication nor on the indexing.

4.3.3. Teach-in response

The Teach-in result provides information about the success of the Teach-in process. As a reaction to a received Teach-in request, the receiver sends an <u>addressed</u> Teach-in response message to the initiating radio partner. This message provides information about the device itself and the Teach-in status.

[1] https://www.enocean-alliance.org/what-is-enocean/specifications/

Teach-in response header		
Manufacturer ID	Result	undefined
11 bits	2 bits	3 bits
	00 = rejected generally 01 = teach-in successful 10 = teach-out 11 = rejected channels outbound or inbound	

FIGURE 4.10: TEACH-IN RESPONSE HEADER

A successful Teach-in or teach-out will be referred by ‘01’ or ‘10’.

If at least one of the inbound or outbound transmitters channels cannot be adopted by the receiver the Teach-in result is ‘11’ and further information about the rejected channels will be given in the channel acknowledgment list following the header. The channel acknowledgement list is described in chapter 4.3.4. Depending on the given list of channels, the transmitter application can decide whether it wants to accept the Teach-in or has to cancel it by sending a Teach-in response message with teach-out result.

In case of acceptance no further action is required. If no cancelation is received by the receiver then the Teach-in is successfully accepted and only those channels will be processed that have been acknowledged.

A Teach-in rejection without a specific reason is ‘00’. In this case no channel acknowledgement list will be given.

Detailed visualisation of the process described above can be seen in the activity diagram in Figure 4.11 and in sequence diagram in Figure 4.12.

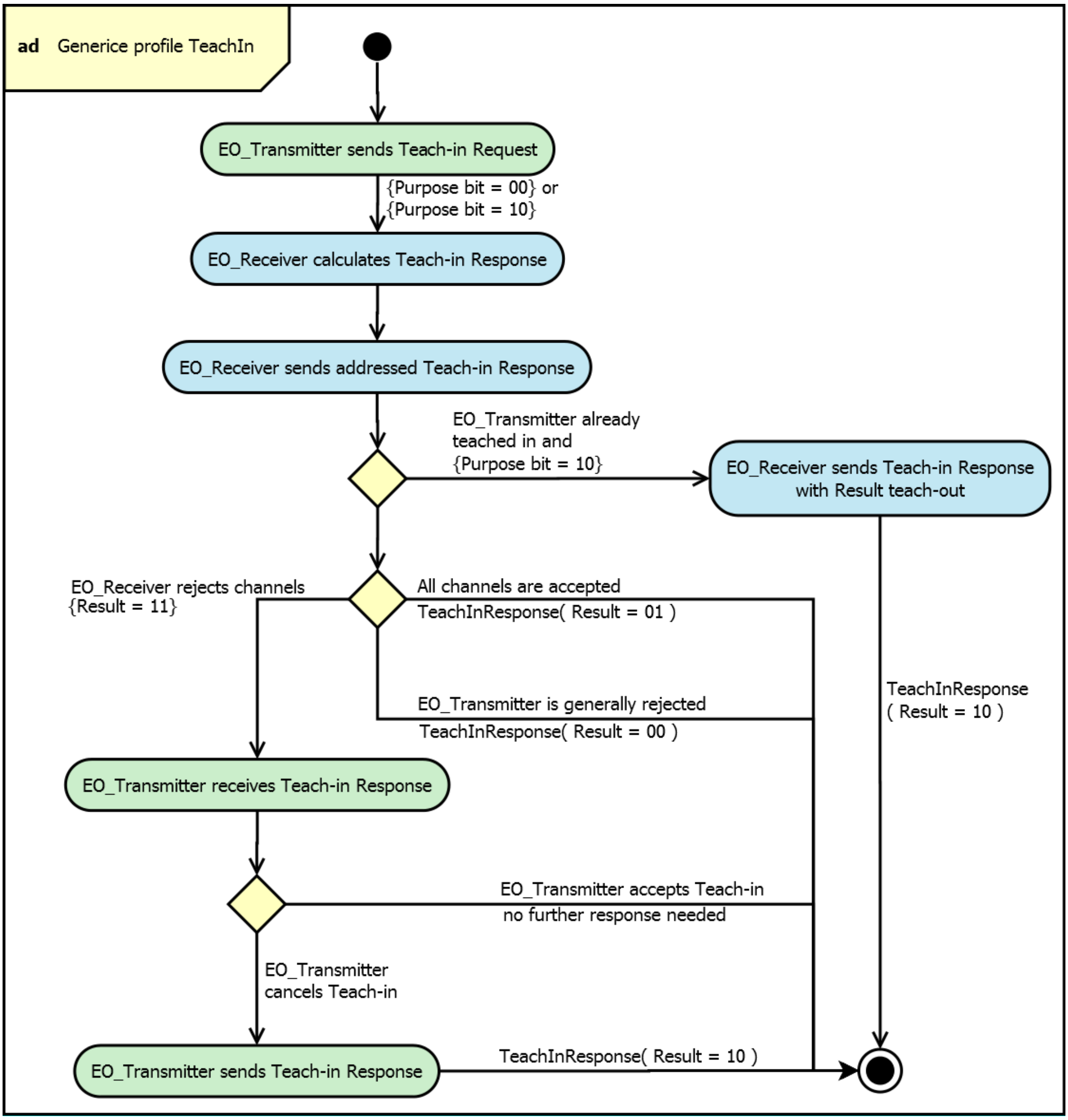

FIGURE 4.11: GENERIC PROFILE TEACH-IN ACTIVITY DIAGRAM

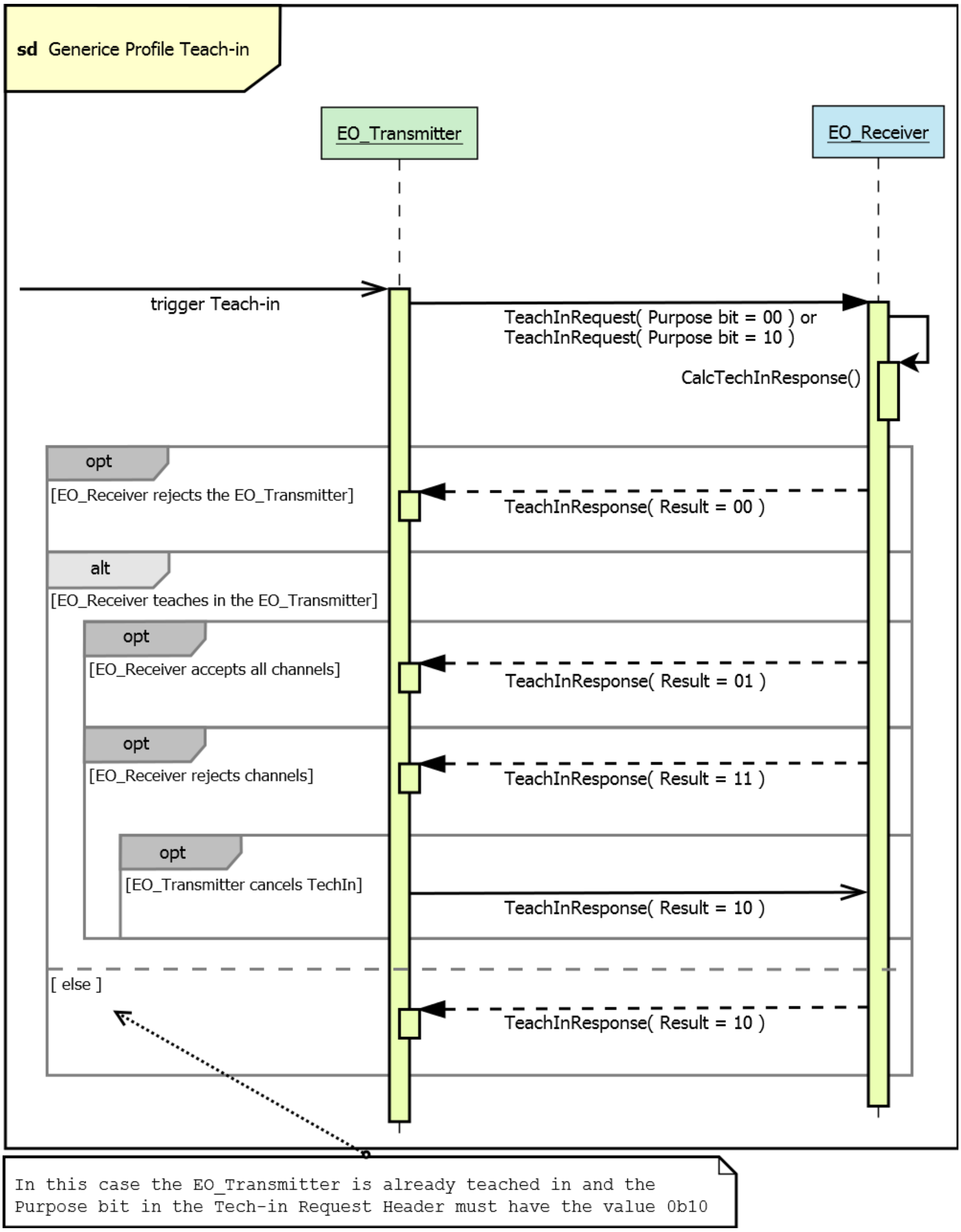

FIGURE 4.12: GENERIC PROFILE TEACH-IN SEQUENCE DIAGRAM

4.3.4. Channel acknowledgement

If not all channels can be adopted by the receiver a list of information about the acknowledgement status of every channel is provided to the transmitter within the Teach-in response message. Therefore, the header is followed by a bit stream that contains one bit for every defined channel transmitting whether the channel is supported or rejected:

- 1 = channel supported;
- 0 = channel rejected

Teach-in response accepted channel list outbound & inbound							
OUTBOUND				INBOUND			
CH 0	CH 1	...	...	....	CH *N*-3	CH *N*-2	CH *N*[2]-1
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

FIGURE 4.13: TEACH-IN RESPONSE CHANNEL LIST

The bit order is exactly the same as the channel definition order. As the transmitter knows this order it can decide if the Teach-in process should be rejected or accepted.

The channel acknowledgement list will only be added to a Teach-in response message if Teach-in was not completely successful (Result = ‘11’). In this case, the complete (inbound if available & outbound if available) channel acknowledgement list is transmitted.

4.4. Channel indexing

A channel index is defined to have a unique numeric reference to the individual channels of a single device. The channel index starts with 0 and will be counted up. Indexing of channels starts with the first defined outbound channel “*index 0*” in the Teach-in request and ends with the last inbound channel “*index N-1*” where *N* is the count of all channels.

Teach-in information channel types are not indexed.

4.5. Message timings

In this chapter, the message timing conventions are defined. The timing defines the maximum timeout in a message exchange process. When the timeout is passed then the message is considered as unreceived. Messages arriving after this timeout have to be processed as not relevant any more. If a message consists of more telegrams, then the timeout describes the transmission / reception of the first of the chained telegrams.

Timing conventions:

[2] *N* is the count of all channels - inbound & outbound. The index is 0 based.

Transmitter timeout: **750 ms**

A Teach-in response should be received within 750ms after transmission of the Teach-in request.

Receiver response time: **500 ms**

A Teach-in response should be transmitted within 500ms after reception of a Teach-in request.

Receiver timeout: **750 ms**

A Teach-in response with teach-out result - '010' should be received within 750ms after transmission of the Teach-in response (some channels are rejected inbound or outbound) from the receiver. If no such Teach-in response was received, it is assumed that the transmitter accepted the teach-in.

Transmitter response time: **500 ms**

A Teach-in response with teach-out result - '010' should be transmitted within 500ms after reception of the Teach-in response (some channels are rejected) from the receiver.

4.6. Using Smart Acknowledge for communication

The use of Smart Acknowledge follows the respective conventions in the 'EEP Specification'. The only difference is the special generic EEP (R-ORG: **B0** FUNC: **00** TYPE: **00**) that generally represents the generic communication. Beside that the procedure is equal to an EEP based Smart Acknowledge Teach-in. The Generic Profiles Teach-in is then executed separated from Smart Acknowledge Teach-in. First the communication link of Smart Acknowledge is build and then Generic Profiles Teach-in is executed. All timing conventions are given by the Smart Acknowledge definition and the application.

EXAMPLE

1. Controller is put to Teach-in mode (with Smart Acknowledge capability).
2. Sensor sends a Smart Acknowledge Teach-in request. The Smart Acknowledge Teach-in request holds the EEP 0xB0 0x00 0x00.
3. Smart Acknowledge Teach-in is executed. Post master is determined and Sensor is successfully taught in.
4. Sensor sendsa Generic Profiles Teach-in request.
5. Controller evaluates the Generic profiles Teach-in and sends a Generic Profiles Teach-in Response through Smart Acknowledge.

NOTE:

The most recent EEP Specification can be downloaded from the website of the EnOcean Alliance. https://www.enocean-alliance.org/eep/ .

Smart Acknowledge Specification is available here:
https://www.enocean-alliance.org/smartack/ .

4.7. Examples

4.7.1. Teach-in request message

	Teach-in request header
bin	1 1 1 1 1 1 1 1 1 1 1 1 0 0 x x
hex	0 x F F F 0
dec	65520
Man. ID	1 1 1 1 1 1 1 1 1 1 1
=	Multi user Manufacturer ID
Data dir.	1
=	Bidirectional
Purpose	0 0
=	teach-in
Undefined	x x

FIGURE 4. 14: SIMPLE BIDIRECTIONAL TEACH-IN REQUEST HEADER

	Teach-in request header
bin	1 1 1 1 1 1 1 1 1 1 1 1 1 0 x x
hex	0 x F F F 8
dec	65528
Man. ID	1 1 1 1 1 1 1 1 1 1 1
=	Multi user Manufacturer ID
Data dir.	1
=	Bidirectional
Purpose	1 0
=	teach-in or deletion of teach-in
Undefined	x x

FIGURE 4. 15: EXAMPLE OF TEACH-IN REQUEST HEADER WITH OPTIONAL DELETION OF TEACH-IN

	Channel definition 1
bin	0 1 0 0 0 0 0 1 1 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 0 0 0 1
hex	0 x 4 1 9 5 0 0 1 0 5 1
dec	281672683601
Chan. type	0 1
=	Data
Sig. type	0 0 0 0 0 1 1 0
=	Current
Val. type	0 1
=	Current value
Res.	0 1 0 1
=	8-bit
Eng. Min.	0 0 0 0 0 0 0 0
=	0
Scal. Min.	0 0 0 1
=	x1
Eng. Max.	0 0 0 0 0 1 0 1
=	5
Scal. Max.	0 0 0 1
=	x1

FIGURE 4. 16: EXAMPLE OF CHANNEL DEFINITION 'DATA'

	Channel definition 'Teach-in information'			
bin	00	000001	00000010	1000001001100000
hex	0x814130			
dec	8470832			
Chan. type	00			
=	Teach-in information			
Sig. type		000001		
=	Teach-in information - Inbound definition follows			
Length of data [Bytes]			00000010	
=	2 Byte			
Data				1000001001100000
=	Channel definition 2 (Occupancy sensor)			

FIGURE 4. 17: CHANNEL DEFINITION 'TEACH-IN INFORMATION' EXAMPLE

	Channel definition 2
bin	1 0 0 0 0 0 1 0 0 1 1 0
hex	0 x 8 2 6
dec	2086
Chan. type	1 0
=	flag
Sig. type	0 0 0 0 1 0 0 1
=	Occupancy
Val. type	1 0
=	Set point absolute

FIGURE 4. 18: CHANNEL DEFINITION 'FLAG' EXAMPLE

4.7.2. Teach-in response message

	Teach-in response header
bin	1 1 1 1 1 1 1 1 1 1 1 0 1 x x x
hex	0 x F F E 8
dec	65512
Man. ID	1 1 1 1 1 1 1 1 1 1 1
=	Multi user Manufacturer ID
Result	0 1
=	teach-in successful
Undefined	x x x

FIGURE 4. 19: SIMPLE TEACH-IN RESPONSE HEADER

	Teach-in response header
bin	1 1 1 1 1 1 1 1 1 1 1 1 1 x x x
hex	0 x F F F 8
dec	65528
Man. ID	1 1 1 1 1 1 1 1 1 1 1
=	Multi user Manufacturer ID
Result	1 1
=	Rejected channels outbound or inbound
Undefined	x x x

FIGURE 4. 20: EXAMPLE FOR A TEACH-IN RESPONSE HEADER WITH REJECTED CHANNELS

	Channel acknowledgement list
bin	1 1 1 1 1 1 0 0 0 0 0 0
=	Channel 0 - 5 supported Channel 6 - 11 rejected

FIGURE 4. 21: TEACH-IN RESPONSE ACKNOWLEDGEMENT LIST

5. Operational mode

This chapter describes how the actual data transfer works.

5.1. Introduction

In operational mode, either complete or selected data messages will be sent transmitted. This chapter describes how the data is arranged.

Each outbound channel of the sensor delivers data with a fixed length of bits which is defined in the Teach-in request. The receiver has the knowledge about the number of bits of each sensors outbound channel data and is able to decode the data correctly after the sensor encoded his measurement values to the data message.

In bidirectional communication, the same principle is applied. The receiver codes its data message according to the inbound channels definition of the Teach-in request. The sensor is then able to decode the message and the data.

A data request mechanism is also part of generic profiles. This topic is more complex. Therefore, the definition of data request mechanism will be added to the appendix after the first field trials.

5.2. Data message definition

There are two different message types:

- **Complete data** message
  It contains all data the sensor can deliver.
- **Selective data** message
  It contains data only of selected channels.

The layering model selects the type of the EnOcean radio telegram(s) applied depending on the length of the message. Messages can consist of

- single radio telegram – payload fits into one telegram.
- more radio telegrams = chained radio telegrams – payload does not fit into one telegram.

Details to the chaining process can be found at chapter 2.3.1.

In the data messages only data, flag or enumeration channel type are included. The Teach-in information channel type is only used during Teach-in process. It is NOT included in the data communication.

5.2.1. Complete Data message

The data of each channel will be compiled into a complete data message and consisting of a bit stream. There is no channel number information in the complete data message, only the measurements.

The rules to add the measurements to the bit stream are:

- Starting with channel 0, all used bits of every channel are concatenated together to a bit stream.
- The bit order will NOT be changed, i.e. MSB stays MSB in the stream.
- After connecting all bits of the sensor, the message will be filled with unused bits (0) till the next byte border is reached.
- Every channel has to be added to the stream.
- A complete message can be either outbound or inbound.

Example:

Three outbound channels of a sensor are defined in the Teach-in request:

- Channel 0 with a 6 bit measurement value
- Channel 1 with a 8 bit measurement value
- Channel 2 with a 5 bit measurement value

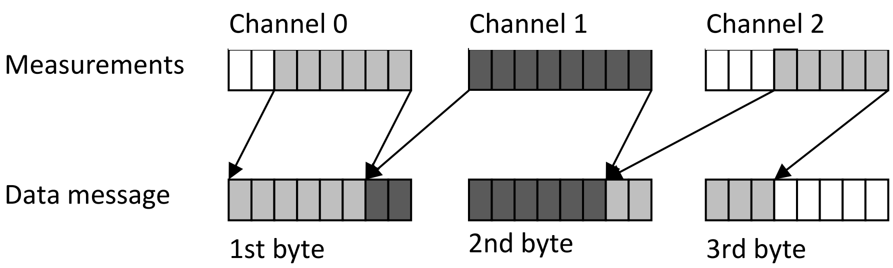

FIGURE 5.1: EXAMPLE OF A COMPLETE DATA MESSAGE

The data message consists of the sum of all measurement bits of the sensor, i.e. 19 bits. There are 3 bytes necessary to transmit. The 5 LSB of the 3rd data message byte are unused (□).

5.2.2. Selective data message

The selective data message starts with a 4-bit header containing the number of channels of the message. To relate the channel to the value the channel index will be inserted prior every channel data. The channel index is 6 bit long. The indexing of channels is described in chapter 4.4.

The rules of adding a channel to the selective data message bit stream are:

- The channel index is 6 bits wide.
- Starting with the first measurement channel to be transmitted, the 6 bit channel number and the used bits of that channel are concatenated together to a bit stream.
- Further channels are added to the bit stream adequately.
- The bit order will NOT be changed, i.e. MSB stays MSB in the stream.
- After connecting all data to transmit, the message will be filled with unused bits till the next byte border is reached.
- A selective message can be either outbound or inbound.

Example:

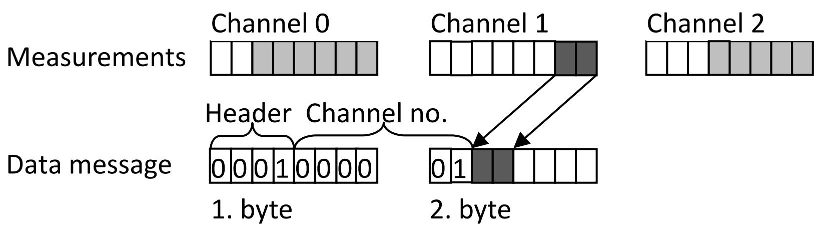

FIGURE 5.3: EXAMPLE OF A SELECTIVE DATA MESSAGE

Three outbound channels of a sensor are defined in the Teach-in message:
FIGURE 5.2: EXAMPLE SELECTIVE DATA MESSAGE

- Channel 1 with a 2 bit measurement value
- Channel 2 with a 5 bit measurement value

Only channel 1 measurement value changed and should be transmitted

The selective data message consists of the 4 bit header (0001), the 6 bit channel number (000001) and the 2 bit measurement of channel 1 of that sensor. The message length is only two bytes.

6. Compatibility with EEP

In this chapter the further coexistence and development of Generic Profiles and EnOcean Equipment Profiles is explained.

6.1. Introduction

Establishing a new concept of radio communication in a world of existing and highly integrated systems requires a strategy to connect devices from both the recent and the new approach. This means that upcoming product introductions into the market needs to consider the 'EnOcean Equipment Profiles Specification' as well as the 'Generic Profiles'.

6.2. Coexistence

As the *Generic Profiles* (GP) are not meant to replace the *EnOcean Equipment Profiles* (EEP) immediately, the coexistence of both concepts is mandatory.

Communication between EEP devices is standardized and so is the communication between GP devices. A mixed data exchange is possible but will not be enforced by the *Generic Profiles* approach. Therefore the special *Generic Profiles* R-ORG's allow to identify generic telegrams and manufacturers are free to implement both or just one of the concepts in their products. During the Teach-in process the two selected devices have to determine which approach they will follow for their data exchange. Unsuccessful Teach-in attempts cannot be prevented by the new approach.

By that a general compatibility of the two different profile approaches can be guaranteed even though it is not necessary that all devices have to be able to connect to each other.

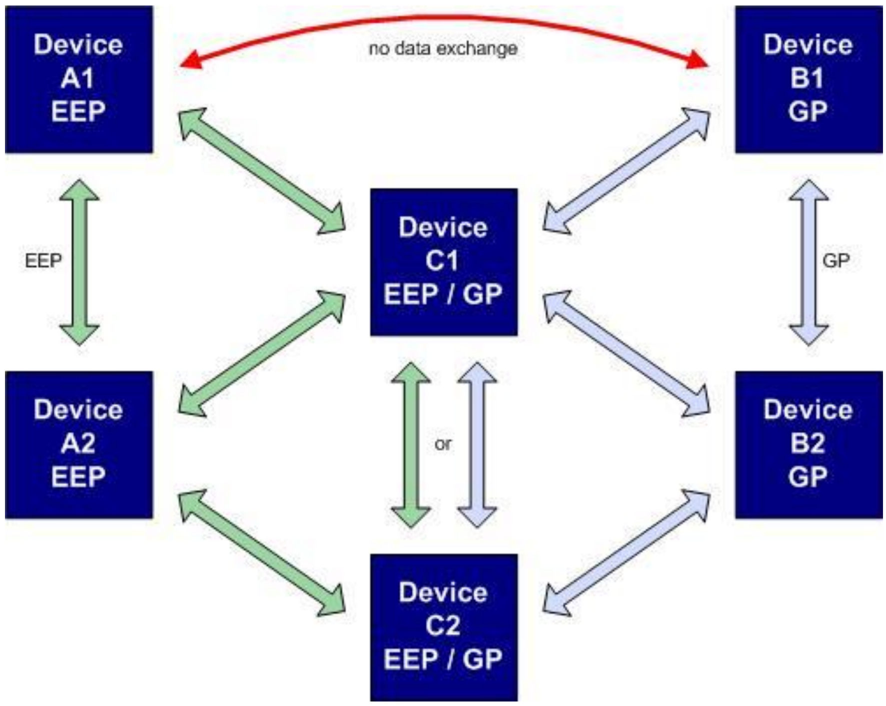

FIGURE 6.1: CONNECTION SCENARIOS

6.3. Transition Plan

Without an official pressure by the definition of the *Generic Profiles* it is up to the market and the different manufacturers of EnOcean devices to establish generic based devices and systems.

EEP's will be valid in the future but GP's offer additional functionality for flexible radio systems with growing requirements concerning data exchange.

FIGURE 6.2: TRANSITION FROM EEP TO GP

7. Remote Management

Generic Profile based devices that are continuously powered shall support a minimum feature set from the EnOcean Remote Management specification. In addition to the Remote Management Control Commands (RMCC) such devices shall support Remote Procedure Calls (RPC) defined in Remote Commissioning Specification.

Minimal required features from Remote Commissioning Specification:

- operations to control Teach-in process
- operations to access and read link table

In case any new RPC definition will be required, it shall be handled in accordance with the standardization process of the EnOcean Alliance TWG and defined within the System Specification on Remote Commissioning.

NOTE:

Remote Management Specification is available here:
https://www.enocean-alliance.org/reman/

Remote Commissioning Specification is available here:
https://www.enocean-alliance.org/recom-spec/