

---

---

**Information technology — Real-time  
locating systems (RTLS) —**

**Part 5:**

**Chirp spread spectrum (CSS) at 2,4 GHz  
air interface**

*Technologies de l'information — Systèmes de localisation en temps réel  
(RTLS) —*

*Partie 5: Spectre étalé de compression d'impulsions (CSS) à une  
interface d'air de 2,4 GHz*

**PDF disclaimer**

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

IECNORM.COM : Click to view the full PDF of ISO/IEC 24730-5:2010



**COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2010

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office  
Case postale 56 • CH-1211 Geneva 20  
Tel. + 41 22 749 01 11  
Fax + 41 22 749 09 47  
E-mail [copyright@iso.org](mailto:copyright@iso.org)  
Web [www.iso.org](http://www.iso.org)

Published in Switzerland

# Contents

Page

Foreword .....	vi
Introduction.....	vii
1 Scope .....	1
2 Normative references .....	1
3 Terms and definitions .....	2
4 Symbols and abbreviated terms .....	3
5 Overview.....	6
5.1 Components.....	6
5.2 Purpose .....	6
5.3 Not covered by the standard .....	7
5.4 System .....	7
5.5 Document structure .....	7
6 Requirements.....	7
6.1 Frequency range.....	7
6.2 2,4 GHz spread spectrum air interface specifications.....	7
6.3 Compliance requirements .....	8
6.4 Manufacturer tag ID .....	8
6.5 Physical layer parameters .....	8
7 Physical (PHY) layer specification.....	9
7.1 Modulations .....	9
7.2 Data rates .....	9
7.2.1 General PHY packet format .....	9
7.3 2-ary orthogonal CSS.....	9
7.3.1 Reference modulator diagram .....	10
7.3.2 Bandwidths and Transmit power spectral density (PSD) mask .....	10
7.3.3 Equivalent baseband representation of the continuous time 2-ary orthogonal CSS signal .....	12
7.3.4 Signal tolerance.....	13
7.3.5 Bit to symbol mapping.....	13
7.3.6 Chirp generator.....	13
7.3.7 Preamble.....	13
7.3.8 Start of frame delimiter .....	14
7.3.9 Bit scrambler.....	14
7.3.10 PHY Header .....	14
7.3.11 Overview (informative).....	14
7.4 DQPSK-CSS .....	16
7.4.1 Reference modulator diagram .....	16
7.4.2 Bandwidth and transmit Power Spectral Density (PSD) mask .....	17
7.4.3 Equivalent baseband representation of the continuous time DQPSK-CSS signal.....	18
7.4.4 Signal tolerance.....	20
7.4.5 Overview (informative).....	20
7.4.6 Demultiplexer (DEMUX) .....	22
7.4.7 Serial to Parallel mapping (S/P) .....	22
7.4.8 Data Symbol - to - Bi-Orthogonal code word mapping .....	22
7.4.9 Parallel - to - Serial converter (P/S) and QPSK symbol mapping .....	25
7.4.10 Differential-QPSK (DQPSK) coding .....	25
7.4.11 DQPSK to DQPSK-CSS modulation .....	26
7.4.12 Chirp generator.....	26
7.4.13 Bit interleaver.....	26

7.4.14	Preamble .....	26
7.4.15	Start of frame delimiter .....	27
7.4.16	PHY Header .....	27
8	MAC sub-layer specification.....	27
8.1	Overview .....	27
8.2	General packet format.....	27
8.3	Packet types.....	27
8.4	MAC frame formats.....	28
8.4.1	MAC frame format for Data packet.....	28
8.4.2	MAC frame format for ACK packet.....	28
8.4.3	MAC frame format for Broadcast packet.....	28
8.4.4	MAC frame format for RTS packet .....	29
8.4.5	MAC frame format for CTS packet .....	29
8.4.6	MAC frame fields.....	29
8.5	MAC Timing .....	31
8.5.1	2-way handshake .....	31
8.5.2	3-way handshake .....	32
8.5.3	Ranging-related time measurements.....	33
8.5.4	Media access .....	33
9	Tag application layer specification .....	36
9.1	Overview .....	36
9.1.1	Example scenario .....	37
9.2	Tag application states .....	38
9.2.1	Default state .....	38
9.2.2	Wait state .....	38
9.2.3	Range state.....	39
9.2.4	Sleep state .....	39
9.2.5	Blink state .....	39
9.2.6	State transitions.....	40
9.3	Commands .....	41
9.3.1	SwitchState command .....	42
9.3.2	SetConfigVector command.....	45
9.3.3	GetConfigVector command .....	47
9.3.4	SetRangingPeers command .....	47
9.3.5	AddRangingPeers command.....	47
9.3.6	GetRangingPeers command.....	48
9.3.7	User defined command .....	48
9.3.8	Command prioritization .....	48
9.4	Tag application packet formats.....	49
9.4.1	Application blink packet .....	50
9.4.2	Application command packet.....	51
9.4.3	Application report packet .....	51
9.4.4	GetConfigVector report.....	52
9.4.5	GetRangingPeers report .....	52
9.4.6	Ranging report .....	52
9.4.7	Application ranging packet.....	53
9.5	Ranging packet exchanges .....	57
9.5.1	Ranging packet exchange type 1 .....	57
9.5.2	Ranging packet exchange type 2 .....	58
9.5.3	Ranging packet exchange type 3.....	60
9.5.4	Ranging packet exchange type 4.....	61
9.6	Timing values.....	62
9.7	Default profile.....	62
9.8	Error handling .....	62
Annex A (informative)	Time base tolerances in two-way ranging.....	63
Annex B (informative)	Coexistence .....	66
Annex C (informative)	Computing location values from range values .....	69

<b>Annex D</b> (informative) <b>Location and roaming of tags</b> .....	<b>70</b>
<b>Bibliography</b> .....	<b>72</b>

IECNORM.COM : Click to view the full PDF of ISO/IEC 24730-5:2010

## Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 24730-5 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 31, *Automatic identification and data capture techniques*.

ISO/IEC 24730 consists of the following parts, under the general title *Information technology — Real-time locating systems (RTLS)*:

- *Part 1: Application program interface (API)*
- *Part 2: 2,4 GHz air interface protocol*
- *Part 5: Chirp spread spectrum at 2,4 GHz air interface*

## Introduction

CSS is a technique for spreading the bandwidth of a digital signal by using chirp pulses. Chirp pulses are pulses with a monotonically increasing or decreasing instantaneous frequency. Chirp pulses were originally used for radar applications. Recently, systems and standards have been developed which use chirp pulses also for communication applications. This part of ISO/IEC 24730 includes ranging and bidirectional communication between tags and infrastructure. Bidirectional communication enables the infrastructure to control the behaviour of tags in a timely manner.

IECNORM.COM : Click to view the full PDF of ISO/IEC 24730-5:2010

IECNORM.COM : Click to view the full PDF of ISO/IEC 24730-5:2010

# Information technology — Real-time locating systems (RTLS) —

## Part 5:

## Chirp spread spectrum (CSS) at 2,4 GHz air interface

### 1 Scope

ISO/IEC 24730 defines air interface protocols and an application programming interface (API) for real-time locating systems (RTLS). This part of ISO/IEC 24730 defines an air interface protocol which utilizes chirp spread spectrum (CSS) at frequencies from 2,4 GHz to 2,483 GHz. This protocol supports bidirectional communication and two-way ranging between the readers and tags of an RTLS. The mandatory default mode ensures interoperability between tags and infrastructure from various manufacturers, while the availability of several options offers flexibility to the developer of the infrastructure to adapt the behaviour of the overall system to the specific needs of his application.

### 2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 15963, *Information technology — Radio frequency identification for item management — Unique identification for RF tags*

ISO/IEC 19762-1, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 1: General terms relating to AIDC*

ISO/IEC 19762-3, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 3: Radio frequency identification (RFID)*

ISO/IEC 19762-4, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 4: General terms relating to radio communications*

ISO/IEC 19762-5, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 5: Locating systems*

ISO/IEC 24730-1, *Information technology — Real-time locating systems (RTLS) — Part 1: Application program interface (API)*

*Guidelines on Limiting Exposure to Non-Ionizing Radiation*, International Commission on Non-Ionizing Radiation Protection (ICNIRP), Munich, 1999

IEC 62369-1 ed1.0, *Evaluation of human exposure to electromagnetic fields from short range devices (SRDs) in various applications over the frequency range 0 GHz to 300 GHz — Part 1: Fields produced by devices used for electronic article surveillance, radio frequency identification and similar systems*

IEEE Std C95.1-2005, *IEEE Standard for Safety Levels with Respect to Human Exposure to Radio Frequency Electromagnetic Fields, 3 kHz to 300 GHz*

### 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19762-1, ISO/IEC 19762-3, ISO/IEC 19762-4, ISO/IEC 19762-5 and the following apply.

- 3.1**  
**chirp spread spectrum**  
technique for spreading the bandwidth of a digital signal using linear frequency sweep signals
- 3.2**  
**Class I**  
system that operates at a radiated power of up to 10 mW EIRP
- 3.3**  
**Class II**  
system that operates at a radiated power higher than 10 mW up to the maximum defined by local regulations
- 3.4**  
**ranging**  
process of determining the distance between two RTLS transceivers through the exchange of a specific set of messages
- 3.5**  
**ranging peer**  
RTLS transceiver with which to perform ranging
- 3.6**  
**RF channel**  
combination of a centre frequency value and bandwidth value
- 3.7**  
**RTLS tag**  
RTLS transceiver that accepts commands from RTLS readers and sends blinks and/or reports to the RTLS readers
- 3.8**  
**RTLS transmitter**  
part of an RTLS transceiver which is capable of sending messages
- 3.9**  
**demultiplexer**  
equipment for reversing the process of multiplexing
- 3.10**  
**medium**  
wireless channel
- 3.11**  
**trilateration**  
method of determining the relative positions of objects using the known locations of three reference points and the measured distance between the object to be located and each reference point
- 3.12**  
**interleaving**  
rearrangement or transposition of data to enhance the effectiveness of error control schemes
- 3.13**  
**interleaver**  
unit that performs interleaving (3.12)

**3.14****baseband**

frequency band occupied by the aggregate of the signals used to modulate a carrier before they combine with the carrier in the modulation process

**3.15****orthogonal**

inner product being close to zero

**3.16****peer X**

x'th peer in a description of a situation with multiple peers

**4 Symbols and abbreviated terms**

ACK	acknowledge
ARQ	Automatic Repeat Query
BTS	Backoff Time Slot
CIFS	Carrier sense Inter Frame Space
CTS	Clear To Send
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance
CSS	Chirp Spread Spectrum
dBr	decibel relative
DEMUX	demultiplexer
DQPSK	Differential Quadrature Phase Shift Keying
DQPSK-CSS	Differential Quadrature Phase Shift Keying over Chirp Spread Spectrum
Dst	Destination address
EIRP	Equivalent Isotropical Radiated Power
LFSR	Linear Feedback Shift Register
LSB	Least Significant Bit
MMSE	Minimum Mean Square Error
MAC	Medium Access Control
NAV	Network Allocation Vector
PHR	PHY header
PHY	physical layer
PPDU	PHY Protocol Data Unit

PSDU	PHY Service Data Unit
QPSK	Quadrature Phase Shift Keying
RTS	Request To Send
RTLS	Real Time Locating System
SFD	Start of Frame Delimiter
SHR	synchronization header
SIFS	Short Inter Frame Space
Src	source address
TWR	Two Way Ranging
SDS-TWR	Symmetric Double Sided Two Way Ranging
$e$	Euler constant
$j$	imaginary unit
$\tilde{s}^{M_0}(t)$	continuous time baseband representation of 2-ary orthogonal CSS signal
$\tilde{r}^{M_0}(t)$	implemented version of $\tilde{s}^{M_0}(t)$
$\tilde{s}_m^{M_1}(t)$	continuous time baseband representation of DQPSK-CSS signal
$\tilde{r}_m^{M_1}(t)$	implemented version of $\tilde{s}_m^{M_1}(t)$
$m$	configuration constant determining the type (one out of four possibilities) of sub-chirp sequence used
$M_0$	superscript indicating that 2-ary orthogonal CSS is described
$M_1$	superscript indicating that DQPSK-CSS is described
$k$	index variable
$n$	index variable
$b_n$	n'th symbol to be transmitted
$c_b(t)$	continuous time baseband representation of chirp pulse b for 2-ary orthogonal CSS
$\mu_0$	configuration constant determining the chirp rate for 2-ary orthogonal CSS
$\mu_1$	constant determining the chirp rate for DQPSK-CSS
$T_{base}$	timebase

$T_{SBIFS}$	time in between two sub blinks
$T_{Blink}$	average blink repetition time
$T_{Rand}$	random time
$T_{Rxon}$	duration of time interval for during which the receiver of a tag is activated
$T_{Contact}$	maximum expected duration between a tag receiving any packets from infrastructure if such is present
$T_{TimeoutApplication}$	duration of time interval during which a tag application shall respond to certain requests
$T_{WaitAfterRange}$	duration of time for which a tag shall go to Wait state after leaving Range state
$T_0$	configuration constant determining the duration of a chirp pulse for 2-ary orthogonal CSS
$T_1$	duration of sub-chirp sequence
$T_{sub}$	duration of sub-chirp
$T_{n,k,m}$	time position of k'th sub-chirp of n'th sub-chirp sequence of type m
$W_T(t)$	raised cosine window of duration $T$
$\alpha$	roll off factor of the raised cosine window
$A$	amplitude variable which is minimized in minimum mean square error computation
$\tau_d$	time delay variable that is minimized in minimum mean square error computation
$\varphi$	phase variable that is minimized in minimum mean square error computation
$d_{n,k}$	information sample of k'th sub-chirp in n'th sub-chirp sequence
$C_{k,m}^{sub}(t)$	continuous time baseband representation of k' sub-chirp of sub-chirp sequence type m
$\tau_m$	timing constant that determines the time-gap between subsequent sub-chirp sequences for the sub-chirp sequence type m
$f_{k,m}$	offset centre frequency of k'th sub-chirp in sub-chirp sequence type m
$\zeta_{k,m}$	chirp direction of k'th sub-chirp in sub-chirp sequence type m
$S_m^{sub}(t)$	continuous time baseband representation of sub-chirp sequence of type m

5 Overview

5.1 Components

The major components of a real-time locating system (RTLS) and the relationship of those components are shown in Figure 1. As shown in this Figure the tags communicate with an infrastructure. The infrastructure provides an application program interface (API) through which an application can control the RTLS and retrieve information about location and state of tags.

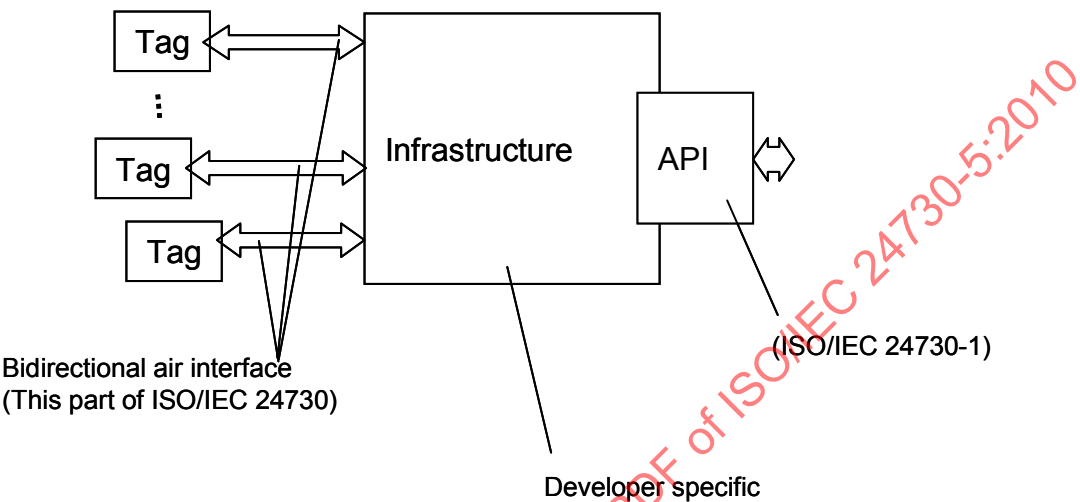


Figure 1 – RTLS components

As indicated in Figure 1 tags communicate with infrastructure over an air interface. Generally the air interface includes the definition of waveforms, formats of packets as well as commands and reports to be exchanged between tags and infrastructure. This can be depicted in a layered approach as shown in Figure 2. Similar interpretations can be found in other standards e.g. in ISO/IEC 18000-1<sup>[1]</sup>.

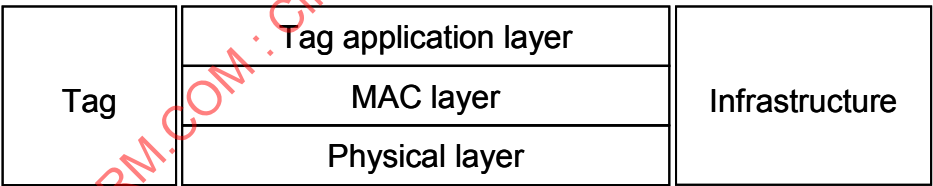


Figure 2 – Air interface layers

5.2 Purpose

This part of ISO/IEC 24730 defines an air interface protocol that optimizes small scale RTLS with an installation that enables simple, and also handheld, RTLS readers. Although the infrastructure itself is not defined in this part of ISO/IEC 24730, it is anticipated that the air interface protocol has a strong impact on the realization of the infrastructure and the related installation effort.

The key condition for simple installation is the possibility of 'autonomous' infrastructure nodes. In this part of ISO/IEC 24730, "autonomous" means there is not a requirement for these nodes to be synchronized with other infrastructure nodes. After being placed at a fixed location, an autonomous node simply responds to requests from RTLS tags.

This condition is achieved by specifying bidirectional communication and two-way ranging. As a consequence, the tag must also support bidirectional communication. Although this requirement increases the complexity of

the tag in one area, it also decreases the complexity in other areas, as additional interfaces for programming and conditioning the tag are not required. Thus bidirectional communication with the tag is seen as beneficial for many applications. Finally in order to utilize existing state of the art communication technology this part of ISO/IEC 24730 includes parts that correspond with IEEE 802.15.4a<sup>[5]</sup>, which is a PHY amendment of IEEE 802.15.4<sup>[4]</sup>, a successful standard for low power, low data rate wireless communication.

### 5.3 Not covered by the standard

The design of the infrastructure is left completely to the developer, e.g. the density of RTLS reader nodes, how the RTLS readers are controlled and communicate with each other, how the infrastructure is set up, etc. may be different in various scenarios and for systems from different vendors. For typical RTLS applications, at least three RTLS readers will communicate with each tag, measuring time of flight in order to locate the tag. For more details on this interaction, see Clause 9, Tag application layer specification.

### 5.4 System

After power on, a tag uses a default profile in which it blinks periodically. With each blink the tag signals its physical address, its capabilities and information about when it will be ready to receive commands from the infrastructure.

The infrastructure decides whether it needs to send commands to the tag while the tag is listening. By sending commands to the tag, the infrastructure controls which RTLS readers are part of the infrastructure the tag performs ranging with. Furthermore the infrastructure can adapt the behaviour of the tags to the actual conditions such as the number of tags in range, number of infrastructure nodes available, etc. For example, the infrastructure is able to instruct the tag to change to another mode (bandwidth, centre frequency, data rate) according to the actual environment or to perform ranging with a specific set of RTLS readers.

When the tag assumes that it has lost connection to the infrastructure e.g. because it doesn't receive any commands for a certain time, it reverts to the default profile. A more detailed description of complete system behaviour can be found in Annex D.

### 5.5 Document structure

The remainder of this part of ISO/IEC 24730 follows the "layered structure" mentioned above. This means that after the Requirements clause the three layers that form an air interface protocol (the Physical Layer [PHY], Media Access Control [MAC] and the Tag application layer) are addressed and specified separately. Additional information for the user of this part of ISO/IEC 24730 is provided in the informative annexes.

## 6 Requirements

### 6.1 Frequency range

This part of ISO/IEC 24730 addresses real-time locating systems (RTLS) operating in the 2,400 to 2,4835 GHz frequencies.

### 6.2 2,4 GHz spread spectrum air interface specifications

The minimum requirements shall include:

- RTLS transceivers shall autonomously generate a chirp spread spectrum frequency beacon indicating when the receiver will be activated.
- RTLS transceivers shall be able to perform two-way ranging when the receiver is activated.
- RTLS transmitters shall be fully compliant with local regulatory requirements.

- Class I RF transmissions shall not exceed 10 mW EIRP
- Class II RF transmissions shall not exceed 100 mW EIRP or the maximum EIRP according to the local radio regulations.

### 6.3 Compliance requirements

To be fully compliant with this part of ISO/IEC 24730, real-time locating systems (RTLS) shall also comply with ISO/IEC 24730-1.

Device manufacturers claiming conformance to this part of ISO/IEC 24730 shall self-certify RF emissions do not exceed the maximum permitted exposure limits recommended by either IEEE C95.1: 2005 or ICNIRP according to IEC 62369-1. If a device manufacturer is unsure as to which recommendation to be cited for compliance the manufacturer shall self-certify to ICNIRP limits.

### 6.4 Manufacturer tag ID

The manufacturer's tag identification number identifies a particular manufacturer and consists of 16 bits. A manufacturer may have more than one ID number. The frame format used in this part of ISO/IEC 24730 mandates a MAC address of at least 48 bits for each device. The first 16 bits of the MAC address are designated for the manufacturer's identification number and shall be assigned according to ISO/IEC 15963 Annex D, under Allocation Class 0000 0000.

### 6.5 Physical layer parameters

For the purposes of this part of ISO/IEC 24730, the following parameter definitions apply in Table 1. These parameters are referenced by parameter name. These operating parameters are to be defined for the temperature range of minus 30 degrees Celsius to 50 degrees Celsius.

**Table 1 – 2,4 GHz CSS link parameters**

Parameter name	Description
Operating frequency range	As permitted by local radio regulations in the band from 2400 to 2483,5 MHz.
Operating frequency accuracy	± 70 ppm (2-ary orthogonal CSS) ± 40 ppm (DQPSK –CSS)
Maximum phase noise	-85 dBc/Hz @ 1 MHz
Occupied 20 dB channel bandwidth	80 MHz (Default value) 22 MHz (Configuration option)
Centre frequency, bandwidth combination	According to Table 2 and Table 6
Time base, $T_{base}$	31,25 ns
Time base accuracy	± 40 ppm
Transmit power	Class 1: 10 dBm EIRP. max. Class 2: Maximum in accordance to local regulations.
Transmitter spectrum mask	According to Table 3 and Figure 13
Spurious emission, out of band	The device shall transmit in conformance with spurious emissions requirements defined by the country's regulatory authority within which the system is operated.
Modulation	2-ary orthogonal CSS DQPSK-CSS (optional)

Parameter name	Description
Data bit rates	1 Mbit/s 250 kbit/s
Symbol rates	$10^6$ symbols/s, 250000 symbols/s 166667 symbols/s (with DQPSK-CSS)

## 7 Physical (PHY) layer specification

### 7.1 Modulations

The PHY layer specification shall contain two modulations (2-ary orthogonal CSS and DQPSK-CSS). The support of 2-ary orthogonal shall be mandatory while the support of DQPSK-CSS is optional.

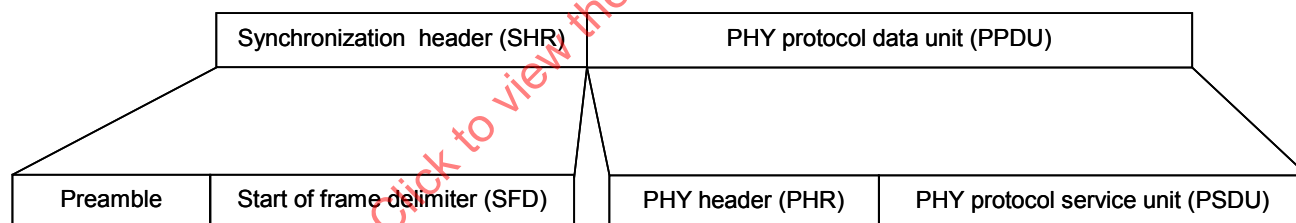
### 7.2 Data rates

2-ary orthogonal CSS shall support data rates of 1 Mbits/s and 250 kbit/s.

DQPSK-CSS, when implemented, shall support data rates of 1 Mbits/s and 250 kbit/s.

#### 7.2.1 General PHY packet format

A PHY packet shall consist of a synchronization header and a PHY protocol data unit as shown in Figure 3.



**Figure 3 – General PHY packet structure**

The definitions of preamble, SFD and PHY are given in 7.3.7, 7.3.8, 7.3.10 and 7.4.14, 7.4.15, 7.4.16.

The PSDU shall contain the MAC a frame defined Clause 8.

### 7.3 2-ary orthogonal CSS

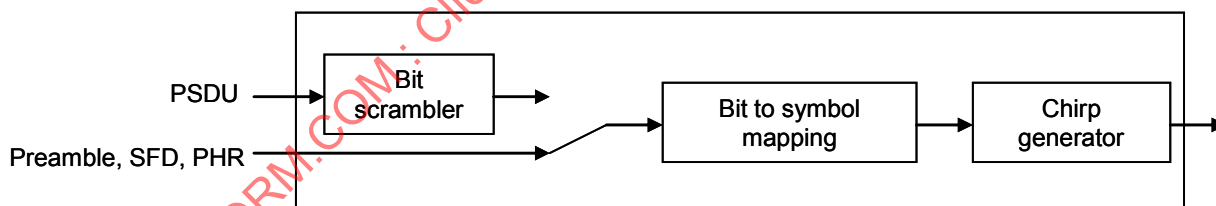
2-ary orthogonal CSS shall be the mandatory PHY mode. A combination of centre frequency and bandwidth shall called "RF channel". The possible RF channels are defined in Table 2.

**Table 2 – Possible combinations of centre frequency and bandwidth for 2-ary orthogonal CSS**

RF channel number	Centre frequency	Bandwidth
0	2441,75 MHz	80 MHz
1	2441,75 MHz	22 MHz
2	2412 MHz	22 MHz
3	2417 MHz	22 MHz
4	2422 MHz	22 MHz
5	2427 MHz	22 MHz
6	2432 MHz	22 MHz
7	2437 MHz	22 MHz
8	2442 MHz	22 MHz
9	2447 MHz	22 MHz
10	2452 MHz	22 MHz
11	2457 MHz	22 MHz
12	2462 MHz	22 MHz
13	2467 MHz	22 MHz
14	2472 MHz	22 MHz
15	2484 MHz	22 MHz

### 7.3.1 Reference modulator diagram

The functional block diagram in Figure 4 is provided as a reference for specifying 2-ary orthogonal CSS for both data rates 1 Mbits/s and 250 kbit/s. The functionality of the Bit scrambler block is specified in 7.3.9. The functionality of the Bit to symbol mapping block is specified in 7.3.5. The functionality of the chirp generator block is specified in 7.3.6.

**Figure 4 – Reference modulator for 2-ary orthogonal CSS**

### 7.3.2 Bandwidths and Transmit power spectral density (PSD) mask

The supported bandwidth values shall be 80 MHz and 22 MHz. The transmitted spectral products shall be less than the limits specified in Table 3, Figure 5 and Figure 6. For both relative and absolute limits, average spectral power shall be measured using a 100 kHz resolution bandwidth. For the relative limit, the reference level shall be the highest average spectral power measured within  $\pm \text{Bandwidth}/2$  of the centre frequency,  $f_c$ . For testing the transmitted spectral power density a pseudo-random binary sequence shall be used as input data.

Table 3 – Transmit PSD limits for 2-ary CSS

Frequency	Relative limit	Absolute limit
$ f-f_c  > \text{Bandwidth}/2$	-20 dBr	-30 dBm

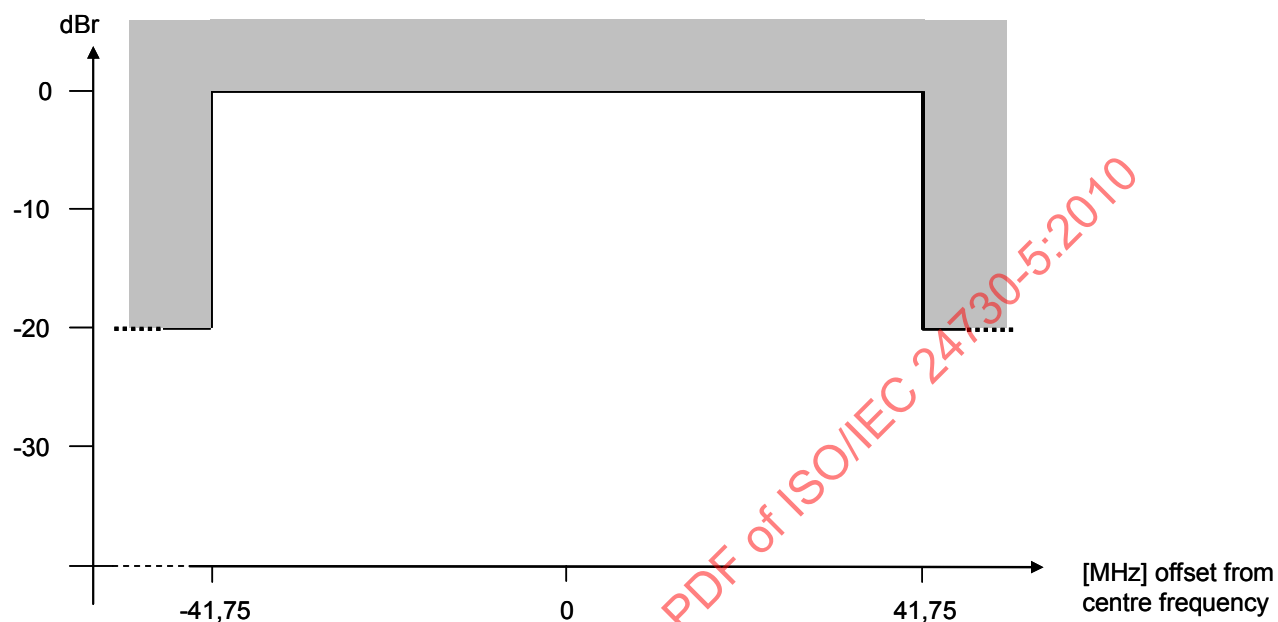


Figure 5 – Transmit PSD limits for 2-ary orthogonal CSS at 80 MHz bandwidth

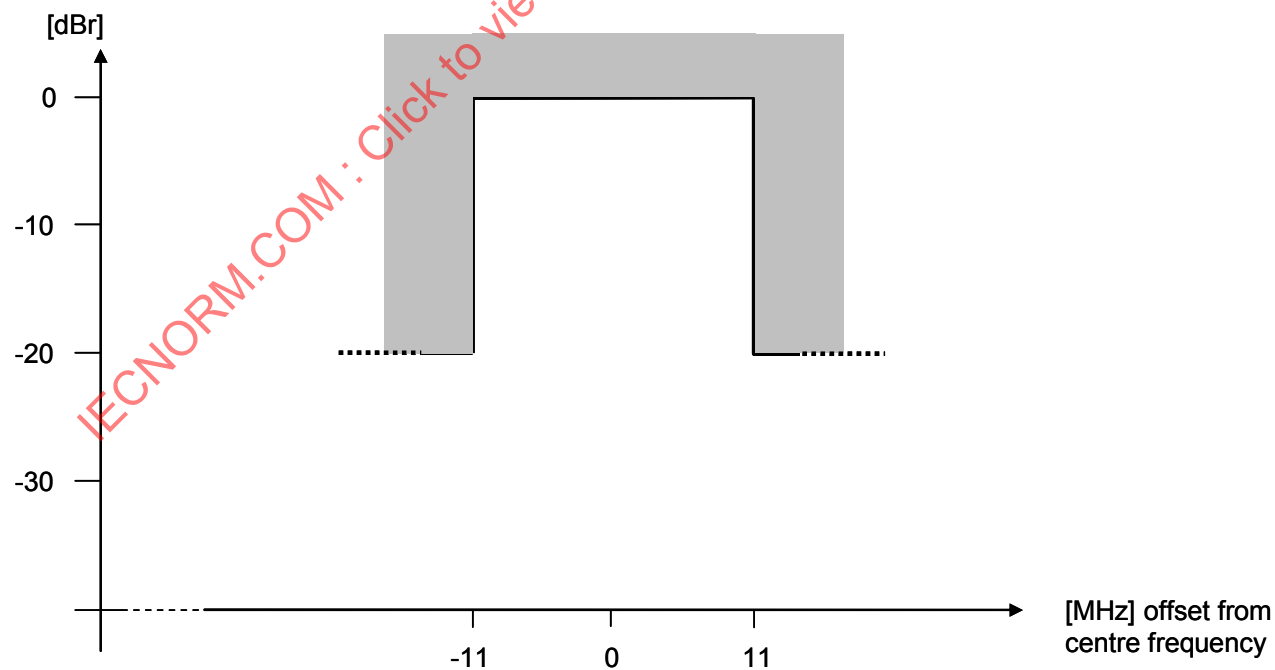


Figure 6 – Transmit PSD limits for 2-ary orthogonal CSS at 22 MHz bandwidth

### 7.3.3 Equivalent baseband representation of the continuous time 2-ary orthogonal CSS signal

The mathematical representation of the continuous time-domain baseband signal  $\tilde{s}^{M_0}(t)$  for 2-ary orthogonal CSS shall be given by Equation (1).

$$\tilde{s}^{M_0}(t) = \sum_{n=0}^{\infty} c_{b_n} \left( t - nT_0 - \frac{T_0}{2} \right) \quad (1)$$

Where

$M_0$  is indicating that 2-ary orthogonal CSS is described,

$n$  is the index of the symbol,

$b_n$  are symbols to be transmitted, which can take the values from [1, -1] and determine which of the two possible pulses is actually realized.

$c_b(t)$  are the continuous time-domain baseband versions of the two possible pulses, which are required for 2-ary orthogonal modulation and which shall be chirp pulses as described by Equation (2).

$$c_b(t) = \begin{cases} \exp \left[ j \cdot b \cdot \frac{\mu_0}{2} \cdot t^2 \right] \cdot W_{T_0}(t) & \text{for } |t| \leq \frac{T_0}{2} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Where

$\mu_0$  is a constant which either takes the value  $2\pi \cdot \frac{80\text{MHz}}{T_0}$  or  $2\pi \cdot \frac{22\text{MHz}}{T_0}$  depending on the bandwidth selected.

$T_0$  is the duration of the chirp pulse which can take a value of 1  $\mu\text{s}$  or 4  $\mu\text{s}$ , depending on the data rate,

$$j = \sqrt{-1}$$

and

$W_T(t)$  is the raised cosine window as described in Equation (3).

$$W_T(t) = \begin{cases} 1 & \text{for } |t| \leq \frac{(1-\alpha)T}{(1+\alpha)2} \\ \frac{1}{2} \left[ 1 + \cos \left( \frac{(1+\alpha)\pi}{\alpha T} \left( |t| - \frac{(1-\alpha)T}{(1+\alpha)2} \right) \right) \right] & \text{for } \frac{(1-\alpha)T}{(1+\alpha)2} < |t| \leq \frac{T}{2} \\ 0 & \text{for } |t| > \frac{T}{2} \end{cases} \quad (3)$$

with  $\alpha = 0.25$

Where

$T$  is the duration of the raised cosine window.

$\alpha$  is the roll off factor of the raised cosine window.

### 7.3.4 Signal tolerance

In addition to the limits specified in Table 3, the Minimum Mean Square Error, MMSE, shall be used as criterion for the compliance of the signal. Let  $\tilde{s}^{M_0}(t)$  be the signal that is defined by (1). Then  $\tilde{r}^{M_0}(t)$ , the implemented version of  $\tilde{s}^{M_0}(t)$ , shall satisfy (4) for  $b_0=+1$  and  $b_0=-1$ .

$$MMSE = \min_{A, \tau_d, \varphi} \left[ \frac{\int_0^{T_0} |\tilde{s}^{M_0}(t) - A \cdot \tilde{r}^{M_0}(t - \tau_d) \cdot e^{j\varphi}|^2 dt}{\int_0^{T_0} |\tilde{s}^{M_0}(t)|^2 dt} \right] \leq 0.005 \quad (4)$$

where the variables  $A$ ,  $\tau_d$ , and  $\varphi$  are used to minimize the mean squared error.

### 7.3.5 Bit to symbol mapping

A bit value of 0 shall be mapped to  $b=-1$ .

A bit value of 1 shall be mapped to  $b=1$ .

### 7.3.6 Chirp generator

The chirp generator shall generate subsequent pulses as defined in Equation (2).

### 7.3.7 Preamble

The preamble for 2-ary orthogonal CSS shall consist of 30 alternating bits starting with bit 0 as shown in Table 4.

**Table 4 – Preamble bit sequence for 2-ary orthogonal CSS**

Bit 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

### 7.3.8 Start of frame delimiter

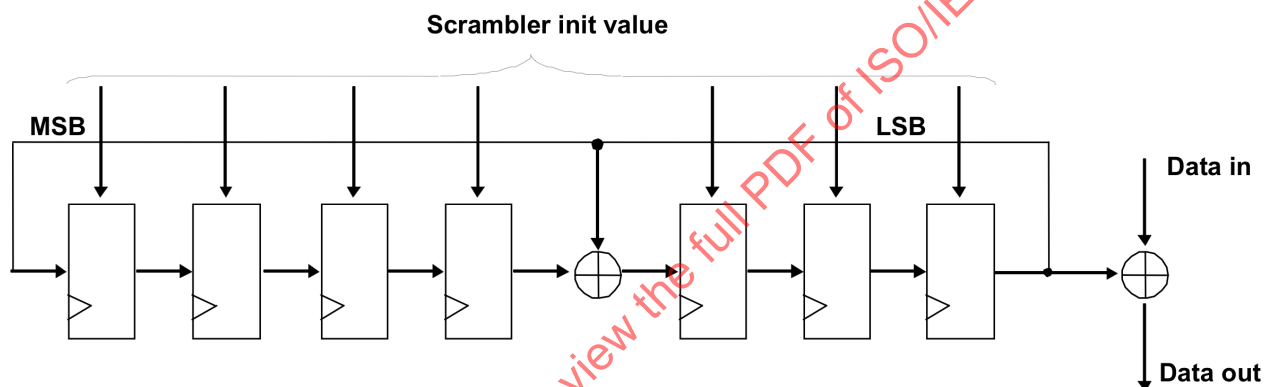
The start of frame delimiter (SFD) for 2-ary orthogonal CSS shall consist of 64 bits that correspond to 8 octets as specified in Table 5. The SFD shall start with the least significant bit of octet 0 and end with the most significant bit of octet 7.

**Table 5 – Start of frame delimiter for 2-ary orthogonal CSS**

Octet 0	Octet 1	Octet 2	Octet 3	Octet 4	Octet 5	Octet 6	Octet 7
64h	52h	52h	9Ah	5Ah	4Bh	DBh	54h

### 7.3.9 Bit scrambler

Before transmission, the bit sequence shall be scrambled with a pseudo-random noise (PRN) sequence. The PRN-sequence shall be generated with the polynomial  $g(D) = D^7 + D^4 + D^0$  (221 in octal representation) and is subsequently EXORed with the bit stream. The 127-bit-length PRN-sequence shall be generated with the linear feedback shift register, as illustrated in Figure 7.



**Figure 7 – Bit scrambler**

### 7.3.10 PHY Header

The PHY Header for 2-ary orthogonal CSS shall consist of 8 bits. The first bit shall be reserved. The subsequent 7 bits shall contain the Scrambler init value as depicted in Figure 7 starting with the MSB.

### 7.3.11 Overview (informative)

For illustrative purposes Figure 8 shows an up chirp pulse and a down chirp pulse in passband domain, while Figure 9 shows the same pulses in time frequency domain. A bit sequence modulated with 2-ary orthogonal CSS is depicted in Figure 10 and Figure 11.

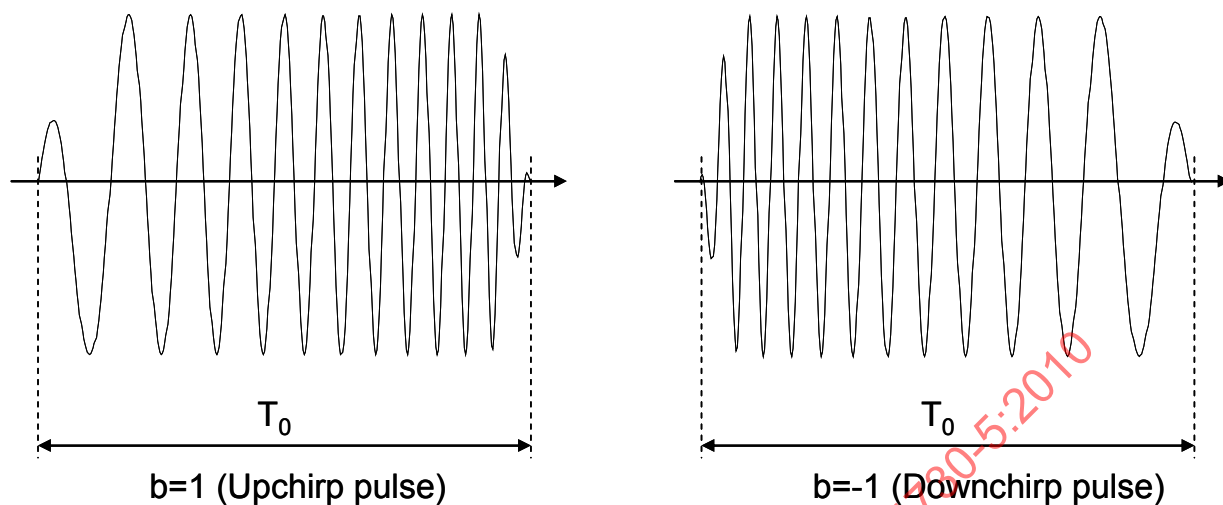


Figure 8 – Chirp pulses in passband domain

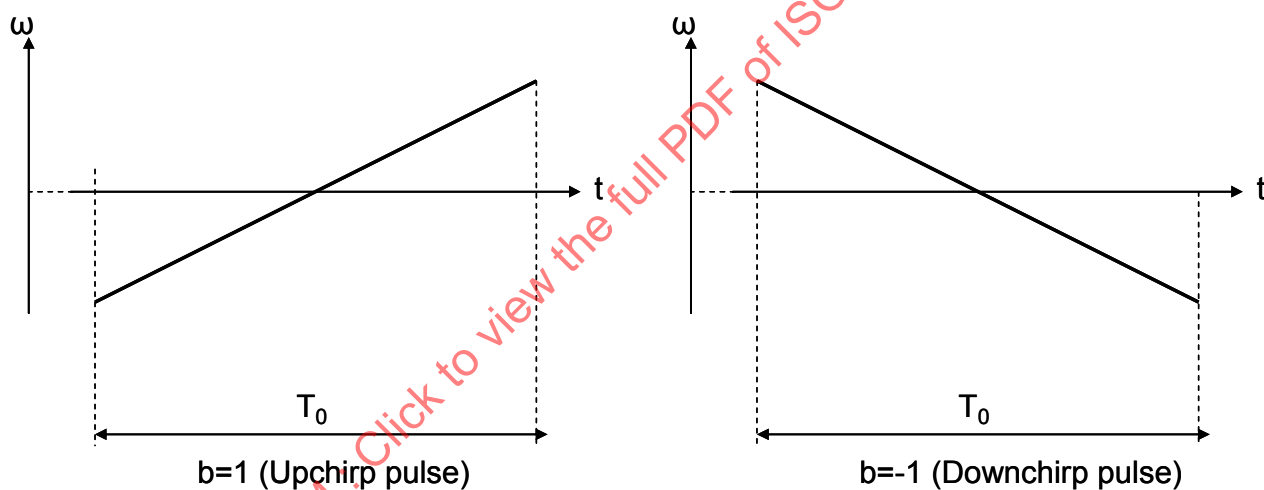


Figure 9 – Chirp pulses in time frequency domain

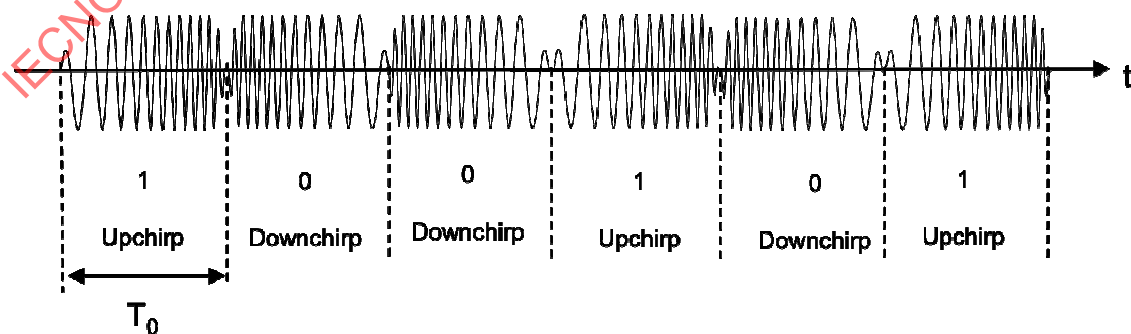


Figure 10 – 2-ary orthogonal CSS modulated bit sequence in passband domain

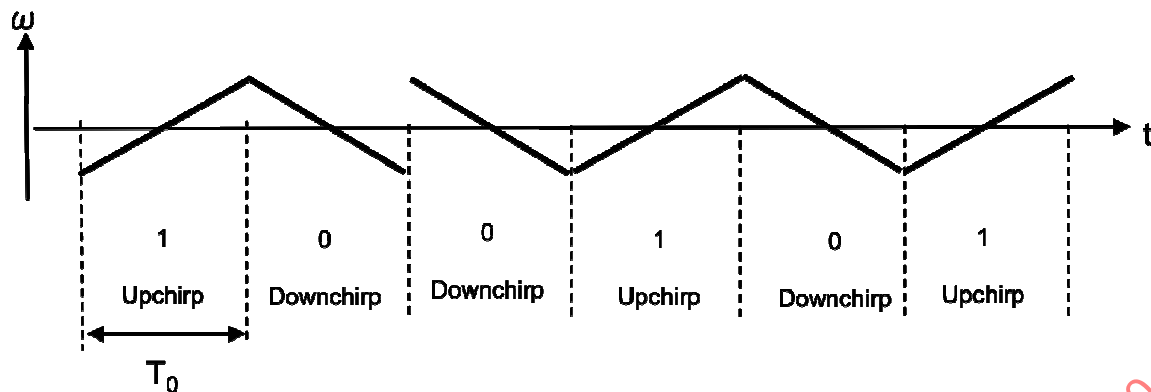


Figure 11 – 2-ary orthogonal CSS modulated bit sequence in time frequency domain

#### 7.4 DQPSK-CSS

Differential Quadrature Phase Shift Keying, in combination with chirp spread spectrum, is an optional PHY mode. The possible combinations of centre frequencies and bandwidths are defined in Table 6.

Table 6 – Possible combinations of centre frequency and bandwidth for DQPSK-CSS

Centre frequency	Bandwidth
2412 MHz	22 MHz
2417 MHz	22 MHz
2422 MHz	22 MHz
2427 MHz	22 MHz
2432 MHz	22 MHz
2437 MHz	22 MHz
2442 MHz	22 MHz
2447 MHz	22 MHz
2452 MHz	22 MHz
2457 MHz	22 MHz
2462 MHz	22 MHz
2467 MHz	22 MHz
2472 MHz	22 MHz
2484 MHz	22 MHz

##### 7.4.1 Reference modulator diagram

The functional block diagram in Figure 12 is provided as a reference for specifying the DQPSK CSS mode for both data rates 1 Mbit/s and 250 kbit/s. The functionality of the 1:2 DEMUX block is specified in 7.4.6. The functionality of the S/P block is specified in 7.4.7. The functionality of the Symbol Mapper block is specified in 7.4.8. The functionality of the Interleaver block is specified in 7.4.13. The functionality of the P/S block is specified in 7.4.9. The functionality of the QPSK mapper is specified in 7.4.9. The functionality of the chirp generator block mapper is specified in 7.4.12.

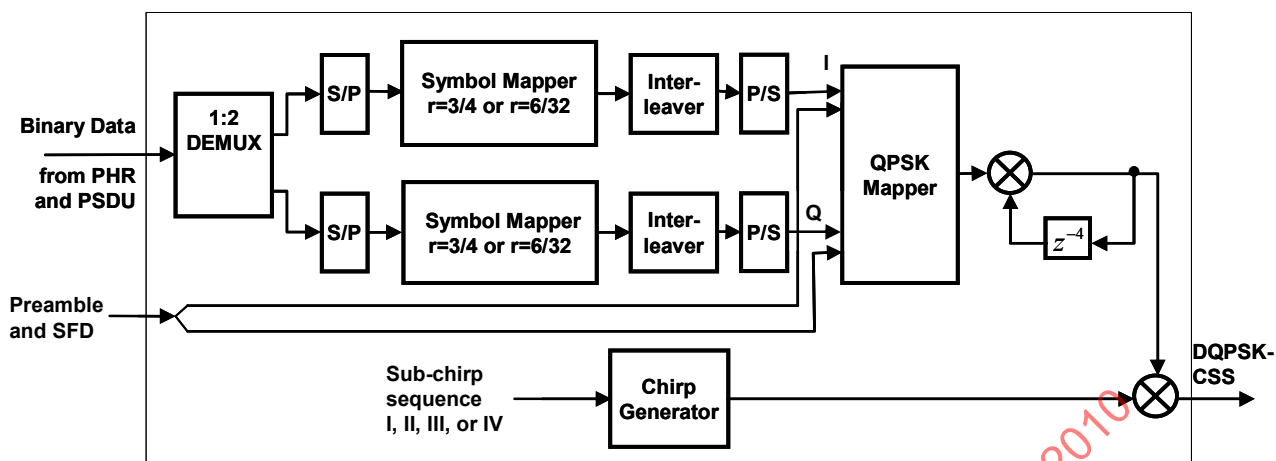


Figure 12 – DQPSK CSS reference modulator

#### 7.4.2 Bandwidth and transmit Power Spectral Density (PSD) mask

The supported bandwidth value shall be 22 MHz. The transmitted spectral products shall be less than the limits specified in Figure 13. The average spectral power shall be measured using a 100 kHz resolution bandwidth. For the relative limit, the reference level shall be the highest average spectral power measured within  $\pm 11$  MHz of the centre frequency. For testing the transmitted spectral power density, a pseudo-random binary sequence shall be used as input data.

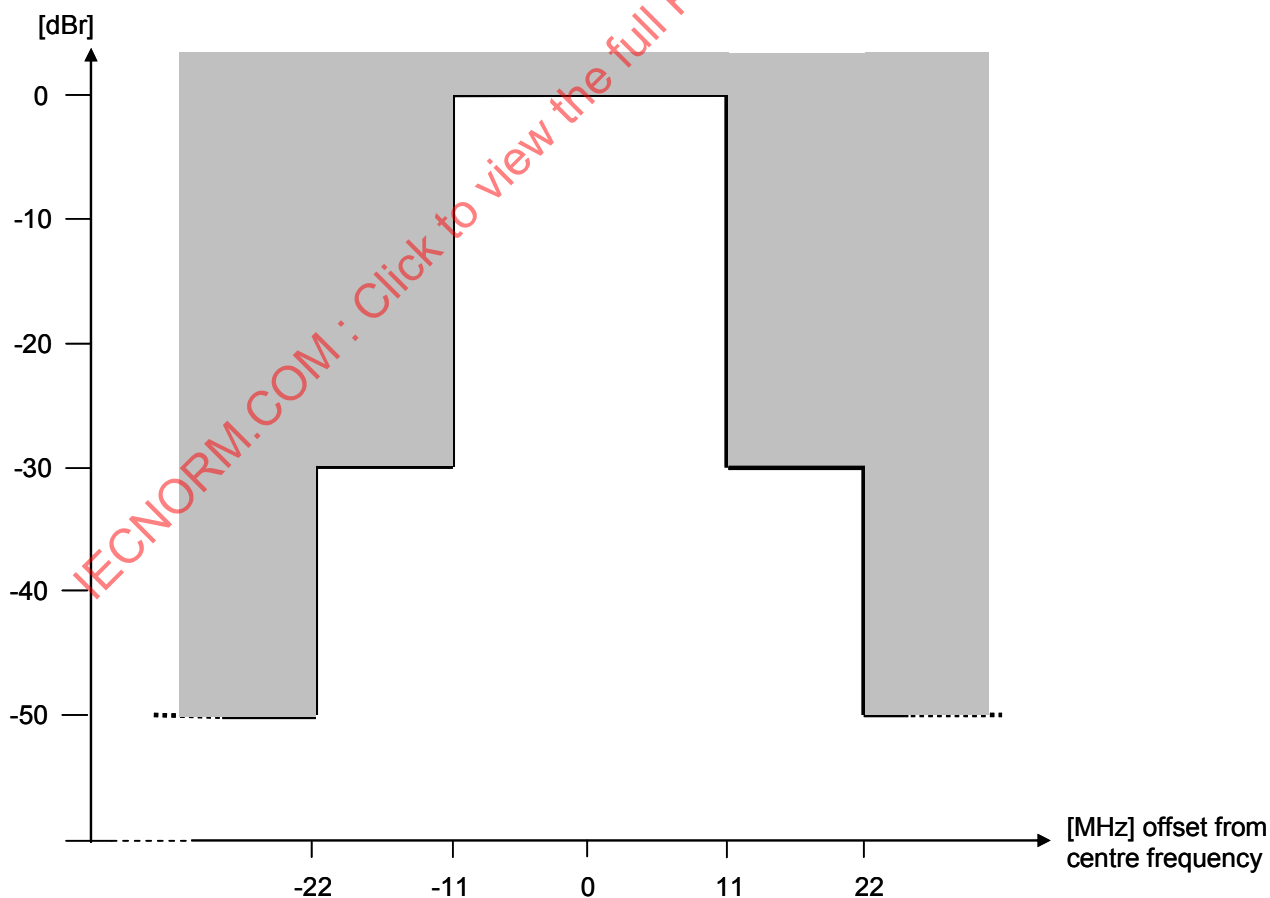


Figure 13 – Transmit PSD limits for DQPSK-CSS

### 7.4.3 Equivalent baseband representation of the continuous time DQPSK-CSS signal

The equivalent baseband representation of the continuous time DQPSK-CSS signal,  $\tilde{s}_m^{M_1}(t)$ , shall be defined by Equation (5)

$$\tilde{s}_m^{M_1}(t) = \sum_{n=0}^{\infty} \sum_{k=1}^4 d_{n,k} C_{k,m}^{sub}(t - T_{n,k,m}) \quad (5)$$

Where

$M_1$  is indicating that DQPSK-CSS is described.

$m$  is a configuration constant from [1,2,3,4] (corresponding to [I, II, III, IV] in Figure 14) that defines which of the four possible sub-chirp sequences has been selected.

$n$  is the index of the sub-chirp sequence.

$k$  is the index of the sub-chirp (see 7.4.3.1)

$d_{n,k}$  is the stream of DQPSK samples which can take values from  $[1+j, 1-j, -1+j, -1-j]$  with  $j = \sqrt{-1}$ .

$C_{k,m}^{sub}(t)$  is a sub-chirp as defined in 7.4.3.1

$T_{n,k,m}$  is the time position of the sub-chirp as defined by Equation (6).

$$T_{n,k,m} = (k-1)T_{sub} + nT_1 - (1 - (-1)^n)\tau_m \quad (6)$$

Where

$T_{sub}$  is the duration of the sub-chirp as defined in Table 9.

$T_1$  is the duration of the sub-chirp sequence as defined in Table 9.

$\tau_m$  is a constant as defined in Table 9 that determines which time-gap is applied between the previous and the actual sub-chirp sequence (see Figure 15).

#### 7.4.3.1 Sub-chirp

A sub-chirp shall be defined by (7)

$$C_{k,m}^{sub}(t) = \exp \left[ j \left( 2\pi \cdot f_{k,m} + \frac{\mu_1}{2} \zeta_{k,m} \left( t - \frac{T_{sub}}{2} \right) \right) \left( t - \frac{T_{sub}}{2} \right) \right] \cdot W_{T_{sub}} \left( t - \frac{T_{sub}}{2} \right) \quad (7)$$

Where

$f_{k,m}$  is a constant that defines the offset centre frequency of the sub-chirp as defined in Table 7.

$\zeta_{k,m}$  is a constant that defines direction of the sub-chirp as defined in Table 8.

$T_{sub}$  is the duration of the sub-chirp as defined in Table 9.

$\mu_1$  is a constant that takes the value  $2\pi \cdot \frac{8.6875125 \text{ MHz}}{T_{\text{sub}}}$ .

$W_{T_{\text{sub}}}$  is the raised cosine window with a duration of  $T_{\text{sub}}$  as defined in Equation (3).

#### 7.4.3.2 Sub-chirp sequence

A sub-chirp sequence shall be defined by (8)

$$S_m^{\text{sub}}(t) = \sum_{k=1}^4 C_{k,m}^{\text{sub}}(t - (k-1)T_{\text{sub}}) \quad (8)$$

Where

$m$  is a configuration constant that defines which of the four possible sub-chirp sequences has been selected.

$C_{k,m}^{\text{sub}}(t)$  is a sub-chirp as defined in 7.4.3.1

$T_{\text{sub}}$  is the duration of the sub-chirp as defined in Table 9.

#### 7.4.3.3 Parameters

Table 7 defines the offset centre frequencies of the sub-chirps.

Table 8 defines the chirp directions of the sub-chirps.

Table 9 defines the timing parameters of the sub-chirps.

**Table 7 – Offset centre frequencies of sub-chirps**

$f_{k,m}$ [MHz]				
$m \backslash k$	1	2	3	4
1	-3,15	+3,15	+3,15	-3,15
2	+3,15	-3,15	-3,15	+3,15
3	-3,15	+3,15	+3,15	-3,15
4	+3,15	-3,15	-3,15	+3,15

**Table 8 – Sub-chirp directions**

$\xi_{k,m}$				
$m \backslash k$	1	2	3	4
1	+1	+1	-1	-1
2	+1	-1	+1	-1
3	-1	-1	+1	+1
4	-1	+1	-1	+1

Table 9 – Timing parameters

Name	Value
$T_1$	$192 T_{\text{base}}$
$T_{\text{sub}}$	$38 T_{\text{base}}$
$\tau_1$	$15 T_{\text{base}}$
$\tau_2$	$10 T_{\text{base}}$
$\tau_3$	$5 T_{\text{base}}$
$\tau_4$	0

#### 7.4.4 Signal tolerance

In addition to the limits specified in Figure 13, the minimum mean square error shall be used as criterion for the compliance of the signal. Let  $\tilde{s}_m^{M_1}(t)$  be the signal that is defined by Equation (5). Then  $\tilde{r}_m^{M_1}(t)$ , the implemented version of  $\tilde{s}_m^{M_1}(t)$ , shall satisfy Equation (9).

$$MMSE = \min_{A, \tau_d, \varphi} \left[ \frac{\int_0^{T_0} |\tilde{s}_m^{M_1}(t) - A \cdot \tilde{r}_m^{M_1}(t - \tau_d) \cdot e^{j\varphi}|^2 dt}{\int_0^{T_0} |\tilde{s}_m^{M_1}(t)|^2 dt} \right] \leq 0.005 \text{ for } m=1,2,3,4 \quad (9)$$

where the constants  $A$ ,  $\tau_d$ , and  $\varphi$  are used to minimize the mean squared error.

The  $d_{n,k}$  in  $\tilde{s}_m^{M_1}(t)$  in Equation (5) are constantly equal  $(1+j)$  for the measurement for all  $n$  and  $k$ .

#### 7.4.5 Overview (informative)

DQPSK-CSS uses chirp spread spectrum (CSS) techniques in combination with Differential Quadrature Phase Shift Keying (DQPSK) and 8-ary or 64-ary Bi-Orthogonal Coding for 1 Mbit/s data rate or 250 kbit/s data rate, respectively.

##### 7.4.5.1 Waveform and sub-chirp sequences

Four individual chirp pulses, here called *sub-chirps*, shall be concatenated to form a sub-chirp sequence that occupies two adjacent frequency sub-bands. Each sub-chirp is weighted with a raised cosine window in the time domain.

By using alternating time-gaps in conjunction with sub-chirp sequences, this DQPSK-CSS provides sub-chirp sequence division as well as frequency division. Four different sub-chirp sequences are defined. Figure 14 shows the four different sub-chirp sequences in time frequency domain. It can be seen that four sub-chirps that have either a linear down-chirp characteristic or a linear up-chirp characteristic and a centre frequency which has either a positive or a negative frequency offset against the centre frequency of the signal are concatenated. The frequency discontinuities between subsequent chirps will not impact the spectrum because the signal amplitude will be zero at these points due to the raised cosine window applied to each sub-chirp.

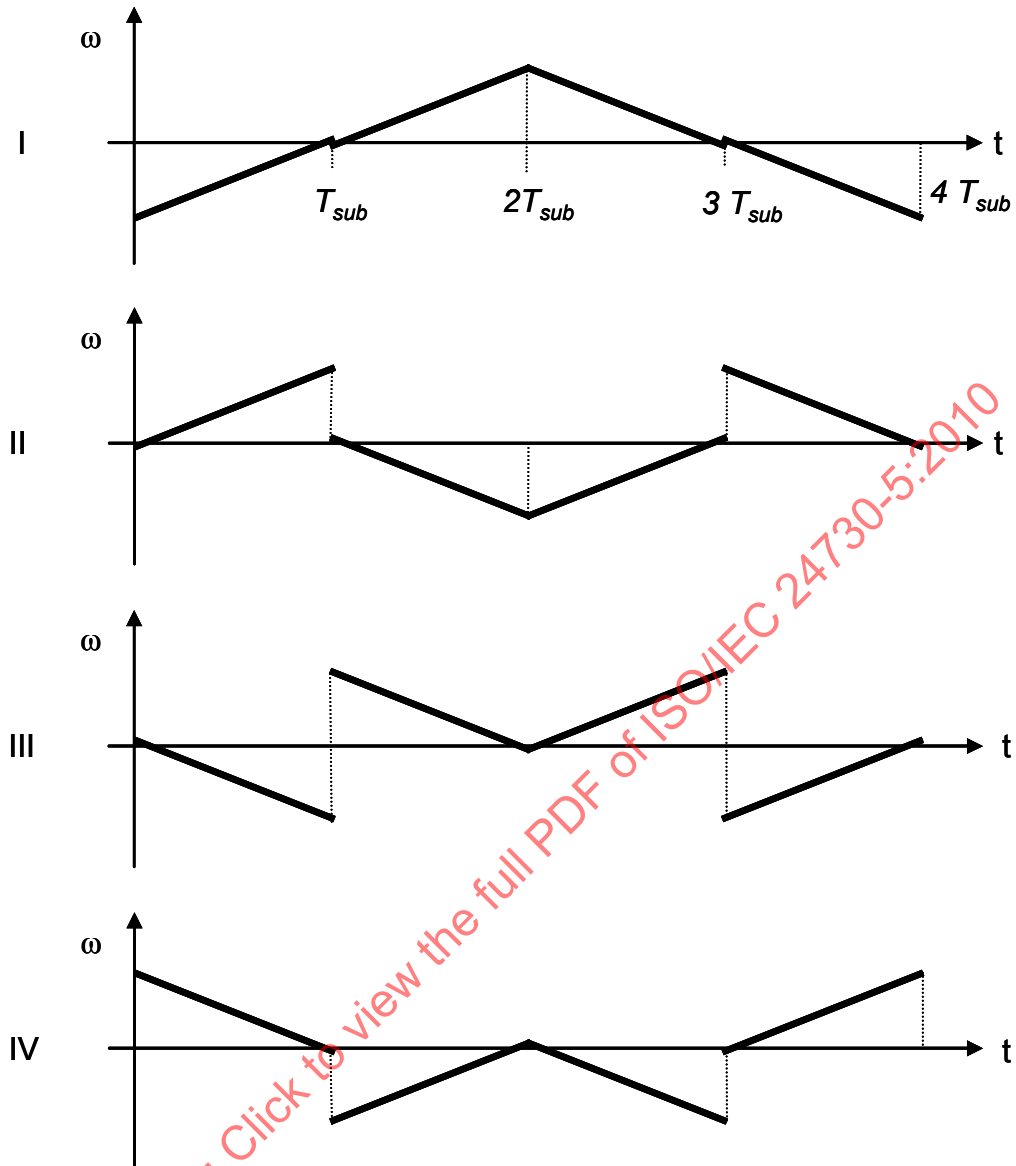


Figure 14 – The four defined sub-chirp sequences

#### 7.4.5.2 Active usage of time gaps

In conjunction with the sub-chirp sequences, pairs of time-gaps are defined. The time-gaps are chosen to apply a sharper orthogonality to the four sub-chirp sequences. The time-gaps are applied alternatively between subsequent chirp symbols as shown in Figure 15.

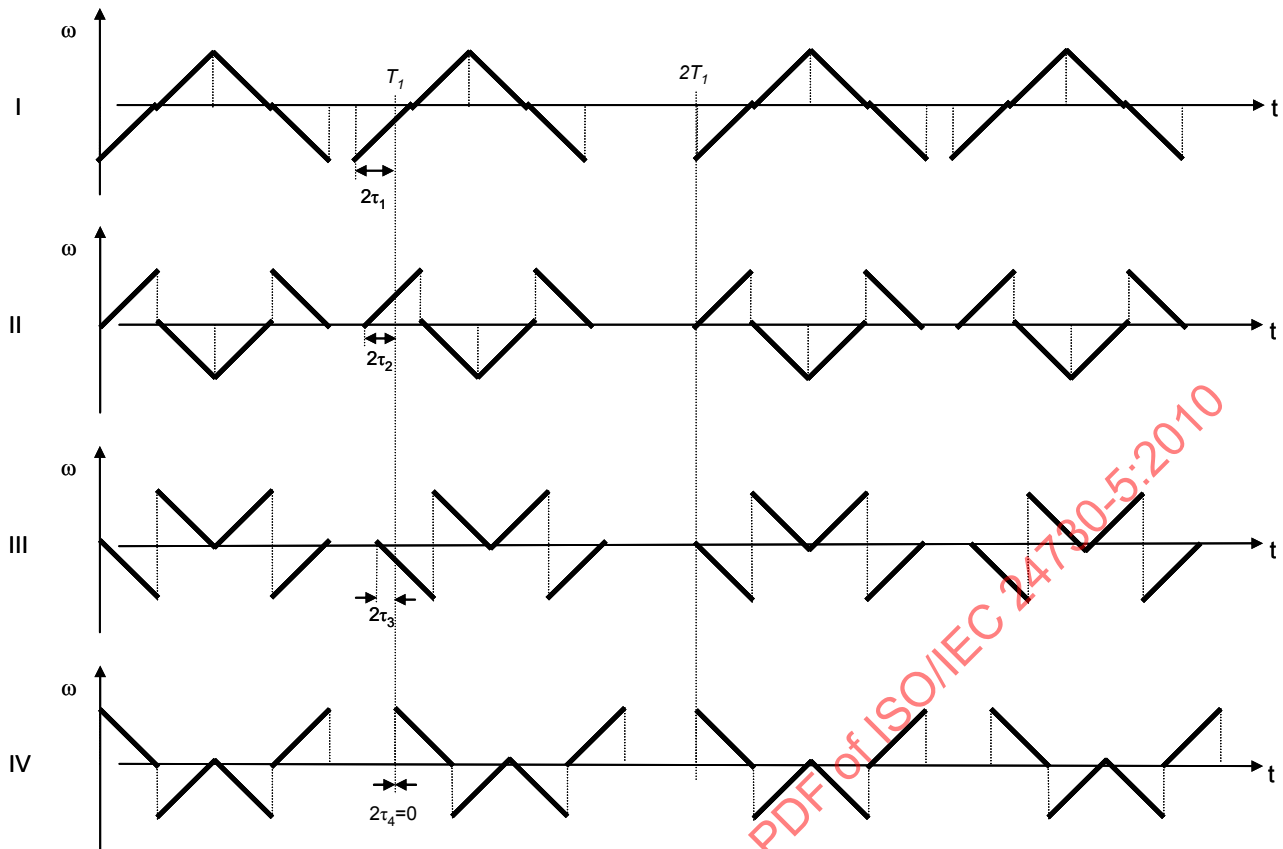


Figure 15 – The four time-gap pairs for the four defined sub-chirp sequences

#### 7.4.6 Demultiplexer (DEMUX)

For each packet the initial position of the DEMUX shown in Figure 12 shall be set to serve the I path (upper path). Thus, the first bit of the incoming stream of information bits of a packet shall be switched to the I path and the second bit shall be switched to the Q path.

#### 7.4.7 Serial to Parallel mapping (S/P)

By using two serial to parallel converters, the sub-streams are independently partitioned into sets of bits to form data symbols. For the data rate of 1 Mbit/s a data symbol shall consist of three bits. Within the binary data symbol ( $b_0, b_1, b_2$ ), the first input data bit for each of I and Q is assigned  $b_0$  and the third input data bit is assigned  $b_2$ . For the data rate of 250 kbit/s a data symbol shall consist of 6 bits, within the binary data symbol ( $b_0, b_1, b_2, b_3, b_4, b_5$ ), the first input data bit for each of I and Q is assigned  $b_0$  and the 6th input data bit is assigned  $b_5$ .

#### 7.4.8 Data Symbol - to - Bi-Orthogonal code word mapping

Each 3-bit data symbol shall be mapped onto a 4-chip Bi-Orthogonal code word ( $c_0, c_1, c_2, c_3$ ) for the data rate of 1 Mbit/s as specified in Table 10. Each 6-bit data symbol shall be mapped onto a 32-chip Bi-Orthogonal code word ( $c_0, c_1, c_2, \dots, c_{31}$ ) for the data rate 250 kbit/s, as specified in Table 11.

Table 10 – 8-ary Bi-Orthogonal Mapping Table ( $r = 3/4$ )

8-ary Bi-Orthogonal $r = 3/4$ , 1Mbit/s		
Data Symbol (Decimal)	Data Symbol (Binary) ( $b_0 b_1 b_2$ )	Code Word ( $C_0 C_1 C_2 C_3$ )
0	000	1 1 1 1
1	001	1 -1 1 -1
2	010	1 1 -1 -1
3	011	1 -1 -1 1
4	100	-1 -1 -1 -1
5	101	-1 1 -1 1
6	110	-1 -1 1 1
7	111	-1 1 1 -1

Table 11 – 64-ary Bi-Orthogonal mapping table ( $r = 6/32$ ) (Optional)

64-ary Bi-Orthogonal $r = 6/32$ , 250kbit/s		
Data Symbol (Decimal)	Data Symbol (Binary) ( $b_0 b_1 b_2 b_3 b_4 b_5$ )	Code Word ( $c_0 c_1 c_2 \dots c_{31}$ )
0	000000	1 1
1	000001	1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1
2	000010	1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1
3	000011	1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1
4	000100	1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1
5	000101	1 -1 1 -1 -1 -1 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1
6	000110	1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1
7	000111	1 -1 -1 1 -1 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1
8	001000	1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
9	001001	1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1
10	001010	1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1
11	001011	1 -1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 -1 -1
12	001100	1 1 1 1 -1 -1 -1 -1 -1 -1 1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1 1 1
13	001101	1 -1 1 -1 1 -1 1 -1 1 -1 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 -1
14	001110	1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1
15	001111	1 -1 -1 1 -1 1 1 -1 1 -1 1 1 -1 1 -1 1 1 -1 -1 1 -1 1 1 -1 -1 1 1 -1 1 -1 -1 -1
16	010000	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1

[illegible]

64-ary Bi-Orthogonal $r = 6/32$ , 250kbit/s		
Data Symbol (Decimal)	Data Symbol (Binary) (b0 b1 b2 b3 b4 b5)	Code Word (c0 c1 c2 ... c31)
51	110011	-1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1
52	110100	-1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 1 1 1 1 1 1 -1 -1 -1 1 1 1 1 1 -1 -1 -1 1 1 1 -1 -1
53	110101	-1 1 -1 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1
54	110110	-1 -1 1 1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 1 1 -1 -1 -1 -1 1 1 1 1 -1 -1 -1 1 1 1 -1 1
55	110111	-1 1 1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1
56	111000	-1 -1 -1 -1 -1 -1 -1 1 1 1 1 1 1 1 1 1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
57	111001	-1 1 -1 1 -1 1 -1 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1
58	111010	-1 -1 1 1 -1 -1 1 1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 1 1
59	111011	-1 1 1 -1 -1 1 1 -1 1 -1 1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 -1 1 1 -1 1 -1
60	111100	-1 -1 -1 -1 1 1 1 1 1 1 1 1 -1 -1 1 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1
61	111101	-1 1 -1 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 -1
62	111110	-1 -1 1 1 1 1 -1 -1 1 1 -1 -1 1 1 1 1 -1 -1 -1 -1 1 1 -1 -1 1 1 -1 1 1 1 -1 -1 -1
63	111111	-1 1 1 -1 1 -1 -1 1 1 -1 -1 1 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 -1 1

#### 7.4.9 Parallel - to - Serial converter (P/S) and QPSK symbol mapping

Each Bi-Orthogonal code word shall be converted to a serial chip sequence. Within each 4-chip code word (c0, c1, c2, c3) for the data rate of 1 Mbit/s, the least significant chip c0 is processed first and the most significant chip c3 is processed last for I and Q, respectively. Within each 32-chip code word (c0, c1, c2, ..., c31) for the data rate of 250 kbit/s, the least significant chip c0 is processed first and the most significant chip c31 is processed last for I and Q, respectively. Each pair of I and Q chips shall be mapped onto a QPSK symbol as specified in Table 12.

Table 12 – QPSK Mapping Table

QPSK Symbol Mapping		
Input chips ( $I_{n,k}$ $Q_{n,k}$ )	Magnitude	Output Phase $\varphi_{n,k}$ [rad]
1, 1	1	0
-1, 1	1	$\pi/2$
1, -1	1	$-\pi/2$
-1, -1	1	$\pi$

#### 7.4.10 Differential-QPSK (DQPSK) coding

The stream of QPSK symbols shall be differentially encoded by using a differential encoder with a QPSK symbol feedback memory of length 4. (This means that the phase differences between QPSK symbol 1 and 5, 2 and 6, 3 and 7, 4 and 8 and so on are computed.)

DQPSK Output:  $e^{j\theta_{n,k}} = e^{j\theta_{n-1,k}} \times e^{j\varphi_{n,k}}$

where  $e^{j\varphi_{n,k}}$  is the QPSK input,

$e^{j\theta_{n-1,k}}$  is stored in feedback memory.

For every packet, the initial values of all 4 feedback memory stages of the differential encoder shall be set  $e^{j\pi/4}$  or equivalently  $\theta_{0,k} = \pi/4$  [rad].

#### 7.4.11 DQPSK to DQPSK-CSS modulation

The stream of DQPSK symbols shall be modulated onto the stream of sub-chirps generated by the chirp generator. One DQPSK symbol shall be multiplied with one sub-chirp.

#### 7.4.12 Chirp generator

The chirp generator shall periodically generate the one of the four defined sub-chirp sequences as specified in 7.4.3.2, which has been configured.

#### 7.4.13 Bit interleaver

The bit interleaver shall be applied only for the data rate of 250 kbit/s. The 32 chip Bi-Orthogonal code words shall be interleaved prior to the parallel to serial converter. The input-output relationship of this interleaver shall be given by:

Input:

even-symbol (c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10, c11, c12, c13, c14, c15,  
c16, c17, c18, c19, c20, c21, c22, c23, c24, c25, c26, c27, c28, c29, c30, c31)  
odd-symbol (d0, d1, d2, d3, d4, d5, d6, d7, d8, d9, d10, d11, d12, d13, d14, d15,  
d16, d17, d18, d19, d20, d21, d22, d23, d24, d25, d26, d27, d28, d29, d30, d31)

Output:

even-symbol (c0, c1, c2, c3, d20, d21, d22, d23, c8, c9, c10, c11, d28, d29, d30, d31,  
c16, c17, c18, c19, d4, d5, d6, d7, c24, c25, c26, c27, d12, d13, d14, d15)  
odd-symbol (d0, d1, d2, d3, c20, c21, c22, c23, d8, d9, d10, d11, c28, c29, c30, c31,  
d16, d17, d18, d19, c4, c5, c6, c7, d24, d25, d26, d27, c12, c13, c14, c15)

As shown in Figure 12, coding (symbol mapping) is applied to every bit following the SFD. The first codeword generated shall be counted as zero, and thus is even.

#### 7.4.14 Preamble

The preamble for 1Mbit/s shall consist of 32 bits as specified in Table 13. The preamble for 250 kbit/s shall consist of 80 bits as specified in Table 13. The preamble sequence from Table 13 shall be applied directly and in parallel to the I input and the Q input of QPSK Mapper, as indicated in Figure 12.

**Table 13 – Preamble Sequence**

Data rate	Preamble Sequence
1 Mbit/s	ones(0:31)
250 kbit/s	ones(0:79)

where ones(0:N) for integer number N shall define a 1-by-N matrix of ones.

#### 7.4.15 Start of frame delimiter

A different start of frame delimiter (SFD) shall be used for each of the two data rates. According to the data rate, the SFD sequences as specified in Table 14 shall be applied starting with bit 0 directly and in parallel to the I input and the Q input in of QPSK Mapper, as indicated in Figure 12.

**Table 14 – SFDs for DQPSK-CSS**

Data rate	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9	Bit 10	Bit 11	Bit 12	Bit 13	Bit 14	Bit 15
1 Mbit/s	-1	1	1	1	-1	1	-1	-1	1	-1	-1	1	1	1	-1	-1
250 kbit/s	-1	1	1	1	1	-1	1	-1	-1	-1	1	-1	-1	-1	1	1

#### 7.4.16 PHY Header

The PHY Header for QPSK CSS shall be empty (consist of 0 bits).

## 8 MAC sub-layer specification

### 8.1 Overview

As generally described in literature, e.g. Tanenbaum<sup>[6]</sup>, the MAC sub-layer is the layer above the PHY layer. The MAC sub-layer controls access to the Medium (wireless channel) using methods such as ALOHA or CSMA/CA. The MAC sub-layer performs operations like address filtering, checksum generation and evaluation. Furthermore, the MAC sub-layer processes handshake packets for 2-way handshake and 3-way handshake e.g. IEEE 802.11<sup>[3]</sup>. In this part of ISO/IEC 24730 the process of ranging (determining the distance between two RTLS transceivers) is realized using handshake packets. Thus ranging-related time measurements are specified inside the MAC sub-layer specification.

### 8.2 General packet format

The PHY protocol service unit shown in Figure 3 shall also be called “MAC frame”.

### 8.3 Packet types

The RTLS transceiver shall support the packet types shown in Table 15.

**Table 15 – Packet types**

Type	Name	Description	Format
Data	Data	The MAC frame of a Data packet can contain up to 8192 octets of data payload.	Figure 16
Acknowledgement	ACK	Acknowledges the successful reception of a Data packet.	Figure 17
Broadcast	Broadcast	Transmits information to all stations in range	Figure 18
Request to Send	RTS	Requests a frame transmission	Figure 19
Clear to Send	CTS	Confirms a requested frame transmission and indicates that a Data packet can be transmitted.	Figure 20

## 8.4 MAC frame formats

### 8.4.1 MAC frame format for Data packet

The Data packet shall use the MAC frame format defined in Figure 16.

Reserved	Type	Dst	Src	Length	Ctrl	CRC1	MAC payload	CRC2
4	4	48	48	13	3	16	8...8*8192	16

**Figure 16 – MAC frame format for Data packet**

### 8.4.2 MAC frame format for ACK packet

The ACK packet shall use the MAC frame format defined in Figure 17.

Reserved	Type	Dst	CRC1
4	4	48	16

**Figure 17 – MAC frame format for ACK packet**

### 8.4.3 MAC frame format for Broadcast packet

The Broadcast packet shall use the MAC frame format defined in Figure 18.

Reserved	Type	Blink info	Src	Length	Ctrl	CRC1	MAC payload	CRC2
4	4	48	48	13	3	16	8...8*8192	16

**Figure 18 – MAC frame format for Broadcast packet**

#### 8.4.4 MAC frame format for RTS packet

The RTS packet shall use the MAC frame format defined in Figure 19.

Reserved	Type	Dst	Src	Length	Ctrl	CRC1
4	4	48	48	13	3	16

Figure 19 – RTS packet

#### 8.4.5 MAC frame format for CTS packet

The CTS packet shall use the MAC frame format defined in Figure 20.

Reserved	Type	Dst	Length	Ctrl	CRC1
4	4	48	13	3	16

Figure 20 – CTS packet

#### 8.4.6 MAC frame fields

##### 8.4.6.1 Reserved field

The reserved bit field shall be reserved for future enhancements. The bits of the reserved field shall be written as specified in Table 16.

Table 16 – Reserved field definitions

Bit 0	Bit 1	Bit 2	Bit 3
0	0	0	0

##### 8.4.6.2 Type field

The Type field shall contain a code that identifies the actual packet type as specified in Table 17.

Table 17 – Type field definitions

Packet type	Bit 0	Bit 1	Bit 2	Bit 3
Data	0	0	0	0
ACK	1	0	0	0
Broadcast	1	1	0	0
RTS	0	0	1	0
CTS	1	0	1	0

8.4.6.3 Blink info

The Blink info field shall be used by the upper layer according to the specification in Figure 40.

8.4.6.4 Dst field

The Dst field shall contain the 48-bit destination address starting with bit 0.

8.4.6.5 Src field

The Src field shall contain the 48-bit source MAC address starting with bit 0. The MAC address shall be the Tag ID described in 6.4.

8.4.6.6 Length field

In a Data or Broadcast packet, the Length field shall contain the number of octets of MAC payload. The minimum size of the MAC payload is one octet. A value 0 of the Length field shall indicate that the MAC payload contains 8192 octets.

In a RTS or CTS packet, the Length field shall contain the number of octets that the initiating RTLS transceiver intends to transmit in the MAC payload of the subsequent Data packet.

8.4.6.7 Ctrl field

The Ctrl field shall be used by the upper layer according to the specification in Clause 9.

8.4.6.8 CRC calculation

CRC1, as well as CRC2, shall be calculated using the polynomial  $X^{16}+X^{12}+X^5+1$ . For reference purposes, a structure for CRC calculation using a Linear Feedback Shift Register (LFSR) is depicted in Figure 21 and Table 18.

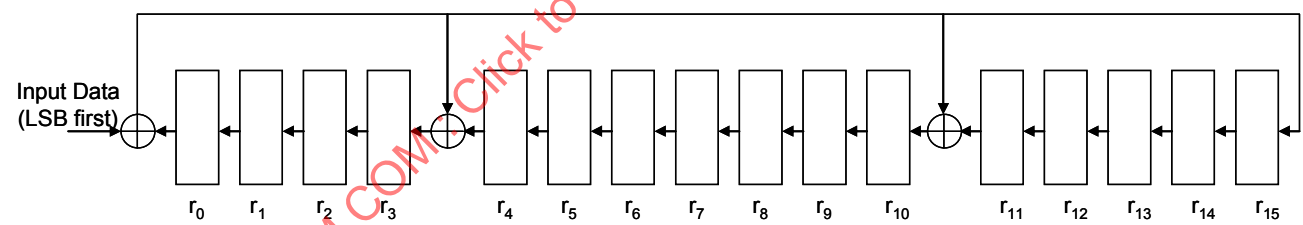


Figure 21 – CRC calculation with LFSR

Table 18 – Steps for CRC calculation with LFSR

Step	Operation
1	Initialize all registers ( $r_0$ through $r_{15}$ ) to one
2	Shift data portion for which CRC is to be calculated into the LFSR
3	After the data portion has been shifted in, invert bits $r_0$ through $r_{15}$ .
4	Bits $r_0$ through $r_{15}$ contain the CRC value

#### 8.4.6.9 CRC1 field

The CRC1 field shall contain the CRC1 checksum starting with bit  $r_{15}$ . CRC1 shall be calculated starting with the first bit of the MAC frame until the bit just before the CRC1 field.

#### 8.4.6.10 MAC payload field

The MAC payload field shall be used by the upper layer according to the specification in Clause 9.

#### 8.4.6.11 CRC2 field

The CRC2 field shall contain the CRC2 checksum starting with bit  $r_{15}$ . CRC2 shall be calculated starting with the first bit following the CRC1 field until the last bit of the MAC payload.

### 8.5 MAC Timing

Table 19 specifies timing values which shall be used by the MAC sub-layer as specified in the subsequent sub clauses.

**Table 19 – Basic MAC timing parameters**

Parameter name	Value	Tolerance
AirInterface PropagationTimeMax	8 $\mu$ s	Relative timebase tolerance as specified in Clause 6
SIFS (Short Inter Frame Space)	8 $\mu$ s	Relative timebase tolerance as specified in Clause 6.
CIFS (Carrier sense Inter Frame Space)	SIFS +2 AirInterface PropagationTimeMax	Relative timebase tolerance as specified in Clause 6.
BTS (Backoff Time Slot)	24 $\mu$ s	Relative timebase tolerance as specified in Clause 6

#### 8.5.1 2-way handshake

2-way handshake shall consist of acknowledging a received Data packet by transmitting an ACK packet back to the RTLS transmitter that sent the Data packet. Upon reception of a Data packet, the MAC sub-layer of the responding RTLS transceiver shall start its ACK packet after SIFS time, as depicted in Figure 22.

An RTLS transceiver shall be capable of performing a 2-way handshake as initiating RTLS transceiver.

An RTLS transceiver shall be capable of performing a 2-way handshake as responding RTLS transceiver.

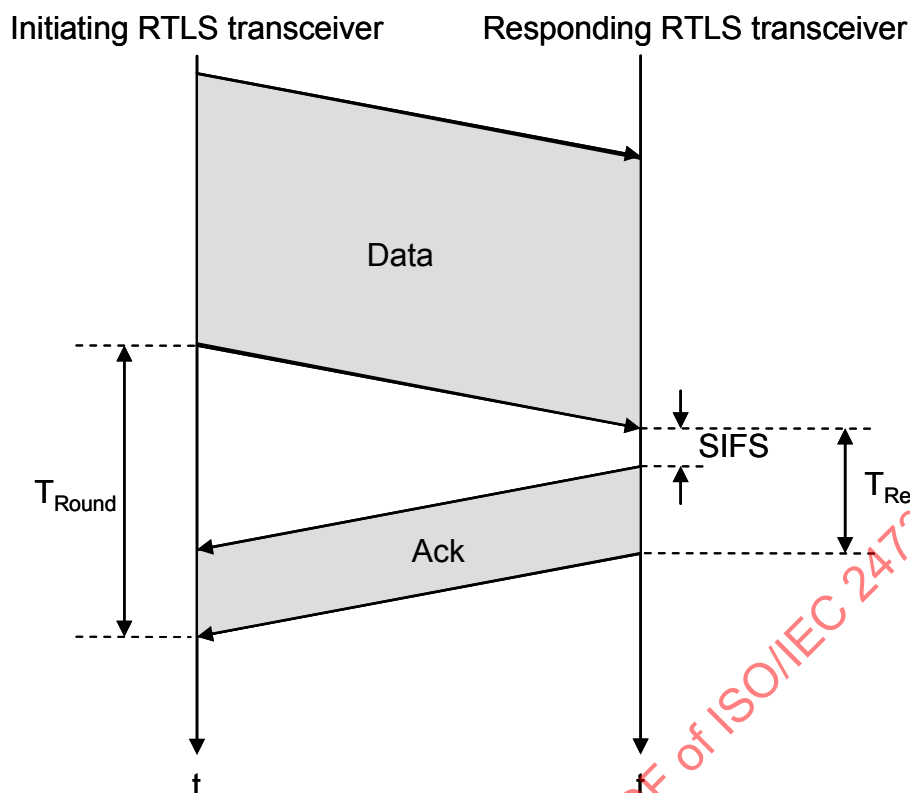


Figure 22 – Data ACK timing

### 8.5.2 3-way handshake

An RTLS transceiver shall be capable of performing 3-way handshake. 3-way handshake shall consist of the steps specified in Table 20 and depicted in Figure 23.

Table 20 – 3-way handshake procedure

Step	
1	The initiating RTLS transceiver sends the RTS packet to the responding RTLS transceiver
2	If the responding RTLS transceiver has received the RTS, the responding RTLS transceiver sends the CTS packet to the initiating RTLS transceiver
3	If the initiating RTLS transceiver has received the CTS packet, the initiating RTLS transceiver sends the Data packet to the responding transceiver.
4	If the responding RTLS transceiver has received the Data packet, the responding RTLS transceiver sends an ACK packet to the initiating RTLS transceiver

3-way handshake shall be performed according to the timing specified in Figure 23.

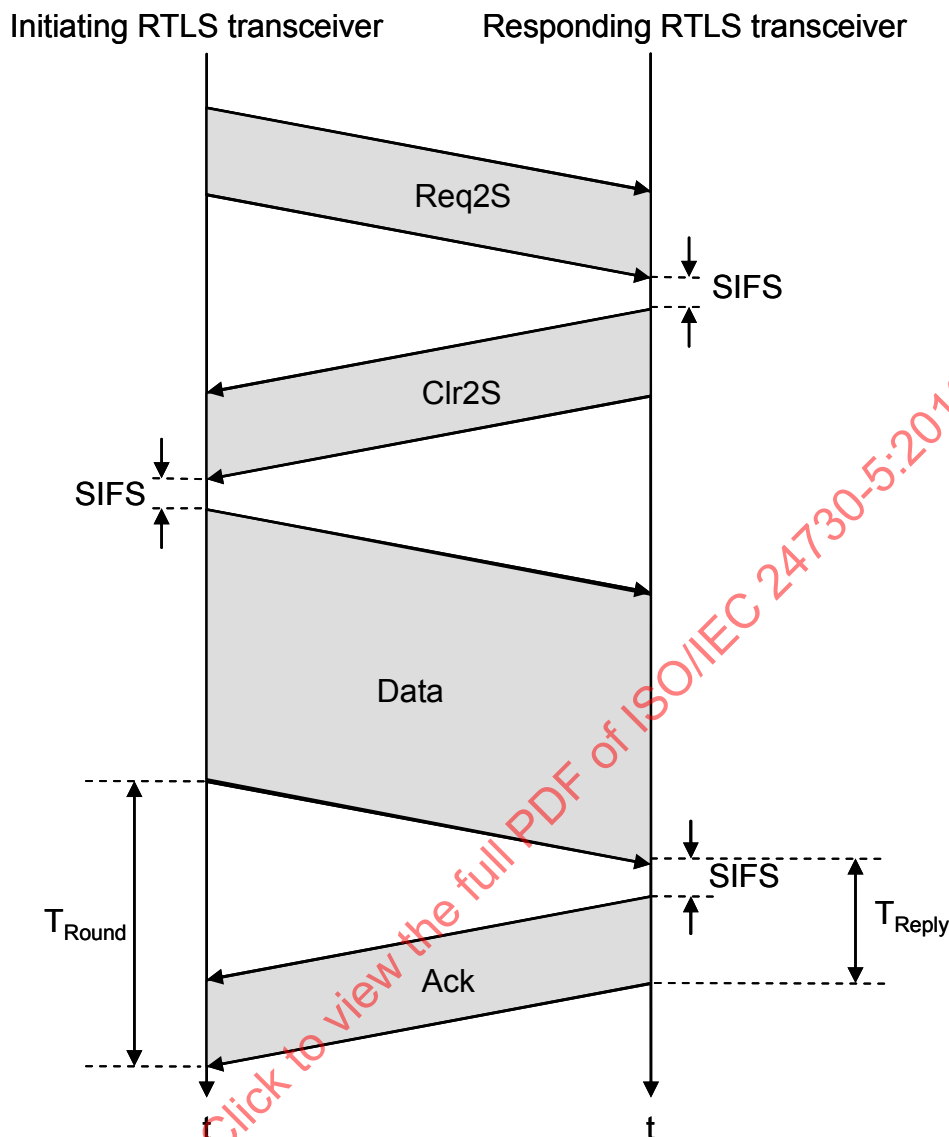


Figure 23 – RTS-CTS timing

### 8.5.3 Ranging-related time measurements

An RTLS transceiver shall be capable of measuring  $T_{\text{Reply}}$  and  $T_{\text{Round}}$  as depicted in Figure 22 and Figure 23 within the time base accuracy specified in Clause 6.

### 8.5.4 Media access

An RTLS transceiver shall support ALOHA as specified below.

An RTLS transceiver shall support CSMA/CA (Carrier Sense Multiple Access/Collision avoidance) as specified below.

#### 8.5.4.1 ALOHA

When using the ALOHA protocol, an RTLS transceiver shall immediately access the Medium without prior listening for any activity.

#### 8.5.4.2 CSMA/CA

When using CSMA/CA, an RTLS transceiver shall be able to sense whether the Medium is busy by using one or more of the following methods:

- Energy detect
- Physical carrier sense
- Virtual carrier sense

When the Medium is busy, the RTLS transceiver shall back-off, as specified in 8.5.4.2.5.

##### 8.5.4.2.1 Usage of CSMA/CA

CSMA/CA shall be applicable only for 2 way handshake and 3-way handshake.

In case of 2-way handshake CSMA/CA shall be used only for the data packets.

In case of 3-way handshake CSMA/CA shall be used only for the RTS packets.

CSMA/CA shall be used on request of the upper layer.

##### 8.5.4.2.2 Energy detect

When using energy detect, the RTLS transceiver shall sense the average power during a CIFS period. If the average power is above a threshold of  $ED_{thres}$  the Medium shall be assumed busy.  $ED_{thres}$  shall be selectable by the upper layer from [ -30 dBm, -50 dBm, -70 dBm, -90 dBm ].

##### 8.5.4.2.3 Physical carrier sense

When using physical carrier sense, an RTLS transceiver shall sense for CIFS duration before transmission. When chirp symbols are detected, the Medium shall be assumed busy.

##### 8.5.4.2.4 Virtual carrier sense

When using virtual carrier sense, an RTLS transceiver shall maintain a network allocation vector (NAV) by evaluating the type field and the length field of any incoming traffic. Thus, the NAV contains the information related to how long the Medium will be kept busy by the ongoing packet exchange as specified in Figure 24, Figure 25 and Figure 26.

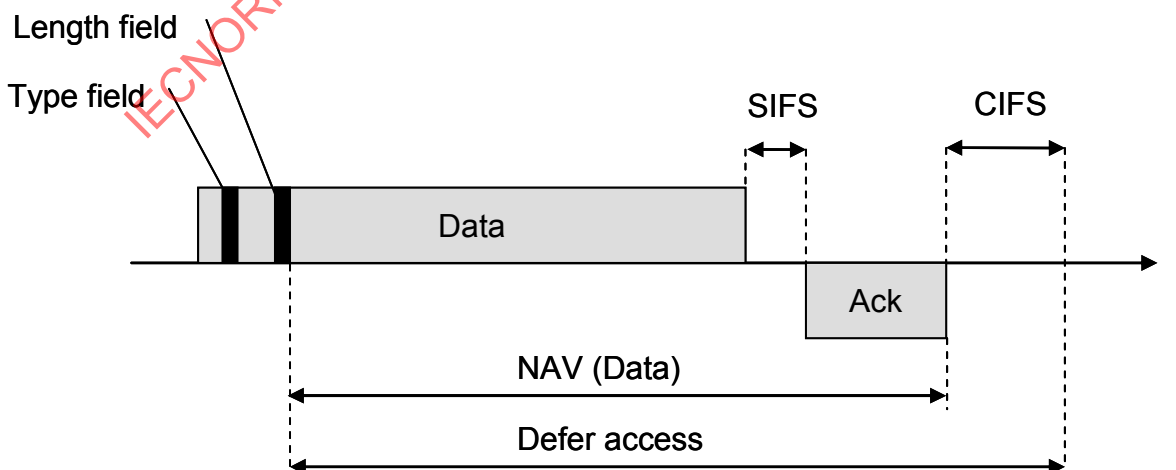


Figure 24 – NAV for Data packet

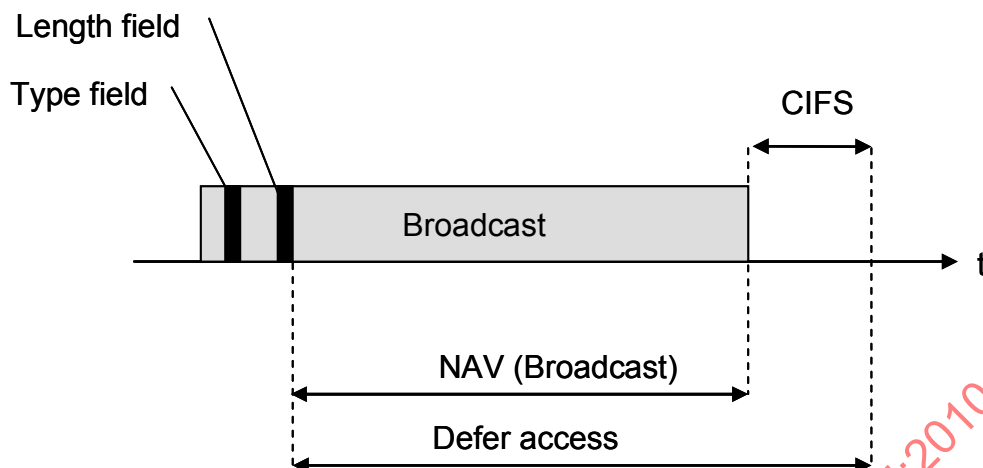


Figure 25 – NAV for Broadcast packet

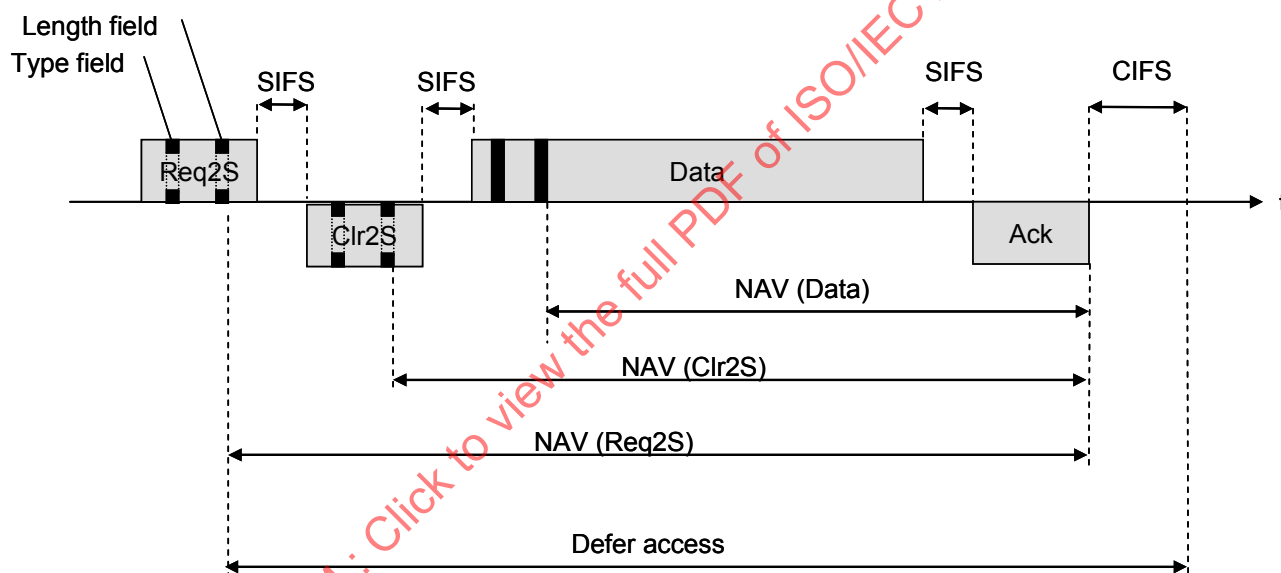


Figure 26 – NAV for 3 way handshake

#### 8.5.4.2.5 Back-off

Back-off shall be applied in combination with CSMA/CA in order to reduce the probability of collisions between transceivers that try to access the Medium when it becomes idle.

##### 8.5.4.2.5.1 Back-off procedure

If the Medium is idle, the RTLS transmitter shall wait a specific period of time of CIFS before it attempts or reattempts a transmission. This time period is the contention window and is determined by a back-off counter that counts defined time slots, each of which is a back-off time slot (BTS). This procedure is illustrated in Figure 27 and specified as follows.

While the Medium is idle, the RTLS transmitter shall decrement the back-off counter using physical carrier sensing and/or energy detect as configured by the upper layer until either the Medium again becomes busy or the back-off counter reaches 0. While the Medium is busy, the back-off counter shall be frozen until the Medium is detected idle again.

An RTLS transmitter shall access the Medium and transmit a packet only when the back-off counter has reached “0” and the Medium is idle.

When a station makes the first transmission attempt, the back-off counter shall be initialized with a pseudo-random integer in the range of 0 to 7. For every new reattempt, this range shall be doubled. For the second transmission attempt, the back-off counter shall be initialized with a pseudorandom integer in the range of 0 to 15; for the third attempt, it shall be initialized in the range of 0 to 31. If the third attempt (second retransmission) also no ACK is received, the transaction shall be considered to have failed.

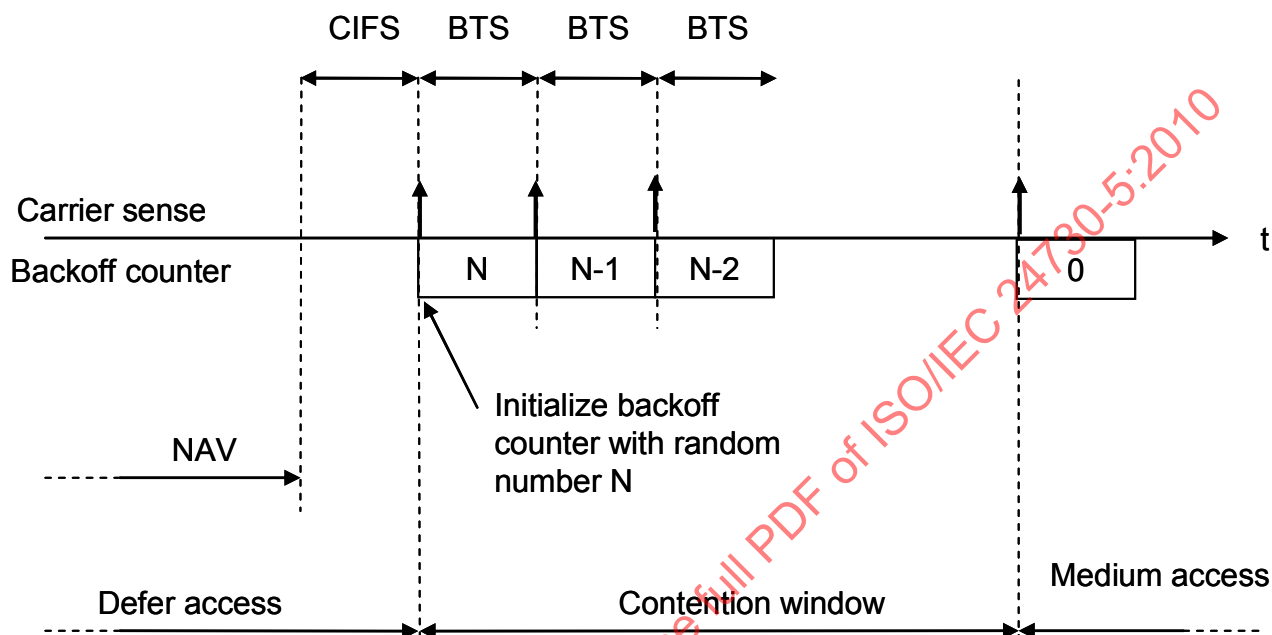


Figure 27 – CSMA/CA Back-off procedure

## 9 Tag application layer specification

### 9.1 Overview

The tag application layer is the layer above the MAC sub-layer. The tag application layers of an RTLS transceiver in a system communicate through application packets. Four types of application packets are defined:

- 1) Command packets are used by the infrastructure to transmit instructions to tags.
- 2) Report packets are used by tags to transmit any kind of information or notification to the infrastructure.
- 3) Ranging packets are used for ranging packet exchanges.
- 4) Blink packets are Broadcast packets transmitted by tags.

The tag application can be switched between certain states, as illustrated in Figure 28. A set of commands is defined through which the infrastructure can instruct the tag application to go to desired states or to go through sequences of states. Details on states and state transitions are specified in 9.2 and 9.2.6 respectively.

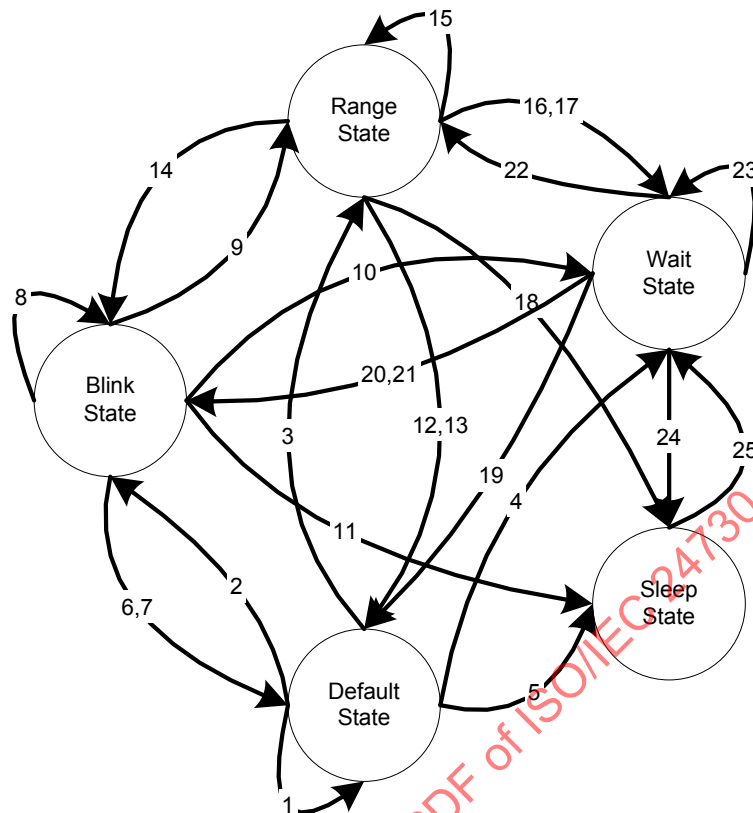


Figure 28 – Tag application states

### 9.1.1 Example scenario

The following example scenario is for illustrative purposes:

A tag is travelling through the world. Since most of the world is not equipped with any wireless infrastructure, the tag will be in the default state, which means the tag sends out broadcast packets (blinks)) on a regular basis.

When the tag enters an area equipped with an infrastructure which is compliant with this part of ISO/IEC 24730, the infrastructure will receive one or several broadcast packets (blinks). The infrastructure then instructs the tag how it should behave by sending one or several application packets, each containing one or several commands. One of these commands will be the SetRangingPeers command by which the infrastructure tells the tag which of the readers the tag should use as ranging peers. Subsequently, by sending the command SwitchState with “Ranging” as argument, the infrastructure instructs the tag to perform ranging with these peers. The tag now cyclically ranges with the peers. The number of cycles, as well as the time between these cycles, is determined by the parameters of the previous SwitchState(Ranging) command. Another parameter of the previous SwitchState(Ranging) command controls if and how the tag sends out an application report packet containing a ranging report. When the tag travels through the area covered by the infrastructure, the readers to range with might change. Thus the infrastructure will update the list of readers to range with (the list of ranging peers) using the commands SetRangingPeers and/or AddRangingPeers.

It should be noted that in order to accommodate varying numbers of tags inside the infrastructure area, the infrastructure can individually instruct each tag how often per time unit the tag shall range with its peers through the SwitchState(Ranging) command.

When the infrastructure recognizes from position changes of the tag, that the tag is about to leave the infrastructure area, a reader will instruct the tag to switch to the default state.

If the instruction to switch to the default state is missed, the tag itself will decide to switch to default state after not having received any commands for a certain time.

## 9.2 Tag application states

An RTLS tag application shall support the states listed in Table 21.

**Table 21 – Tag application states**

Name	State code
Default state	0
Blink state	1
Wait state	2
Range state	3
Sleep state	4

Reserved and user-defined tag application state codes shall be the ones defined in Table 22.

**Table 22 – Reserved and user-defined tag application states**

Status	State code
Reserved	5..8
User defined	9..15

### 9.2.1 Default state

A tag shall enter the Default state either upon receiving a SwitchState(Default) command from the infrastructure or upon having detected the "Out of infrastructure area condition" defined 9.2.1.1. Upon entering the default state, the tag shall select the default profile as specified in 9.7 and shall blink periodically as defined in 9.2.5. For parameters in the default state, see Table 42.

#### 9.2.1.1 Out of infrastructure area condition

If the tag receives no Command or ACK packets from an RTLS transceiver for a duration of  $T_{\text{Contact}}$  the tag shall assume that it has left the infrastructure area.

### 9.2.2 Wait state

A tag shall enter the Wait state upon receiving a SwitchState(Wait) command or upon waking up from Sleep state. Upon entering the Wait state the tag application shall check the latest Command packet received for any pending commands. If pending commands are found, these are executed. If, after executing the last command, no command has altered the state of the tag, the tag shall activate its receiver and listen for further instruction(s).

A tag shall leave the Wait state upon receiving a state changing command or, if after WaitMaxDuration the tag has not received any state changing command, the tag shall switch from Wait state to Blink state.

### 9.2.3 Range state

A tag shall enter the Range state upon receiving a SwitchState(Range state) command. In this state, the tag shall range X times with the RTLS transceivers specified to the tag by a previous SetRangingPeers command. For details of the parameter X, see 9.3.1.6.3.

A tag shall leave the Range state upon any state changing command.

A tag shall switch from Range state to Wait state with  $\text{WaitMaxDuration} = T_{\text{WaitAfterRange}}$  after the number of ranging cycles X is complete

### 9.2.4 Sleep state

A tag shall enter the Sleep state for a duration of X ms upon receiving the corresponding command. During the Sleep state, the tag shall deactivate its receiver and transmitter.

Upon completion of the sleep duration, the tag shall enter the Wait state for  $\text{WaitMaxDuration} = T_{\text{Waitdefault}}$ . For details of the parameter X see 9.3.1.7.1.

### 9.2.5 Blink state

A tag shall enter the Blink state upon receiving a SwitchState(Blink) command or upon detecting a timeout condition in Wait state. In blink state, a tag shall periodically transmit a burst of packets (one blink) consisting of  $N_{\text{sub}}$  identical Broadcast packets (sub blinks) as specified in Figure 29. Each  $M_{\text{Blink}}$ th blink shall consist of only one sub-blink followed by a receive window during which the receiver of the RTLS transceiver shall be activated for at least  $T_{\text{Rxon}}$ . During the receive window the tag listens for commands from any infrastructure RTLS transceiver. Any commands shall be executed immediately after the receive window has closed. In case of having received inconsistent commands from different infrastructure RTLS transceivers the commands shall be prioritized as defined in 9.3.8.

A tag shall leave the Blink state upon receiving a SwitchState command or upon detecting the “Out of infrastructure area condition” specified in 9.2.1.1.

#### 9.2.5.1 Randomization of blink time interval

The time interval between the start of subsequent blinks shall be randomized by a random offset  $T_{\text{Rand}}$ , which is added to the constant  $T_{\text{Blink}}$ .

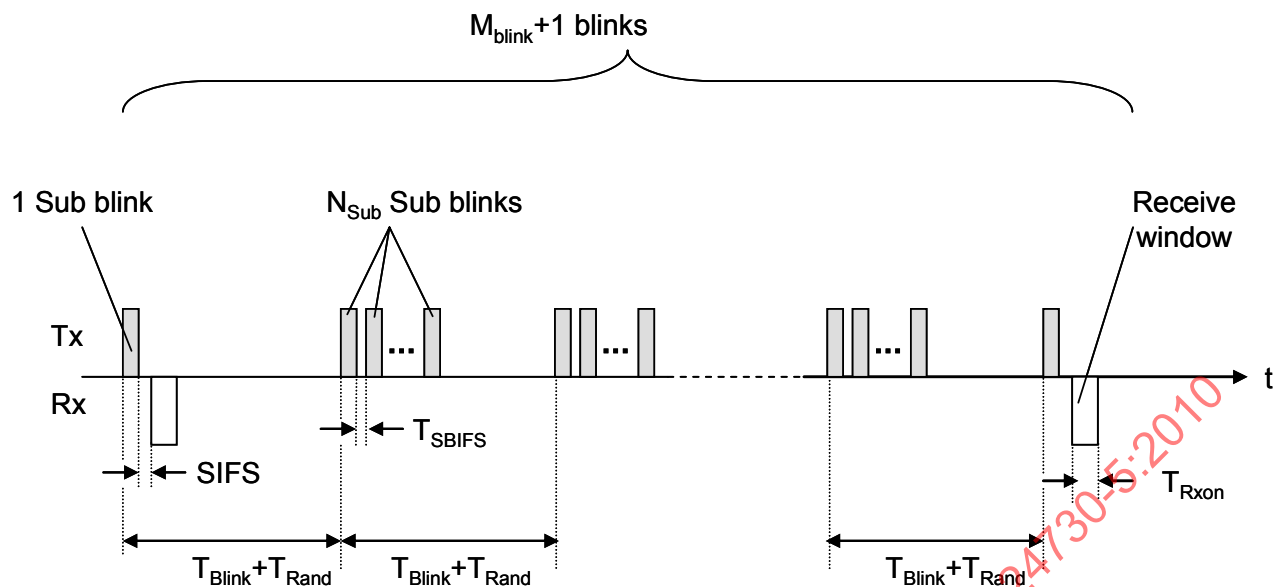


Figure 29 – Blink mode

For timing values see Table 41.

#### 9.2.6 State transitions

The possible transitions among the states defined above are summarized in Table 23. In most cases a SwitchState command will be the trigger for a state change.

Besides the SwitchState command, the following events may trigger state transitions:

1. Detection of the "Out of infrastructure area" condition: Upon this event a tag shall switch to Default state.
2. WaitMaxDuration timeout: Upon this event a tag shall switch to Blink state.
3. Sleep duration over: Upon this event a tag shall switch to Wait state
4. Ranging cycles complete: Upon this event a tag shall switch to Wait state.

Table 23 – Application layer state transitions

Previous\new	Default	Blink	Range	Wait	Sleep
<b>Default</b>	1 SwitchState(Default)	2 SwitchState(Blink)	3 SwitchState(Range)	4 SwitchState(Wait)	5 SwitchState(Sleep)
<b>Blink</b>	6 SwitchState(Default)	8 SwitchState(Blink)	9 SwitchState(Range)	10 SwitchState(Wait)	11 SwitchState(Sleep)
	7 Out of infrastructure area				
<b>Range</b>	12 SwitchState(Default)	14 SwitchState(Blink)	15 SwitchState(Range)	16 SwitchState(Wait)	18 SwitchState(Sleep)
	13 Out of infrastructure area			17 Ranging cycles complete	
<b>Wait</b>	19 SwitchState(Default)	20 SwitchState(Blink)	22 SwitchState(Range)	23 SwitchState(Wait)	24 SwitchState(Sleep)
		21 WaitMaxDuration timeout			
<b>Sleep</b>				25 After sleep duration is over	

### 9.3 Commands

Commands shall start with a command code field containing the command code as specified in Table 24.

Table 24 – Tag application commands

Command name	Command code
SwitchState	01h
SetConfigVector	02h
GetConfigVector	82h
SetRangingPeers	03h
AddRangingPeers	04h
GetRangingPeers	83h

Command codes reserved or user defined shall be the codes defined in Table 25.

**Table 25 – Reserved and user-defined tag application command codes**

Status	Command code range
Reserved	0
Reserved	05...40h
User defined	41h...7Fh
Reserved	80h
Reserved	84h...C0h
User defined	C1...FFh

### 9.3.1 SwitchState command

The SwitchState(New state) command shall be formatted as specified in Figure 30.

Command code	Reserved	New state code	Parameters
8	4	4	Format depends on New state code

**Figure 30 – SwitchState command format**

#### 9.3.1.1 New state code field

The State code field shall switch to the new state code as specified in Table 21.

#### 9.3.1.2 Parameters field

The Parameters field shall contain parameters for the new state as specified in Table 26.

**Table 26 – Switch state parameters**

New state	Parameter
Default state	-
Blink state	$T_{\text{Blink}}$ , $M_{\text{Blink}}$ , $T_{\text{Rxon}}$
Wait state	$T_{\text{WaitMaxDuration}}$
Range state	Type, report, repetition
Sleep state	Duration

#### 9.3.1.3 SwitchState(Default)

The SwitchState command with Default as argument shall be formatted as specified in Figure 31.

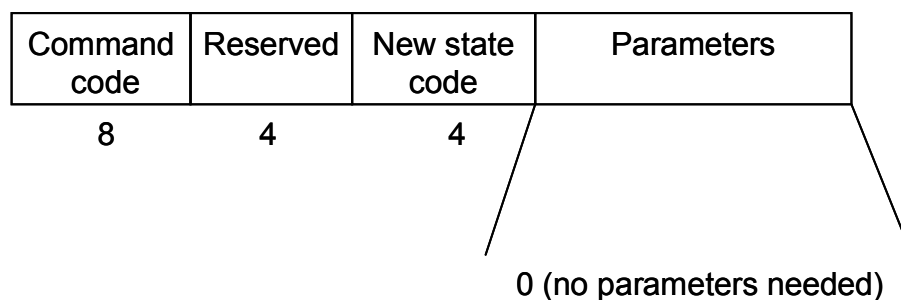


Figure 31 – SwitchState(Default)

#### 9.3.1.4 SwitchState(Blink)

The SwitchState command with Blink as argument shall be formatted as specified in Figure 32.

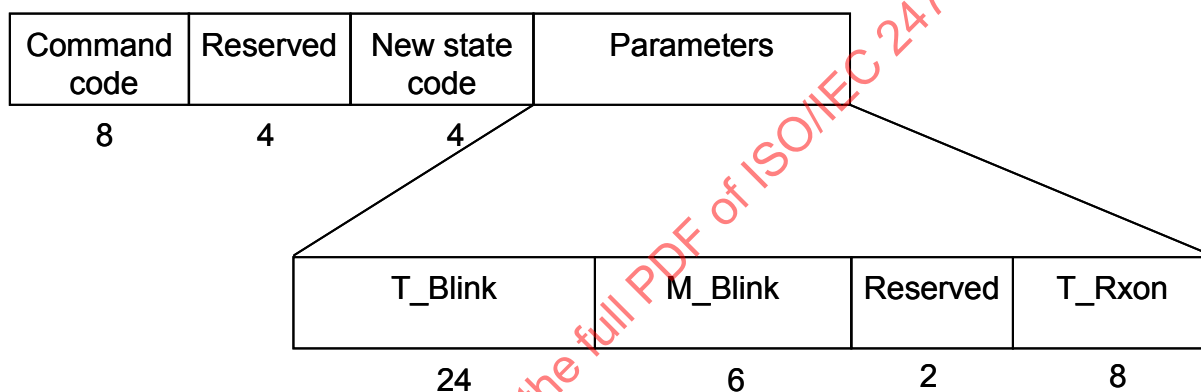


Figure 32 – SwitchState(Blink)

##### 9.3.1.4.1 T\_Blink subfield

The T\_Blink subfield shall be an unsigned 24-bit integer. It shall contain the duration parameter  $T_{\text{Blink}}$  in ms, as specified in Figure 29.

##### 9.3.1.4.2 M\_Blink subfield

The M\_Blink subfield shall be an unsigned 6-bit integer starting with the LSB. It shall contain the dimensionless parameter  $M_{\text{blink}}$  defined in 9.2.5.

##### 9.3.1.4.3 T\_Rxon subfield

The T\_Rxon subfield shall be an unsigned 8-bit integer starting with the LSB. It shall contain the duration of the receive window in ms, as described in Figure 29.

#### 9.3.1.5 SwitchState(Wait)

The SwitchState to Wait state command with Wait as argument shall be formatted as specified in Figure 33.

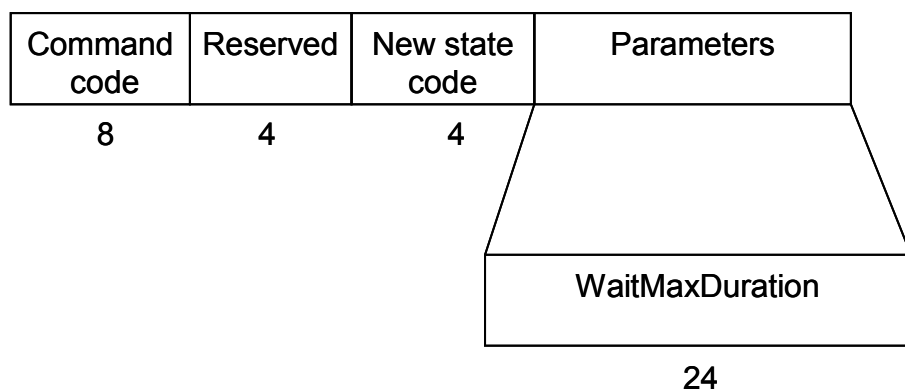


Figure 33 – SwitchState(Wait)

#### 9.3.1.5.1 WaitMaxDuration subfield

The WaitMaxDuration subfield shall be an unsigned 24-bit integer starting with the LSB. The WaitMaxDuration subfield shall contain the maximum time in ms for which the tag application layer stays in Wait state after having received the SwitchState to Wait state command. If no state changing command is received during this time, the tag application layer shall switch to default state.

#### 9.3.1.6 SwitchState(Range)

The SwitchState to Range state command with Range as argument shall be formatted as specified in Figure 34.

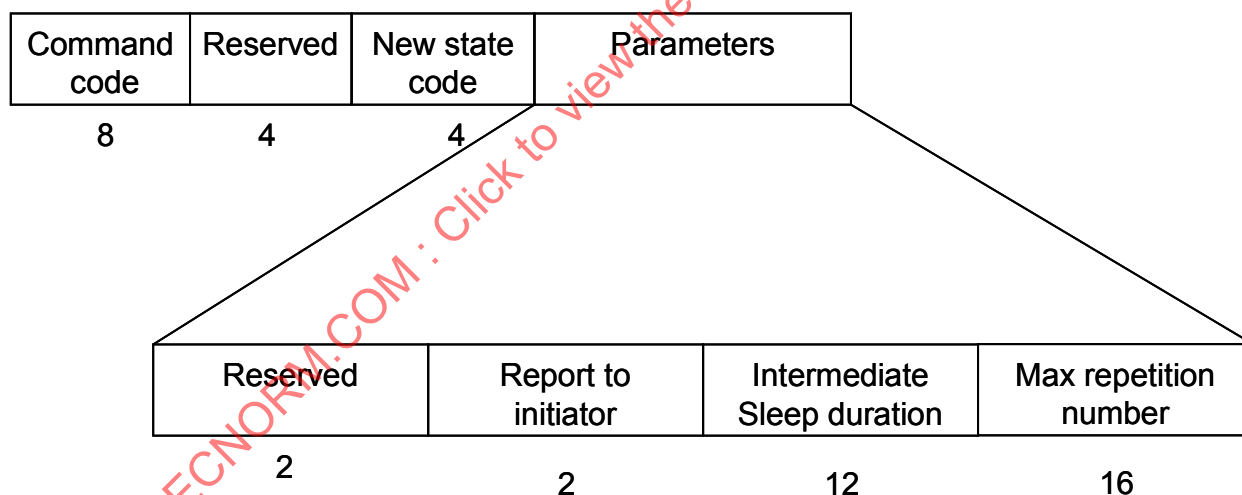


Figure 34 – SwitchState(Range)

Upon receiving the SwitchState to Range state command, the RTLS transceiver shall start ranging with the ranging peers as specified by a previous SetRangingPeers command.

##### 9.3.1.6.1 Report to initiator subfield

The Report to initiator subfield shall be a 2-bit unsigned integer starting with the LSB.

The value of 0h shall define that no ranging report will be sent.

The value 1h shall define that the RTLS transceiver sends a Ranging report to the reader that sent the last SwitchState(Range) command after performing a ranging packet exchange with each ranging peer.

The value of 2h shall identify that the RTLS transceiver sent the Ranging report as broadcast.

The value of 3h shall be reserved.

#### 9.3.1.6.2 Intermediate sleep duration subfield

The Intermediate sleep duration subfield shall be a 12-bit unsigned integer starting with the LSB.

The Intermediate sleep duration subfield shall define a time duration in ms for which the RTLS transceiver goes to sleep state after having performed a ranging packet exchange with each ranging peer and having sent its Ranging report (if requested).

#### 9.3.1.6.3 Max repetition number subfield

The Max repetition number subfield shall be a 16 bit unsigned integer starting with the LSB.

After the having stayed in Sleep state the RTLS transceiver starts again to range with all ranging peers. This process is performed as often as specified in the Max repetition number subfield.

#### 9.3.1.7 SwitchState(Sleep)

The SwitchState command with Sleep as argument shall be formatted as specified in Figure 35.

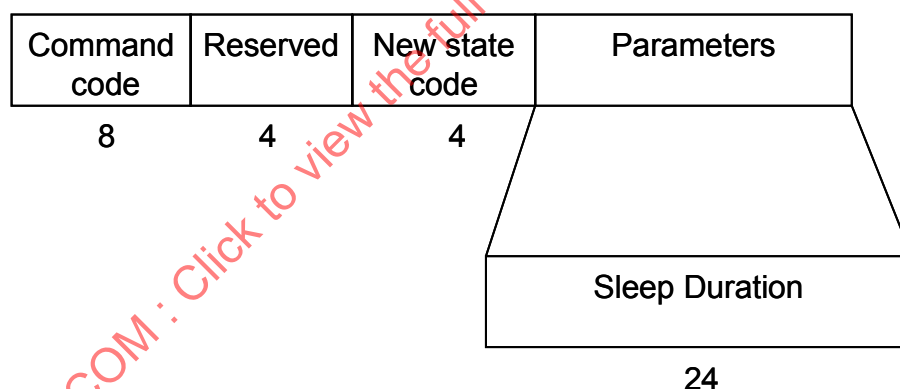


Figure 35 – SwitchState(Sleep)

#### 9.3.1.7.1 Sleep Duration subfield

The Sleep Duration subfield shall be an unsigned 24-bit integer starting with the LSB. The Sleep Duration subfield shall specify the duration in ms for which the RTLS transceiver is instructed to stay in sleep state.

### 9.3.2 SetConfigVector command

The SetConfigVector command shall be formatted as specified in Figure 36.

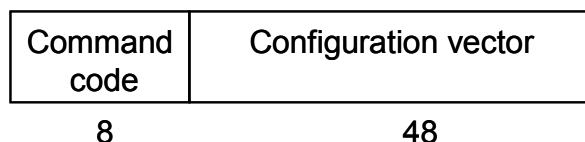


Figure 36 – SetConfigVector command format

The Configuration vector shall be defined by Table 27.

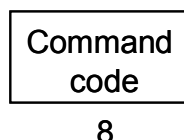
Table 27 – Configuration vector

Parameter	Bit width	Possible values
Modulation	1	0= 2-ary orthogonal CSS 1= DQPSK CSS
Bandwidth/centre frequency combination	4	0..15, according to Table 2
Data rate	1	0= 1 Mbit/s 1= 250 kbit/s
Default $T_{\text{Blink}}$	24	Unsigned integer from $[0..2^{24}-1]$
Default $M_{\text{Blink}}$	6	Unsigned integer from $[0..2^6-1]$
CSMA/CA	1	0= CSMA/CA off 1= CSMA/CA on
Carrier sense criteria: Energy detect	1	0= off 1= on
Energy detect threshold	2	Unsigned integer from $[0..3]$ 0h= -30dBm 1h= -50dBm 2h= -70dBm 3h= -90dBm
Carrier sense criteria: Physical carrier sense	1	0= off 1= on
Carrier sense criteria: Virtual carrier sense	1	0= off 1= on
3 way handshake	1	0= 3 way handshake off 1= 3 way handshake on
Reserved	1	
$T_{\text{WaitAfterRange}}$	4	Unsigned integer from $[0..15]$
Sub-chirp sequence for DQPSK-CSS	2	Unsigned integer from $[0..3]$ 0= sub-chirp sequence I 1= sub-chirp sequence II 2= sub-chirp sequence III 3= sub-chirp sequence IV
Reserved	6	

After start of execution of a SetConfigVector command, the RTLS transceiver shall have updated the configuration within 25 ms.

### 9.3.3 GetConfigVector command

The GetConfigVector command shall be formatted as specified in Figure 37.

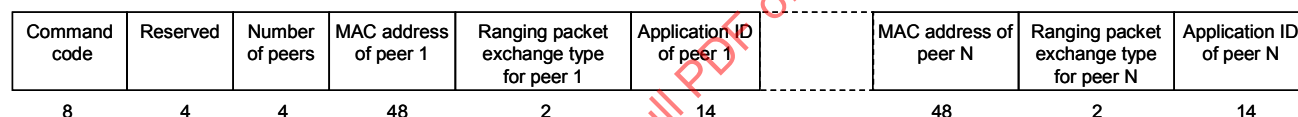


**Figure 37 – GetConfigVector command format and GetRangingPeers command format**

Upon receiving a GetConfigVector command, the RTLS transceiver shall respond with the corresponding report packet within  $T_{\text{TimeoutApplication}}$ .

### 9.3.4 SetRangingPeers command

The SetRangingPeers command shall be formatted as specified in Figure 38.



**Figure 38 – SetRangingPeers command format, AddRangingPeers command format**

The SetRangingPeers shall be used by the infrastructure to instruct a tag with which peers to exchange ranging packets upon subsequent "SwitchState to Range state commands".

Upon reception of the SetRangingPeers command, a tag shall overwrite the internal list of ranging peers.

### 9.3.5 AddRangingPeers command

The AddRangingPeers command shall be formatted as specified in Figure 38.

The AddRangingPeers command shall be used by the infrastructure to instruct a tag to add peers to the list of peers to exchange ranging packets upon subsequent "SwitchState to Range state commands".

Upon reception of the AddRangingPeers command, a tag shall add the peers to its internal list of peers to range with (ranging peer list). If the size of the ranging peer list, which a tag can support is exceeded, the first entry of the list shall be removed and a new entry shall be appended. A tag shall support a list of up to 15 ranging peers.

#### 9.3.5.1 MAC address of peer X subfield

The MAC address of peer X subfield shall contain the 48-bit address of peer X.

#### 9.3.5.2 Ranging packet exchange type for peer X

The "Ranging packet exchange type for peer X" subfield shall be a 2 bit unsigned integer starting with the LSB. As specified in Table 28, its value plus one shall determine which of the four defined ranging packet exchanges shall be used with this ranging peer.

Table 28 – Ranging packet exchange type subfield values

Value of Ranging packet exchange type subfield	Ranging packet exchange type
0h	1
1h	2
2h	3
3h	4

### 9.3.5.3 Application ID of peer X subfield

The Application ID of the peer X subfield shall contain a 14-bit ID of the peer which has been determined by the infrastructure.

NOTE The purpose of the ID is to identify the ranging peer in the ranging report (see 9.4.6).

### 9.3.6 GetRangingPeers command

The GetRangingPeers command shall be formatted as specified in Figure 37.

Upon receiving a GetConfigVector command, the RTLS transceiver shall respond with the corresponding report packet within  $T_{\text{TimeoutApplication}}$ .

### 9.3.7 User defined command

Any user-defined command shall be formatted as specified in Figure 39.

Command code	Number of octets	User defined
8	8	8 * Number of octets

Figure 39 – User defined command

#### 9.3.7.1 Number of octets subfield

The Number\_of\_octets subfield shall contain the remaining number of octets of the command.

### 9.3.8 Command prioritization

In case that a tag while it has activated its receiver, receives different commands from different infrastructure RTLS transceivers the following rules shall apply.

1) Any commands that do not change the configuration vector or the state of the tag application (e.g. SetRangingPeers, AddRangingPeers) shall be executed upon reception.

2) Any commands that might change the configuration vector or the state of the tag application shall only be executed from the RTLS transceiver which sent the command with the highest priority according to Table 29. The commands shall be executed following the order in which the commands are given in the selected application command packet.

**Table 29 – Application command priorities**

Command	Priority
SetConfig	7
GetConfig	6
Get RangingPeers	5
SwitchState(Sleep)	4
SwitchState (Wait)	3
SwitchState (Range)	2
SwitchState (Blink or Default)	1

3) In the case that a tag, while it has activated its receiver, receives commands with identical priorities from different infrastructure RTLS transceivers, the tag shall consider only the latest received command packet.

#### 9.4 Tag application packet formats

The tag application layer shall support the four application packet types listed in Table 30.

**Table 30 – Tag application packet types**

Application ranging packet type	Application control code
Ranging	1h
Command	2h
Report	3h
Blink	4h

Reserved and user-defined tag application packet type codes shall be those specified in Table 31.

**Table 31 – Reserved and user defined tag application packet types**

Status	Application control code
Reserved	0h
User defined	5h...7h

The type of a tag application packet shall be indicated by the application control code placed in the Ctrl field of the MAC frame.

### 9.4.1 Application blink packet

The application blink packet shall be a Broadcast packet formatted as specified in Figure 40.

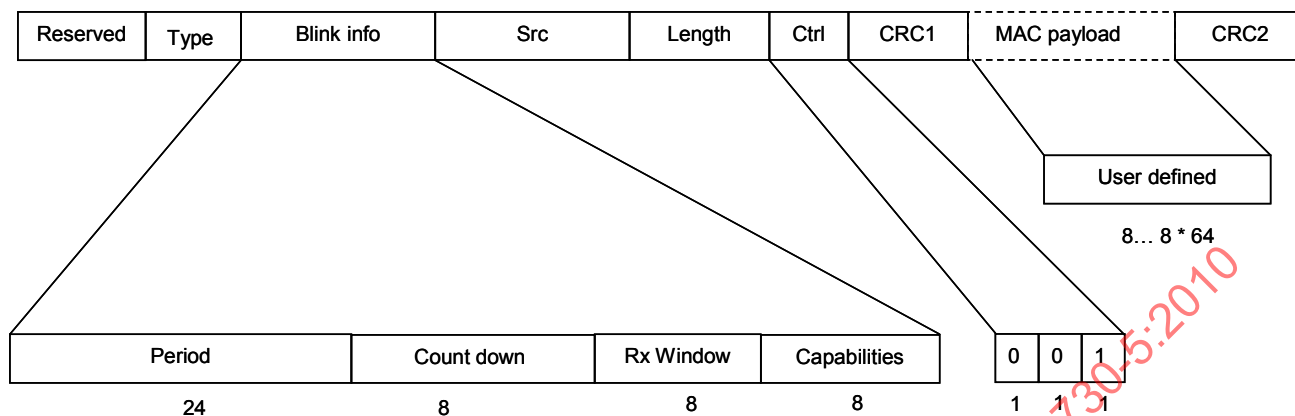


Figure 40 – Blink packet

#### 9.4.1.1 Period subfield

The Period subfield shall contain the value of  $T_{\text{blink}}$  in ms, as an unsigned integer starting with the LSB.

#### 9.4.1.2 Count down subfield

The Count down subfield shall contain the remaining number of blinks after which the tag will turn on its receiver. A value of zero shall indicate that immediately after this sub blink the RTLS transceiver will activate its receiver for a duration of at least  $T_{\text{Rxon}}$ . This subfield shall be an unsigned integer, starting with the LSB.

#### 9.4.1.3 Rx Window subfield

The Rx Window subfield shall contain the width of the receive window in ms as an unsigned integer starting with the LSB.

#### 9.4.1.4 Capabilities subfield

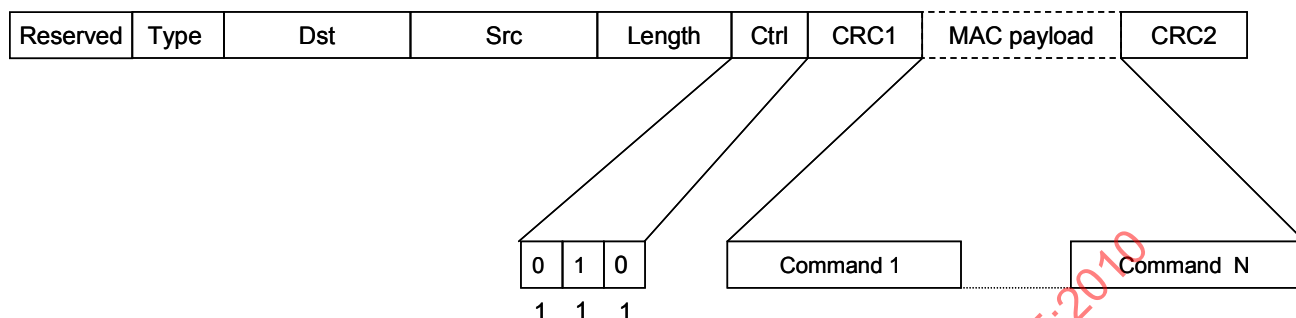
The Capabilities subfield shall contain a bit field starting with bit 0, indicating the capabilities of the tag as specified in Table 32.

Table 32 – Capabilities subfield

Bit number	Capability	Usage
0	DQPSK-CSS for communication	1 = capability available 0 = capability <b>not</b> available
1	DQPSK-CSS for ranging	1 = capability available 0 = capability <b>not</b> available
2	$T_{\text{Blink}}$ overwritable	1 = capability available 0 = capability <b>not</b> available 0
3	$M_{\text{Blink}}$ overwritable	1 = capability available 0 = capability <b>not</b> available 0
4,5,6,7	Reserved	0

### 9.4.2 Application command packet

An application command packet shall be formatted as specified in Figure 41.



**Figure 41 – Command packet format**

It shall be possible to place one or several commands in the MAC payload of a Data packet. Commands shall be executed by the tag application layer strictly in the order in which they are received.

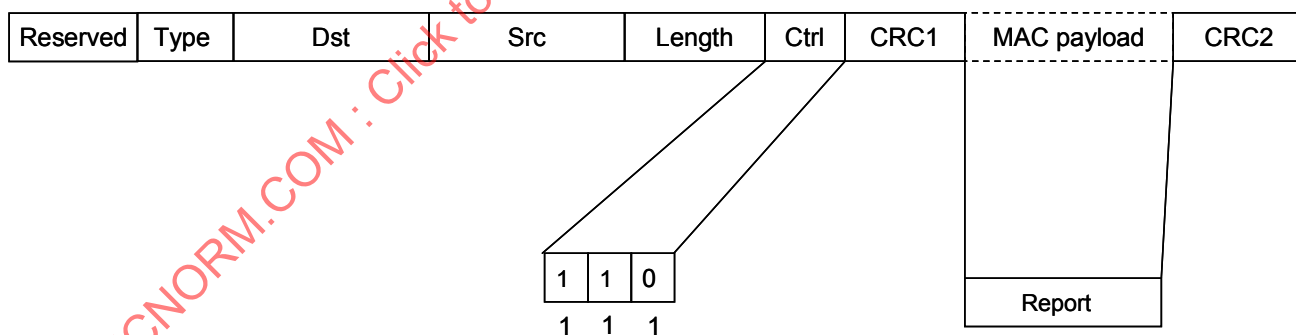
The length of the MAC payload of an application command packet shall not exceed 128 octets.

**NOTE** As mentioned in 9.1, it is anticipated that command packets are transmitted from infrastructure stations to tags. Thus, it is the responsibility of the developer of the infrastructure to choose commands or sequences of commands appropriate for his purpose.

### 9.4.3 Application report packet

Application report packets shall be transmitted by tags in response to Get commands.

A report packet shall be formatted as specified in Figure 42.



**Figure 42 – Report packet format**

The tag application reports listed in Table 33 shall be supported.

**Table 33 – Application report codes**

Report name	Report code
Ranging report	81h
GetConfigVector report	82h
GetRangingPeers report	83h

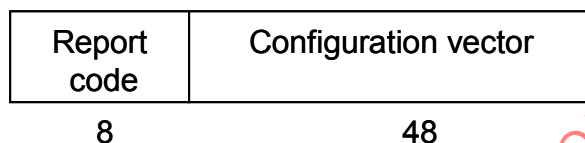
Reserved and user-defined application report codes shall be those specified in Table 34

**Table 34 – Reserved and user defined application report codes**

Status	Report code
Reserved	80h
Reserved	84h...FFh
User defined	0h...7Fh

#### 9.4.4 GetConfigVector report

The GetConfigVector report shall be formatted as specified in Figure 43.

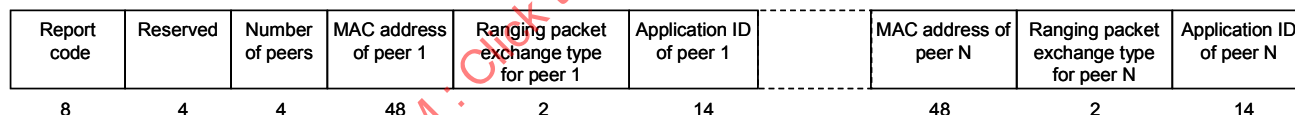


**Figure 43 – GetConfigVector report**

The Configuration vector shall be defined by Table 27.

#### 9.4.5 GetRangingPeers report

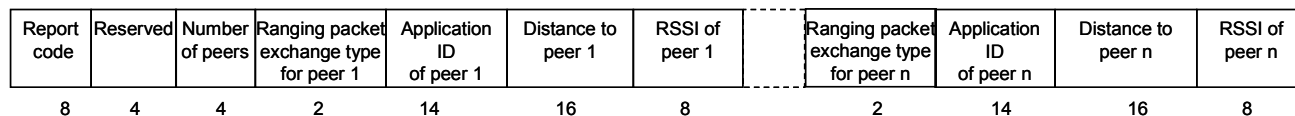
The GetRangingPeers report shall be formatted as specified in Figure 44. The subfields MAC address of peer, Ranging packet exchange type for peer and Application ID of peer shall correspond with the subfields defined in 9.3.4.



**Figure 44 – GetRangingPeers report format**

#### 9.4.6 Ranging report

The Ranging report shall be formatted as specified in Figure 45. The subfields Ranging packet exchange type for peer and Application ID of peer shall correspond with the subfields defined in 9.3.4.



**Figure 45 – Ranging report format**

#### 9.4.6.1 Distance to peer subfield

The Distance to peer subfield shall be a 16-bit signed integer starting with the LSB. Values equal or greater than 0 shall contain an estimate of the distance to the peer in decimetres. This means that one integer unit corresponds to 1E-1 meters. Any negative value shall indicate that no ranging result is available. Negative values shall be available for user-defined error codes.

#### 9.4.6.2 RSSI of peer subfield

The RSSI of peer subfield shall be an 8-bit signed integer starting with the LSB. Values equal or smaller than zero shall be a measurement in dBm within +6 dBm of the signal strength with which packets of the peer have been received. Values larger than zero shall be user-defined.

#### 9.4.7 Application ranging packet

An application ranging packet shall contain an RTLS code and a ranging packet code as specified in Table 35.

**Table 35 – Ranging packet codes**

Application ranging packet type	RTLS control code	Ranging packet code
T1R1	1h	1h
T1R2	1h	2h
T1R3	1h	3h
T2R1	1h	4h
T2R2	1h	5h
T2R3	1h	6h
T3R1	1h	7h
T3R2	1h	8h
T4R1	1h	9h
T4R2	1h	Ah

Reserved and user-defined ranging packet codes shall be those specified in Table 36.

**Table 36 – Reserved and user defined ranging packet codes**

Status	RTLS control code	Ranging packet code
Reserved	1h	0h
User defined	1h	80h...FFh

#### 9.4.7.1 Application ranging packet fields

An application ranging packet may contain one or more of the following fields.

#### 9.4.7.2 Ranging packet code field

The ranging packet code field shall be an unsigned 8-bit integer value as specified in Table 35, starting with the LSB.

9.4.7.3 Treply field

The Treply field shall be an unsigned 24-bit integer value starting with the LSB. It shall contain a time value in units of 0.1 ns, meaning that one integer unit corresponds to 1E-10 seconds.

9.4.7.4 Tround field

The Tround field shall be an unsigned 24-bit integer value starting with the LSB. It shall contain a time value in units of [0.1 ns].

9.4.7.4.1 Type 1, ranging 1 (T1R1)

The T1R1 packet shall be formatted as specified in Figure 46.

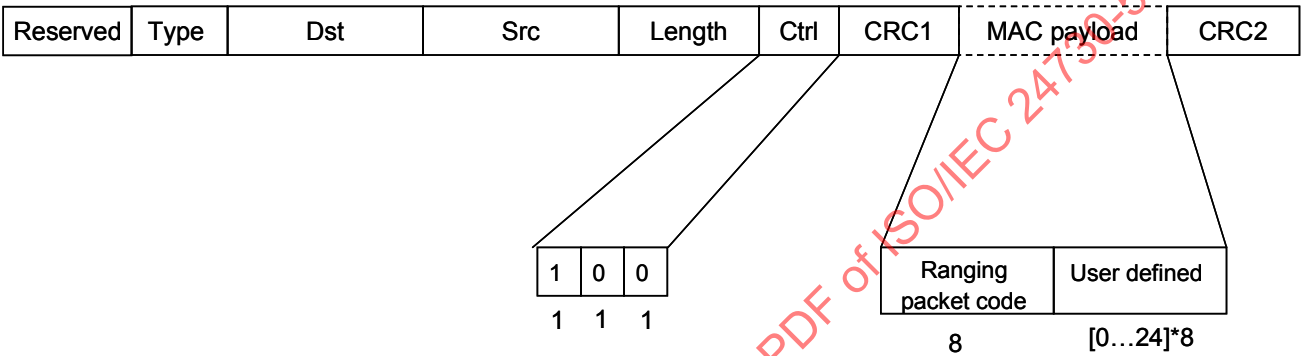


Figure 46 – Ranging packet T1R1

9.4.7.4.2 Type 1, ranging 2 (T1R2)

The T1R2 packet shall be formatted as specified in Figure 47.

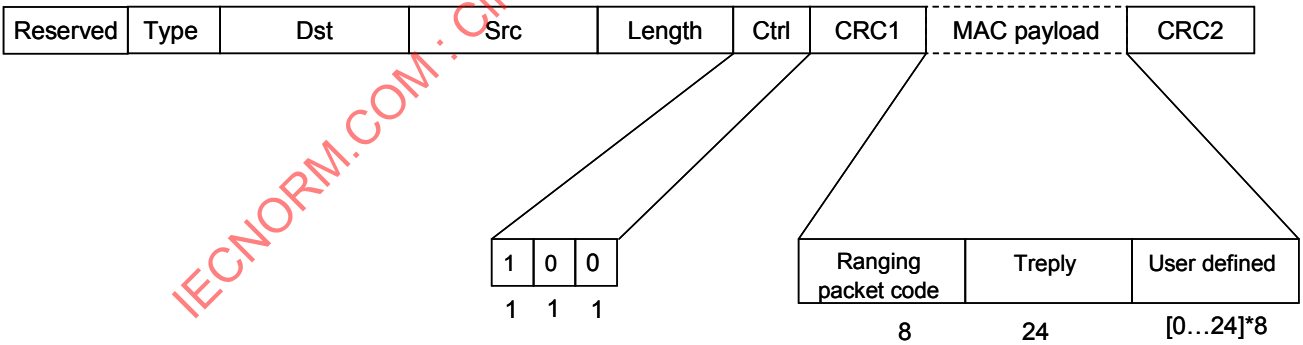


Figure 47 – Ranging packet T1R2

#### 9.4.7.4.3 Type 1, ranging 3 (T1R3)

The T1R3 packet shall be formatted as specified in Figure 48.

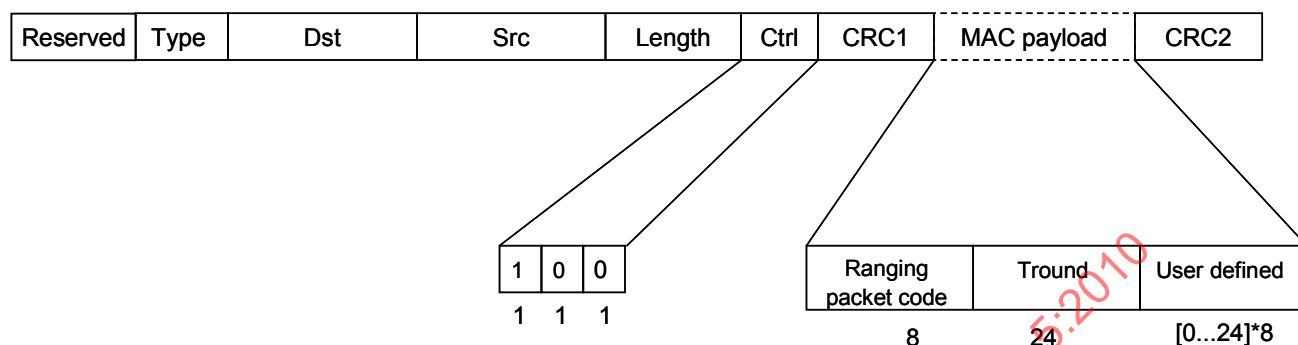


Figure 48 – Ranging packet T1R3

#### 9.4.7.4.4 Type 2, ranging 1 (T2R1)

The T2R1 packet shall be formatted as specified in Figure 49.

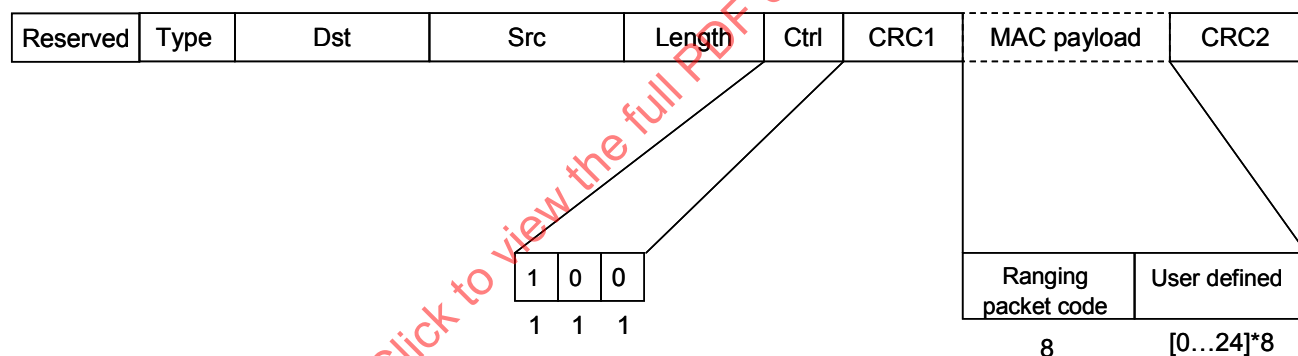


Figure 49 – Ranging packet T2R1

#### 9.4.7.4.5 Type 2, ranging 2 (T2R2)

The T2R2 packet shall be formatted as specified in Figure 50.

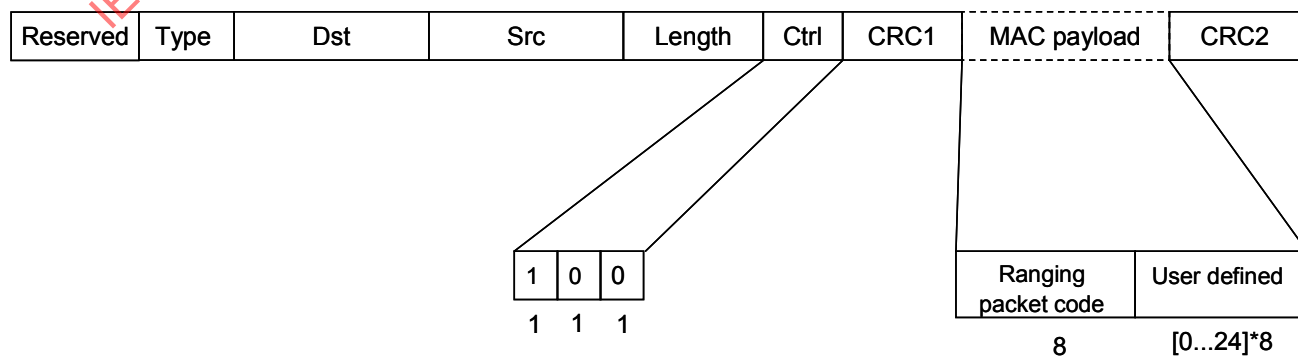


Figure 50 – Ranging packet T2R2

#### 9.4.7.4.6 Type 2, ranging 3 (T2R3)

The T2R3 packet shall be formatted as specified in Figure 51.

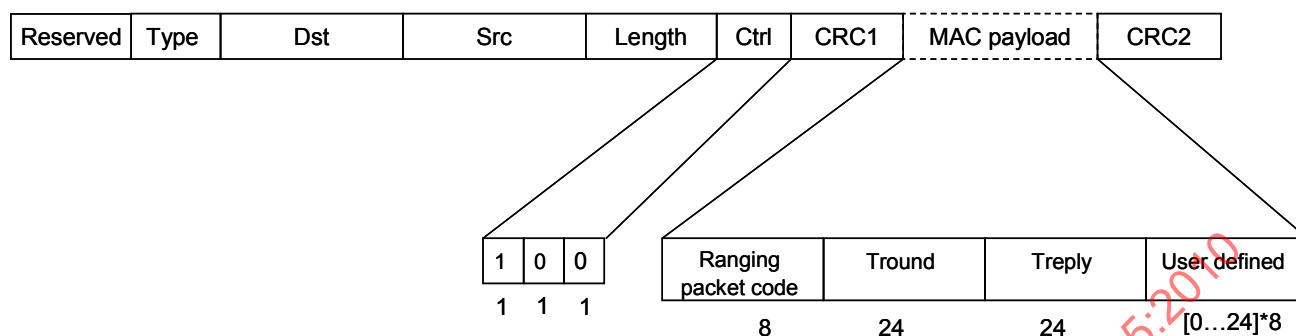


Figure 51 – Ranging packet T2R3

#### 9.4.7.4.7 Type 3, ranging 1 (T3R1)

The T3R1 packet shall be formatted as specified in Figure 52.

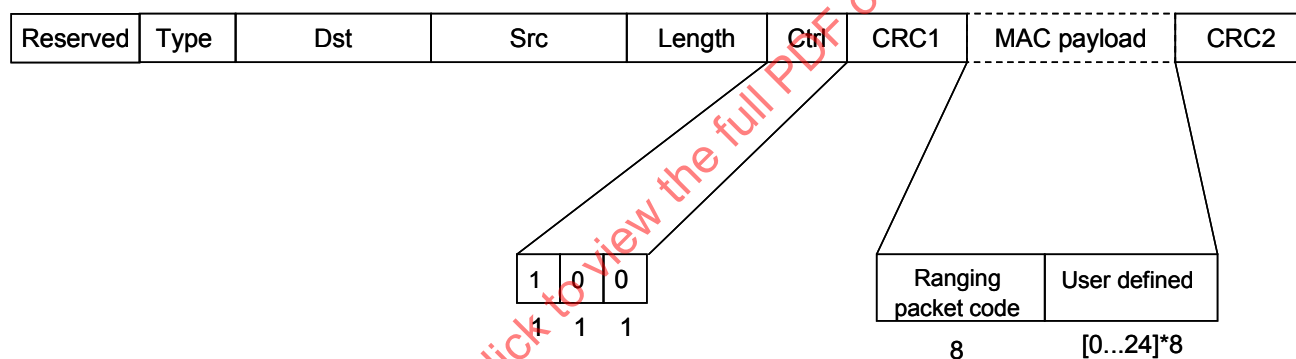


Figure 52 – Ranging packet T3R1

#### 9.4.7.4.8 Type 3, ranging 2 (T3R2)

The T3R2 packet shall be formatted as specified in Figure 53.

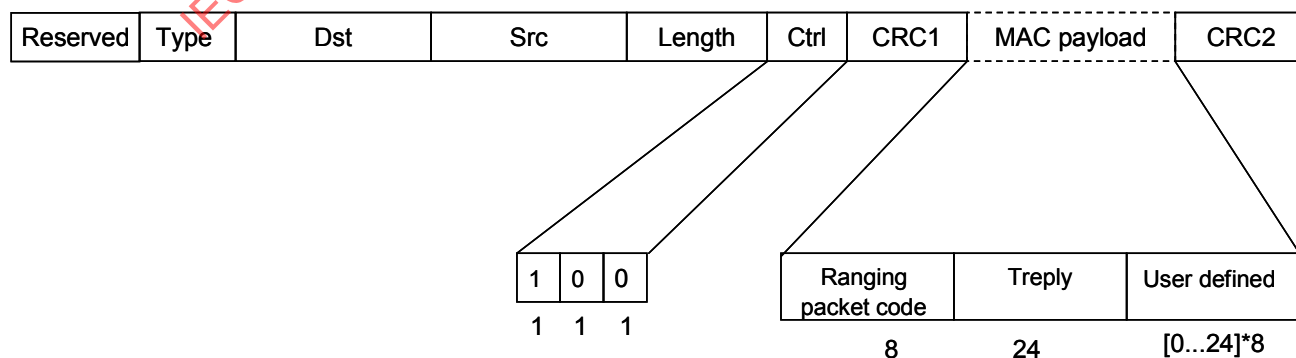


Figure 53 – Ranging packet T3R2

#### 9.4.7.4.9 Type 4, ranging 1 (T4R1)

The T4R1 packet shall be formatted as specified in Figure 54.

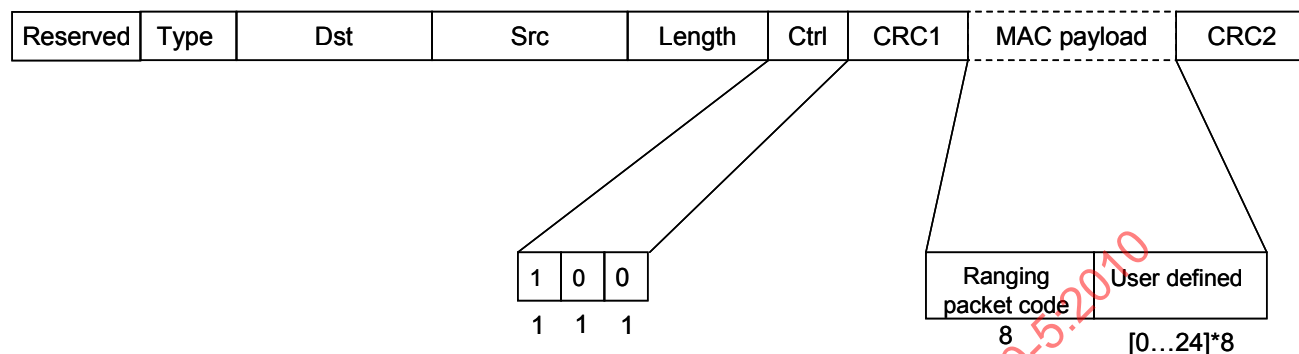


Figure 54 – Ranging Packet T4R1

#### 9.4.7.4.10 Type 4, ranging 2 (T4R2)

The T4R2 packet shall be formatted as specified in Figure 55.

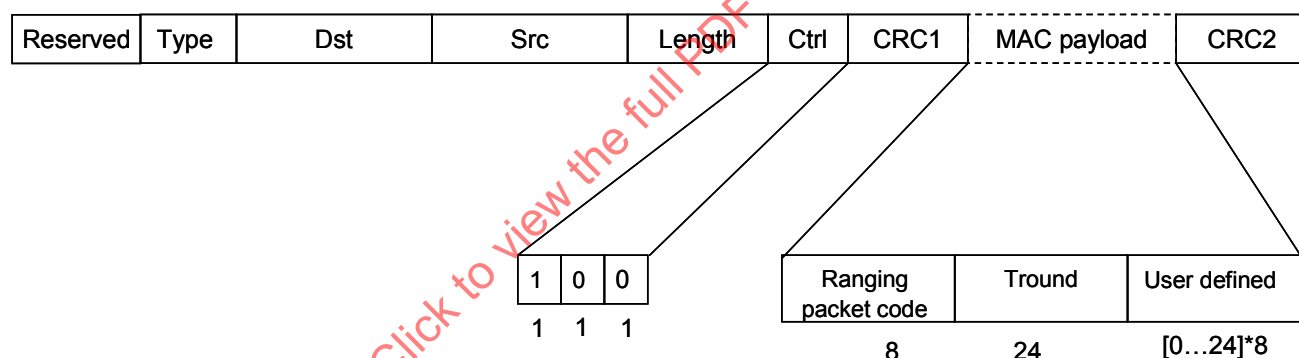


Figure 55 – Ranging packet T4R2

### 9.5 Ranging packet exchanges

Four types of ranging packet exchanges between two RTLS transceivers shall be supported. The four types differ in the number of packets exchanged and in which of the two RTLS transceivers have all information required to calculate the distance between the RTLS transceivers after completion of the ranging packet exchange.

#### 9.5.1 Ranging packet exchange type 1

Ranging packet exchange type 1 shall consist of the three steps specified in Table 37 and depicted in Figure 56. After completion of this ranging packet exchange, RTLS transceiver A has all of the information required to calculate the distance between the two RTLS transceivers.