System Specifications

Communication Media

Radio Frequency

3

2

5

Summary

This document specifies the medium specific Physical Layer for the Radio Frequency (RF) medium.

Version 01.06.03 is a KNX Approved Standard.

This document is part of the KNX Specifications v2.1.

Document updates

Version	Date	Modifications
01	xxxx.yy.zz	Document creation based on • S22. STATUS: See document with integration marks. • KSG327-01 RF additional issues (KSG conclusion 2004.03.23-24) STATUS: DONE • Foreword and introduction added.
04	2007.11.16	• AN043 “LTE on RF” integrated
1.3		• Draft Proposal
1.4.00	2010.11.02	• Creation of the Draft for Voting
1.4.02	2011.07.28	• Publication of the Approved Standard
1.5.00	2011.10.25	• Draft for Voting.
1.5.01	2012.01.31	• TFRF review
1.6.00	2012.03.06	• Preparation of the Approved Standard.
1.6.01	2012.04.25	• Editorial update of Figures 50 and 51 according conclusion of TF RF Communication Medium meeting of 2012.03.16.
01.06.03	2013.10.29	Editorial updates for the publication of KNX Specifications 2.1.

References

[01] Chapter 3/3/2 “Data Link Layer – General Requirements”
[02] Chapter 3/3/4 “Transport Layer”
[03] Chapter 3/3/7 “Application Layer”
[04] Chapter 3/5/1 “Resources”
[05] Chapter 3/5/3 “Configuration Procedures”
[06] Part 10/1 “Logical Tag Extended”

Filename: 03_02_05 Communication Medium RF v01.06.03 AS.docx
Version: 01.06.03
Status: Approved Standard
Savedate: 2013.10.29
Number of pages: 81

Contents

Foreword

<Placeholder, to be filled in by CENELEC>

Introduction

According to OSI, Physical Layers consist of the medium, the cable, the connectors, the transmission technology, etc., which refers to their hardware requirements. In this European Standard however, the status of the Physical Layer as "communication medium" is emphasized.

The RF multichannel mechanism, named KNX RF Multi, specified below shall provide the following features.

- More reliability in Frame transmissions in presence of interferers.
- More efficiency when KNX RF products increase at the same location.
- Mixing of permanent and non-permanent receiving products.
- Mixing of fast and slow reaction time devices.
- Listen Before Talk mechanism.

After tests on products controlling lights that need fast reaction time, a good compromise is reached between the time for RF channel hopping and the reaction time by using 3 "fast" RF channels. To increase the number of possible applications and to introduce battery-powered receivers, 2 additional "slow" RF channels have been added. Therefore, with the constraint of mixing slow and fast products, 5 RF channels is a good compromise.

The two categories of RF channels define explicitly two application domains. Fast RF channels are mainly intended to be used with human controlled applications like for example lights, shutters… Slow RF channels are mainly intended to be used with non-permanent receivers for automatic applications like sensors (smoke, temperature, wind, etc.), heating control, etc.

Compatibility issues with products in compliance with the former KNX RF specification (KNX RF 1.1) and the new versions are considered in clause 7 at the end of this document.

1 Scope

This European Standard defines the mandatory and optional requirements for the medium specific Physical and Data Link Layer of Radio Frequency.

Data Link Layer interfaces and general definitions that are medium independent are given in EN 50090-4-1.

This European standard defines the requirements for KNX RF Ready, KNX RF Multi, KNX BiBat and KNX BiBat 2 devices.

- KNX RF Ready is a single RF channel system.
- KNX RF Multi is an RF multichannel evolution of KNX RF Ready system with 2 additional RF channels for fast reaction time products and 2 RF channels for slow reaction time products.
- KNX BiBat is a system for synchronised products based on KNX 1.1 specification
- KNX BiBat 2 is an evolution of BiBat with two RF channels based on KNX 1.1 specification.

All those four KNX RF specifications are used depending on the application. As some frequencies are common, products can easily communicate together if needed. In case of a gateway, the certification for two or more specifications is mandatory.

Six months after vote, all new certification shall be done with this current specification.

A compliance guide is available in clause 4.

Manufacturing and selling products previously certified with KNX RF1.1 specification are allowed after the vote of the present specification, but it is recommended to convert them to KNX RF Ready specification.

For configuration details see [04] and [05].

2 Normative references

This specification incorporates by dated or undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this specification only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies (including amendments).

EN 300 220-1

EN 300 220-2

EN301 489-1

EN 301 489-3

CEPT/ERC/REC 70-03

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this part the terms and definitions given in EN 50090-1 and the following apply.

Synchronous BiBat System

A Synchronous BiBat System consists of a single BiBat Master and one or several BiBat Slaves (devices). Optionally there can be BiBat Retransmitters in addition. All these components shall belong to a common unique KNX RF Domain with one KNX RF Domain Address. Multiple Synchronous BiBat Systems can coexist, but cannot communicate in a synchronous way with each other. Currently any application is limited to a KNX RF Domain.

A Synchronous BiBat System shall consist of a time scheme and shall partly use asynchronous RF Telegrams and partly dedicated Synchronous RF Telegrams.

Synchronous RF Telegrams

Synchronous RF Telegrams shall be identified by bit 6 of the KNX Ctrl field set to 1. Synchronous RF Retransmitters shall retransmit Synchronous RF Telegrams with a fixed delay time (without checking the medium). Synchronous RF Telegrams normally are ignored by standard KNX RF devices and by standard KNX RF Retransmitters. Exceptions may be e.g. bus monitoring devices.

BiBat Master

In this context the BiBat Master in a Synchronous BiBat System (domain) is a bidirectional device with continuous reception capability that can transmit synchronous Telegrams for the communication with the BiBat devices of its domain. In addition it supports all asynchronous communication.

Synchronous RF Retransmitter (BiBat Retransmitter)

A Synchronous RF Retransmitter shall have permanent RF reception capabilities (normally mains powered). It shall retransmit RF synchronous Telegrams with a fixed time delay without the usual collision avoidance of asynchronous KNX RF. It shall handle asynchronous RF Telegrams like an asynchronous RF Retransmitter.

BiBat Slave

A BiBat Slave shall be a device that shall support bidirectional RF communication. It shall be able to receive synchronous RF Telegrams in receive windows from a BiBat Master and shall be able to resynchronise its internal clock and hence the position of its receive windows.

BiBat Slave expecting feedback to own action

A *BiBat Slave expecting feedback to own action* asynchronously (not periodically but rarely i.e. activated by human interaction or an alarm) sends a Telegram to a mains powered BiBat Master. It expects a fast (typically < 100 ms) asynchronous Data Link Layer acknowledge from its BiBat Master. In addition status information may be transmitted to the initiating BiBat Slave and to other components of the application using the standard principles of synchronous BiBat Slaves.

BiBat device with long header

A *BiBat device with long header* shall emit its (rare) alarm Telegrams with a 3,5 s header, thus allowing not only a mains powered BiBat Master but also battery driven RF receivers with a wake-up period of < 3,4 s. A Bidirectional RF alarm sensor shall implement such a wake-up period of ≤ 3,4 s. In addition it may optionally use synchronized communication with a BiBat Master according the methods of synchronous BiBat Slaves.

BiBat device

A BiBat device shall be either

- a BiBat Master, a BiBat Slave or a BiBat Retransmitter, or
- a BiBat device with long header (Bidirectional RF alarm sensor), or
- a BiBat device expecting feedback to own action

Direction “down”

The communication direction from the BiBat Master to the BiBat Slave is denoted as “down” (can be via a BiBat Retransmitter).

Direction “up”

The communication direction from the BiBat Slave to the BiBat Master is denoted as “up” (can be via a BiBat Retransmitter).

RF Channel hopping

Action to change the RF channel during or after transmitting a Frame.

Budget link

The budget link of a device is the difference expressed in dB between the max radiated power and the radiated sensitivity. The higher is the budget link, the better radio range is.

3.2 Abbreviations

AFA	Adaptive Frequency Agility
BER	Bit Error Rate
BiBat	Bidirectional battery driven devices (specification for synchronized battery driven systems)
CC	Connection Code
D.C.	Duty Cycle
DLL	Data Link Layer
ERP	Effective Radiated Power
F1	F1 RF channel with a preamble of 15 ms in the transmitted Frame
F1r	F1 RF channel with a preamble of 4,8 ms in the transmitted Frame
F1sh	F1 RF channel with a preamble of 1ms in the transmitted Frame
FSK	Frequency Shift Keying
F_x	One of F1, F2 or F3 RF channels
F_x	One of F1, F2 or F3 RF channels
GFSK	Gaussian Frequency Shift Keying
LBT	Listen Before Talk
NPRM	Non-Permanent Reception Mode
PB	Push Button
PhL	Physical Layer
PRM	Permanent Reception Mode
Rx	Receiver

SN	KNX Serial Number
S_x	One of S1 or S2 RF channels
TRx	Transceiver
Tx	Transmitter

4 Introduction

As described in the scope, this European standard defines the RF Physical Layer requirements for:

- KNX RF Ready
- KNX RF Multi
- KNX BiBat
- KNX BiBat 2

All those four KNX RF specifications are used depending on the application. As some frequencies are common, products can easily communicate together if needed. In case of a gateway, the certification for two or more specifications is mandatory.

For KNX certification in Europe, the products shall be in compliance with at least one line of the following clauses.

Table 1 - Guide for compliance

System	Physical Layer	Data Link Layer
KNX RF Ready	clause 5.1	clause 6.1 and clause 6.2
KNX RF Multi	clause 5.2	clause 6.1 and clause 6.6
KNX RF BiBat	clause 5.1	clause 6.1, clause 6.2 and clause 6.3
KNX RF BiBat 2	clause 5.3	clause 6.1, clause 6.4 and clause 6.5

For KNX certification where specific frequency sub-band is allowed, the physical parameters of clause 8 shall apply.

The overview of KNX RF possibilities is described in Table 2.

Table 2 – Overview and naming

Region examples	Frequency sub-band	KNX RF Ready	KNX RF Multi	KNX RF BiBat	KNX RF BiBat 2
Europe	868 MHz	**KNX RF1.R**	**KNX RF1.M**	**KNX RF1.B**	**KNX RF1.B2**
Russia, Australia, Arab countries, Europe	433 MHz	**KNX RF2.R**	Under study	Not defined.	Not yet defined
China	433 MHz limited bandwidth	**KNX RF3.R**	Not defined	Not defined	Not defined
US	915 MHz	**KNX RF4**	Not defined	Not defined	Not defined

Devices not having the same RF sub-bands are not compatible.

To ease the installation and avoid any compatibility problems, the KNX RF devices shall be marked as specified in clause 9.

5 KNX RF Physical Layer

5.1 Physical Layer for KNX RF Ready and BiBat

5.1.1 Signalling for KNX RF1

Table 3- General requirements for Physical Layer Type KNX RF1 Ready and BiBat

Characteristic	Value or applicable standard
Tx centre frequency	f_c = 868,300 MHz
Bandwidth	600 kHz
Max. Tx frequency tolerance	± 25 ppm [a]
Tx duty cycle max	1 %
Tx modulation type	FSK
FSK deviation	f_{DEV} = ± 48 kHz to ± 80 kHz typically 60 kHz
Tx chip rate	32 768 chips per second
Maximum Tx chip rate tolerance	± 1,5 %
Maximum Tx jitter per transition	± 5 µs
Tx ERP	Typical : 0 dBm Min : -3 dBm Max: +14dBm
Rx blocking performance	according EN 300 220-1, category 2 receivers [b]
Rx centre frequency	f_c = 868,300 MHz
Rx frequency tolerance	± 25 ppm KNX Tx to KNX Rx [a, b] ± 60 ppm Metering Tx to KNX Rx [a, b]
Minimal Rx chip rate tolerance	±2,0 % [b]
Rx radiated sensitivity	typical: -95 dBm [b] minimal: -80 dBm [b]
Minimal operating temperature range	0°C to 45°C [c]

[a] This frequency tolerance includes tolerances due to temperature variations within the operating temperature range and tolerances due to crystal aging.

[b] At Bit Error Rate (BER) 10^{-4} in optimum antenna direction.

[c] KNX Physical Layer parameters shall be met on the entire product temperature range declared by the manufacturer. (e.g. : -10°C to 70°C for outdoor usage)

NOTE 1 A link budget of 100 dB is recommended.

5.1.2 Telegram structure for RF Ready and BiBat

5.1.2.1 Telegram structure for KNX Ready systems

Table 4- KNX Ready systems Telegrams definition

Characteristics	Value	Notes
Data encoding	Manchester	chip "0" means f_{LO} (= f_C - f_{DEV}) chip "1" means f_{HI} (= f_C + f_{DEV}) bit "0" is coded as f_{HI} to f_{LO} transition, chip sequence "10" bit "1" is coded as f_{LO} to f_{HI} transition, chip sequence "01"
Preheader	consists of Preamble, Manchester violation, Sync word	see below
Preamble	79x chip sequence "01" sent by Tx	learning sequence for Rx, number of preamble chips is not checked by Rx (~4.8 ms)
Manchester violation	chip sequence "000111"	necessary for capture effect
Sync word	chip sequence "011010010110"	useful for synchronisation on chip rate
Postamble	2 chips to 8 chips	software reasons, mandatory for all Tx, number of postamble not checked by Rx.
Capture effect	optional	Preheader allows it; Rx may use it

5.1.2.2 Telegram structure for BiBat systems

The Frame structure is the same as in 5.1.2.1 except for the preamble length.

Table 5- BiBat systems Telegrams definition

Characteristics	Value	Notes
Preamble	min. 15x chip sequence "01" sent by Tx	learning sequence for Rx, number of preamble chips is not checked by Rx (~1 ms)

5.1.3 Medium access RF Ready &BiBat

5.1.3.1 Definition and use

Medium access control shall serve for prevention of collisions on the RF medium. For two reasons medium access cannot be completely controlled on RF.

1. Unidirectional senders access the medium at non-predictable times.
2. Non KNX RF devices access the medium at non-predictable times.

Bidirectional devices shall be able to sense whether the medium is free before they transmit. The inter-Frame time shall be the time interval during which a bidirectional device shall wait for a free medium (regardless of whether it is addressed by a preceding Frame or not). If no preamble is detected during this interFrame time the device may start sending.

If a Frame is received while the Physical Layer gets a request to send, the interFrame time shall start after the Frame reception is completed, this is after the last CRC is received. The same shall count for sending: if the Physical Layer gets a send request while it is sending, the interFrame time shall start when the last CRC is transmitted.

NOTE 2 RF supports no collision avoidance; therefore the transmission priorities are not coded in the Frame.

5.1.3.2 Medium access times

Table 6 – Medium access times

Type of frame	InterFrame time [Tint]	Random time [Trd]	Total medium access Time [Tma]
REPEATED Ready/BiBat frame	5ms	0 ms ≤ Trd < 10 ms	5 ms ≤ Tma < 15 ms
Ready / BiBat frame Bidirectional devices	15 ms	0 ms ≤ Trd < 15 ms	15 ms ≤ Tma < 30 ms
Ready / BiBat frame Unidirectional devices	150 ms	0 ms ≤ Trd < 10 ms	150 ms ≤ Tma < 160 ms

The assumed typical ‘blind time’ for devices is 1 ms.

The step for the random time shall be 1 ms.

5.2 Physical Layer for KNX RF Multi

5.2.1 General requirements (KNX RF Multi)

The RF channels used in the KNX RF Multi shall be composed of the following 3 + 2 RF channels.

Table 7 – RF channels of the KNX RF Multi Physical Layer

RF channel name	Abbreviation	signalling speed kbps	encoding	preamble length ms
Primary fast RF channel	F1	16,384	Manchester	15
Second fast RF channel	F2	16,384	Manchester	15
Third fast RF channel	F3	16,384	Manchester	15
Primary slow RF channel	S1	8,192	Manchester	500
Second slow RF channel	S2	8,192	Manchester	500

The KNX RF frequency used by former devices in compliance with the KNX RF1.1 specification is called F1sh.

The RF channels shall be divided in two categories.

1. The first category shall contain RF channels for “fast” RF Telegrams. The fast Telegrams shall be composed of a short **15 ms** wake-up at **16,384 kbps** signalling speed.
2. The second category contains RF channels for “slow” RF Telegrams. The slow Telegrams are composed of a long **500 ms** wake-up at **8,192 kbps** signalling speed.

The two categories of RF channels define explicitly two application domains.

EXAMPLE 1 Non-permanent receivers: smoke sensors, heating control.

EXAMPLE 2 Permanent receivers: all devices concerning human interaction

The first three Fast RF channels are primarily used in application for fast permanent and non-permanent receivers and the last two RF channels are primarily used for slow non-permanent Rx devices. The receiver reception capability determines the preamble length.

Devices from both categories can cohabitate independently without link or with links done by specific mains powered products receiving all the 5 RF channels.

Fast Telegrams are Telegrams transmitted on any of the Fast RF channels; slow Telegrams are Telegrams transmitted on a Slow RF channel.

Typically NPRM devices using slow Telegrams can only receive RF Telegrams with long wake-up. This enables the NPRM devices to stay in low power mode most of the time and to become periodically active for the reception of a long preamble. It is suggested that battery powered devices use mainly the RF channels S1 and S2 with the lower data rate.

PRM devices supporting the Fast RF channels shall be in permanent scanning and receiving mode.

NPRM devices supporting the Fast RF channels shall be in a scanning mode of each RF channel every 15 ms (1 ms for hopping, 1 ms for scanning one RF channel for an example). Is there a preamble for me?" If yes, listen to the Frame, or go to next RF channel).

A receiver on only one RF channel without scanning the other is not allowed.

In the first three Fast RF channels, the first RF channel is the one used by existing RF KNX 1.1 devices and the two others Fast RF channels are escape RF channels used in case of a busy RF channel.

In the last two Slow RF channels, the first one is S1 for slow products and the other Slow RF channels is an escape RF channel used in case of a busy RF channel.

Devices will mainly use F1, F2, F3 or S1, S2 or possibly the 5 RF channels in very seldom applications.

5.2.2 Physical Layer type RF Multi

5.2.2.1 KNX RF1 channel definitions for RF channels F1, F2 and F3

Table 8 – RF channel definitions for RF1 channels F1, F2 and F3

Parameter	Value		Comment
Tx centre frequency	Channel F1	868,300 MHz	**KNX Tx to KNX Rx** [b] Tx : ±25 ppm Rx : ±25 ppm **Metering Tx to KNX RF Multi** [c] Tx : ±60 ppm Rx : ±60 ppm
	Channel F2	868,950 MHz	
	Channel F3 [a]	869,850 MHz	
TX radiated power	Typical: 0 dBm Min: -3 dBm Max: +14 dBm		
Deviation	± 48 kHz to ± 80 kHz		Typical : 60 kHz
Max allowed bandwidth	500 kHz 300 kHz		For F1 and F2 For F3 [a]
Tx max duty cycle	1 % for F1 0,1 % for F2 100 % for F3		For F3 only, Duty Cycle is 100 % up to a maximum radiated power of 5 mW and restricted to 1 % from 5 mW to 25 mW
Tx chip rate	32 768 chips per second		
Maximum Tx chip rate tolerance	±1,5 %		
Maximum Tx jitter per transition	±5 µs		
Sensitivity max	-95 dBm typical -80 dBm min		Radiated test [d] BER : 10^{-4}
Minimal Rx chip rate tolerance	± 2 %		
Preamble length	247x chip sequence “01”		~15 ms, number of preamble chips is not checked by Rx
Receiver blocking performance	Minimum category 2		Category 2 according EN 300 220
Minimal operating range	0°C to 45°C [e]		

[a] RF channel F3 is optional. It might not be implemented by hardware.

[b] This frequency tolerance includes tolerances due to temperature variations within the operating temperature range and tolerances due to crystal aging.

[c] Frequency error correction may be needed in the case of Tx metering to KNX Rx specific products. Metering only applies to RF channel F1.

[d] At Bit Error Rate (BER) 10^{-4} in optimum antenna direction.

[e] KNX Physical Layer parameters shall be met on the entire product temperature range declared by the manufacturer. (e.g. : -10°C to +70°C for outdoor usage)

5.2.2.2 KNX RF1 channel definitions for RF channels S1 and S2

Table 9 – RF channel definitions for RF1 Multi channels S1 and S2

Parameter	Value		Comment
Tx centre frequency	Channel S1	869,850 MHz	Tx: ±25 ppm [a] Rx: ±25 ppm
	Channel S2	869,525 MHz	
Tx radiated power	Typical: 0 dBm Min: -3 dBm Max: +14 dBm		
Deviation	±20 kHz to ±65 kHz		The usage of frequency error correction may be needed to guarantee good receiver performances.
Max allowed bandwidth	300 kHz 250 kHz		For S1 For S2
Tx max duty cycle	100 % for S1 10 % for S2		For S1 only, Duty Cycle is 100 % up to a maximum radiated power of 5 mW and restricted to 1 % from 5 mW to 25 mW
Tx chip rate	16 384 chips per second		
Maximum Tx chip rate tolerance	Tx: ±1,5 %		
Maximum Tx jitter per transition	±5 µs		
Sensitivity max	-95 dBm typical -80 dBm min		Radiated test [b] BER = 10^{-4}
Minimal Rx chip rate tolerance	± 2 %		
Preamble length	4111 chip sequence “01”		~500 ms, number of preamble chips is not checked by Rx
Receiver blocking performance	Minimum category 2		Category 2 according EN 300 220
Minimal operating range	0°C to 45°C [c]		

[a] This frequency tolerance includes tolerances due to temperature variations within the operating temperature range and tolerances due to crystal aging.

[b] At Bit Error Rate (BER) 10^{-4} in optimum antenna direction.

[c] KNX Physical Layer parameters shall be met on the entire product temperature range declared by the manufacturer. (e.g. : -10°C to +70°C for outdoor usage)

5.2.3 Telegram structure for KNX RF Multi systems

Table 10 – KNX RF Multi Telegrams definition

Characteristics	Value	Notes
Data encoding	Manchester	chip “0” means f_{LO} (= f_C – f_{DEV}) chip “1” means f_{HI} (= f_C + f_{DEV}) bit “0” is coded as f_{HI} to f_{LO} transition, chip sequence “10” bit “1” is coded as f_{LO} to f_{HI} transition, chip sequence “01”
Preheader	consists of Preamble, Manchester violation, Sync word	see below
Preamble	See above	Depends on which RF channel is used.
Manchester violation	chip sequence “000111”	necessary for capture effect
Sync word	chip sequence “011010010110”	useful for synchronisation on chip rate
Postamble	2 chips to 8 chips	software reasons, mandatory for all Tx, number of postamble not checked by Rx. (Only applicable if no Fast Ack is used, refer to 6.6.4.3.)
Capture effect	optional	Preheader allows it; Rx may use it

5.3 Physical Layer for RF BiBat 2

5.3.1 General requirements for RF BiBat 2

RF BiBat 2 shall use the RF channels as specified in Table 11.

NOTE 3 RF BiBat 2 does not use any of the RF channels F1sh, F2, F3, S1 or S2. The RF channel F4 is exclusively used by RF BiBat 2.

Table 11 – KNX RF1 channels of the KNX RF BiBat 2 Physical Layer

RF channel name	Abbreviation	signalling speed kbps	encoding	preamble length ms
Primary fast RF channel	F1	16,384	Manchester	1
Fourth fast RF channel	F4	16,384	Manchester	1

The access characteristics are such that BiBat systems are extended to two RF channels, while keeping the synchronous, fast access to an RF channel. Therefore, the two RF channels are accessed following a duty cycling approach and not using LBT. Doing so, we support a communication scheme for battery-driven devices with only sparse communication needs, but with a high demand for long battery lifetime.

The general idea of the proposal is to use the same time-slot-system as proposed in the BiBat extension to the KNX RF1.1 standard, extended with a mechanism working on multiple frequencies. The use of a frequency is fixed for a synchronized period consisting of 64 blocks and a total duration of a maximum of 5 minutes and 20 s. Besides the below-stated changes, the BiBat standard shall be followed.

5.3.2 KNX RF1 channel definitions for RF channel F4

The technical parameters of escape RF channel F4 to the historical RF channel 1 are specified in Table 12.

Table 12 – KNX RF1 channel definition for RF channel F4

Parameter	Value		Comment
Tx centre frequency	F4	869,525 MHz	Tx: ±25 ppm Rx: ±25 ppm
Radiated power	Typical: 0 dBm Min: -3 dBm Max: +14 dBm		
Deviation	±50 kHz to ±60 kHz		
Max allowed bandwidth	250 kHz		
Data rate [a]	16,384 kbps Manchester		Tx: ±1,5 % Rx: ±2 %
Max Tx jitter per transition	±5 µs		
Sensitivity max	Typ. -95 dBm Max -80 dBm		Radiated test
Medium access	10 % duty cycle		
Preamble length	Min 15 chip sequence “01”		~1 ms

[a] The specification of the data rate combined with the maximum allowed bandwidth for F4 is a challenge for today’s RF chips. However, achieving this data rate within this bandwidth should not be a problem in the future.

This RF1 channel F4 is located at the same centre frequency as S2, but shall only be used with RF BiBat 2 systems.

NOTE 4 Neither management of the RF channel selection nor RF channel changing algorithms for BiBat 2 systems are described in this document.

6 KNX RF Data Link Layer

6.1 KNX RF Data Link Layer for all KNX RF devices

6.1.1 Differences to existing (bidirectional) KNX protocol

6.1.1.1 Extended Group Address

The Extended Group Address (8 octets) in a KNX RF Frame shall be the combination of the standard KNX Group Address (2 octets) with the KNX Serial Number of the sender of the Frame (6 octets). Every group addressed KNX RF Frame shall contain an Extended Group Address.

Any received Frame shall be taken in account by the receiver only if the Extended Group Address of the sender is known by the receiver.

NOTE 5 According to the RF Frame, these 8 octets are not transmitted consecutively.

The RF Frame shall contain the KNX Serial Number of the sender for the following communication modes:

- point-to-multipoint, connectionless (multicast) and
- point-to-system, connectionless (system broadcast).

This shall be indicated by the value 0 of the field AddrExtensionType in the second block of the RF Frame. Multicast Telegrams received with the wrong value of the AddrExtensionType shall be discarded by the receiving Data Link Layer instance.

For other communication modes, the RF Domain Address shall be used.

In any Frame in system broadcast communication mode the Destination Address shall be 0000h and the Address Type shall be “group”.

6.1.1.2 Predefined Extended Group Addresses for transmit-only devices

Transmit only devices shall use Extended Group Addresses. As transmit-only devices only have sending Datapoints (only one Group Address per Datapoint), all addresses can and shall be factory set.

- For Group Addresses

  For all unidirectional sensors, Datapoint 1 shall have Group Address = 0001h, Datapoint 2 shall have Group Address = 0002h, Datapoint N will have Group Address = N, with as result on the bus Extended Group Address (KNX Serial Number of sensor, 0001h) , (KNX Serial Number of sensor, 0002h) and (KNX Serial Number of sensor, N). These Group Addresses shall be unique for each sender.

- For Individual Addresses

  All devices shall have the default Individual Address (05FFh).

- Device Descriptor type 2

  DD2 shall show the currently configured status of the RF channel. Please refer as well to [05], clause 1.2 “Pre-assigned Group Addresses in unidirectional devices”.

  In case generic channels are used, after sending the DD2, the transmit only device shall spontaneously send its parameters via A_NetworkParameter_Write-PDUs (PID_CONFIG_LINK[Channel_Param_Response] [1]. This parameter block shall contain the currently configured state.

[1] This procedure is specified for PB-Mode. Please refer to [05], clause “Link procedure”.

6.1.1.3 Pre-defined Extended Group Addresses for bidirectional devices

For PB-Mode, also the sending Datapoints of bidirectional devices shall use Extended Group Addresses that shall be composed in the same way: Device KNX Serial Number, Group Object Number.

This requirement does not hold for S-Mode devices.

The consequence from this is that groups consist of one sender and n receivers, hence form a 1-to-n relationship. If several senders control a group of actuators, each of these actuators shall listen to the sending addresses of all senders.

6.1.1.4 RF Domain Address

The RF Domain Address shall be a 6 octet number. The RF Domain Address in an RF installation shall always be identical to the KNX Serial Number of one of the devices in the installation. This shall guarantee that the RF Domain Address is a unique number.

The RF Frame shall contain the RF Domain Address for the following communication modes:

- point-to-point, connectionless,
- point-to-point, connection-oriented and
- point-to-all-points, connectionless (broadcast).

This shall be indicated by the value 1 of the field AddrExtensionType in the second block of the RF Frame. Point-to-point connectionless and point-to-point connection-oriented Telegrams received with the wrong value of the AddrExtensionType shall be discarded by the receiving Data Link Layer instance.

For other communication modes, the KNX Serial Number shall be used.

In any Frame in broadcast communication mode the Destination Address shall be 0000h and the Address Type shall be “group”.

6.1.1.5 RF Broadcast and RF System Broadcast

Broadcasts can be broadcasts within an installation or system broadcasts. Whether a broadcast is a system broadcast shall be indicated by the AddrExtensionType field in the second block of the RF Frame.

- 0: system broadcast (shall not be restricted to the RF installation = domain; the Frame shall contain the KNX Serial Number of the sender).
- 1: broadcast (shall be restricted to the installation = domain; the Frame shall contain the Domain Address).

6.1.2 Data Link Layer Frame

6.1.2.1 General

This clause specifies the Frame format of the KNX-RF system.

NOTE 6 No difference is made in coding Standard Telegrams and Extended Telegrams as on the other media.

6.1.2.2 Structure

The Frame format builds on the FT3 Data Link Layer (IEC 870-5). The Frame consists of a preamble (Physical Layer), several data blocks, each followed by 2 octets CRC, and a postamble (Physical Layer).

The first data block has a fixed length of 10 data octets. The following blocks contain 16 data octets, except the last block, which may contain less than 16 octets (the remainder).

NOTE 7 The Frame structure of Extended Telegrams is used, this is, there is no difference between standard and Extended Telegrams as on other media. The KNX-Ctrl octet in the second data block contains the 4 bits “Frame format”.

	10 octets	2 octets	16 octets	2 octets		2 octets	
preamble	data block 1	CRC	data block 2	CRC	…	CRC	postamble

Figure 1 - Overview of the Data Link Layer Frame

6.1.2.3 Bit and octet order

Data shall be transmitted most significant bit (msb) first.

For data fields consisting of multiple octets (e.g. KNX Serial Number/Domain Address and Device Addresses) the most significant octet (MSB) shall be transmitted first.

6.1.2.4 First block

octet 1	octet 2	octet 3	octet 4								octet 5 to octet 10	octet 11 and octet 12
Length	C	Esc	RF-info								SN/DoA	CRC
			7	6	5	4	3	2	1	0		
			reserved				received signal strength		battery state	unidir		
	44h	FFh	0									

Figure 2 - Structure of the first block

- **Length**

Description: According to IEC870-5: total number of user octets counted from the C-field (excluding the CRCs). FFh value is reserved for future use." See [01].
For Media Couplers, see clause 6.1.6.4.

- **C ()**

Description: According to IEC870-5. KNX only uses SEND/NO REPLY (C = 44h)

- **Esc ()**

Description: This field shall have the fixed value FFh.

- **RF-info – bit 7**

Description: This bit is reserved for KNX system and shall be set to 0.

- **RF-info – bit 6**

Description: This bit shall be set to 0 by the sender. See specific usage for BiBat.

- **RF-info – bits 5 and 4**

Description: These bits shall be set to 00b by the sender.

- **RF-info – bits 3 and 2**

Description: This field shall contain the received signal strength indication.
This field shall be filled in by the Retransmitter with the lowest received signal strength; other senders shall always fill in the value 00h for this field.
The Retransmitter shall not change the value if it cannot measure the signal strength.

Encoding:
00b: void (no measurement)
01b: weak
10b: medium
11b: strong

- **RF-info – bit 1**

Description: This field shall contain the battery state of the sender of the Frame.

Encoding:
0: battery is weak
1: battery is ok

- **RF-info – bit 0: Unidir**

Description: Unidir

Encoding:
0: Frame sent by bidirectional device
1: Frame sent by unidirectional device

- **SN/DoA()**

Description: KNX Serial Number or Domain Address of the sender.
The field AddrExtensionType in the LPCI in the second block shall indicate whether this field contains the KNX Serial Number or the Domain Address.

- **CRC**

Description: CRC according to IEC870-5-1
Encoding: For information: The CRC according to FT3 of IEC 870-5-1 uses

$$2^{16}+2^{13}+2^{12}+2^{11}+2^{10}+2^{8}+2^{6}+2^{5}+2^{2}+2^{0}$$

as a generator polynomial. It starts with zero and treats the data msb first. The CRC result is complemented. The MSB of the 16-Bit CRC is transmitted first.

EXAMPLE 3 The sequence 01 02 03 04 05 06 07 08 has the CRC FCBCh.

6.1.2.5 Second block for Standard Telegrams

octet 1							
KNX Ctrl							
7	6	5	4	3	2	1	0
f	f	f	f	EFF	EFF	EFF	EFF

octet 2								octet 3							
Source Address															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

octet 4								octet 5							
Destination Address															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

octet 6							
LPCI							
7	6	5	4	3	2	1	0
AT	RC	RC	RC	LFN	LFN	LFN	AET

octet 7							
7	6	5	4	3	2	1	0
TPCI	TPCI	Seq. number	Seq. number	Seq. number	Seq. number	APCI	APCI

octet 8								octets 9 to octet n								octet n+1 and n+2							
								data								CRC							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
APCI	APCI	APCI/Data	APCI/Data	APCI/Data	APCI/Data	APCI/Data	APCI/Data																

Figure 3 - Structure of the second block for Standard Telegrams

- **ffff field**

Description: The Frame format field is described in clause 6.1.3.

- **Extended Frame Format (EFF)**

Description: The Extended Frame Format field shall specify the format of the Frame.
Encoding: 0000b = Standard Frame
Other values shall be used for the LTE-Frame (see 6.1.2.6) or are reserved (see [01]).

- **Source Address (SA)**

Description: The Source Address shall contain the Individual Address of the device that initiates the transmission of the Frame.
Encoding: See [01] clause “Individual Address”.

- **Destination Address (DA)**

Description: This shall be the Destination Address of the Frame and shall be an Individual Address or a Group Address.
Encoding: See [01] for the format of the Individual Address and the Group Address.

- **Address Type (AT)**

Description: This field shall specify whether the Destination Address is an Individual Address, a Group Address, an LTE Group Address or the broadcast address.
Encoding: For the Standard Frame, the encoding shall be as follows:
0: Individual Address
1: Group Address

- **Repetition counter (RC)**

Description Specifies the maximum number of repetitions allowed for one Frame.

- **Data Link Layer Frame Number (LFN)**

Description Sequence counter to discriminate successive Frames.

- **Address Extension Type (AET)**

Description: For the Standard Frame, the AET shall be used as follows:
Encoding: 0: The field SN/DoA in the first block shall be interpreted as the KNX Serial Number of the sender.
1: The field SN/DoA in the first block shall be interpreted as the RF Domain Address.

- **TPCI**

Description: The TPCI field shall contain the Transport Layer service indication.
Encoding: See [02].
00b: unnumbered data
01b: numbered data
10b: unnumbered control
11b: numbered control

- **Sequence Number**

Description: This field shall contain the Sequence Number of the Frame.

- **APCI**

Description: The APCI field shall contain the Application Layer service indication.
Encoding: See [03].

- **Data**

Description: Up to 8 data octets in this block (16 octets max block length), subsequent data octets in following blocks (each block 16 octets, except the last block, which may contain less than 16 data octets.)

6.1.2.6 Second block for LTE Extended Telegrams

		7	6	5	4	3	2	1	0
octet 1	KNX Ctrl	f	f	f	f	EFF	EFF	EFF	EFF
octet 2	Source Address								
octet 3	Source Address								
octet 4	Destination Address								
octet 5	Destination Address								
octet 6	LPCI	AT	RC	RC	RC	LFN	LFN	LFN	AET
octet 7		TPCI	TPCI	Seq. number	Seq. number	Seq. number	Seq. number	APCI	APCI
octet 8		APCI	APCI	APCI	APCI	APCI	APCI	APCI	APCI
octet 9	Interface Object Type	IOT	IOT	IOT	IOT	IOT	IOT	IOT	IOT
octet 10	Interface Object Type	IOT	IOT	IOT	IOT	IOT	IOT	IOT	IOT
octet 11	IO instance	IOInst	IOInst	IOInst	IOInst	IOInst	IOInst	IOInst	IOInst
octet 12	PID								
octets 13 to octet n	data								
octet n+1 and n+2	CRC								

Figure 4 - Structure of the second block LTE Extended Telegrams

- **ffff field**

Description: The Frame format field is described in clause 6.1.3.

- **Extended Frame Format (EFF)**

Description: The Extended Frame Format field shall specify the format of the Frame.
Encoding: 01xxb = LTE Extended Frame
Other values shall be used for the Standard Frame (see 6.1.2.5) or are reserved (see [01]).

- **Source Address (SA)**

Description: The Source Address shall contain the Individual Address of the device that initiates the transmission of the Frame.
Encoding: See [01] clause "Individual Address".

- **Destination Address (DA)**

Description: For the LTE Extended Frame, this field shall contain the LTE Group Address.
Encoding: See [06] clause "LTE Group Address Extension".
The DA shall be evaluated as specified in 6.1.5.3.

- **Address Type (AT)**

Description: This field shall specify whether the Destination Address is an Individual Address, a Group Address, an LTE Group Address the broadcast address or the broadcast address.
Encoding: For the LTE Extended Frame, the encoding shall be as follows:
0: Individual Address
1: LTE Group Address

- **Repetition counter (RC)**

Description Specifies the maximum number of repetitions allowed for one Frame.

- **Data Link Layer Frame Number (LFN)**

Sequence counter to discriminate successive Frames.

- **Address Extension Type (AET)**

Description: For the LTE Extended Frame, the AET shall be used as follows:
– A unidirectional sender shall use its own KNX Serial Number
⇒ AddrExtensionType = 0
– A bidirectional sender shall use the Domain Address
⇒ AddrExtensionType = 1
Encoding: 0: Frame shall contain the KNX Serial Number of the sender.
1: Frame shall contain the Domain Address.

- **TPCI**

Description: The TPCI field shall contain the Transport Layer service indication.
Encoding: For the LTE Extended Frame, the TL service shall have the fixed value:
00b: unnumbered data

- **Sequence Number**

Description: For the LTE Extended Frame, the TL service shall have the fixed value:
0001b: T_DATA_TAG_GROUP (See T_Data_Tag_Group in [02])

- **APCI**

Description: APCI for LTE Telegrams: see [06].

- **Interface Object Type (IOT)**

Description: See [06].

- **Interface Object Instance (IOInst)**

Description: See [06].

- **Property Identifier (PID)**

Description: See [06].

- **Data**

Description: Up to **4 data octets in this block** (16 octets max block length), subsequent data octets in following blocks (each block 16 octets, except the last block, which may contain less than 16 data octets.)

6.1.3 Use of the KNX Ctrl Field

6.1.3.1 Overview

Table 13 – KNX CTRL field values

KNX Ctrl bit								Frame Type
7	6	5	4	3	2	1	0	
0	0	0	0	e	e	e	e	Asynchronous Data Frames L_Data…
0	0	0	1	e	e	e	e	Fast_ACK
0	0	1	0	r	r	r	r	Reserved
0	0	1	1	r	r	r	r	Reserved
0	1	0	0	e	e	e	e	Synchronous Data Frames L_Data…:
0	1	0	1	0	0	0	0	BiBat Sync Frame
0	1	0	1	r	r	r	r	Reserved range xxxx ≠0000
0	1	1	0	0	0	0	0	BiBat Help Call
0	1	1	0	r	r	r	r	Reserved range xxxx ≠0000
0	1	1	1	0	0	0	0	BiBat Help Call Response
0	1	1	1	r	r	r	r	Reserved range xxxx ≠0000
1	0	0	0	e	e	e	e	RF Multi asynchronous Data Frames L_Data
1	0	0	1	e	e	e	e	RF Multi asynchronous Data Frames L_Data with Fast_ACK requested
1	0	1	0	0	0	0	0	RF Multi Repeater Acknowledge Frame
1	0	1	0	r	r	r	r	Reserved rrrr ≠ 0000
1	0	1	1	r	r	r	r	Reserved
1	1	r	r	r	r	r	r	Reserved range for BiBat 2, tbd. (xxxxxx ≠ 111111)
1	1	1	1	1	1	1	1	Escape value for future KNX Ctrl Extensions (2 Octet KNX Ctrl FFxx)
eeee = EFF field, see [01] r = reserved. Reserved bits are reserved for the KNX system. Reserved or unknown coding of KNX Ctrl shall not be used by the transmitter and shall be ignored by the receiver.								

6.1.4 Data Link Layer protocol

6.1.4.1 AddrExtensionType

The AddrExtensionType bit shall be a parameter of the Data Link Layer instance.

In transmission direction the sending Data Link Layer instance shall evaluate the AddrExtensionType and set the correct data in the field SN/DoA in block 1 of the transmitted Frame: either the devices' own KNX Serial Number or the Domain Address,

In reception direction the receiving Data Link Layer instance shall use this bit for a correct interpretation of the field SN/DoA in block 1 of the received Frame as either KNX Serial Number or Domain Address.

This Data Link Layer parameter shall be set by the Application Layer and shall be passed through the communication stack by the other layers as input to the Data Link Layer.

6.1.4.2 Duplication prevention

6.1.4.2.1 General

Please refer to [01] for medium independent requirements for error and exception handling.

In the presence of Retransmitters in the system, Frame duplications can occur in the receivers if both the original sender and repetitions sent by the Retransmitters are received. Therefore a mechanism is foreseen in Layer-2 to prevent the evaluation of duplicated Telegrams in receivers.

6.1.4.2.2 Transmitters

The Data Link Layer of each transmitter shall insert a Link layer Frame Number (LFN) into the LPCI of each sent Frame (see clause 6.1.2.5). The LFN shall be a 3 bit counter that shall be incremented for each transmitted Frame. After 8 Frames the counter shall wrap around and shall start off again from zero.

In order to increase the probability to have no Frame lost, the Frame with the same LFN can be resent.

6.1.4.2.3 Receivers

The receiver shall discard subsequent Telegrams that contain the identical LFN from the same sender. In case the LFN differs, the newly received LFN shall be stored.

Every receiver shall have a table to store the KNX Serial Number and the LFN of previously received Telegrams (no matter from where they are sent). The table length shall be less or equal to 7, because the LFN counts from 0 to 7. This avoids unintentional discarding of Telegrams, even if Telegrams from only one device are received.

This mechanism ensures that repeated Telegrams originating from the same sender up to within the following 8 Telegrams are discarded.

6.1.4.2.4 Error and exception handling for LTE Extended Telegrams

- **Invalid Frame formats**

  In the Ctrl field only Frame formats EFF = 01xxb for LTE Telegrams and EFF = 0000b for standard Telegrams shall be accepted. Other values of EFF are reserved and the receiver shall neglect the Frame without further action.

  Further error and exception handling for LTE Telegrams is specified in [06].

6.1.5 Data Link Layer services

6.1.5.1 L_Data service and protocol

In addition to the general Data Link Layer protocol requirements specified in [01], for the L_Data-service, the following shall apply.

The L_Data service on KNX RF shall either be an unconfirmed or a confirmed Telegrams service. If the local Data Link User prepares an LSDU for one or more remote Data Link Layer users, it shall apply the L_Data.req service to pass the LSDU to the local Data Link Layer. The local Data Link Layer shall accept the service request and shall try to send the LSDU to the remote Data Link Layer Users. The Destination Address may be an Individual Address, a Group Address or the Broadcast Address. The Local Data Link Layer shall pass an L_Data.con primitive to the Local Data Link Layer User that shall indicate a correct or erroneous data transfer.

The Local Data Link Layer shall accept the L_Data.req service request. The fields SN/DoA, AT and AET shall be filled in as follows.

If the service parameter address_type denotes an Individual Address then

- the SN/DoA field in the Frame shall be filled with the RF Domain Address;
- the Address Extension Type (AET) shall be set to 1; this shall indicate that the SN/DoA field shall be interpreted as the RF Domain Address
- the Address Type field shall be set to 0; this shall denote that the Destination Address shall be interpreted as an Individual Address.

If the address_type denotes a Group Address then

- if the Destination Address value equals the broadcast address 0000h then
  - the SN/DoA field in the Frame shall be filled with the RF Domain Address;
  - the Address Extension Type (AET) shall be set to 1; this shall indicate that the SN/DoA field shall be interpreted as the RF Domain Address
  - the Address Type field shall be set to 1; this shall denote that the Destination Address shall be interpreted as a Group Address.
- if the Destination Address value does not equal the broadcast address 0000h but is a normal Group Address, then
  - the SN/DoA field in the Frame shall be filled with the KNX Serial Number of the sender;
  - the Address Extension Type (AET) shall be set to 0; this shall indicate that the SN/DoA field shall be interpreted as the KNX Serial Number of the sender;
  - the Address Type field shall be set to 1; this shall denote that the Destination Address shall be interpreted as a Group Address.

The service parameter Frame_format shall further indicate

- whether the Frame shall be transmitted using a standard Telegrams (see 6.1.2.5), or
- whether the Frame shall be transmitted using an LTE Extended Telegrams (see 6.1.2.6).

L_Data.req(address_type, destination_address, Frame_format, lsdu, octet_count, priority, source_address)

address_type:	This parameter shall be used to indicate whether the Destination Address is an Individual Address or a Group Address.
destination_address:	This parameter shall be used to indicate the Destination Address of the Frame to be transmitted; it shall be either an Individual Address or a Group Address.
Frame_format:	This parameter shall be used to indicate an extended Frame format.
lsdu:	This parameter shall be used to contain the user data to be transferred by Layer-2.
octet_count:	This parameter shall be used to indicate the length information of the requested Frame.
priority:	This parameter shall be used to indicate the priority that shall be used to the transmit the requested Frame; it shall be “system”, “urgent”, “normal” or “low”.
source_address	Individual Address of the device that requests the L_Data-service.

6.1.5.2 L_SystemBroadcast service and protocol

The Local Data Link Layer shall increment the Data Link Layer Frame Number (LFN) by one compared to the previous transmitted L_Data-Frame.

The Source Address shall be filled with the Individual Address of the Local Data Link Layer User.

6.1.5.3 Additional Frame acceptance criteria for LTE Extended Telegrams

In the Data Link Layer of the receiver the KNX Serial Number and the RF Domain Address shall be additional criteria for acceptance of the Frame as specified in Table 14.

Table 14 – Frame acceptance evaluation for RF LTE Extended Telegrams

EFF	LTE Group Address	AddrExtensionType	
		= 0: KNX Serial Number (SN)	= 1: Domain Address (DoA)
0100b	xxxxh	The receiver shall check the **SN** of the sender and discard the Frame if the SN is not contained in the **KNX Serial Number Table** of the receiver, else the Frame shall be processed further as specified in [06].	The receiver shall check the DoA and discard the Frame if the DoA is different from the DoA of the receiver, else the Frame shall be processed as specified in [06].
0101b	xxxxh		
0110b	xxxxh		
0111b	> 0000h		
0111b	= 0000h	**LTE System Broadcast** The receiver shall ignore the SN of the sender and process the LTE broadcast Frame as specified in [06].	**LTE Domain Broadcast** The receiver shall check the DoA and discard the Frame if the DoA is different from the DoA of the receiver, else the LTE broadcast Frame shall be processed further as specified in [06].

6.1.6 The Layer-2 of an RF-TP Media Coupler

6.1.6.1 Introduction

The RF-TP Media Coupler shall perform a bijective translation between the 8 octet RF Extended Group Address and the 2 octet standard Group Address and Individual Address.

NOTE 8 Each device that is required to receive messages from the Media Coupler shall store the KNX Serial Number of the Media Coupler.

Two translations are foreseen:

- Translation configured by a tool

The translation of the addresses uses a bijective table in the Media Coupler, which shall be downloaded by ETS®. It is the responsibility of the installer and ETS® to make sure that the translated Group Addresses are unique in the TP system.

- Automatic translation

The association between an Individual Address or a Group Address and an Extended Address shall be done by using an index number. The index number shall be associated to a KNX Serial Number, an Individual- or a Group Address. In order to build a bijective relation between an Extended Address and one Individual Address or one Group Address and to enable more than one Media Coupler in an installation and according to the requirements, the fields shall have the following size:

- Coupler number: 2 bit
- index: 6 bit

In order to enable more than 2^6 translated Extended Group Addresses, the Group Address part of the Extended Address shall be logically divided in two fields.

1. The first one called @bank on 12 bit shall be used in association with the KNX Serial Number and shall be translated in the index part.
2. The second one called @link on 4 bit shall be directly copied in the lower part of the TP translated Group Address. This coding shall provide the possibility to have more than 16 Extended Group Addresses translated per KNX Serial Number.

<table>
<tr><td>Octet 1</td><td>Octet 2</td><td>Octet 3</td><td>Octet 4</td><td>Octet 5</td><td>Octet 6</td><td>….</td><td colspan="8">Octet 7</td><td colspan="8">Octet 8</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="6">KNX Serial Number</td><td></td><td colspan="12">@bank</td><td colspan="4">@link</td></tr>
</table>

Figure 5 - Logical Interpretation of Extended Group Address in automatic translation

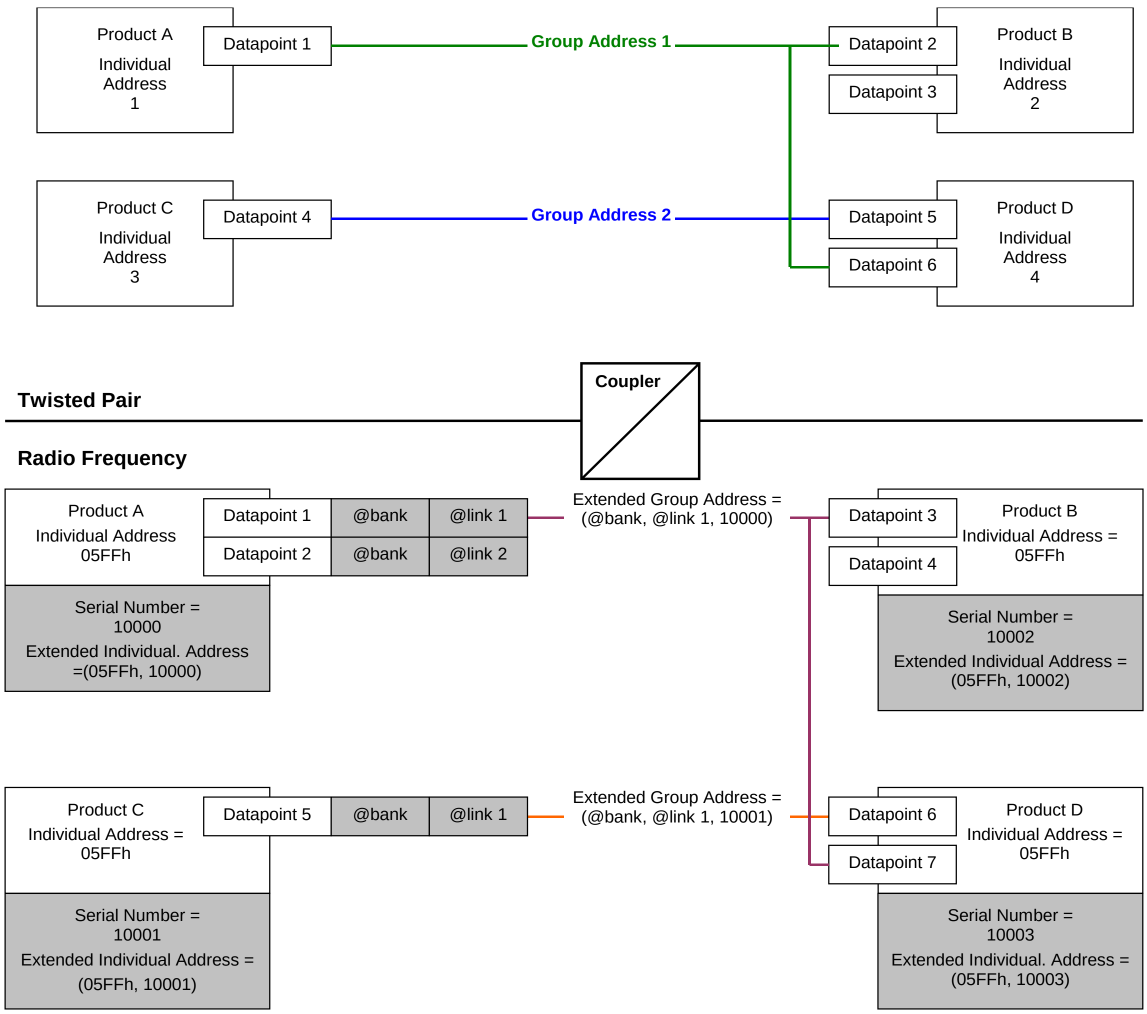


Figure 6 - Coupling a KNX TP and RF system

The different cases of translation are detailed in the next clauses.

6.1.6.2 Automatic translation

6.1.6.2.1 Messages coming from RF to TP

6.1.6.2.2 General requirements

For all messages coming from RF, the Source Address shall be translated according the rule as specified below. Then if the Destination Address is a Group Address the additional Group Address translation from RF to TP shall be used.

6.1.6.2.2.1 Source Address translation from RF to TP

The Source Address shall be replaced by a concatenation of the Coupler Subnetwork Address, the Media Coupler Number and the index.

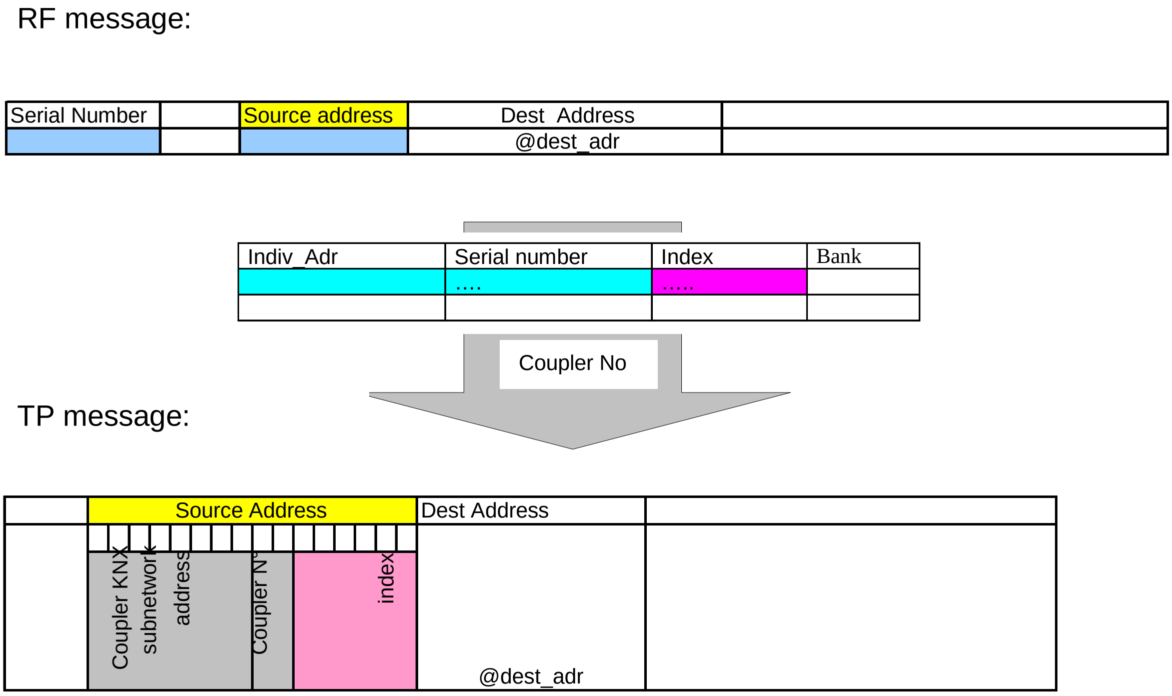


Figure 7 - Automatic translation principle from RF to TP of the source Individual Address

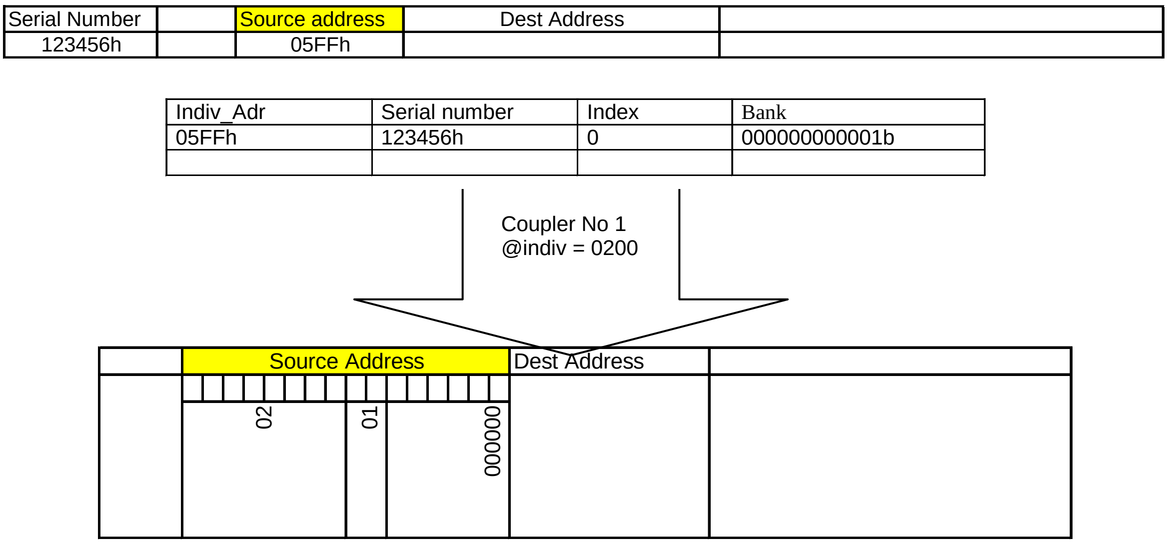


Figure 8 - Example

6.1.6.2.2.2 Group Addresses messages from RF to TP

In this case the Source Address shall be translated according the general rule of translation from RF to TP and the Destination Group Address is translated.

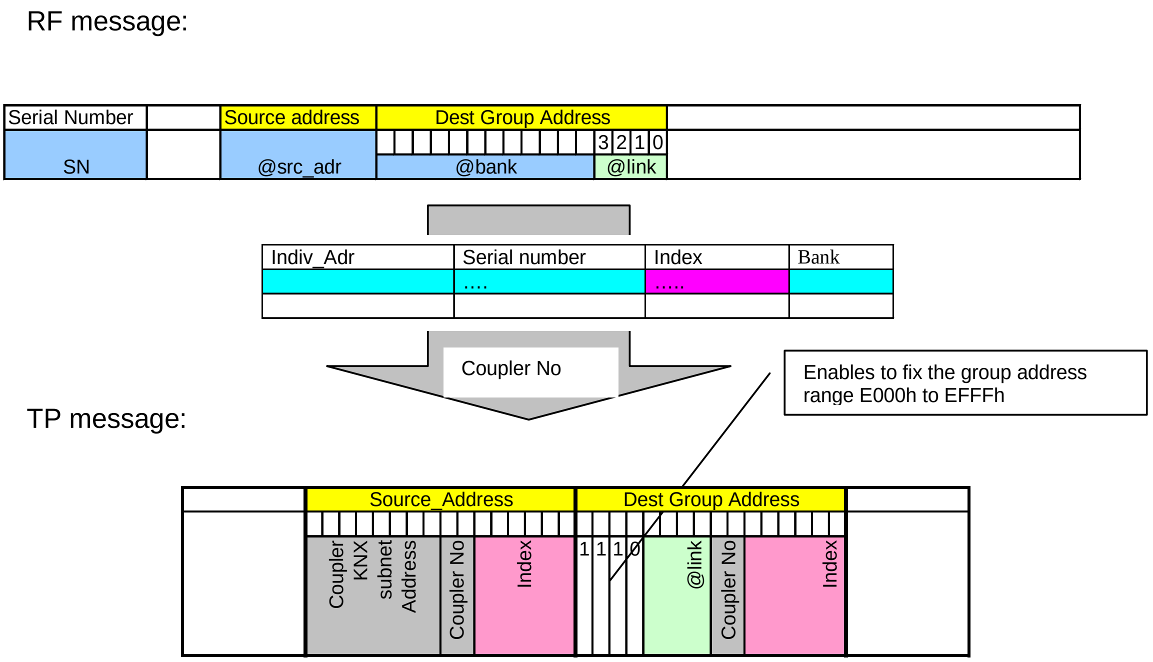


Figure 9 - Automatic translation principle from RF to TP of the Group Address

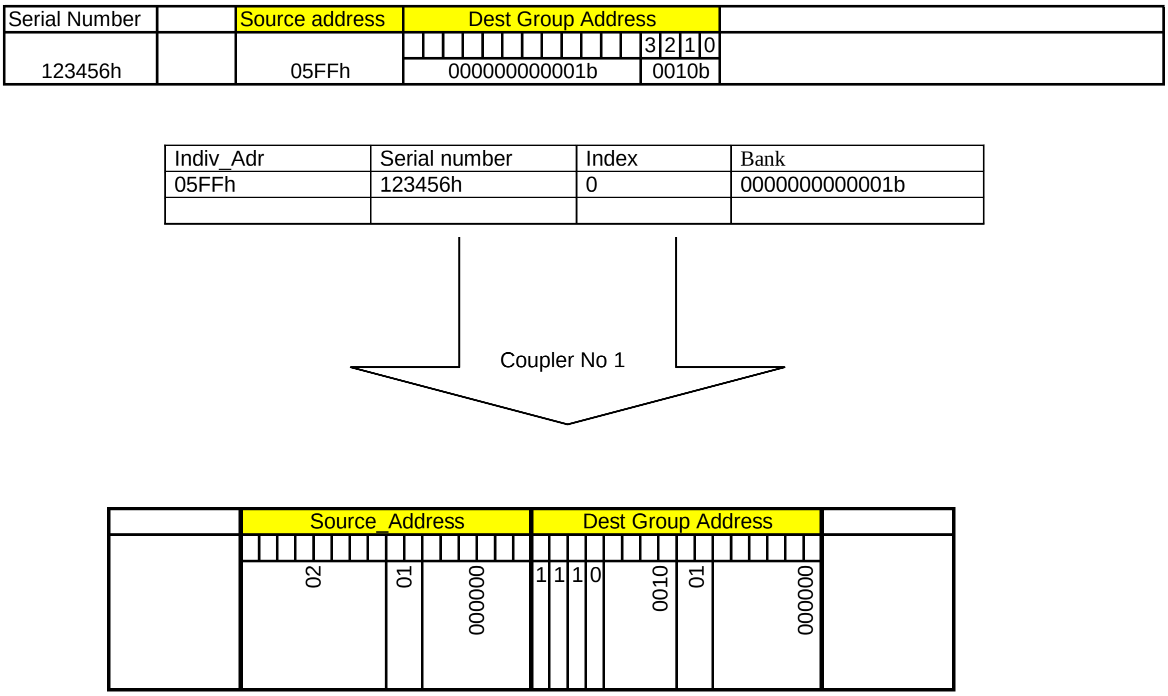


Figure 10 - Example

6.1.6.2.3 Messages coming from TP to RF

6.1.6.2.3.1 General requirements

All messages coming from TP shall keep the Source Address.

6.1.6.2.3.2 Group Address translation from TP to RF

If the destination Group Address is in the range of RF Group Addresses Exxxh

In this case the destination Group Address is translated. The Group Address is replaced by a concatenation of the corresponding KNX Serial Number, the bank number.

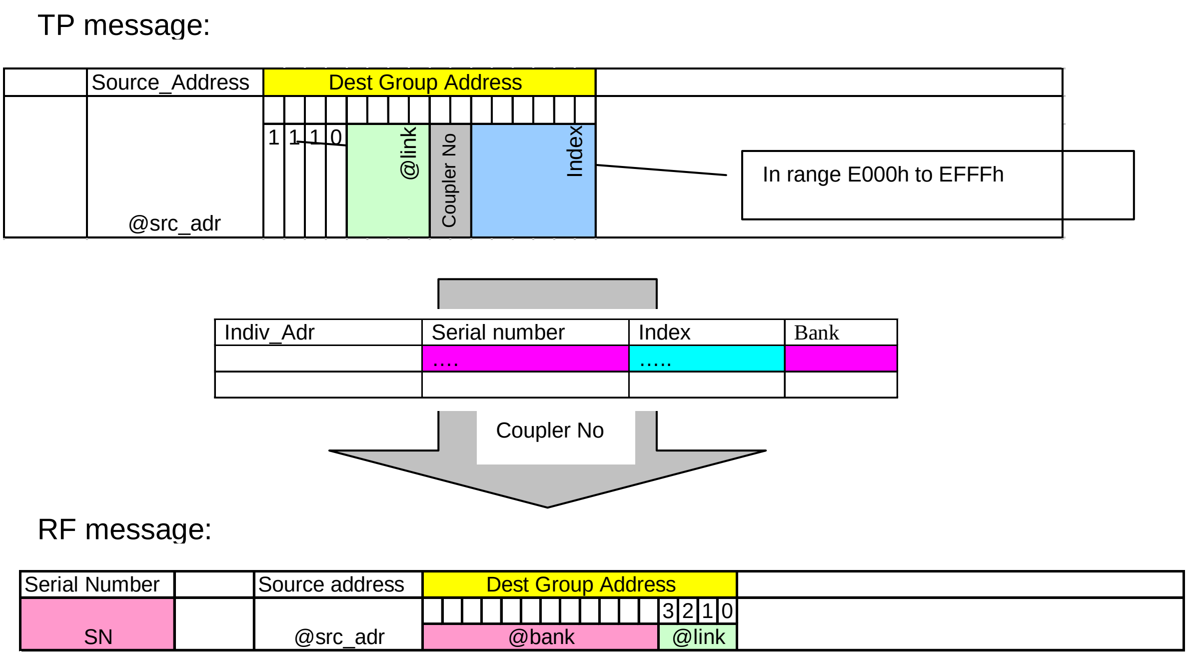


Figure 11 - Automatic translation principle from TP to RF of the Group Address

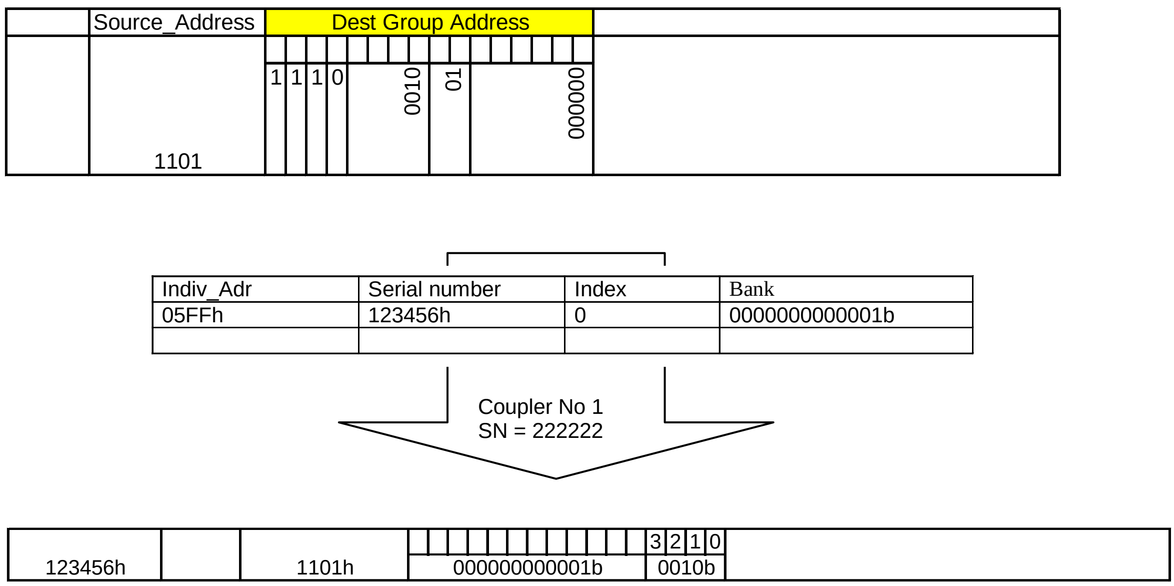


Figure 12 - Example

If the destination Group Address is NOT in the range of RF Group Addresses Exxx

The translation consists of adding the Coupler KNX Serial Number.

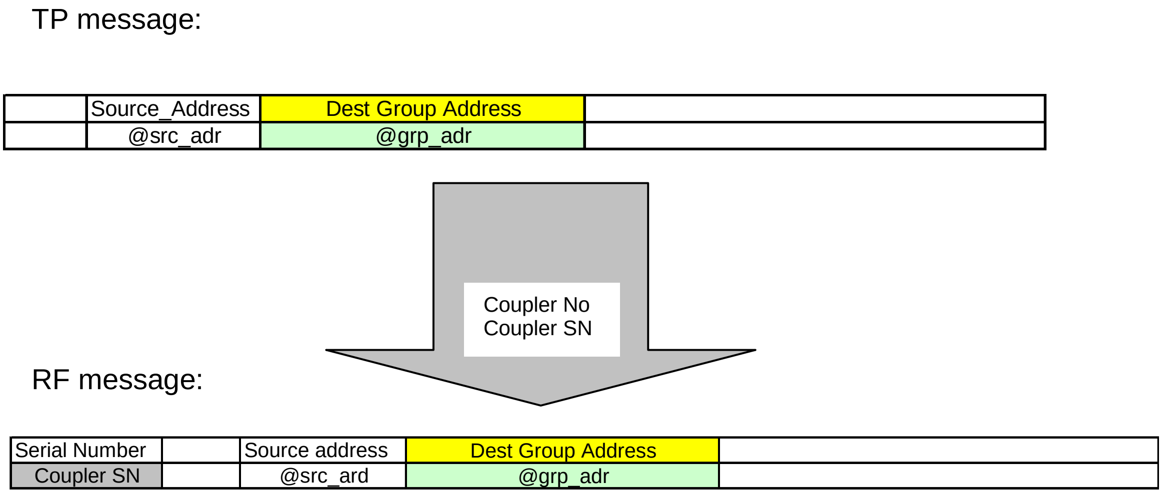


Figure 13 - Automatic translation principle from TP to RF in case the Group Address is not in the range of RF Group Addresses Exxx

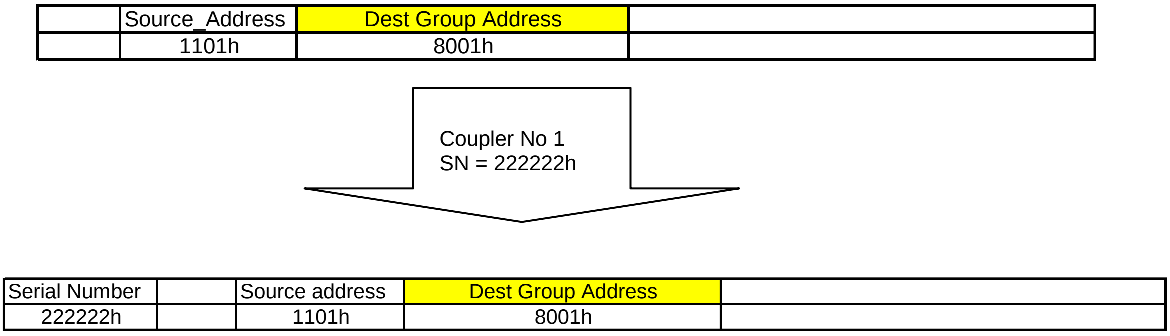


Figure 14 - Example

6.1.6.2.3.3 Individual addressing coming from TP to RF

If the Subnetwork Address of the Destination Individual Address is equal to the Coupler Subnetwork Address

The destination Individual Address is replaced by the one corresponding to the index.

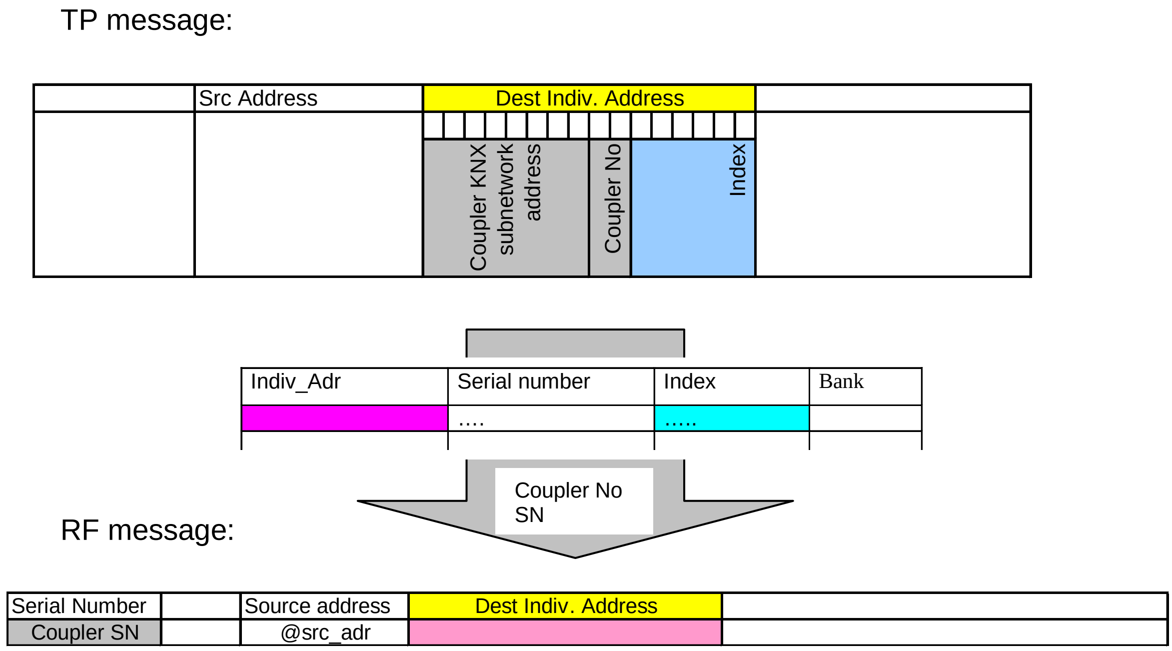


Figure 15 - Automatic translation principle form TP to RF for Individual Addresses

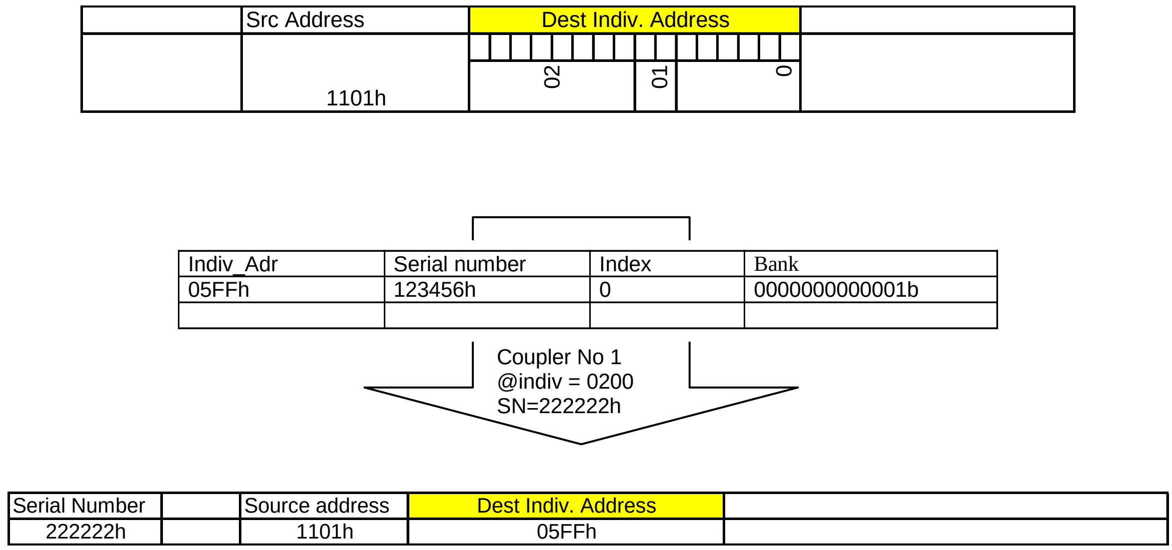


Figure 16 - Example

If the Subnetwork Address of the Destination Individual Address is NOT equal to the Coupler Subnetwork Address

The translation consists of adding the Coupler KNX Serial Number.

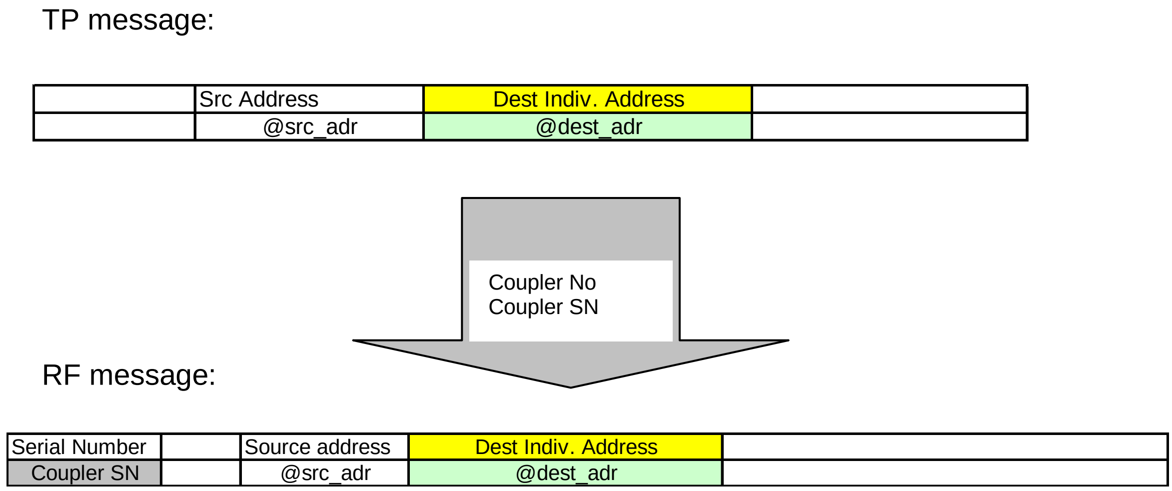


Figure 17 - Automatic translation principle from TP to RF for Individual Addresses if the destination Individual Address is not equal to the Coupler Subnetwork Address

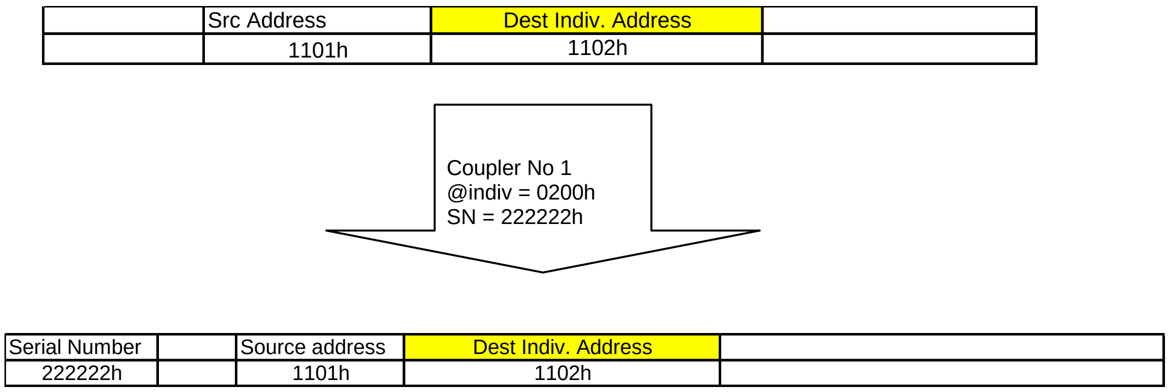


Figure 18 - Example

6.1.6.3 Configuration by a tool

Alternatively to the automatic translation described in clause 6.1.6.2, it is also possible to configure the Media Coupler by a tool, e.g. ETS.

6.1.6.4 Translation between standard and Extended Telegrams and RF Telegrams

Translation from other media to RF

- Ctrl field and extended Ctrl. Field (2 octets) in the L_Data_Extended Frame shall be replaced by the KNX-Ctrl field (extended Frame format + 4 reserved) in octet 1 of the second bloc.
- The length octet in the L_Data_Extended Frame shall be replaced by the LPCI.

Translation from RF to other media

- All Telegrams with length ≤ 20 octets shall be translated into an L_Data_Standard Frame with length (0 to 15) and repetition counter set with parameters from the Media Coupler.
- If the length > 20 octets, then it shall be translated into an L_Data_Extended Frame with "Extended Frame Format" = "standard message", length (0 to 254).

NOTE 9 In the resulting Frame the destination_adress_flag (DAF) is not at the same place, it depends on the Frame format.

Important

Check Octets shall be calculated or verified (according to the way of the message) by the Coupler.

For translation to Powerline (PL110), any router shall use System Broadcast communication mode if the APCI denotes a Domain Address management service.

6.1.7 The Layer-2 of an RF Retransmitter

6.1.7.1 History list

Senders shall set the LFN in the LPCI of every Frame, as stated in clause 6.1.4.2.2.

The Data Link Layer of every *Retransmitter* shall have a history list that shall store information about the previously received Telegrams. The history list shall contain the KNX Serial Number of the sender (not the Domain Addresses) and the LFN. If a Retransmitter receives a Frame, its Data Link Layer shall check if the received KNX Serial Number and LFN in this combination are contained in the history list. If this is the case, this Frame shall be discarded, otherwise the processing shall continue.

The history list shall have

- a minimal length of 3 entries, and
- a maximum length of 7 entries or shall provide a deletion mechanism after a timeout (e.g. 3 s).
- only one entry per KNX Serial Number.

6.1.7.2 RF Repetition counter

If the processing continues, the RF Repeater shall decrement the RF Repetition counter and compare it with a limit value.

If the processing continues, the RF Repetition counter of the received Frame shall be compared with a limit value.

- If the received value of the RF Repetition counter is larger than the limit value, then the RF Repetition counter shall be decremented and the Frame shall be repeated.
- If the received value of the RF Repetition counter equals to or is smaller than the limit value, then the Frame shall not be repeated and be ignored.

The limit value could optionally be a parameter into the RF Repeater to limit the number of RF Repeater levels. If implemented, this parameter shall be set or read using the property PID_RF_REPEAT_COUNTER (PID = 74) in the Device Object (object_type = 0). It shall be used by the RF Repeater in the following way:

```
if rf_repetition_counter(rec_Frame) > 0 and
   rf_repetition_counter(rec_Frame) > rf_repetition_counter_limit(Repeater)
          rf_repetetion_counter(rec_Frame)—
else
   discard(rec_Frame)
endif
```

6.2 KNX RF Data Link Layer for KNX RF Ready and BiBat systems

6.2.1 Data Link Layer protocol

6.2.1.1 Data Link Layer for RF Ready

The general KNX RF Data Link Layer requirements as specified in clause 6.1 “KNX RF Data Link Layer for all KNX RF devices” shall apply.

6.2.1.2 RF Repetition counter for end devices

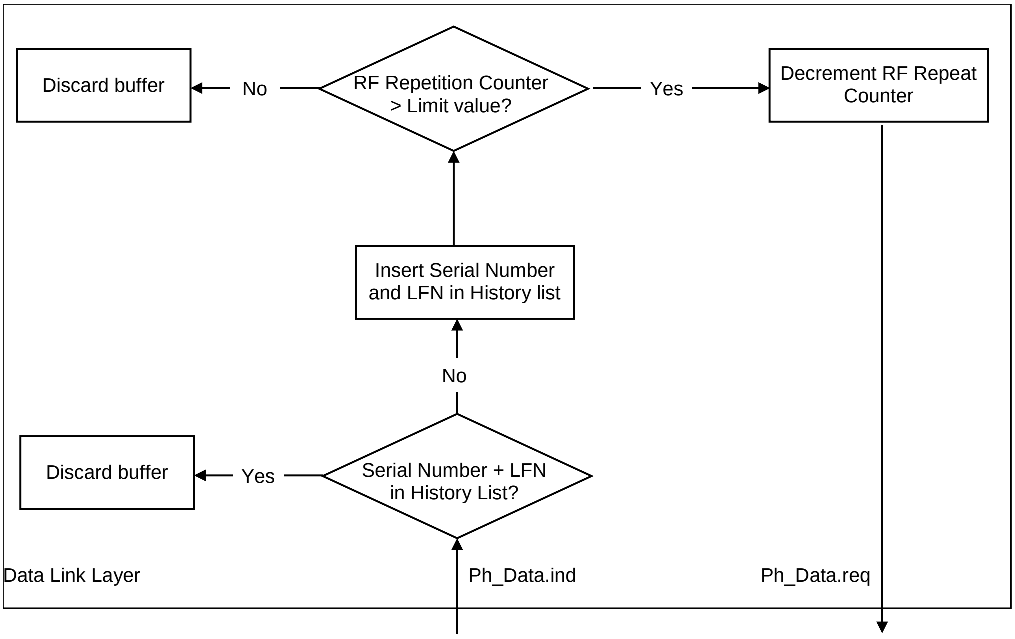
The RF Repetition counter shall be set to 6 for end devices.

6.2.2 The Layer-2 of an RF Retransmitter

6.2.2.1 Filtering

Filtering not implemented in KNX Ready and BiBat Retransmitters.

6.2.2.2 Retransmitter flowchart



NOTE This flowchart only describes the handling of received messages for repeating. The handling of messages to the internal management of the Retransmitter is not shown.

Figure 19 - KNX RF Ready and BiBat flowchart of the Data Link Layer and Network Layer of the Retransmitter

6.3 KNX RF Data Link Layer for BiBat

6.3.1 General

6.3.1.1 BiBat Master - BiBat Slave System

Synchronous communication to BiBat Slaves requires a BiBat Master that shall send all Telegrams to its associated BiBat Slaved in synchronized time-slots. This device is called BiBat Master. If data are pending, the BiBat Master shall send Telegrams in the time-slots that it has assigned at configuration time to its BiBat Slaves. If the application is idle, the BiBat Master shall send Data Link Layer sync Telegrams in predefined time-slots. A BiBat Slave shall use every Telegrams it receives from its BiBat Master or from a BiBat Retransmitter in its Synchronous BiBat System to synchronise its time to the BiBat Master Time.

The Domain Address used in this synchronous system shall be the KNX Serial Number of the BiBat Master.

6.3.1.2 Retransmitters

Optionally up to three Retransmitters supporting synchronous Telegrams may be present in one Synchronous BiBat System.

Synchronous Retransmitters cannot be cascaded, this is, a retransmitted Telegram cannot be retransmitted by another Retransmitter. Therefore a synchronous Telegram with hop count 5 or less shall not be retransmitted.

A Synchronous Retransmitter does not need any general time information; it shall only resend synchronous Telegrams with a fixed Retransmitter-specific time delay. Therefore a unique number 1 to 3 shall be assigned to each Synchronous Retransmitter by the BiBat Master during configuration.

A Synchronous Retransmitters shall retransmit synchronous Telegrams only with the RF Domain Address of the BiBat System or the KNX Serial Number of the BiBat Master.

Asynchronous Telegrams shall be retransmitted as defined in KNX Ready.

6.3.2 Synchronous communication

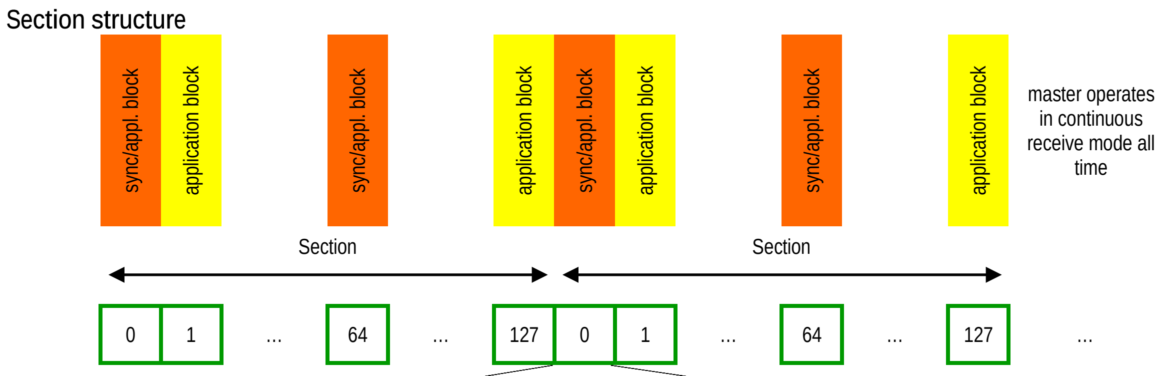
The BiBat Master shall work in continuous receive operation.

Runtime Telegrams from the BiBat Master to BiBat Slaves shall use synchronous Telegrams that shall be transmitted in a given time scheme.

To be compatible with the KNX RF Ready the KNX BiBat Master shall support also all asynchronous communication.

Runtime communication from the BiBat Slave to the BiBat Master or to standard KNX RF devices shall use asynchronous Telegrams (as KNX RF Ready).

For Data Link Layer management, BiBat Slaves may transmit Data Link Layer management Telegrams (help call) at any time. These Telegrams can be transmitted at asynchronous times with the standard KNX RF medium access, but these special help call Telegrams shall be marked as “synchronous” so that a BiBat Retransmitter will retransmit them with a fixed delay.



Block structure

slot-/interval typ:	master timeslot	reserved	timeslot for restransmitter 1	timeslot for restransmitter 2	timeslot for restransmitter 3	recommended time area for spontaneous / asynchronous Telegrams of sync slaves	timeslots for slaves listening to master			timeslots for slave, listening to retransmitter 2			recommended time area for spontaneous / asynchronous Telegrams of sync-slaves	random pause
									timeslots for slave, listening to restransmitter 1		timeslots for slave, listening to restransmitter 3			
direction:	down		down	down	down	up	up	up	up	up	up	up		
function:	sync/appl. Telegram	not used	retransmitted master Telegram	retransmitted master Telegram	retransmitted master Telegram	spontaneous / periodic asynchronous Telegrams of sync slaves	response slave Telegram	response slave Telegram	response slave Telegram	response slave Telegram	response slave Telegram	response slave Telegram	spontaneous / periodic asynchronous Telegrams of sync slaves	preventing continuous collisions
block time: (ms)	0 to 62,5	62,5 to 125,0	125,0 to 187,5	187,5 to 250,0	250,0 to 312,5	312,5 to 2 125,0	2 125,0 to 2 187,5	2 187,5 to 2 250,0	2 250,0 to 2 312,5	2 312,5 to 2 375,0	2 375,0 to 2 437,5	2 437,5 to 2 500,0	2 500,0 to 4 000,0	4 000 to 5 000
slot No.	0	1	2	3	4	5 to 33	34	35	36	37	38	39	40 to 63	64 to 80

time tolerance: ± 0,1 ms ± 100 ppm

Figure 20 - Time structure with sections, blocks and time-slots

A time-slot shall be a period of time with a length of 0,0625 s.

In the time system of the BiBat Master a block shall consists of

- one time-slot (1) for the synchronous communication in “down” direction (i.e. BiBat Master to BiBat Slave), followed by
- one reserved time-slot (2), followed by
- 3 Synchronous Retransmitter “down” time-slots (i.e. Synchronous Retransmitter to BiBat Slave, 3 to 5), followed by
- 29 time-slots (6 to 34) as a recommended time-area for spontaneous and periodic asynchronous Telegrams of any synchronous BiBat Slaves, followed by
- 2 time-slots (35 and 36) for BiBat Slave responses to a synchronous request in slot 1 of the same block for BiBat Slaves synchronised directly to the BiBat Master, followed by
- 4 time-slots (37 to 40) for BiBat Slave responses to a synchronous request in slot 1 of the same block for BiBat Slaves synchronised to a Synchronous Retransmitter, followed by
- 24 time-slots (41 to 64) as a recommended time-area for spontaneous and periodic asynchronous Telegrams of any synchronous BiBat Slaves, followed by
- a pseudo-random pause.

For pre-programmed responses to BiBat Master Telegrams the BiBat Slave shall respond in its time-slot 35 (in the time system of the BiBat Slave). Depending on whether the BiBat Slave is synchronised directly to the BiBat Master or to a Retransmitter, this may be time-slot 35, 37, 38 or 39 respectively in the time system of the BiBat Master. This enables an optional Retransmitter to safely observe the CEPT-admission limit of a dead time of > 1,8 s between pre-programmed transmissions. Standard KNX RF-medium access techniques shall be used for any up Telegrams.

Since a BiBat Slave may be synchronized to one of the BiBat Retransmitters (instead of directly to the BiBat Master) it uses other absolute time windows.

Up communication of BiBat Slaves that is triggered by a user interaction or by a rare technical event (alarm) may be transmitted at any time. It is however recommended that BiBat Slaves should try to avoid transmission in the time-slots 1 to 5 to minimize collisions with the down communication.

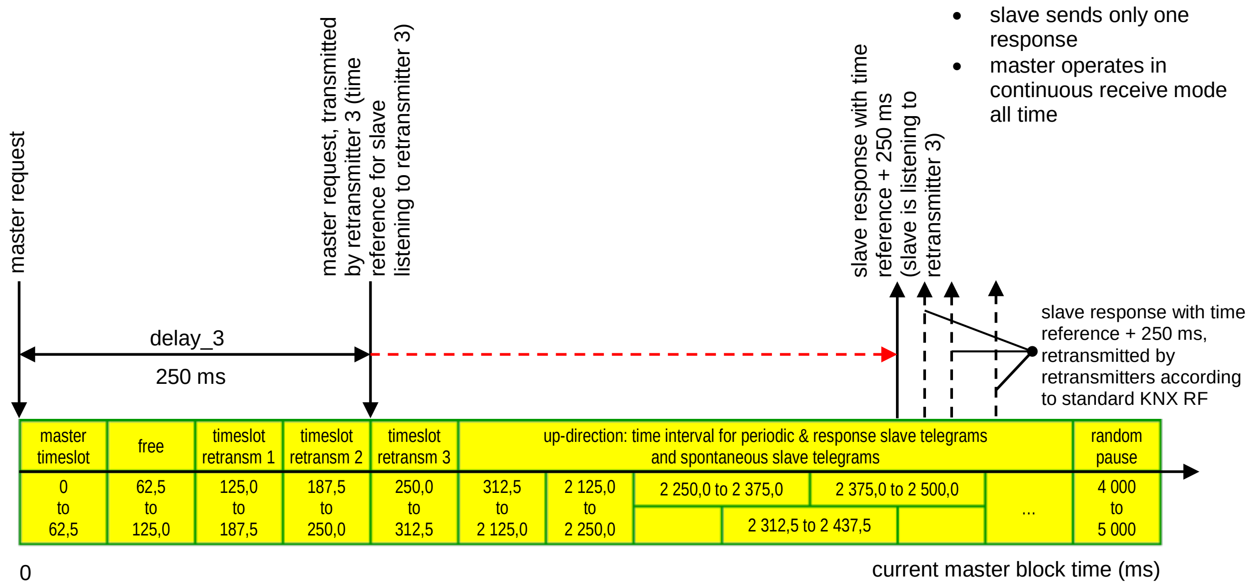


Figure 21 - Example of BiBat Slave response and time delays with Retransmitter No. 3

The time interval between 4 s and 5 s shall be a pseudo random pause. For the calculation of the length of this pseudo random time see clause 6.3.9. Every block shall have a total length between 4 s and 5 s.

All synchronous Telegrams shall be sent to the BiBat Master or to the BiBat Slaves with a clock accuracy better than ±100 ppm plus an additional system timing jitter of < ±100 µs.

The Retransmitter shall delay synchronous Telegrams with a time accuracy of < ±100 µs.

All synchronous Telegrams shall be sent with a preamble of 32 chips.

6.3.3 Telegrams length

To guarantee the 1 % duty cycle limit for all automatic or pre-programmed communication for the BiBat Master and the optional BiBat Retransmitter(s) the BiBat Master has to limit its transmit air time (i.e. including all header, sync and trailer chips) to 1 % minus the length of all received Telegrams, as these may have passed a Retransmitter that cannot limit its own duty cycle.

A time-slot shall have the length of 62,5 ms. Such a time-slot may contain up to three concatenated Data Link Layer Telegrams as long as all Telegrams after the first Telegrams have the same receivers as the first Telegrams. The maximum continuous air time of the three concatenated Telegrams shall be limited to 61 ms in order to cope for tolerances.

Subsequent concatenated Telegrams start directly after the preceding Telegrams (including the postamble – see note) with the preamble of the following Telegrams. This shall allow distribution of data of multiple Datapoints to one BiBat Slave or a group of BiBat Slaves within a single block, thus improving the throughput of the system.

6.3.4 Format of the postamble

In asynchronous KNX RF Telegrams the postamble shall have a length of 2 chips to 8 chips and the chip sequence is not specified (see 5.1.2.2). In case there are subsequent synchronous Telegrams in the same block, the BiBat Master shall generate a postamble with a length of 6 chips with the chip sequence "111000" (Manchester violation). The Manchester violation in the postamble simplifies the detection of the Telegrams end and separation of subsequent Telegrams in the receiver of the BiBat Slave.

The postamble of the last Telegrams in the block shall be the same as for asynchronous Telegrams.

1 block < 61 ms								
1st Telegrams			2nd Telegrams			3rd Telegrams		
Header	Block 1 to n	Postamble	Header	Block 1 to n	Postamble	Header	Block 1 to n	Postamble

…	CRC	Postamble	Header			Block1	…
		Manchester violation ”111000”	Preamble	Manchester violation ”000111”	Sync word "011010010110"		

Figure 22 - Format of the postamble

NOTE 10 Currently the data length in RF Telegrams is restricted to 14 octets for standard group messages and 10 octets for properties. With this restriction the maximum air time of one Telegrams is therefore currently 20,264 ms.
Up to 3 Telegrams in one block are allowed. This leads to max. 61 ms continuous air time in one block

6.3.5 Medium access

Synchronous Telegrams shall be sent at a specific time determined by the sending device. When a device sends a synchronous Frame it shall not check whether the medium is free before sending.

6.3.6 Frame formats

6.3.6.1 Usage of the KNX Ctrl Field in the BiBat system

KNX Ctrl-field is fully specified in clause 6.1.3.

6.3.6.2 Synchronous Frames

6.3.6.2.1 Protocol

Synchronous Frames shall be marked by bit 6 in KNX Ctrl. If this bit is 0, the Frame shall be a standard asynchronous Frame, if this bit is 1, the Frame shall be a synchronous Frame.

All synchronous Frames shall be delayed by BiBat Retransmitters with a fixed time delay.

BiBat Slaves shall use the synchronous Frame for help call to request the resynchronisation via their BiBat Master.

These Frames shall be sent with hop count 6.

6.3.6.2.2 Synchronous Data Telegrams L_Data_…

Synchronous data Frames shall only be used by the BiBat Master for runtime downstream communication.

<table>
<tr><td colspan="8">KNX Ctrl</td><td colspan="16">Source Address</td><td colspan="16">Destination Address</td><td colspan="8">LPCI</td><td colspan="8"></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="8">SA(hi)</td><td colspan="8">SA(lo)</td><td colspan="8">DA(hi)</td><td colspan="8">DA(lo)</td><td colspan="8"></td><td colspan="8"></td></tr>
<tr><td colspan="8">1 octet</td><td colspan="16">2 octets</td><td colspan="16">2 octets</td><td colspan="8">1 octet</td><td colspan="8">1 octet</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

Figure 23 - Synchronous L_Data_… Frame

6.3.6.3 Data Link Layer management Frames

6.3.6.3.1 Use requirements

Data Link Layer management Frames shall be used to handle the synchronisation entirely within the Data Link Layer. This enables a modular implementation both on the BiBat Master- and on the BiBat Slave side.

Bit 4 and bit 5 in the KNX-Ctrl-field shall be used to encode the following Data Link Layer management Frames.

6.3.6.3.2 Sync Frame

<table>
<tr><td colspan="8">KNX Ctrl</td><td colspan="16">Source Address
(BiBat Master)</td><td colspan="16">Destination Address</td><td colspan="8">LPCI</td><td colspan="8">Random pause pointer</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="8">SA(hi)</td><td colspan="8">SA(lo)</td><td colspan="8">DA(hi)</td><td colspan="8">DA(lo)</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="8"></td></tr>
<tr><td colspan="8">1 octet</td><td colspan="16">2 octets</td><td colspan="16">2 octets</td><td colspan="8">1 octet</td><td colspan="8">1 octet</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="8">00h</td><td colspan="8">00h</td><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td colspan="8"></td></tr>
</table>

Figure 24 - Sync Frame

A Sync Frame shall be sent by the BiBat Master twice in a section, in the first time-slot of block 0 and of block 64 if no application Frames are pending for these blocks. This Frame shall be sent as a (domain) broadcast. It shall contain a cyclic random sequence pointer (values 0 to 12) that shall point to the next entry of the random table that shall be used in determining the next random interval after the sync block. A device that is not in sync but coincidentally receives this Frame can by this be helped to synchronise.

A data Frame may be transmitted by the BiBat Master instead of the Sync Frame.

6.3.6.3.3 Help Call

<table>
<tr><td colspan="8">KNX Ctrl</td><td colspan="16">Source
(BiBat Slave)</td><td colspan="16">Destination
(BiBat Master)</td><td colspan="8">LPCI</td><td colspan="8">Retransmitter_ number</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="8">SA(hi)</td><td colspan="8">SA(lo)</td><td colspan="8">DA(hi)</td><td colspan="8">DA(lo)</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="8">1 octet</td><td colspan="16">2 octets</td><td colspan="16">2 octets</td><td colspan="8">1 octet</td><td colspan="8">1 octet</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="8"></td><td colspan="8"></td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td colspan="8">00h</td></tr>
</table>

Figure 25 - Help Call Frame

A BiBat Slave shall send a Help Call Frame if it has not received a valid Sync Frame or a valid Data Frame within 18 minutes. This Help Call Frame shall be sent in point-to-point communication mode to the BiBat Master.

The original Help Call Frame (transmitted by the BiBat Slave) shall be sent with a zero in the Retransmitter_number field.

In a retransmitted Help Call Frame the Retransmitter_number field shall contain the number of the Retransmitter retransmitting the Help Call Frame. This is required so that the BiBat Master can determine the remaining delay for transmitting the Help Call Response Frame.

6.3.6.3.4 Help Call Response

KNX Ctrl	Source (BiBat Master)		Destination (BiBat Slave)		LPCI
	SA(hi)	SA(lo)	DA(hi)	DA(lo)	
1 octet	2 octets		2 octets		1 octet
7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
0 1 1 1 0 0 0 0					0 x x x x x x 1

ticks_until_start_of_next_block	next_block_nr	random_pause_-pointer
3 octets	1 octet	1 octet
7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0

Figure 26 - Help Call Response Frame

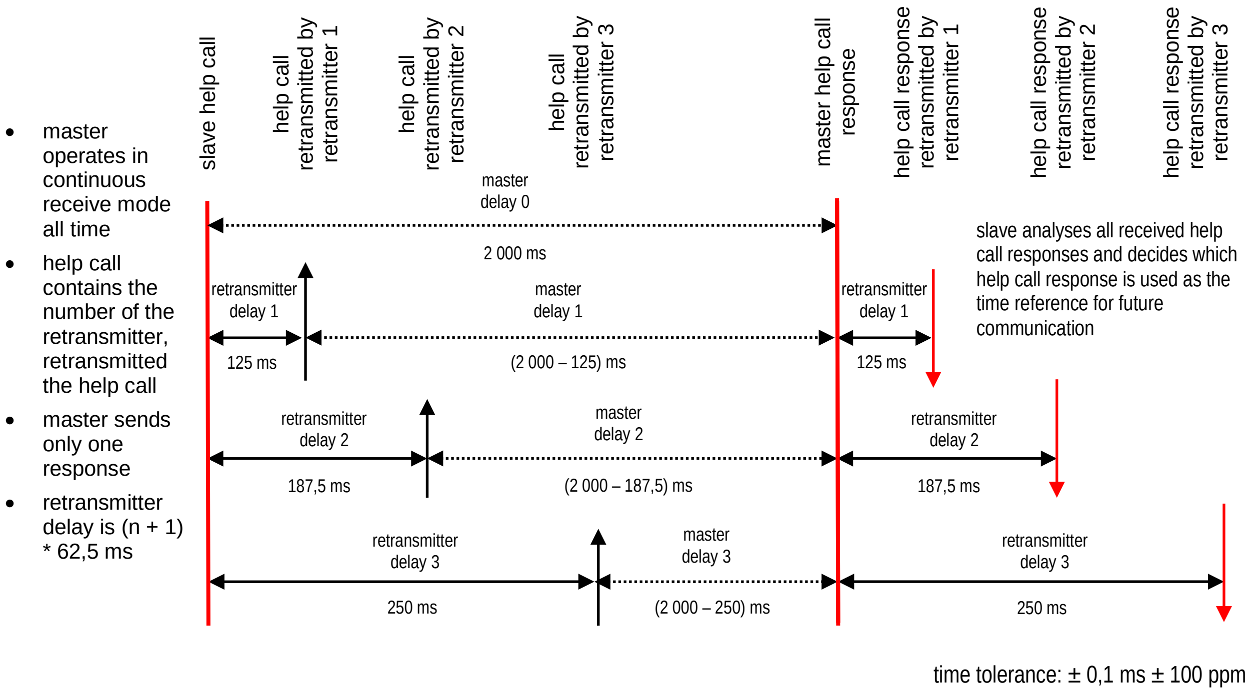


Figure 27 – Sequence of asynchronous Help Call (with Retransmitters)

The BiBat Master shall send a Help Call Response Frame with a delay of 1,75 s to 2 s depending on the number of the Retransmitter through which it received the Help Call Frame. For a direct Help Call Frame this delay shall be 2 s; for a Help Call Frame received via Retransmitter No. n this delay shall be (2 - 0,0625 x (n+1)) s.

This delay shall assure that the BiBat Master's Help Call Response Frame always has a fixed delay with respect to the original Help Call Frame from the BiBat Slave, independent whether the BiBat Master has received the Help Call Frame directly or via one or several Retransmitters.

This Help Call Response Frame shall be sent in point-to-point connectionless communication mode addressed to the BiBat Slave. To allow the BiBat Slave to fully resynchronise to the BiBat Master from this Help Call Response the Help Call Response shall contain:

- the number of 16 384 Hz bit times (24 bit unsigned integer) until the beginning of the next block (max. 5 s) in the field ticks_until_start_of_next_block (maximum value ~81920 ticks; one tick = 1/16384 s), and
- the 8 bit value (0 to 127) of the number of the next block in the field next_block_nr, and
- and the 8 bit value (0 to 12) of the random pause pointer. The random_pause_pointer shall point to the table entry containing the random pause length of the next block.

Behaviour of the BiBat Master

The Help Call Response Frame shall have a fixed delay of 2 000 ms relative to the BiBat Slave's Help Call Frame (see above). Therefore the Help Call Response Frame shall be sent in the currently active block or in the next block, depending on the relative reception time of the BiBat Slave's Help Call within the current block and the current block length, as indicated in the examples below.

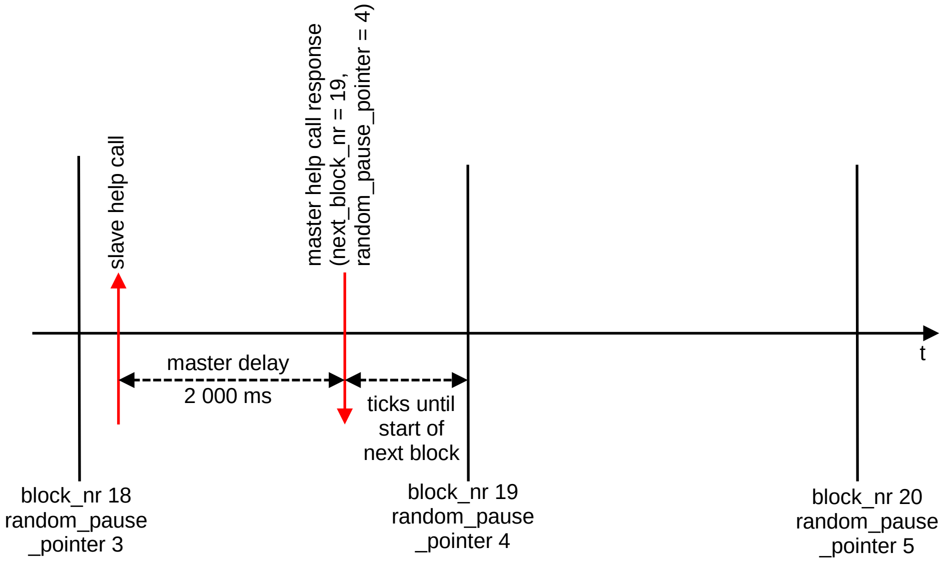


Figure 28 – Help Call Response Frame sent in the same block

The Help Call Response Frame shall not be sent within the last slot of the block in order to have sufficient process time in the BiBat Slave to calculate the beginning of the next block and to activate the receiver in due time. If the delay between Help Call Response Frame and the beginning of the next block is less than one slot length (62,5 ms), the following block shall be indicated as reference (see next Figure 29). Therefore ticks_until_start_of_next_block shall always be larger than 1 024.

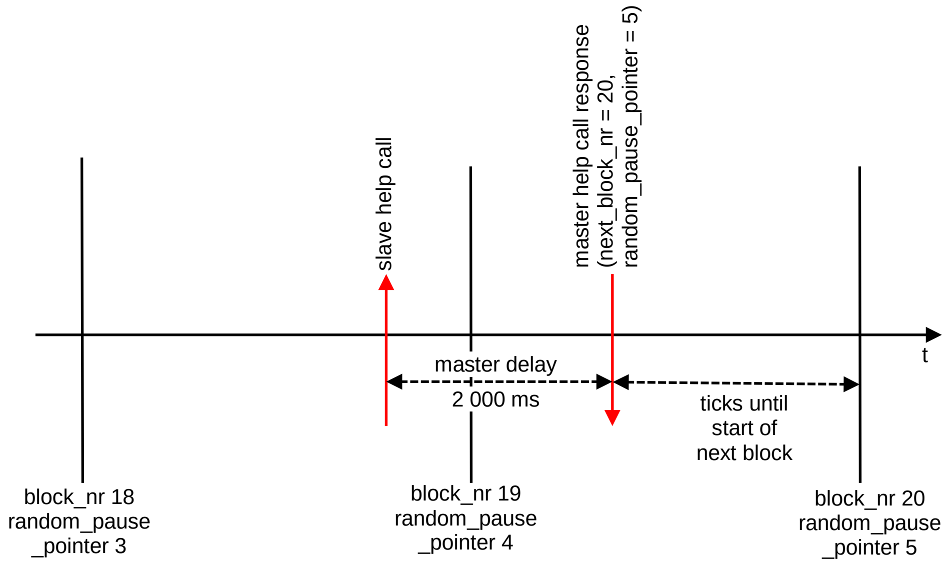


Figure 29 – Help Call Response is sent in the next block

In the block following the Help Call Response Frame the BiBat Master shall continue synchronous downstream communication according to the time-slot scheme.

- If data is pending in the next block, the BiBat Master shall send the corresponding synchronous L_Data Frame.
- In Blocks No. 0 and No. 64 the BiBat Master shall send a Sync Frame if the application is idle.
- No synchronous Frame shall be sent in all other cases. This is, there shall be no synchronous Frame in the next block and the BiBat Slave can only synchronize by way of calculation.

Behaviour of the Slave waiting for the Help Call Response Frame

After reception of the Help Call Response Frame, the BiBat Slave can synchronize its time-slot scheme only by way of calculation.

- In the next block there may be no downstream Frame (because no data is pending in the BiBat Master)
- or the downstream Frame is addressed to other BiBat Slaves.

Therefore it is not appropriate to activate the receiver in the next block in any case.

The BiBat Slave will activate its receiver only in the next active block according to the receive block table. Since any synchronous BiBat Slave shall be able to receive at least the Sync Frames (or Data Frames instead) in block No. 0 and block No. 64 of each section, the next synchronous Frame will be received after a minimum delay of approx. 2 s to 3 s (best case) and a maximum delay of about 5 minutes (worst case, which is sufficient to keep the BiBat Slave clock synchronized with an acceptable time inaccuracy and hence an acceptable width of its receive window).

The receive time point (start) of the Telegrams shall be measured at the transition from the Manchester violation to the Sync-Word. The beginning of the time-slot has a time offset of 38 chips (approx. 1,1597 ms) with respect to this reception time point.

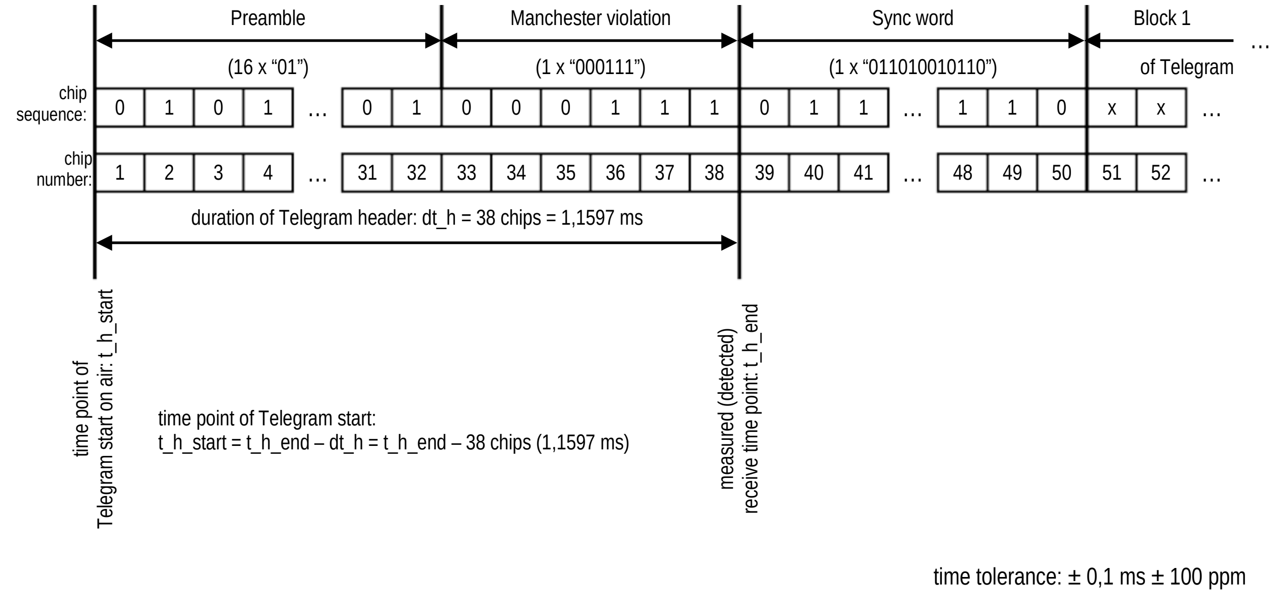


Figure 30 – Calculating the Telegrams receive- and starting time point

The start of these Telegrams shall then be calculated as the detected transition point minus 38 chip times (considering the length of the usually only partially received 32 chip preamble plus the 6 chips of the Manchester-violation).

The reason for this definition is that the receiver normally cannot detect the beginning of the preamble, since an undefined section of the preamble is used to adjust the data slicer. Therefore, for reception timing the end of the Manchester-violation-chip, i.e. the beginning of the Sync-Word shall be used. For the transmit timing of Telegrams however the start of the header shall be used.

6.3.7 Synchronous Retransmitter

6.3.7.1 Requirements

A Retransmitter supporting synchronous communication shall retransmit synchronous Telegrams

- of the BiBat Master that has configured it, and
- of the BiBat Slaves (help calls) that have been assigned to it during configuration.

All synchronous Telegrams shall be retransmitted after a fixed delay time. This delay time of each Retransmitter shall depend on its Retransmitter Number and shall be a multiple of the slot length; this is 0,0625 s ± 0,1 ms.

The nominal delay time of Retransmitter 1, 2 and 3 shall be calculated as:

delay R No. 1 = 2 x 0,0625 s ± 0,1 ms

delay R No. 2 = 3 x 0,0625 s ± 0,1 ms

delay R No. 3 = 4 x 0,0625 s ± 0,1 ms

The Retransmitter Number shall be set by the BiBat Master during configuration time.

A Retransmitter shall fill in its Retransmitter number in the Retransmitter_number field when retransmitting a Help Call.

A synchronous Retransmitter shall filter

- synchronous messages sent with Domain Address – i.e. filtering by Domain Address - and
- synchronous messages sent with KNX Serial Number – i.e. filtering by the KNX Serial Number of the BiBat Master.

Optionally the Retransmitter may also filter other Telegrams according to the Domain Address or a KNX Serial Number table.

At configuration time a synchronous BiBat Slave may receive Telegrams directly from the BiBat Master (and with some delay) from one or several Retransmitters. After configuration the BiBat Slave shall only be able to open its receive window for one of these multiple possible Telegrams receptions. It is recommended that the BiBat Slave analyses all help call responses and decides itself to lock its receive window to one of them. The selection may be either simply the first or better the strongest response.

Additionally to this any synchronous Retransmitter shall retransmit asynchronous Telegrams.

6.3.7.2 Retransmitter configuration

During configuration of the Retransmitter it receives from the BiBat Master

- an Individual Address,
- the RF Domain Address,
- the Retransmitter Number, and
- the BiBat Master KNX Serial Number.

This information shall be exchanged in asynchronous communication.

6.3.8 Receive-blocks

6.3.8.1 Definition

Depending on the application a synchronous BiBat Slave may be interested to receive as fast as possible i.e. to have as many receive-blocks as possible.

On the other hand the number of receive-blocks is limited by the battery lifetime.

Additionally different synchronous BiBat Slaves are interested to have different receive-windows as far as possible, because in a receive-window where a message is sent, all synchronous BiBat Slaves have to listen until they recognise that the message is addressed to other BiBat Slaves. This however stresses the battery-lifetime.

To have the highest flexibility for this, the receive-blocks are assigned to the BiBat Slaves by the BiBat Master at configuration time via a bit-map.

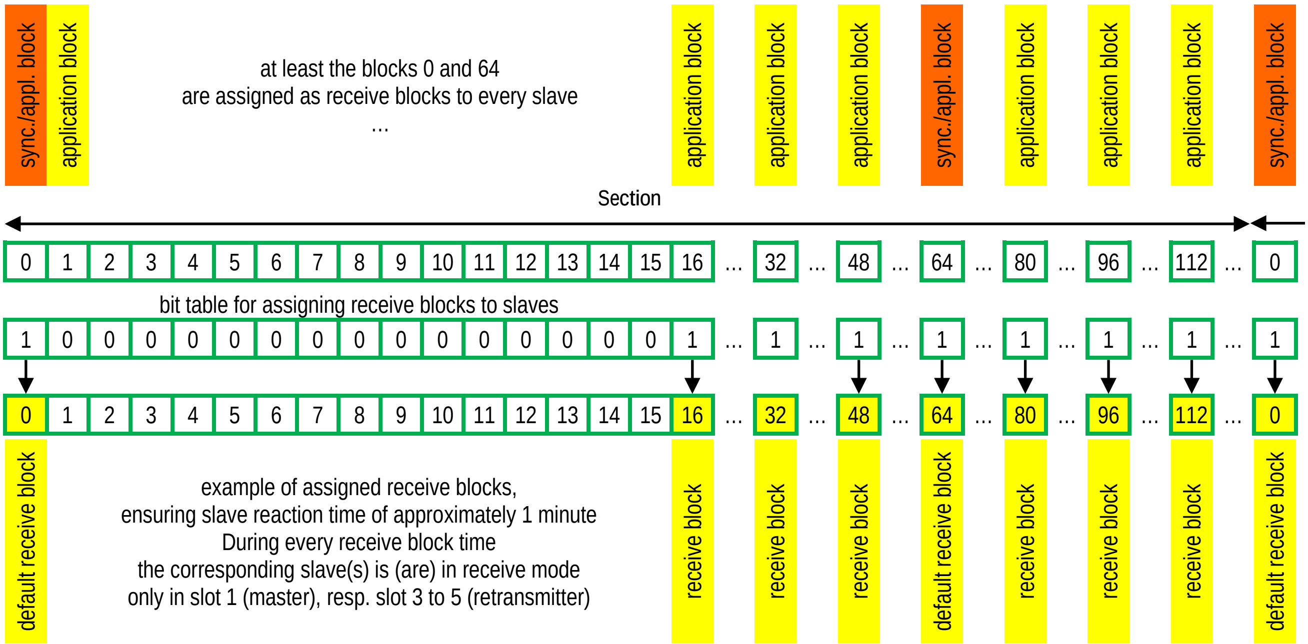


Figure 31 – Assigning receive-blocks to BiBat Slaves

6.3.8.2 Default global receive-block

Any synchronous BiBat Slave shall be able to receive at least the sync Telegrams (or data Telegrams instead) in block No. 0 and block No. 64 of each section.
This corresponds to a reception period of approx. 5 minutes and is sufficient to keep the BiBat Slave clock synchronized with an acceptable time inaccuracy and hence an acceptable width of its receive window. A BiBat Slave shall be able to remain synchronised to the BiBat Master if it receives Telegrams in Telegrams in blocks No. 0 and No. 64 only.

This minimum receive frequency is assumed, if the BiBat Slave does not signal a higher receive frequency capability at configuration time.

NOTE 11 This receive frequency is the allowed minimum due to the resynchronisation requirements.

EXAMPLE 4 If the clock accuracy of the BiBat Slaves is identical to the one of the BiBat Master (i.e. ± 100 mm ± 0,1 ms) the reception time window width shall be ± 2 x (100 ppm of 316 3125 s + 0,2 ms) = 400 ppm + 0,4 ms =117 ms, i.e. in this case the receiver shall be on for a fraction of roughly 1/2500 of the operating time. Thus for a receiver current of e.g. 15 mA, the average current requirement for this minimum would be around 6 µA. To reduce the width of the receive window of the BiBat Slave, the BiBat Slave may be designed for better time accuracy. In addition it may adjust its internal time base to the time base of its BiBat Master during configuration to eliminate the contribution of the difference of the manufacturing tolerance of the BiBat Master and the BiBat Slave quartz to the width of the receive window.

6.3.8.3 Higher receive frequency

A synchronous BiBat Slave may signal a higher receive frequency (= smaller reception period) to the BiBat Master at configuration time.

Therefore the BiBat Master shall read the property PID_RECEIVE_BLOCK_NR of the BiBat Slave. The BiBat Slave shall answer with the number of possible additional receive-blocks (additional to the 2 standard receive-blocks No. 0 and No. 64).

The BiBat Master may then assign receive-blocks to the BiBat Slave via a 128 bit table (property PID_RECEIVE_BLOCK_TABLE).

The BiBat Master may alter this receive-block bit table as long as the average number of receive-blocks signalled by the BiBat Slave is not exceeded.

The BiBat Master may assign each additional receive-block to a single BiBat Slave, a group of BiBat Slaves, to all BiBat Slaves or to no BiBat Slave at all. It can thus efficiently map multicast messages to groups of BiBat Slaves by assigning one or several receive-blocks to a given multicast address.

6.3.9 Pseudo random pause

6.3.9.1 General

To prevent multiple systems (i.e. BiBat Masters) within the RF reception range from systematic continuous collisions, the BiBat Master transmit times (and hence the BiBat Slave receive time-slots) shall not be strictly periodic. This is especially important since such systems are often controlled via high stability quartz crystal oscillators, which may retransmit collisions in periodic systems for hours.

Since the synchronous reception concept does not allow for collision avoidance of the BiBat Master, another way to avoid continuous collision of any two different systems (i.e. different Domain Addresses) is used: a pseudo random pause between blocks. The length of this random pause shall be known to all members of the BiBat System, i.e. the BiBat Master shall inform the BiBat Slaves at configuration-time.

6.3.9.2 Definition of the Pseudo Random Pause

To have the guarantee that two (or multiple) BiBat Systems do not have systematic continuous collisions the worldwide unique KNX Serial Number of the BiBat Master shall be used.

Every BiBat Master shall maintain a unique table of 12 values for the length of 12 subsequent varying random pauses. The random pause No. 13 of 1,0 s (16 time-slots) is fixed.

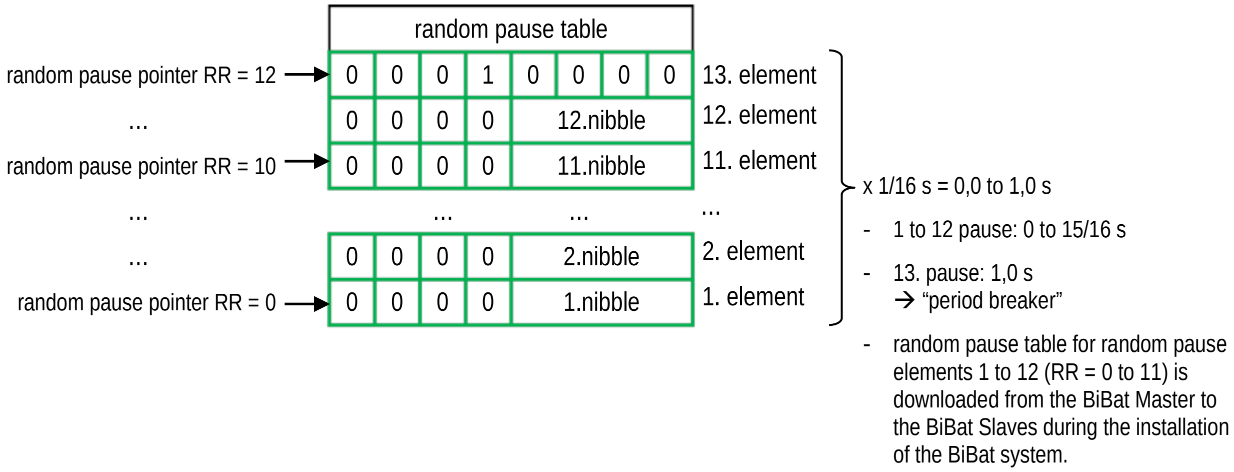


Figure 32 – Generating pseudo random pauses between blocks

The table is derived from the unique 48 bit KNX Serial Number of the BiBat Master in the following way:

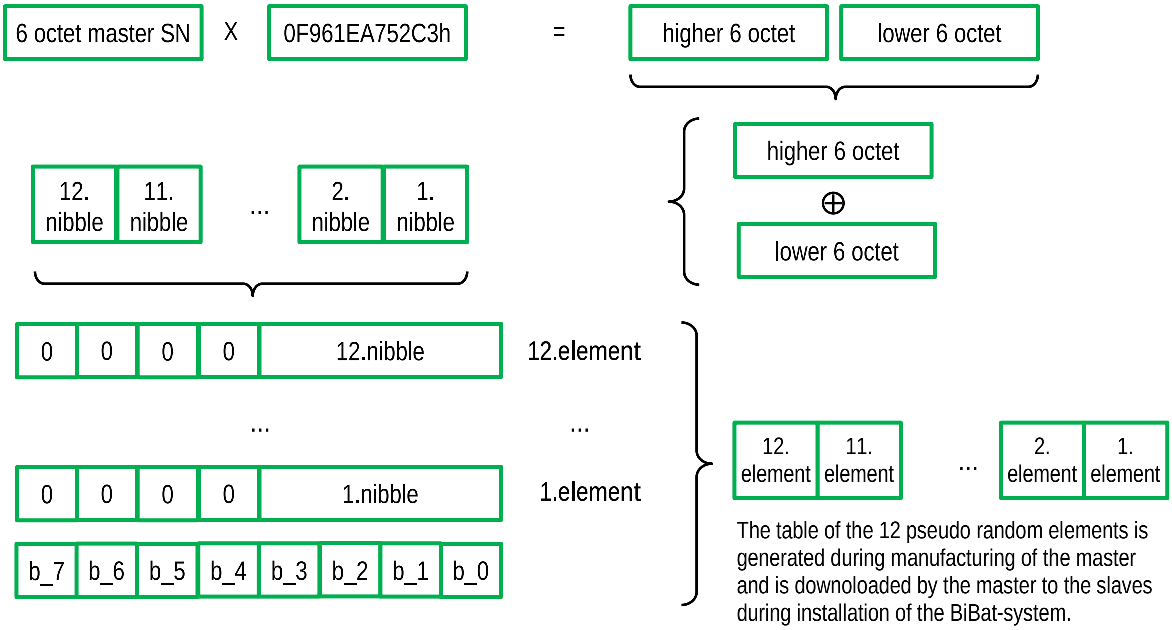


Figure 33 – Calculating pseudo random pause nibbles and octets

For simplicity this unique table can be stored into every BiBat Master during manufacturing together with its unique 48 bit KNX Serial Number.

The BiBat Master shall distribute this random pause table to all its BiBat Slaves during configuration via the property PID_RANDOM_PAUSE_TABLE.

The 13th random pause (RR = 12) shall have a fixed length of 1 s and shall not be downloaded in the random pause table (Period-breaker).

To allow the BiBat Slave to resynchronise fully to the BiBat Master, the ‘Help Call Response’ message shall contain a random pause pointer value (0 to 12).

Random pause pointer value 0 to 11: points to the corresponding element 1 to 12 of the random pause table (n+1).

6.3.9.3 Derivation of the Pseudo Random Pause (informative)

Since the stepping of the length of this pseudo random pause shall be longer than the Telegrams length, the random pause is defined as an integer multiple of the basic time-slot of 62,5 ms. Since the random pause should not be too long compared to the useful block length of 4 s, it is suggested that a 4 bit pseudo random number (0 to 15) is multiplied with the basic time-slot length, yielding a random pause length of 0 s to 0,9375 s.

Using only a 4 bit number as a pseudo-random generator code would allow many different Domain Addresses with identical periods. Therefore the random pause after each block is made different instead of constant. For the total unique address space of a Domain Address of 48 bit, a sequence of 12*bit = 48 bit would provide a unique time interval sequence for each BiBat Master and its BiBat System, thus preventing continuous collision with any other BiBat Master in the universe even for BiBat Master clocks with high stability and accuracy. This random pause length pattern repeats then every 12 blocks (roughly every minute). To prevent two BiBat Masters with Domain Addresses that are only rotated nibbles, a 13th fixed and unique interval of 16 time-slots (i.e. 1,0 s) is used after the 12 varying intervals derived from the Domain Address.

Since the sync period is cyclic over 128 blocks, which is not an integer multiple of this 13 block period of the pseudo random time interval pattern, the actual sync period of any BiBat Master will also not be constant but varying. In general this technique will guarantee that any two BiBat Masters in the universe will not produce a permanent collision sequence for more the 12 attempts, even if their clocks are completely synchronous and for an initial accidental collision phase of their clocks.

Each BiBat Master shall maintain a unique table of 12 4 bit values for the length of the first 12 subsequent random pauses. It will distribute this random table to all its BiBat Slaves during (asynchronous) configuration. For simplicity this unique table is stored into each BiBat Master during manufacturing together with its unique 48 bit KNX Serial Number.

In the manufacturing process this unique random table is generated by the manufacturing equipment with the following algorithm. Using only the 12 x 4 = 48 bit of the unique BiBat Master address directly as the table would not be ideal, since often Domain Addresses are assigned in ascending order. Then frequently a pair of BiBat Master-IDs will only differ in a single (of its total of 12) nibbles. This could produce in this typical case long collision sequences. To avoid this, not the 48 bit Domain Address itself is used as a basis of these 12 pseudo random nibbles, but instead the Domain Address is first multiplied by a suitable 48 bit constant (chosen as 0F961EA752C3h). Of this result the lower 48 bit are XOR-ed with the higher 48 bit and the result is used as the source of the 12 pseudo random nibbles. This multiplication and its factor is chosen so that it will generate even for directly following Domain Addresses (=changes only in one nibble) a nibble list, where most nibbles will be different. The XOR operation guarantees this total change of the nibble list independent on which octet is incremented for subsequent Domain Addresses. To save the BiBat Master from doing this 48 x 48 bit multiplication, the BiBat Master will be assigned the result of this table generation by the manufacturing equipment. During configuration the BiBat Master then transmits this table to its BiBat Slaves.

6.3.10 Management of the synchronous BiBat System

6.3.10.1 Introduction

NOTE 12 The current specification covers only BiBat systems, a specification of the upper layers (network and channel access) of the BiBat 2 system has to be defined.

6.3.10.2 Network configuration

One RF installation may contain more than one synchronous BiBat System

EXAMPLE 5 A BiBat System for the application Heating and a BiBat system for the application Security.

Multiple synchronous BiBat Systems can coexist, but cannot communicate in a synchronous way with each other. Currently any application is limited to one Domain.

Each synchronous BiBat System shall have its own unique RF Domain Address that shall be derived from the KNX Serial Number of the BiBat Master. The BiBat Master shall assign the RF Domain Address to all its BiBat Slaves during configuration.

Asynchronous bidirectional devices may communicate with a BiBat Master in point-to-point connectionless or – connection-oriented communication mode. In this case the asynchronous bidirectional device shall share the RF Domain Address of the corresponding BiBat System. The RF Domain Address of the asynchronous bidirectional device is then typically assigned also by the BiBat Master or alternatively by ETS.

The Individual Addresses of the devices of one Domain shall be assigned completely independent of the other RF Domain(s) in the same RF installation. There is no negotiation of Individual Address ranges between BiBat Masters. The Subnetwork Address (SNA) may be identical in different BiBat Systems or RF Domains. Typically the default SNA for RF medium will be assigned.

If ETS is used to access or configure such an RF installation, then

- it shall be able to handle multiple Domains in one project, or
- the RF installation shall be separated logically into different independent projects.

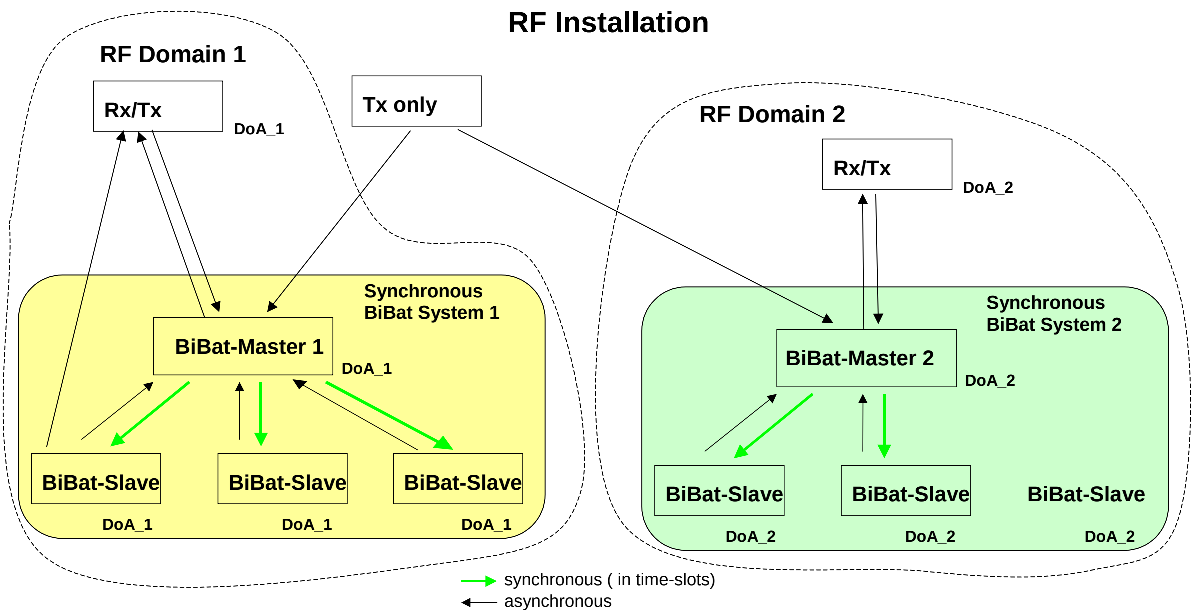


Figure 34 – RF installation with two BiBat Systems and two Domains

6.3.10.3 Configuration of the BiBat Slave

1. Set BiBat Master into configuration mode for synchronous configuration (fallback to normal mode by command or after a default time).
2. Optional configuration of BiBat Retransmitter(s) (see 6.3.10.4).
3. Installer presses teach-in-button on BiBat Slave and device starts configuration mode for asynchronous communication (time-limited permanent reception, 1 minute recommended).
4. BiBat Slave sends (asynchronous) system broadcast with its SN
   (Device Descriptor Type 2 info report).
5. (Optional)
   BiBat Master reads (asynchronous) old DoA + IA.
6. BiBat Master sets (asynchronous) DoA + IA.
7. BiBat Master reads Info from BiBat Slave via Properties (asynchronous)
   (info = number of supported blocks, Property PID_RECEIVE_BLOCK_NR)
   so it is implicitly checked whether BiBat Slave is a synchronous BiBat Slave

   Optional: BiBat Master reads in addition PID_MGT_DESCRIPTOR_01 to check the supported BiBat features: BiBat Slave (mandatory), Fast Ack (optional), Long Header (optional), details see the specification of PID_MGT_DESCRIPTOR in [04].
8. (Optional)
   If BiBat device is no sync BiBat Slave then the BiBat Master sets (asynchronous) old DoA + IA.
9. BiBat Master tells its IA to BiBat Slave (sync BiBat Master address)
   (property PID_BIBAT MASTER_ADDRESS)
10. BiBat Master assigns individual receive-block-table to BiBat Slave
    (property PID_RECEIVE_BLOCK_TABLE)
11. BiBat Master assigns the global random-pause-table to the BiBat Slave
    (property PID_RANDOM_PAUSE_TABLE)

12. BiBat Master prepares its link layer to acknowledge asynchronous sensor BiBat Slave Telegrams (optional)
13. BiBat Master sends connectionless A_Restart to BiBat Slave
14. BiBat Slave sends Help_Call
15. BiBat Master answers with Help_Call_Response
16. BiBat Slave enters time-slotted receive mode

NOTE 13 The BiBat Master can set the BiBat Properties in any order (steps 9 to step 12).

6.3.10.4 Procedure for configuration of BiBat Retransmitters

The complete configuration shall be done in asynchronous communication.

1. The installer sets the Retransmitter into configuration mode by a manufacturer specific action (time-limited, 1 minute recommended).
2. (Optional) BiBat Master reads old DoA + IA from the Retransmitter.
3. BiBat Master reads DD 0 to identify device as Retransmitter.
   BiBat Master reads in addition PID_MGT_DESCRIPTOR_01 to check the supported BiBat features: BiBat Retransmitter (mandatory), details see [04].
4. BiBat Master sets DoA + IA in the Retransmitter
5. BiBat Master assigns the BiBat Master KNX Serial Number to the Retransmitter (element 1 of property PID_SERIAL_NR_TABLE).
6. BiBat Master assigns the Retransmitter number (property PID_RETRANSMITTER_NUMBER).
7. BiBat Master sends connectionless A_Restart to the Retransmitter.
8. Retransmitter leaves configuration mode and starts to retransmit synchronous Telegrams with the correct DoA or KNX Serial Number.
   After this configuration step asynchronous Telegrams containing KNX Serial Number of the sender are not yet filtered; for details please refer to the specification of PID_SERIAL_NR_TABLE in [04].
9. The installer starts from point 1. to install the next Retransmitter.
10. (Optional) After BiBat Slave installation the BiBat Master downloads the rest of the KNX Serial Number table (property elements 2 to N of PID_SERIAL_NR_TABLE) into the Retransmitters and they start to filter asynchronous Telegrams.

NOTE 14 The BiBat Master can set the BiBat-properties in any order (steps 5 and 6).

6.3.10.5 Help Call

Help Calls shall be issued in two cases by a BiBat Slave: during the Configuration Procedure and when the BiBat Slave cannot receive a valid sync or data Frame for a certain time period.

If the BiBat Master receives a Help Call from a BiBat Slave it shall answer with a Help Call Response with a fixed delay of 2 s ± 100 µs. Each BiBat Retransmitter shall delay a Help Call (like any other synchronous communication) by a fixed delay of (n+1) x 62,5 ms, where n shall be the BiBat Retransmitter number (1 to 3). Since the BiBat Retransmitter shall add its Retransmitter number to all Help Call Telegrams the BiBat Master shall transmit only a single Help Call response Telegrams. If it has received the Help Call Telegrams not directly from a BiBat Slave but only (delayed) via a single or multiple BiBat Retransmitters it shall reduce its 2 s delay by the appropriate BiBat Retransmitter delay time of 2 s – 0,0625 x (n+1) s.

As the BiBat Slave does not know from which BiBat Retransmitter it will receive the Help Call, the BiBat Slave shall expect a possible answer to a Help Call with a delay of 2 s + 0,0625 x (n+1) s . It shall thus activate its receiver in these four fixed time windows. The BiBat Slave shall then decide which help call response Telegrams is uses as the time reference for all future communication. It is recommended to use the strongest signal.

The width of this activation window shall depend on the start-up time of the receiver and the time to safely detect that no Telegrams header is received and is typically in the range of 1 ms to 3 ms.

In case the BiBat Slave does not receive a Help Call Response in one of these receive windows it shall turn the receiver off and shall issue another Help Call after a certain delay. This delay shall grow in length until a maximum of 6 minutes.

6.4 KNX RF Data Link Layer extension for BiBat devices with long header

(For Bidirectional fire-alarming devices)

6.4.1 Task

Alarm BiBat Slaves emit very rarely asynchronous alarm Telegrams. These Telegrams can be received by any standard KNX RF device and also by any synchronous BiBat Master.

To also allow battery operated receivers to directly receive safety alarm messages (no other communication like life check) long headers are used in bidirectional fire alarming devices (can be extended to other devices by the KNX Application Specification Groups).

6.4.2 Realization

Therefore these BiBat Slaves shall use a header of a duration of ≥ 3,5 s. The complete Telegrams length of the alarm Telegrams shall be ≤ 3,55 s.

So the maximum Telegrams length allowed in the used frequency band (3,6 s) can always be observed.

The receiver part of such battery operated bidirectional alarm BiBat Slaves then has to be operated with a period of < 3,4 s to safely detect the header of such a long header alarm Telegrams.

Existing RF devices and RF Retransmitters don’t have problems with long headers. They receive the Telegrams and treat it like short-header-Telegrams.

6.4.3 Retransmission of long-header-Telegrams

Retransmission of long-header-Telegrams shall be done in the smoke detectors directly and not in the Retransmitters (this is an application requirement and is usual in such systems).

NOTE 15 There is no problem if standard Retransmitters receive such a Telegrams and retransmit it with a short header.

For reception and retransmitting a Teach-in and filtering by Group Addresses is necessary.

Retransmission is done on application level in the following way.

- If the Alarm-Input-Object receives an alarm then the Alarm-Output-Object sends a message with a long header for passing the alarm to further smoke detectors.
- Thereby the Group Address of the retransmitting device is used (normal KNX mechanism). Thus there is the possibility for building chains.
- The smoke detecting device (and only this) sends once per minute an alarm message.

To avoid collisions of retransmitted Telegrams the following is done.

- The delay between message and repetition consists of a fixed delay and an additional random delay 2).
- Before starting transmission the sender checks the medium.
- If a smoke detector has retransmitted a received fire-alarm it will not retransmit again within a certain mute time 3).

6.4.4 Possible device types with long header

Such a bidirectional alarm BiBat Slave may additionally use any standard synchronous communication to a BiBat Master.

For reasons of Interworking following device types for fire alarming are possible:

- unidirectional devices with short header (standard KNX RF),
- bidirectional devices with long header, and
- bidirectional devices with long header and additional synchronous communication from a BiBat Master.

6.5 KNX RF Data Link Layer extension for BiBat devices with feedback to own action

6.5.1 Task

To allow the integration of battery-driven applications within KNX RF / BiBat that require a fast feedback the following protocol extension shall be used for BiBat devices.

6.5.2 Background

Sensors, remote controls or other devices with user interaction may require a fast response (typically well below 200 ms), which indicates that the command has been received and will be processed now.

This is a kind of acknowledge that indicates “command accepted” – in the following text this is addressed as ***Fast_Ack*** –, but not yet “command processed” – in the following text this is addressed as ***Status_Response***. It is necessary to have this *Fast_Ack* response from the organizing BiBat Master (in this case the BiBat Master device) within a relative short delay. The reason is that, as far as this response appears within 200 ms or less, the user will typically feel almost no delay, and interprets this response as immediate – so this has to be achieved.

The real status information, which indicates that the command has been *processed* by the device that was intended to finally process a given command [4], is usually *not* available immediately (below 200 ms), due to non-deterministic runtime of either the device itself or the delay of one or more transmission paths to that device and vice versa. So this information can in best case be displayed after the intended command has been processed and the acknowledge of this has been transmitted to the (BiBat) Master. Therefore there is no need to hurry in transmitting this information downstream to the BiBat Slave(s).

---

[2] These delays have to be defined in detail together with the corresponding KNX Application Specification Group and the KNX Working Group Interworking.

[3] This mute time has to be defined in detail together with the corresponding KNX Application Specification Group and the KNX Working Group Interworking.

[4] E.g. a light switch now has switched the light on, or a certain dimming position is reached now.

6.5.3 Realization

The BiBat Slave is – as any other KNX RF device – allowed to transmit a command asynchronously to the BiBat Master and in parallel direct to any other (standard) KNX RF device. This Telegrams may be retransmitted by a BiBat Retransmitter.

The BiBat Master shall additionally be able to send a *Fast_Ack* to the BiBat Slave. This *Fast_Ack* is asynchronously related to the BiBat Masters synchronous timing scheme, may override the priority of a pre-planned synchronous time-slot and therefore “kill” the information pre-planned to be transmitted in that time-slot. The transmission of *Fast_Ack* by the BiBat Master is automatically done whenever either the KNX Serial Number (6 octets) or the Individual Address (2 octets) of the BiBat Slave is stored in a table that is held by the BiBat Master. The Domain Address of either BiBat Master and BiBat Slaves that communicate with each other have to be the same, so this is information is redundant and there is no need to store it within the table. This table is part of the BiBat Masters configuration. To keep it of reasonable size, the number of BiBat Slaves requiring a *Fast_Ack* has to be limited. However a minimum storage for at least 16 of these devices shall be provided (16 x 6 octet) by any BiBat Master.

The configuration of this table is manufacturer–specific, possibly local during teach-in by the user.

To allow a *Fast_Ack* (within less than 200 ms) its transmission typically has to be done by a radio pre-processor within the BiBat Master, which only handles low level communication, as it will be insufficient to have this done by the application processor / layer.

6.5.4 Coding of Fast_ACK

The ACK on RF shall be coded as follows (see also clause 6.3.6.1):

Bits 4 to 7 within KNX-Ctrl:

bit 7: 0 (reserved)

bit 6: 0 (asynchronous)

bit 5,4: 01b LL service = Fast_Ack

This means that the coding shall be 0001b in bit 4 to bit 7. Bit 0 to bit 3 shall be as in the normal Frame. The Fast_Ack Frame shall be sent with hop count 6.

6.5.5 Addressing

The Fast_ACK shall be transmitted in point-to-point connectionless communication mode to the original sender, i.e. the Destination Address of the ACK shall be the Source Address of the data Frame. The Source Address of the ACK shall be the Individual Address of the ACK sender, typically the BiBat Master. The first block of the ACK Frame shall contain the Domain Address. The second block shall contain the KNX-Ctrl octet, the Source- and the Destination Address, and the LPCI.

Block 1:

octet 1	octet 2	octet 3	octet 4								octet 5 to octet 10	octet 11 and octet 12
Length	C	Esc	RF-info								SN/DoA	CRC
			7	6	5	4	3	2	1	0		
0Eh	44h	FFh	0	0	x	x	x	x	x	x		

Block 2:

octet 1								octet 2								octet 3								octet 4								octet 5								octet 6								octet 7 and octet 8
Ctrl								Source Address																Destination Address																LPCI+NPCI								CRC
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
t	t	t	t	EFF	EFF	EFF	EFF																																	AT	RC	RC	RC	LFN	LFN	LFN	AET	
0	0	0	1	x	x	x	x																																	0							1	

Figure 35 – Fast_ACK Telegrams

The *Fast_Ack* shall be transmitted by the BiBat Master within less than 200 ms. The BiBat Slave shall wait for 300 ms (200 ms max. BiBat Master delay + 15 ms. Telegrams length plus 2 x one level of Retransmitters) after its own transmission to receive the BiBat Masters *Fast_Ack*. During this time the BiBat Slave shall keep its receiver turned on. This is no problem concerning battery life time, as this happens only very rarely, e.g. after a user interaction. If the BiBat Slave receives no *Fast_Ack* within 300 ms the BiBat Slave shall assume that the asynchronous transmission to the BiBat Master has failed and it shall again asynchronously retransmit the previous Telegrams to the BiBat Master immediately (within 10 ms). There shall be no more than 3 approaches (this is two repetitions), so the BiBat Slave can optionally indicate the failure to connect to the BiBat Master to the user within one second.

As there is usually no need to transmit *Status_Response* information to the BiBat Slave as soon as possible, this information is transmitted within one of the next pre-planned synchronous time-slots of the BiBat Master to the BiBat Slave(s) due to the timing scheme of the BiBat Master.

Optionally the BiBat Master is allowed to “steal” one of the time-slots formerly intended for other devices if the BiBat Slave that initiated the command requires a quick update of the status. The necessity of a faster status – in the following text addressed as ***Fast_Status*** – information by the BiBat Master is turned on/off within the BiBat Masters configuration.

So the BiBat Master temporarily changes the usage of the synchronous time-slots. It “steels” one time-slot, transmits a *Fast_Status* and afterwards returns to normal operation and normal usage of time-slots. To make sure that the initiating BiBat Slave receives the *Fast_Status* it shall keep its receiver active for 2 s after receiving the *Fast_Ack*. So the BiBat Master shall change its timing scheme in order to do a *Fast_Status* response within less than 2 s after the *Fast_Ack* or never. This is also no problem concerning battery life time, as this happens only very rarely, e.g. after a user interaction and the BiBat Slave does not have to listen to this information.

Of course other BiBat Slaves, like other info display units or other BiBat sensors without feedback will have no knowledge of this additional Telegrams, and consequently won't listen to it. So their status display will first be actualized as intended by the predefined synchronous time scheme. This may result in a discrepancy between BiBat info displays for a short time. Standard KNX (RF) info displays are not affected by this mechanism. The maximum delay for the BiBat info displays is given by the synchronous time scheme of the BiBat Master, typically some 10 s to 60 s [5]. As this feature is optional customers having the necessity of possibly delayed but always synchronous display of status information at several info displays may turn this feature off within the configuration of the BiBat Master; see above.

6.6 KNX RF Data Link Layer specific to KNX RF Multi systems

6.6.1 Medium access RF Multi

KNX RF Multi devices medium access is based on a medium free check before transmission.

Table 15 - Medium access times

Type of Frame	InterFrame time [Tint]	Random time [Trd]	Total medium access time [Tma]
Ready Frame	15 ms	0 ms ≤Trd<15 ms	15 ms ≤ Tma < 30ms
Multi Frame (FAST)	30 ms	0 ms ≤Trd<20 ms	30 ms ≤ Tma <50 ms
Multi Frame (SLOW)	60 ms	0 ms ≤Trd<40ms	60 ms ≤ Tma <100ms

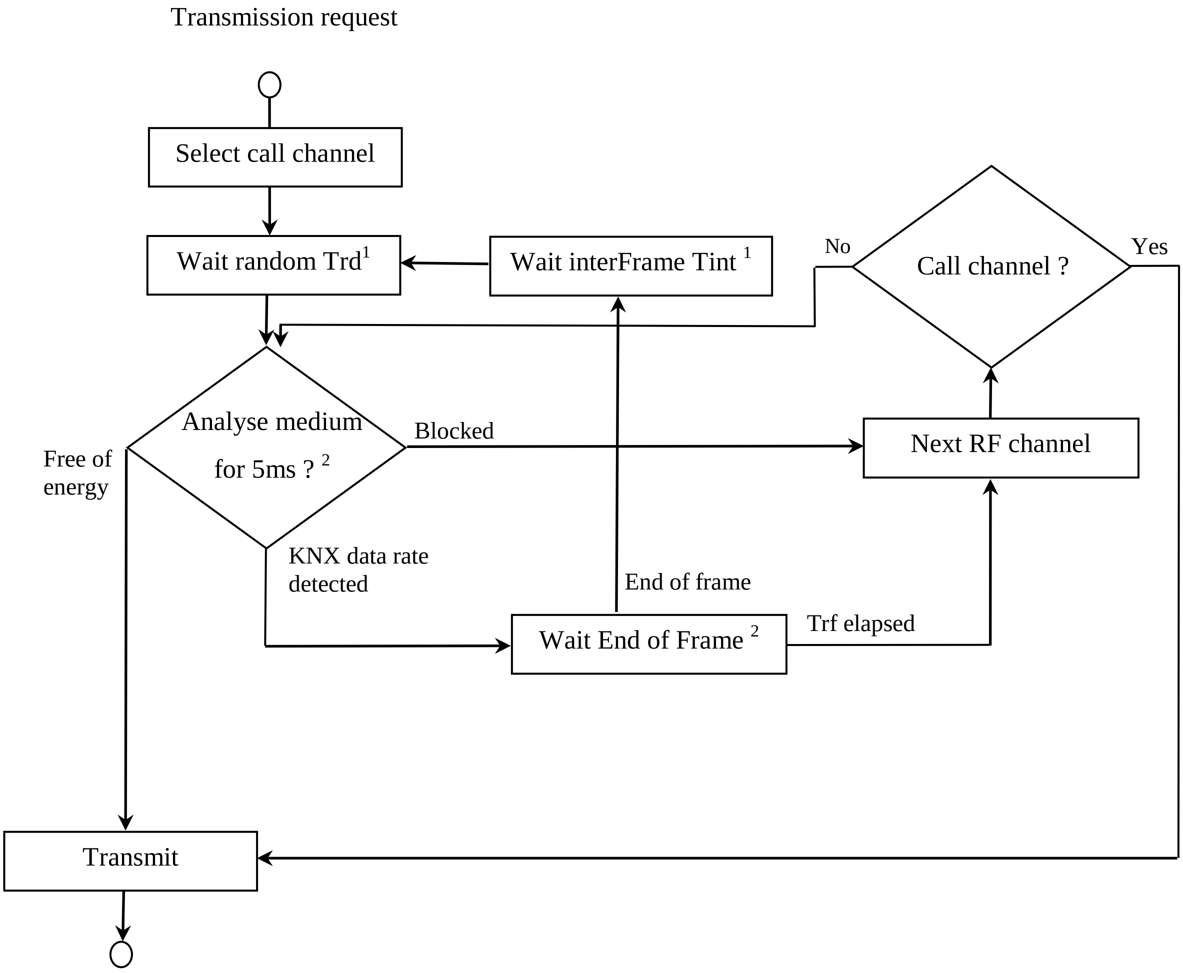
The assumed typical 'blind time' for devices is 1 ms.

The step for the random time shall be 1 ms.

If the transmit request just follows a Frame reception, an additional interFrame time Tint shall apply before any new transmission request described in Figure 36.

The following descriptions shall apply to all transmit requests.

[5] The maximum is 4 minutes.



1: Full reception mode (scanning) shall be enabled in parallel.
2: Reception in the current RF channel shall be enabled in parallel.

Figure 36 - Medium access algorithm for a Fast RF channel

Where Trf = time out value = 500 ms for a fast RF channel.

For Slow RF channels, all "fast" timings shall be multiplied by 2, except Trf = 1,5 s

6.6.2 Frame format

Only the KNX CTRL field is specific to KNX RF Multi Frames. The KNX Ctrl field is fully specified in clause 6.1.3.

During configuration phase, a KNX RF multi device will discriminate a Multi device from a KNX RF Ready. The table of configuration shall be updated accordingly.

6.6.3 RF Multi-channel usage

6.6.3.1 Slow and fast devices

6.6.3.1.1 Introduction (normative)

A fast device shall receive fast Telegrams and may receive slow Telegrams.

A slow device shall receive slow Telegrams and shall not receive fast Telegrams.

All devices shall be able to transmit fast and slow Telegrams.

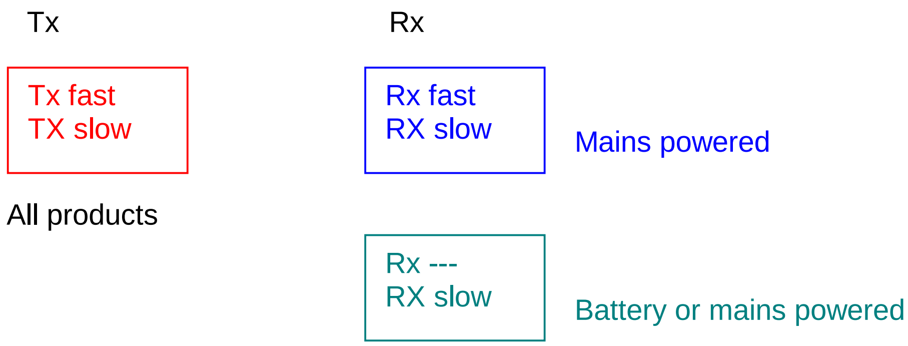
Regardless of unidirectional or bidirectional devices, the communication matrix is the same.

Table 16 - Communications matrix

		Receivers	
		Mains powered (PRM or NPRM)	Battery powered (NPRM)
Transmitters	Mains powered	Fast Frame	Slow Frame
	Battery powered	Fast Frame	Slow Frame

NOTE Other combinations may happen.

The following capabilities scheme is retained whatever may be the KNX RF Multi product:



Some exceptions may occur for the transmitters, in case of solar powered device for example.

6.6.3.1.2 Unidirectional products

A unidirectional product is by nature not informed of the Rx capabilities of the associated actuators. So the unidirectional product has to send the RF Frame in a way that maximizes the chance for the actuator to receive the Frame. The solution chosen is to send at least one fast or one slow Frame according to the configuration capability. If there is no way to indicate to the product to which type of actuator it is linked, the product shall send one Frame in a Fast RF channel and one in a Slow RF channel. This enables the slow or fast associated actuators to receive the Frame according to their reception capabilities. This mechanism avoids Frame congestion in other RF channels and need less energy.

6.6.3.1.3 Bidirectional products

6.6.3.1.3.1 General requirements

The speed shall always be related to the device reception capability. The transmitter shall adapt itself to the receiver capability during configuration phase. In addition, a transmitter associated to several receivers with different capabilities shall adapt itself to the slowest capability. It means that if at least one associated device is slow, then the sensor shall transmit one fast Frame on one free Fx channel (i.e. F1 to F3) and the same Frame (with a longer preamble) on one free Sx channel to all actuators in the group.

6.6.3.1.3.2 Transmission mode

During configuration phase, the preferred Fast RF channel shall be chosen between F1, F2 and F3. This RF channel, named the Call Channel (CC), shall be the same in the whole installation. It will prevent installation troubles due to a permanent interfering transmission on F1, F2 or F3 that makes one of them unusable. If F3 is selected, it shall be checked that all products in the installation are F3 capable.

The same principle shall apply to S1, S2 with the Slow Call Channel (SCC).

The Call Channel shall remain the primary RF channel in which every KNX product shall to transmit firstly.

For all products, and especially in case of PB-Mode, the preferred Call Channel shall be set to F1 for Fast RF channels and S1 for the Slow RF channels.

Other Call Channels can be chosen by appropriate means or using ETS advanced configuration if needed.

The new RF channels are considered as escape RF channels. In case of a busy RF channel the product shall detect if the RF channel is occupied by a KNX Frame or not. If there is a KNX Frame, it is useless to change to another RF channel immediately because the receiver is probably already listening to the Frame. In this specific case, it is better to wait until the RF channel remains free. In case of a non-KNX signal, it shall hop immediately to next RF channel.

After each transmission on an escape RF channel, the next transmission shall firstly be tried on the Call Channel.

6.6.3.1.3.3 Receive mode

6.6.3.1.3.3.1 General requirements

The receive mode and scanning sequences shall depend on the power supplies capabilities.

To keep the compatibility with KNX RF Ready devices, the scanning sequence of RF channel 1 shall be different from a full KNX RF multi system.

In case of presence of at least one KNX RF Ready devices in a GA installation, all KNX RF multi devices shall be in permanent receive mode.

In the case of only KNX RF multi devices, the low energy power mode of power supply may be used. Devices might be in non-permanent received mode.

6.6.3.1.3.3.2 Fast Rx

Usually, products use only fast or only Slow RF channels, depending on the application and power supply. The following describes the scanning algorithms for several products as PRM, NPRM, accessing 3 or more RF channels.

For a PRM fast receiver product that has no time constraint, the scanning algorithm shall be the as specified in Figure 37.

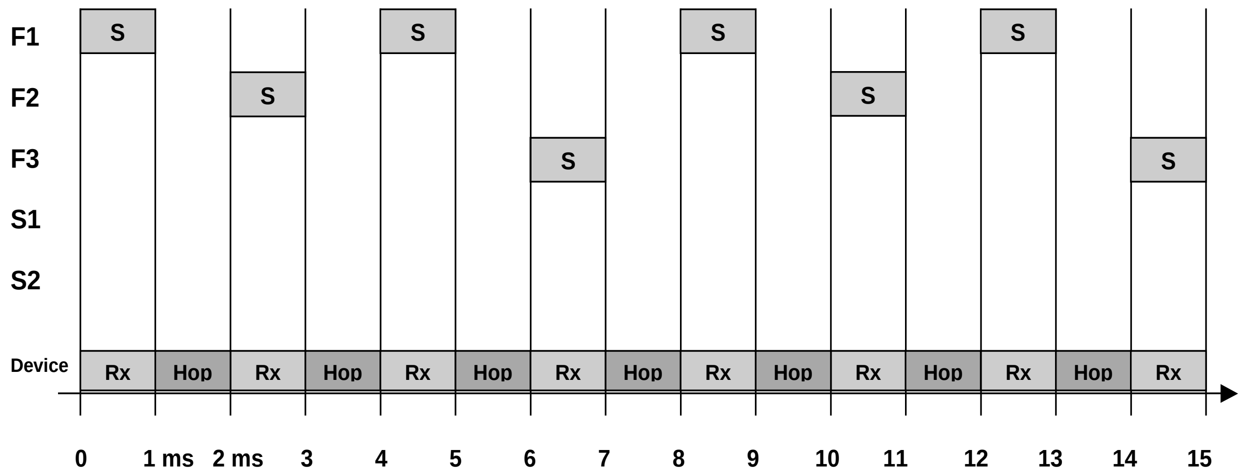


Figure 37 - Scan sequence for PRM fast receiver

S for scan, Hop for Hopping, Rx for receiving mode

1 ms for hopping, 1 ms for listening or scanning or less.

For a PRM fast receiver product that has no time constraint and that have to scan all the RF channels, the scanning algorithm shall be as specified in Figure 38.

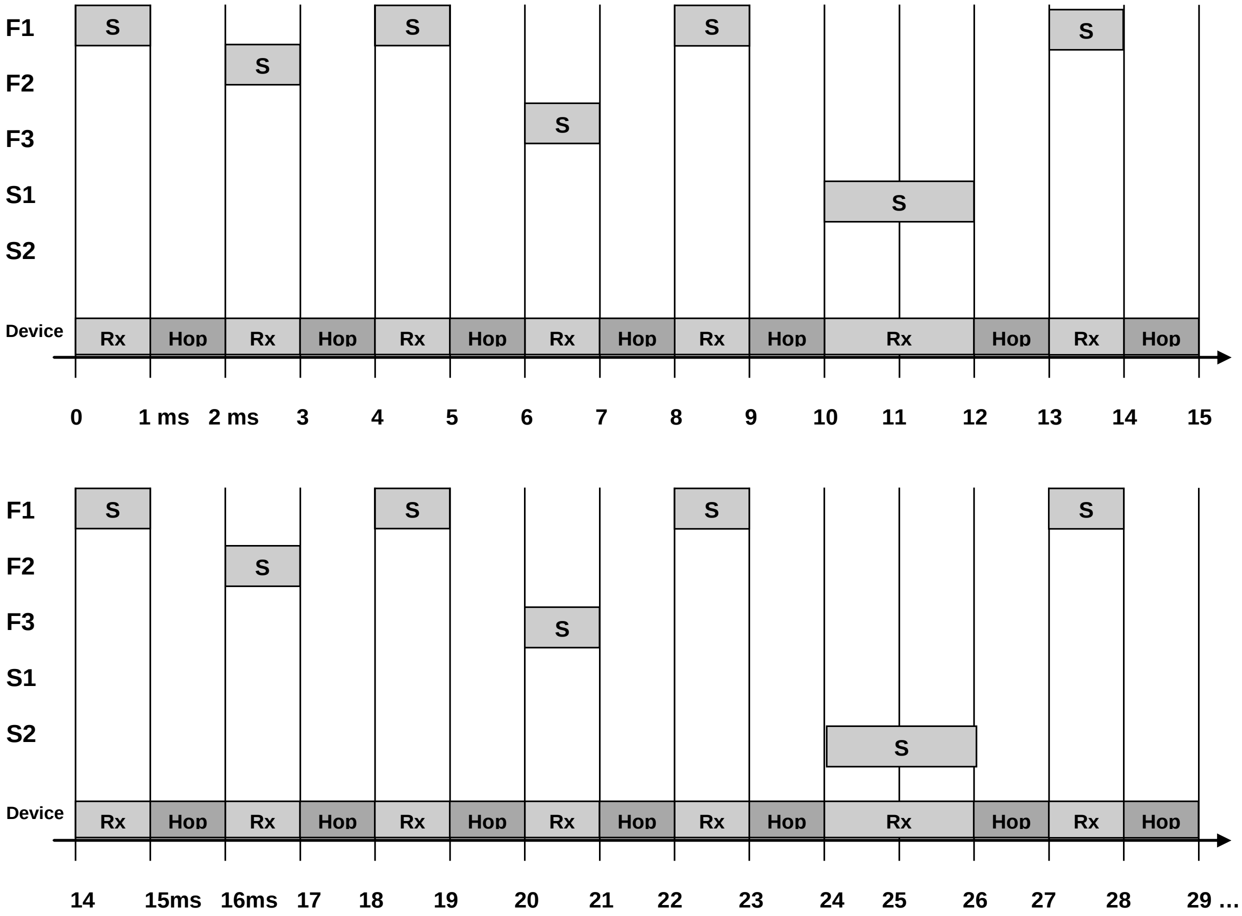


Figure 38 - Scan sequence for PRM fast & slow receiver

S for scan, Hop for Hoping, Rx for receiving mode

A NPRM fast receiver product can be in low power mode most of the time during a 14 ms period. The scanning algorithm shall be as specified in Figure 39.

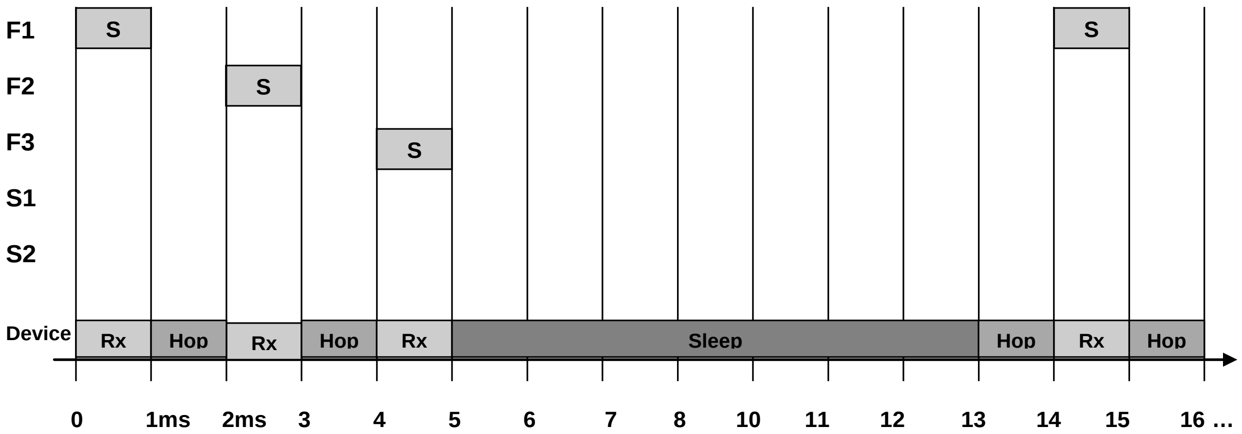


Figure 39 - Scan sequence for NPRM fast receiver

After 14 ms, the scanning sequence restarts.

A NPRM fast receiver product that has to scan all the RF channels can be in low power mode most of the time during a 14 ms period. A scanning algorithm could be as specified in Figure 40.

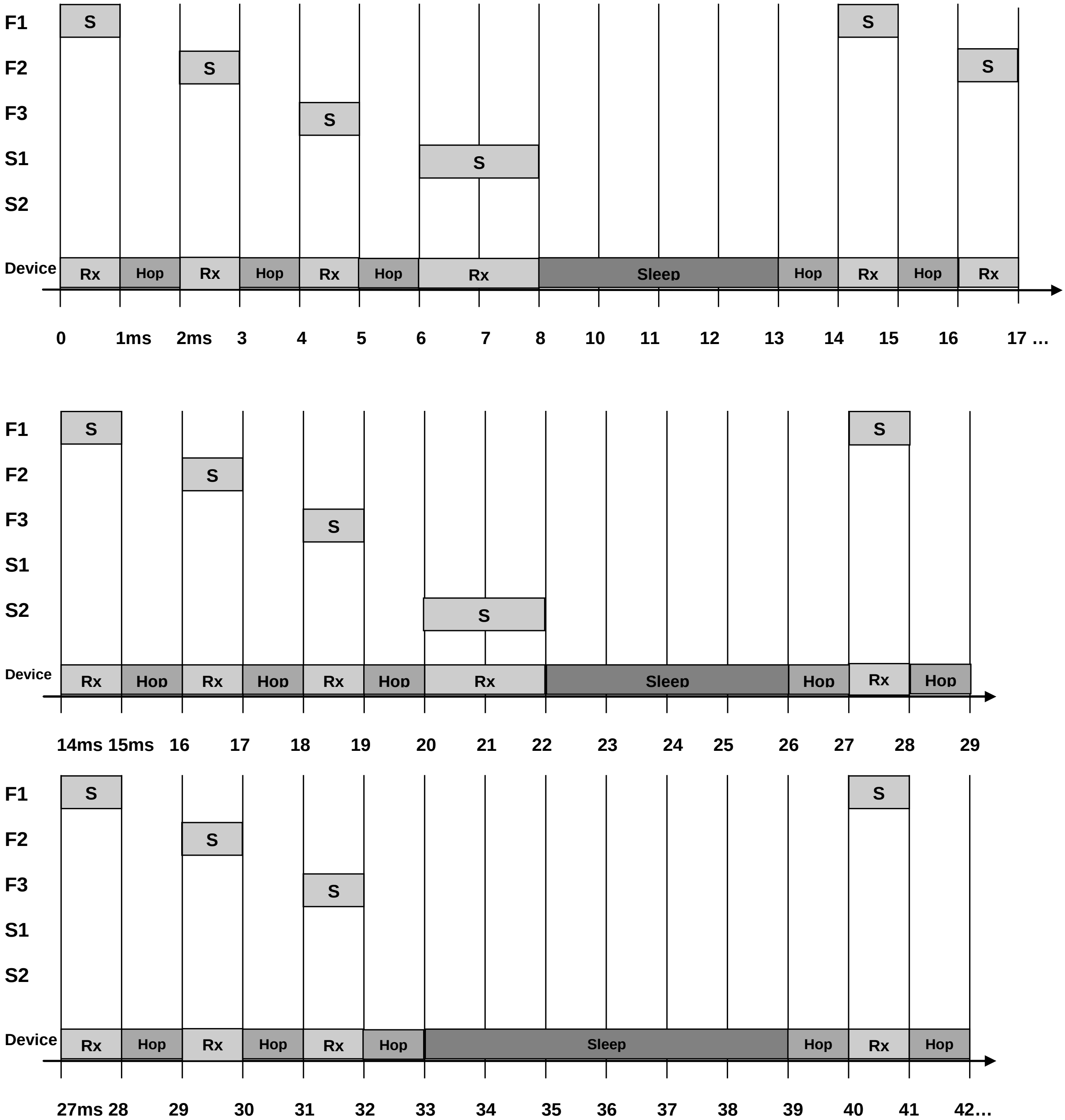


Figure 40 - Scan sequence for NPRM fast & slow receiver

The next sequence of S1 and S2 scan may happen at 490 ms.

Ts = RF channel switching time = about 1 ms

Tw = Time for scanning the wake-up = about 1 ms

The time spent by the fast receiver for the scanning of the 5 RF channels in a 15 ms period (wake-up of the fast RF channels) is:

- Case 1: 4xTs + 5Tw = **9 ms** (when scanning also a slow RF channel)
- Case 2: 3xTs + 3Tw = **6 ms** (when not scanning a slow RF channel)

6.6.3.1.3.3.3 Slow Rx

A NPRM slow device is a product that can only receive 500 ms wake-up Telegrams. This means that it scans only the two Slow RF channels. For these devices, a scanning algorithm can be as specified in Figure 41.

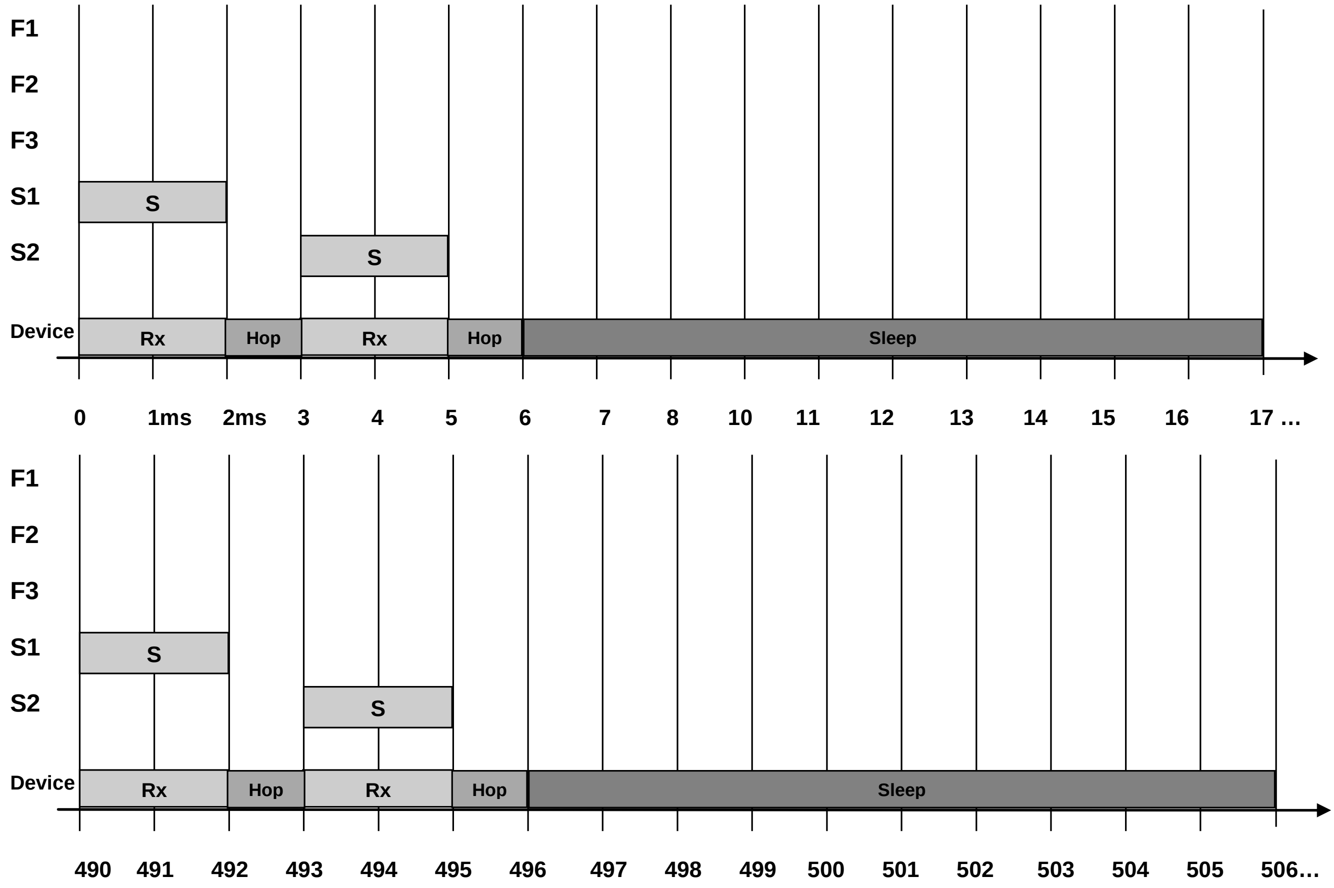


Figure 41 - Scan sequence for NPRM slow receiver

Ts = Switching time = about 1 ms

Tw = Time for scanning the wake-up = about 2 ms

The time spent by the slow receiver for the scanning of the 2 slow RF channels in a 500 ms period (wake-up of the slow RF channels) is:

- 2xTs + 2Tw= **6 ms**

A PRM slow device is meaningless; it will always be a fast PRM that will scan all the 5 RF channels.

6.6.3.2 Telegrams sent between products

During the configuration, the information on the reception capabilities of the devices shall be exchanged. The reception capabilities are a property associated to the whole device.

Internally, the device shall store an attribute associated to each Datapoint as one Datapoint can be linked to a unique fast device, or to a unique slow device or to several mixed devices.

This information shall be used by each device to determine whether it has to transmit a short Frame, a long Frame or both Frames in runtime for a specific Datapoint.

Table 17 specifies the Telegrams that shall be sent by a sensor according to the reception capabilities of the actuators associated to the sensor.

Table 17 – Transmission matrix

Transmitter	Telegrams sent	Receiver
Bidir sensor	One Frame in one **Fast**RF channel	**Only fast devices**
	One Frame in one **Fast**RF channel **+** One Frame in one **Slow**RF channel	**Fast and slow devices**
	One Frame in one **Slow**RF channel	**Only slow devices**

6.6.4 Fast Acknowledgment

6.6.4.1 Basic requirements

The main risk in such an environment is to have 2 transmissions in 2 different RF channels at the same time. As it is not possible to avoid this, the system shall detect those situations and retry to reach the destination therefore an acknowledgement is necessary. A Fast Ack is allowed by EN300220, provided that it is sent by the destination within the first 5 ms after the end of the received Frame.

The Fast Ack is optional; every manufacturer chooses to allow Fast Ack service or not.

A transmitter shall be aware of the Fast Ack capability during the configuration phase. Therefore, in the case of a multicast communication with a Fast Ack requested and both type of devices answering, the transmitter shall only consider expected Ack. Devices without the Ack capability shall be considered by default as having received the transmission.

6.6.4.2 Principle

In order to indicate to the destination that a Fast Ack is expected, the KNX field bit 4 shall be set to 1.

Value of Bit 4 KNX CTRL field	Meaning
1	Fast Ack requested
0	Fast Ack not requested

Moreover, to have enough time to check the content of the received Frame a postamble extension to 9 ms is necessary.

6.6.4.3 Postamble with End of Ack

This postamble shall occupy the medium until the destinations products are able to answer. A synchro word and the number of Ack expected are necessary for installations with Repeaters.

The format of the Postamble shall be as specified in Figure 42.

Start	Synchro Word	Number of Ack expected	Number of Ack expected	CRC	End

Figure 42 - Postamble with End of Ack message content

- **Start**

Description: 123x chip sequence “01”

- **Synchro word**

Description: Manchester violation “000111” chip sequence followed by the sequence “011011011011”

- **Number of Fast Ack expected (1B)**

Description: Number of Ack expected by the sender excluding Repeaters
Encoding:

- **Redundant number of Fast Ack expected (1B)**

Description: Number of Ack expected by the sender excluding Repeaters

- **CRC (1B)**

Description: The last 8 bits from CRC according to IEC870-5-1
$(2^{16}+2^{13}+2^{12}+2^{11}+2^{10}+2^{8}+2^{6}+2^{5}+2^{2}+2^{0})$
CRC is calculated using the 2 previous octets (number of Fast Ack)

- **End**

Description: 4 chip sequence “01”
Encoding:

Postamble with EOA message total length is 20 octets, that is 9,76 ms with 32 786 kchips.

The "number of Fast Ack expected" value shall be within the range from 1 to 64 to avoid increasing the latency for the application.

In case of transmission error in the number of Ack expected, it is advised to keep the smallest value of the two octets.

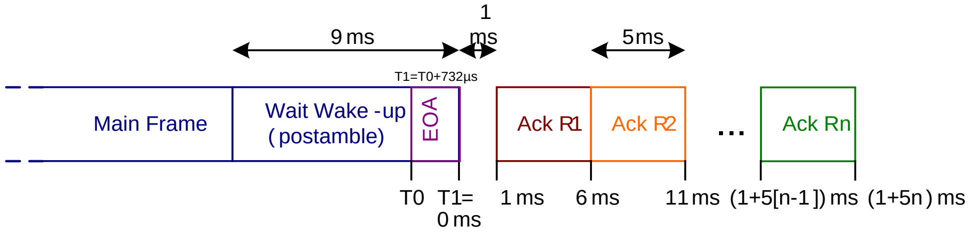


Figure 43 - General Ack insertion

10 ms after the end of the main Frame (i.e. 1 ms after the end of the postamble), the first receiver shall sends its acknowledge message. After that, the second receiver shall send its message and so on until the last concerned receiver. Every concerned receiver shall have its own time-slot to send its message. All receivers shall be synchronized on the same starting point at the end of the postamble synchro word. The time windows shall be 5 ms. The Receiver’s Ack order (for receivers) as well as receivers list (for transmitter) shall be defined by configuration. The Ack transmission shall be sent between 0,1 ms and 0,3 ms after the beginning of the time-slot. This requires a maximum tolerance on the microcontroller oscillator of ±0,05 %.

- For a group communication, the transmitter shall be configured with the list of the destinations devices and their Ack slot number in the configuration; therefore, the device knows how many Acks are expected. The transmitter of the initial Frame shall check the received acknowledges and handle the error if any.

NOTE 16 Be careful on links that may be deleted thus creating "empty holes" in the Ack list.

The same acknowledgment mechanism applies also to Slow RF channels but with a lower data rate. All timings are changed accordingly.

The limit of 64 devices by group for Fast Ack can be overridden by a specific installation.

6.6.4.4 Ack Frame format

The Ack Frame shall always be sent on the same RF channel as the Frame that it acknowledges.

The format of the Ack Frame shall be as specified in Figure 44.

Short Preamble	Synchro	KNX Ctrl	Info	CRC
18 chips	6 chips	1 octet	1 octet	2 octets

Figure 44 - Ack Frame format

- **Short Preamble**

Description: Short preamble for Frame transmitter synchronisation.
Encoding: 18 sequence of “01” chip

- **Synchro**

Description: The use shall be the same as in the main KNX Frame.
Encoding: chip sequence "011010010110"

- **KNX Ctrl**

Description: KNX Control (same use as main KNX Frame)
Encoding: 1 octet

- **Info**

Encoding: 00h (Default value if not used. Otherwise optional values below shall be used)
0 rrr rrrr: Value:

- 000 x rrrr : RSSI value
  Reception level is calculated by [-113 dBm + 3 × RSSI Value (1 to 31)].
  If RSSI value = 31 the reception level is ≥ -20 dBm
- 0 01 r rrrr : Temperature (internal of the device, board temp)
- 0 10 r rrrr : Reserved for future use
- 0 11 r rrrr : Reserved for future use

1rrr rrrr: Technical alarm/ error:

- 1000 0rrr : Environmental issue
  - 1000 0001 : Frozen
  - 1000 0010 : Overheat
  - 1000 0rrr : All values except the 2 above are reserved.
- 1000 1000 : Lamp failure (defective output)
- 1000 1001 : Application busy
- 1rrr rrrr : All values except above are reserved.

FFh value is reserved for Repeaters.
Reserved bits (r) are reserved for the KNX system and shall be set to zero.

- **CRC**

Description: CRC according to IEC870-5-1 ($2^{16}+2^{13}+2^{12}+2^{11}+2^{10}+2^{8}+2^{6}+2^{5}+2^{2}+2^{0}$)

The Fast Ack Frame shall have a total length of7 octets; that is 3,41 ms at 32,786 kchips.

The same acknowledgment mechanism shall apply also to Slow RF channels but with a lower data rate. All timings shall be changed accordingly.

6.6.4.5 Time-slot filling

In case of lack of one or several acknowledges, time-slot(s) left free shall be filled in order not to lose the RF channel. Any free time slot shall be filled by the transmitter. The fill transmission shall be 8 to 10 times a sequence of “0011” chips corresponding roughly to 1 ms.

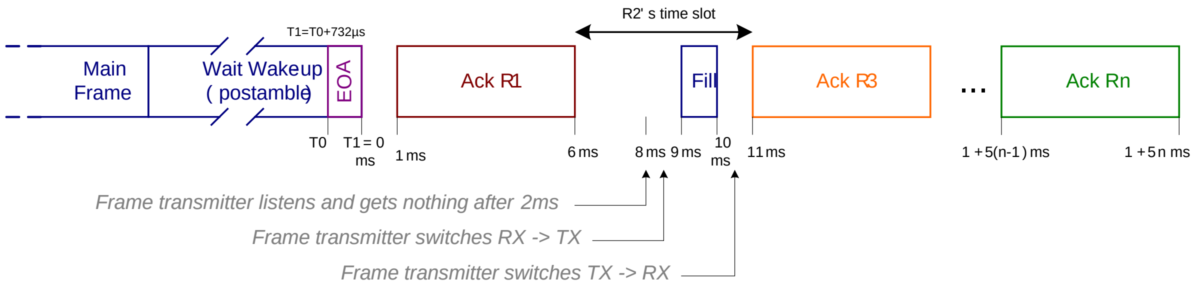


Figure 45 - Time-slot filling (R2 KO)

This way, the maximum gap is 4 ms (two consecutive missing Ack Frames).

The same acknowledgment mechanism shall apply also to Slow RF channels but with a lower data rate. All timings shall be changed accordingly.

6.6.4.6 Runtime flowcharts

6.6.4.6.1 Transmitter side

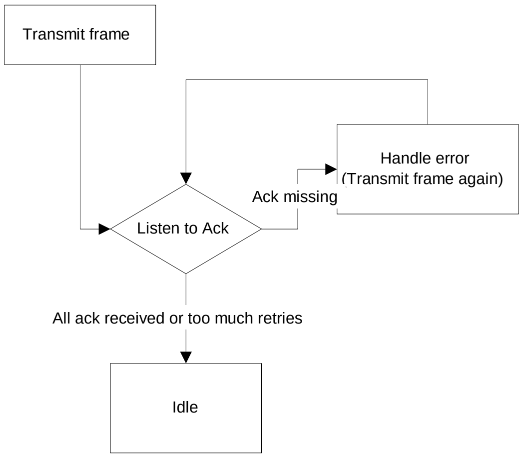


Figure 46 - Acknowledge algorithm, transmitter side

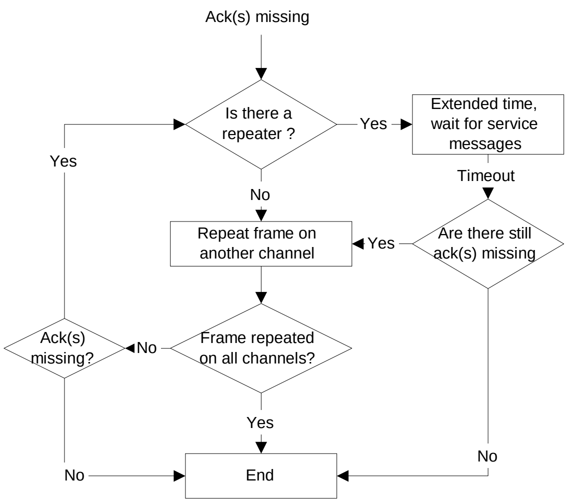


Figure 47 - Acknowledge algorithm, handle error

If a Frame is transmitted for which Fast Acks are expected then the sender shall set a timeout to check if a Repeater is present in the installation or not. To determine if there is a repeater, the device shall switch to reception mode to wait for the Echo frame (i.e. repeated frame). The timeout for the reception of the echo Frame is defined as follows:

> Repeater interFrame (5 ms) + max random Repeater (10 ms) + sufficient time to process the repeated Frame (60 ms) = 75 ms

Fast RF channel

If a Repeater is detected because of the reception of the echo Frame, the device shall set another timeout for the reception of the Ack Rep Frames. This timeout shall be dependant of the length Frame and the number of expected Fast Acks. The timeout shall be the following.

- 1000 ms if less than 32 Acks are expected
- 2000 ms more than 32 Fast Acks are expected

Slow RF channel

If a Repeater is detected by the reception of the echo Frame, the sender shall set another timeout for the reception of the Ack Rep Frames. This timeout shall be dependant of the length Frame and the number of expected Fast Acks. The timeout is the following:

- 4000 ms if less than 32 Fast Acks are expected
- 5000 ms if more than 32 Fast Acks are expected

Of course these timeouts are not used if all the expected Fast Acks have been received.

In the case of group communication, if one or several Ack messages are missing, the device shall send the Frame again but on the next RF channel. The Ack mechanism shall stop after 3 successive immediate retries on the two next RF channels after the Call Channel, and the Call Channel. A typical sequence would be F1, and retries on F2, F3 and F1. The same applies for Slow RF channels with S1 and S2.

6.6.4.6.2 Receiver side

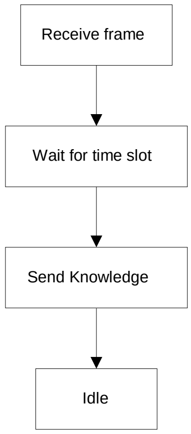


Figure 48 - Acknowledge algorithm, receiver side

6.6.5 Data Link Layer protocol

6.6.5.1 RF Repetition counter for end devices

The RF Repetition counter shall be set to 2 for end devices.

6.6.5.2 The Layer-2 of an RF Retransmitter

6.6.5.2.1 General requirements

Any RF Retransmitter shall be either one of the following two types: a fast Repeater or a slow Repeater. A Repeater shall repeat once any Telegrams received coming from the device it is configured with.

For KNX RF Multi, filtering is the rule.

Fast Repeaters shall be mains powered but slow Repeaters may be battery driven.

Operating on fast KNX RF Multi Telegrams, it shall receive and repeat any fast Frame. It shall use the frequencies F1, F2 and F3. Operating on slow KNX RF Multi Telegrams, it shall receive and repeat any slow Frame. It shall use the frequencies S1 and S2.

The KNX RF Repeater can be generic device. The selection of fast or slow Repeater function is done by an HMI.

A Repeater repeats frames whatever the LFN value is but doesn’t repeat a frame that is already in the history list. There is only one entry per KNX Serial Number; the LFN is overwritten.

On fast RF channels Repeater shall adapt the preamble length according to the product it is linked with. This is, the preamble length shall be 15 ms or 4,8 ms. This implies a scanning sequence according to Figure 36 for a fast Repeater or Figure 37 for fast and slow Repeater or Figure 40 for slow Repeaters.

A fast and slow Repeater may lose or delay the repetition of fast Frames in case on simultaneous slow Frame occurring, therefore this implementation is not encouraged. The interFrame time of 5 ms between a reception and the following transmission apply to Repeaters.

6.6.5.2.2 Working principle

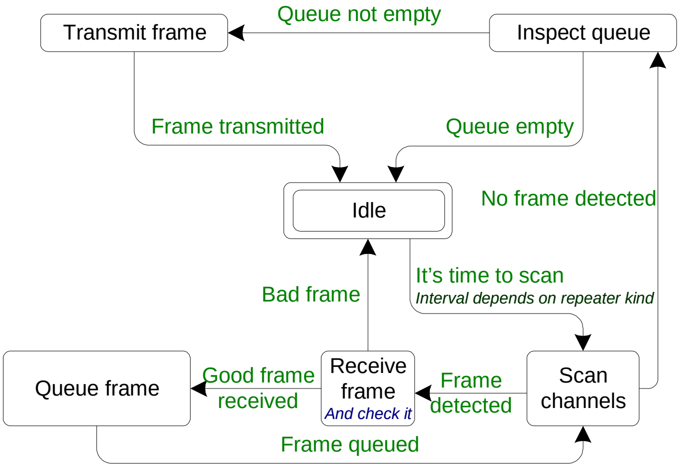


Figure 49 - General Repeater algorithm

By default, the system is scanning the RF channels.

At regular intervals (relative to the kind of Repeater it is), the Repeater shall scan its RF channels to see if a KNX Frame is being transmitted. If not, it shall inspect the Frame queue. In case of a non-empty queue, the first Frame in the queue shall be transmitted. Else it shall get back to its scanning idle state.

If a Frame is detected, the Repeater shall receive the Frame. It shall then check the Frame to determine if the Frame is a correct KNX Frame. It can also check if it has yet sent the same Frame recently to avoid any distortion or echo effect. If the Frame is considered as bad, the Repeater shall get back to its scanning idle state.

If the Frame is correct then the Repeater shall queue that Frame. The Repeater shall then get back to its scanning idle state, waiting for the next Frame.

6.6.5.2.3 Fast Repeaters

Fast Repeaters shall have two working modes, KNX RF Ready (legacy compatible KNX RF 1.1) or KNX RF Multi. For the KNX RF Ready, please refer to the relevant clauses. Nevertheless, in this mode, all the Telegrams to be repeated shall be located on the same frequency. As the preamble received may be only 1 ms-long, the Repeater shall listen to the RF channel continuously and the queued Telegrams can be immediately transmitted.

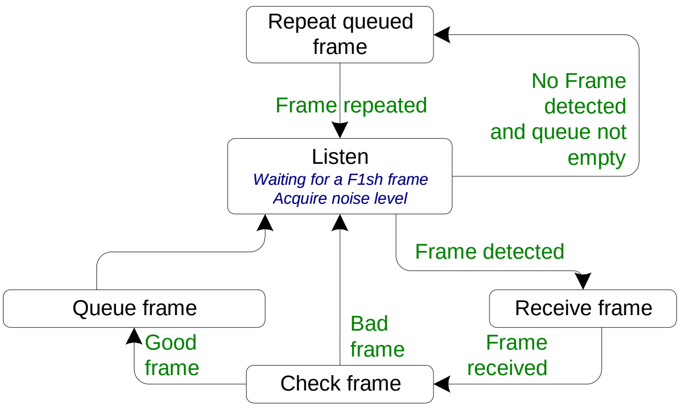


Figure 50 - KNX RF Ready algorithm (KNX RF 1.1 legacy)

In the KNX RF Multi mode, the Telegrams to be repeated can be located on different frequencies (F1, F2, F3). The preamble of each Frame shall be 15 ms long; this allows the Repeater to be in complete low power mode most of the time.

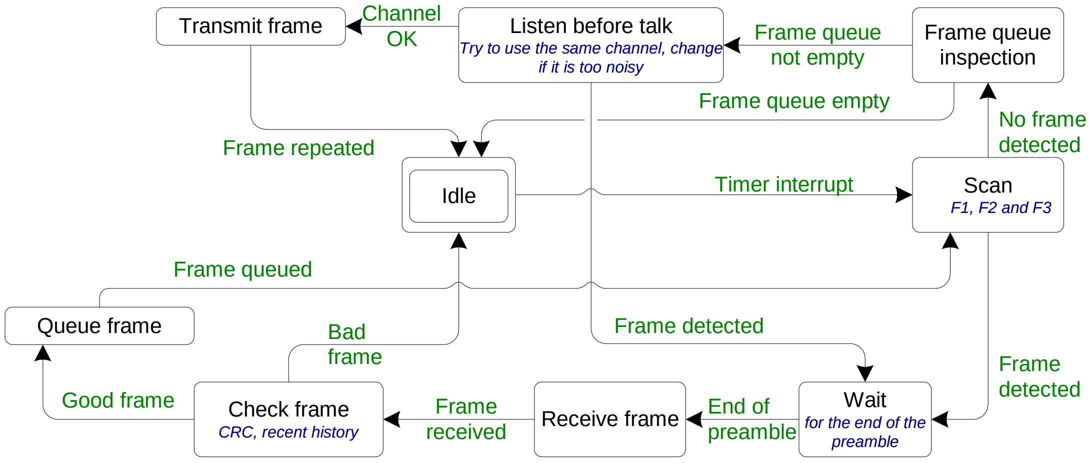


Figure 51 - Fast KNX RF Multi Repeater algorithm

The scan described above is a real scan amongst the RF channels not used (e.g. F2 and F3 if the Frame is received on F1). It is mandatory to do a scan on each RF channel prior transmitting due to the hidden node problem as it may result in KNX Frames at the same moment on different RF channels.

6.6.5.2.4 Slow Repeaters

With this Repeater, the Telegrams to be repeated can be located on different frequencies (S1, S2). The preamble of each Frame shall be 500 ms long; this allows the Repeater to be in complete low power mode most of the time.

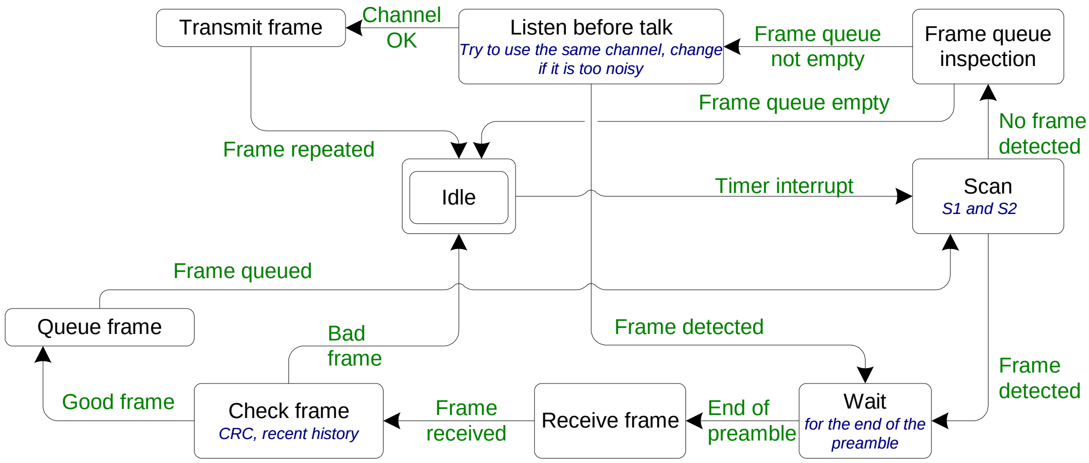


Figure 52 - Slow Repeaters algorithm

The scan described above is a real scan amongst the RF channels not used (e.g. S2 if the Frame is received on S1). It is mandatory to do a scan on each RF channel prior transmitting due to the hidden node problem as it may result in KNX Frames at the same moment on different RF channels.

6.6.6 Runtime with an RF Repeater and Fast Ack requested

A RF Repeater shall be dynamically aware that a second RF Repeater is present thanks to the repeated Frame with the same KNX Serial Number (Echo Frame). To wait for the Echo Frame, the Repeater shall be kept in receive mode at least 75 ms after the End Of Ack. In absence of an RF Repeater, the retry shall be sent as soon as possible on the next free RF channel. Only Frames learnt by the Repeater shall be repeated.

After a retransmission, the RF Repeater shall wait for all the Fast Ack and an Acknowledge Repeater (Ack Rep) Frame shall be sent back to the sender.

The Ack Rep Frame shall be a copy of the initial Frame sent by the sensor except for the following.

C field	00h (Ack Frame defined in IEC 870-5)
Start	See clause 6.1.3.
RC	Repetition counter is set to 2
Data field	Field of n octets (n = number of Fast Ack requested in the EOA) • In case a Fast Ack is received, the octet contains the Info Octet of the Fast Ack. • In case a Fast Ack is NOT received, the octet is filled by FFh value.

The length of the Frame shall be adapted to the number n of Ack requested in the EOA received.

The max number of expected Fast Ack expected shall be limited to 64.

An Ack Rep Frame shall have high priority compared to other application Frames if the Repeater is embedded in a device with another function.

A repeater which receives a Ack Rep frame shall always repeat it provided that the repetition counter is not equal to 0.

6.6.7 InterFrame delays for RF Repeaters

To be able to work efficiently, the repeated and Ack Rep Frames must be sent before standard Frames. Table 18 gives the interFrame and random times used for different frame types.

Table 18 – Transmission matrix

Type of frame	InterFrame time [Tint]	Random time [Trd]	Total medium access time [Tma]
REPEATED Multi fast frame	5 ms	0 ms ≤ Trd <5 ms	5 ms ≤ Tma <10 ms
REPEATED Multi slow frame	10ms	0 ms ≤ Trd <10 ms	10 ms ≤ Tma <20 ms
REPEATED Ready frame	5ms	0 ms ≤ Trd < 10 ms	5 ms ≤ Tma <15 ms
Ack Rep frame	10 ms	0 ms ≤ Trd <5 ms	10 ms ≤ Tma <15ms
Ready / BiBat frame	*15 ms*	*0 ms ≤ Trd <15 ms*	*15 ms ≤ Tma <30 ms*
Multi frame(FAST)	*30 ms*	*0 ms ≤ Trd <20 ms*	*30 ms ≤ Tma <50 ms*
Multi frame(SLOW)	*60 ms*	*0 ms ≤ Trd <40 ms*	*60 ms ≤ Tma <100ms*

6.6.8 Repetition counter

6.6.8.1 General rule

Each KNX RF Frame carries a repetition counter bit field.

For all KNX RF ready and KNX RF Multi repeaters, on each repeated Frame, the repetition counter is decremented. If the repetition counter equals 0, the Frame shall not be repeated. The repetition counter is used to avoid endless retransmission of Frames in multi repeater installations.

RC(repeated frame) = RC(received frame)-1

In KNX RF Ready installations, a maximum of 6 chained repeaters is allowed. Then the maximum repetition counter value is 6 for Ready Frames. In case of KNX RF Multi devices downsized into KNX RF ready mode, the repetition counter has to be set to 6.

In KNX RF Multi installations, a maximum of 2 chained repeaters are allowed. Then the maximum repetition counter value is 2 for Multi Frames.

6.6.8.2 Repetition counter value for Ack Rep frame

In addition to clause 6.6.8.1 rule, each Ack Rep Frame has its own repetition counter value.

When generating an Ack Rep frame, a repeater shall calculate the repetition counter value using the following rule taking into account the repetition counter value of the frame with Fast Ack requested received from the sender or the previous repeater:

RC(generated Ack Rep frame) = 2 – RC(receivedframe request)

6.6.9 Media Coupler

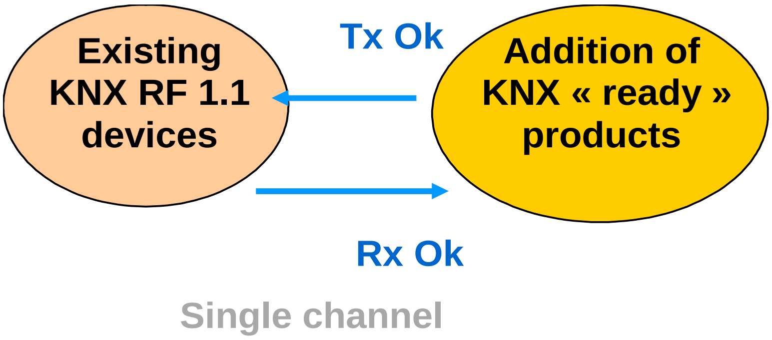
A Media Coupler shall be transparent in configuration and in runtime mode. No specific action is required.

A Media Coupler shall take into account the Fast Ack service in its Radio Frequency Data Link Layer instance if available. However, this service will only be available with a configuration tool.

7 Compatibility between KNX RF 1.1, KNX RF Ready and KNX RF Multi

7.1 Communications between KNX RF 1.1 and KNX RF1 Ready devices

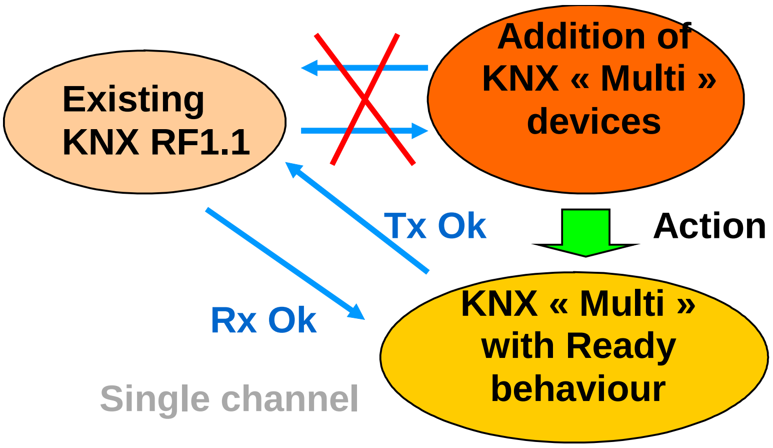
KNX RF 1.1 and KNX RF Ready devices are based on the same hardware. Only the length of preamble for transmissions is higher with Ready devices. Therefore a full compatibility is expected.



7.2 Communication between KNX RF 1.1 and KNX RF1 Multi devices

By using a new length of preamble and RF channel hopping of 1 ms, the backward compatibility is not assured. This is due to the fact that the time needed to switch on an alternate RF channel and to detect that no Frame is currently being received, is about 2 ms long. The current short wake-up is about 1 ms (30 chips), thus going and returning from an alternate RF channel need about 3 ms. During these 3 ms, the product can lose a short wake-up Frame on the primary RF channel.

In an installation with at least one KNX RF 1.1 product, all KNX RF Multi products need to swap to KNX RF Ready behaviour. This swap is done by a physical action on the product. Therefore the devices will only use F1sh and the relevant Data Link Layer. The physical action can be freely imagined by each manufacturer (switch, starting ON/OFF sequence …) If the devices are intended to be configured by a tool or ETS, the swap can be ordered by a software action. For KNX RF multi devices, the low energy mode for the power supply will not be possible anymore.

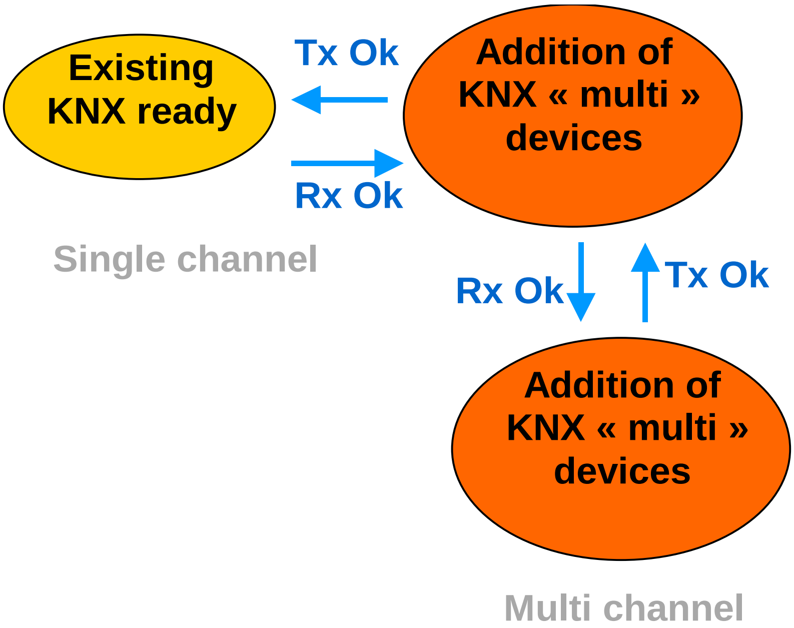


If in an existing installation, there is only one KNX RF 1.1 device and all other are KNX RF Multi, to have the maximum benefit of RF multi-channel feature, it is highly recommended to replace the KNX RF 1.1 device by a multi to avoid this feature reduction.

The possibility to set a KNX RF multi device into the KNX RF Ready is not optional for mains powered devices. In the latter case, each manufacturer has to design a power supply in line with this intended use (PRM). Nevertheless, battery driven RF Multi devices shall be able to transmit and receive in fast mode for the configuration phase.

7.3 Communication between KNX RF Ready and KNX RF Multi devices

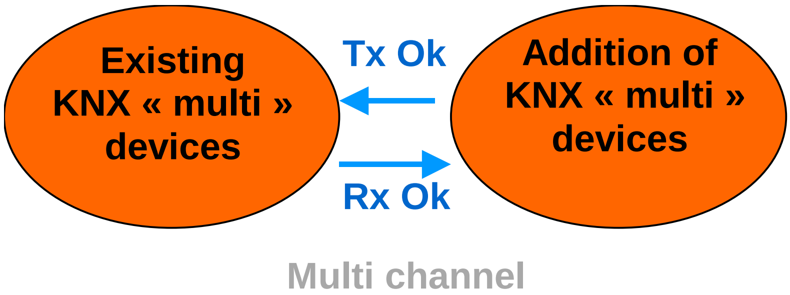
Every KNX RF Multi product will be able to listen to “KNX Ready” products thanks to the scan function. The configuration is always done on F1; therefore KNX RF multi devices will be aware that they are linked to KNX RF Ready devices. First KNX RF multi devices will change their power supply mode to a permanent receive mode; the scanning sequence will change accordingly. For all GA including the KNX RF Ready device, the Frames with KNX CTRL =0x00 will be sent with a preamble of 5 ms and only on RF channel F1. For all other links, the full Multi feature is used (15 ms of preamble and F1 to F3).



For KNX RF multi devices linked with KNX RF Ready devices, the power supply low energy mode will not be possible anymore.

7.4 Communication between KNX RF Multi and KNX RF Multi devices

The configuration is always done on RF channel F1. The Call Channel is F1 by default on devices, which can be configured by PB-Mode. For other devices, the Call Channel can be settled to F2 or F3. Each device is responsible for the choice of the power supply mode (low energy or normal). When the low power mode is available, it is highly recommended to use it. To activate this function or not can be done via ETS or a configuration tool.



8 Physical parameters for other versions

8.1 KNX RF2 Ready

8.1.1 Introduction

For all countries where the frequency band 433 MHz is allowed, the following specific Physical Layer parameters apply.

The attention of manufacturers is drawn to the fact that a local RF standard may apply in addition to the KNX RF2.R certification.

8.1.2 KNX RF2 Ready Physical Layer

8.1.2.1 Signalling

Table 19 - General requirements for Physical Layer Type RF KNX RF 2 Ready

Characteristic	Value or applicable standard
Tx centre frequency	f_c = 433,500 MHz
Bandwidth	500 kHz
Max. Tx frequency tolerance	± 25 ppm [a]
Tx duty cycle max	1 %
Tx modulation type	FSK
FSK deviation	f_{DEV} = ± 48 kHz to ± 80 kHz typically 60 kHz
Tx chip rate	32 768 cps
Maximum Tx chip rate tolerance	± 1,5 %
Maximum Tx jitter per transition	± 5 µs
Typical TX ERP	0 dBm
Min Tx ERP	-3 dBm
Max Tx ERP	+10 dBm
Rx blocking performance	according EN 300 220-1, category 2 receivers [b]
Rx centre frequency	f_c = 433,500 MHz
Rx frequency tolerance	± 25 ppm KNX Tx to KNX Rx [a, b]
Minimal Rx chip rate tolerance	±2,0 % [b]
Rx radiated sensitivity	typical: -95 dBm [b] minimal: -80 dBm [b]
Minimal operating temperature range	0°C to 45°C [c]

[a] This frequency tolerance includes tolerances due to temperature variations within the operating temperature range and tolerances due to crystal aging.

[b] At Bit Error Rate (BER) 10^{-4} in optimum antenna direction.

[c] KNX Physical Layer parameters shall be met on the entire product temperature range declared by the manufacturer. (e.g. : -10°C to 70°C for outdoor usage)

NOTE 17 A link budget of 100 dB is recommended.

8.1.2.2 Telegram structure for KNX RF2 Ready

Clause 5.1.2 shall apply.

8.1.2.3 Medium access for KNX RF2 Ready

Clause 5.1.3 shall apply.

8.1.3 KNX RF2 Ready Data Link Layer

Clauses 6.1 and 6.2 shall apply.

8.2 KNX RF3 Ready

KNX RF3 Ready devices are KNX RF2 Ready devices with a max radiated power limited to 5 mW and a modulation bandwidth range limited to 400 kHz (according EN300220-1 measurement method).

Clause 8.1 shall apply with 2 restrictions:

Modulation bandwidth	400 kHz
Max Tx ERP	+7 dBm

8.3 KNX RF4

8.3.1 KNX RF4 Physical Layer

To be completed

8.3.2 KNX RF4 Data Link Layer

To be completed

9 Marking

KNX RF devices shall have a legible marking of the RF type according to Table 19.

Table 20 – Systems type marking

Frequency sub-band	KNX RF Ready	KNX RF Multi	KNX RF BiBat	KNX RF BiBat 2
868 MHz (clauses 5 & 6)	KNX RF1.R	KNX RF1.M	KNX RF1.B	KNX RF1.B2
433 MHz (clause 8.1)	KNX RF2.R			
433 MHz limited BW (clause 8.2)	KNX RF3.R			
915 MHz (clause 8.3)	KNX RF4			