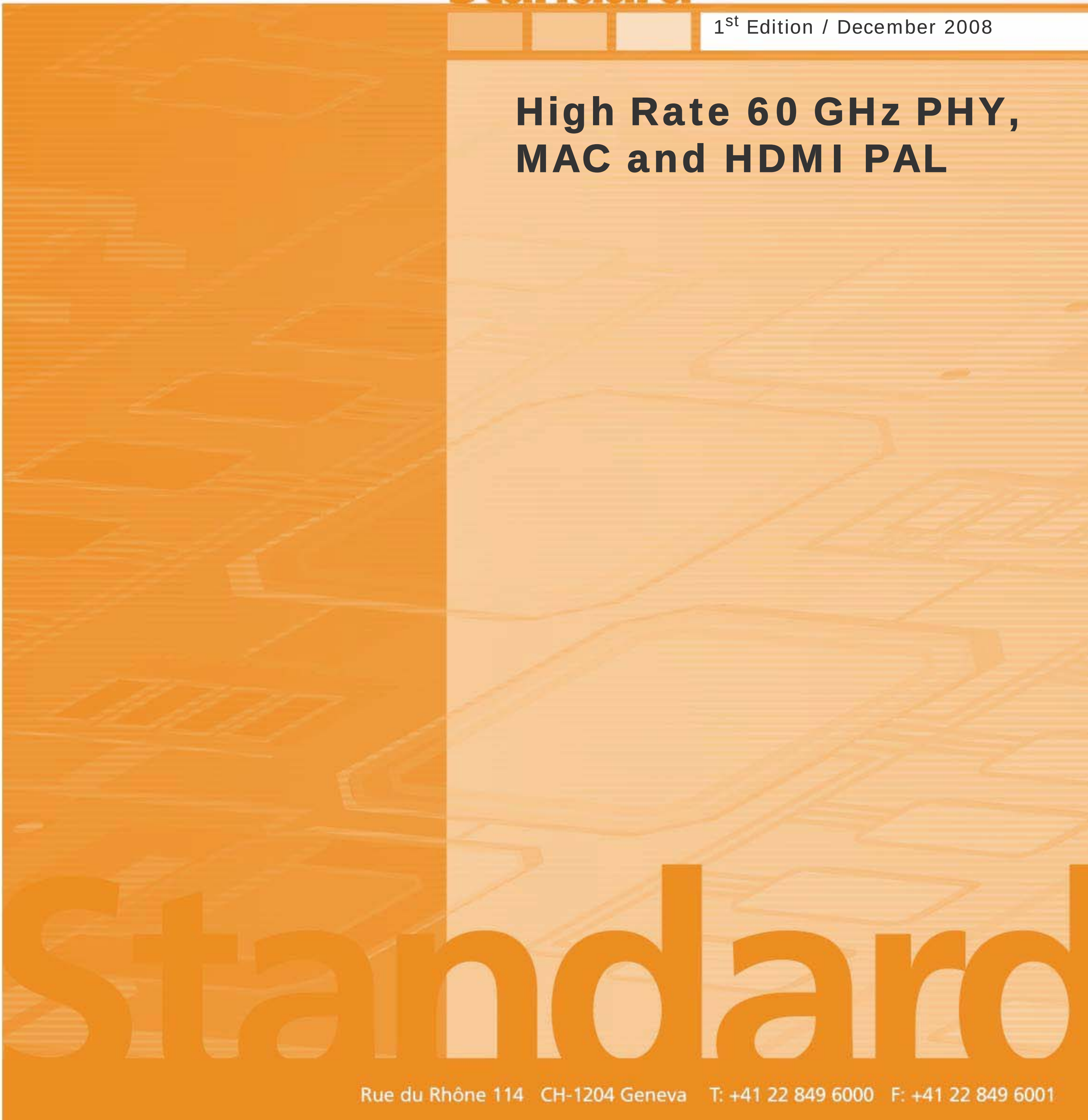
Standard ECMA-387

1st Edition / December 2008

High Rate 60 GHz PHY, MAC and HDMI PAL



Rue du Rhône 114 CH-1204 Geneva T: +41 22 849 6000 F: +41 22 849 6001

Standard
ECMA-387

1st Edition - December 2008

High Rate 60 GHz PHY, MAC and HDMI PAL

Table of contents

Introduction

This is a standard for a 60 GHz PHY, MAC and HDMI PAL for short range unlicensed communications providing high rate wireless personal area network (including point-to-point) transport for both bulk data transfer and multimedia streaming; addressing usages and applications such as high definition (uncompressed / lightly compressed) AV streaming, access point, wireless docking station, and short range sync-and-go.

The standard defines three device types. All device types coexist and interoperate with each other. None of the types requires presence of another type for its operation.

- Device Type A offers video streaming and WPAN applications in 10 meter range LOS/NLOS multipath environments. It uses high gain trainable antennas. This device type is considered as the 'high end - high performance” device.
- Device Type B offers video and data applications over shorter range (1-3 meters) point to point LOS links with non-trainable antennas. It is considered as the “economy” device and trades off range and NLOS performance in favour of low cost implementation and low power consumption.
- Device Type C is positioned to support data only applications over point to point LOS links at less than 1 meter range with non-trainable antennas and no QoS guarantees. This type is considered a 'bottom end' device providing simplest implementation, lowest cost and lowest power consumption.

Four frequency channels are defined and used by all the three device types. The frequency separation for these channels is 2.160 GHz and the symbol rate on each channel is 1.728 Gsps. All device types follow the same frequency plan defined in the major regulatory domains. The standard supports bonding of two or three adjacent channels. The channel bonding allows achieving higher data rates, or the same data rates while using smaller, more efficient constellations.

A single MAC layer protocol is defined within which Type B and Type C devices support limited functionality supported by their respective PHY layers as illustrated in Figure 1. A multiplexing sublayer (MUX) is defined to enable the coexistence of concurrently active higher layer protocols within a single device.

There is an HDMI PAL defined in the standard which places the 60 GHz wireless solution between the HDMI source and HDMI sink. In the wireless HDMI transmitter, the HDMI TMDS coding is removed prior to wireless transmission. The three HDMI data types (video, control and data) are multiplexed together with the video flagged for unequal error protection (UEP) and data/control flagged for equal error protection (EEP). In the wireless HDMI receiver, the three HDMI data types are demultiplexed prior to reapplying the TMDS coding and sending on to the HDMI data sink.

This Ecma Standard has been adopted by the General Assembly of December 2008.

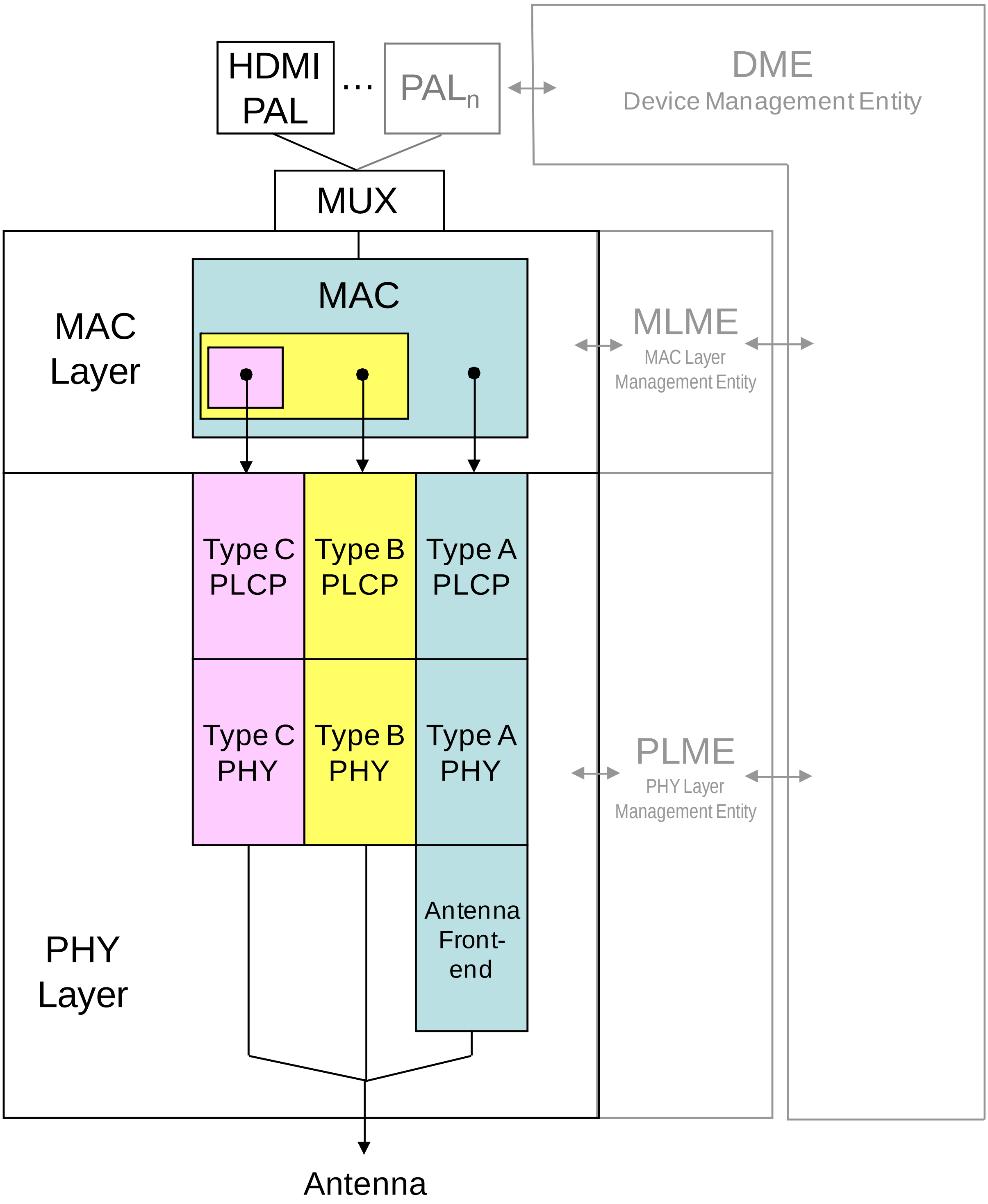


Figure 1 - Protocol structure

NOTE: The DME, MLME, PLME, and PALs (except the HDMI PAL) are outside the scope of this standard and all references to these are informative.

1 Scope

This Standard specifies a physical layer (PHY), distributed medium access control (MAC) sublayer, and an HDMI protocol adaptation layer (PAL) for 60 GHz wireless networks.

2 Conformance

Conforming devices of Type A, B or C shall implement the MAC sublayer and the PHY layer and may implement the HDMI PAL as specified herein.

3 References

This Standard shall be used in conjunction with the following publications. At the time of publication, the editions indicated were valid. All Standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the Standards listed below.

ISO/IEC 18033-3:2005, Information technology - Security techniques - Encryption algorithms - Part 3: Block ciphers

ISO/IEC 8802-11 Amendment 6, "Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications - Amendment 6: Medium Access Control (MAC) Security Enhancements", Second Edition, 2006

ISO/IEC 10646:2003, Information technology -- Universal Multiple-Octet Coded Character Set (UCS)

4 Definitions

For the purposes of this Standard, the following terms and definitions apply. For terms not defined in this Clause, please consult IEEE100, The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition.

4.1 Beacon group (BG)

The set of devices from which a device receives beacons that identify the same beacon period start time (BPST) as the device.

4.2 Beacon period (BP)

The period of time declared by a device during which it sends or listens for beacons.

4.3 Beacon period start time (BPST)

The start of the beacon period.

4.4 Channel

The medium over which cooperating entities exchange information.

4.5 Data integrity

The assurance that the data has not been modified from its original form.

4.6 Device

An entity containing an implementation of this Standard.

4.7 Distributed reservation protocol (DRP)

A protocol implemented in each device to support negotiation and maintenance of channel time reservations binding on all neighbours of the reservation participants.

4.8 Equivalent isotropic radiated power (EIRP):

Effective Isotropic Radiated Power is the amount of power that a theoretical isotropic antenna (that evenly distributes power in all directions) would emit to produce the peak power density observed in the direction of maximum antenna gain.

4.9 Equivalent isotropic received power (EIRxP):

Effective Isotropic Received Power is the amount of power that a theoretical isotropic antenna (that evenly receives power in all directions) would receive.

4.10 Extended beacon group

The union of a device's beacon group and the beacon groups of all devices in the device's beacon group.

4.11 Frame

A unit of data transmitted by a device.

4.12 Frame protection

Security service provided for a frame, including (but not limited to) payload encryption, message authentication, and replay attack protection.

4.13 MAC client

An entity above the MAC sublayer that generates MAC service data units for delivery to corresponding entities in other devices, and receives MAC service data units from such entities.

4.14 MAC command data unit (MCDU)

A unit of data exchanged between peer medium access control sublayers in order to manage medium access control functions.

4.15 MAC protocol data unit (MPDU)

The unit of data exchanged between two peer medium access control sublayers using the physical layer.

4.16 MAC service data unit (MSDU)

Information that is delivered as a unit between medium access control service access points (SAPs).

4.17 Master-slave pair (MSPr)

A device to device link in which a first device acts as the master (initiates polling) and a second device acts as a slave (responds to a polling inquiry).

4.18 Master-slave period (MSP)

The period of channel time used for communication by an MSPr.

4.19 Message integrity code (MIC)

A cryptographic checksum generated using a symmetric key that is typically appended to data in order to provide data integrity and source authentication similar to a digital signature.

4.20 Neighbour

Any device in a device's beacon group.

4.21 Reservation

A named set of one or more medium access slots (MASs) within a superframe during which a device has preferential access to the medium.

4.22 Reservation block

One or more temporally contiguous medium access slots (MASs) within a reservation not adjacent to other MASs in the reservation.

4.23 Secure frame

A frame in which frame protection is applied.

4.24 Stream

A logical flow of MSDUs from one device to one or more other devices.

4.25 Superframe

The periodic time interval used in this Standard to coordinate frame transmissions between devices, which contains a beacon period followed by a data period.

4.26 Switched beam antenna (SBA) Device

A device that employs switching-capable antenna array that forms beams towards pre-defined directions

4.27 Symmetric key

A secret key shared between two or more parties that may be used for both encryption and decryption as well as for message integrity code computation and verification.

4.28 Time domain spreading factor (TDSF)

The bandwidth expansion ratio due to the application of time domain spreading sequence.

5 Notational conventions

The use of the word *shall* is meant to indicate a requirement which is mandated by the Standard, i.e. it is required to implement that particular feature with no deviation in order to conform to the Standard.

The use of the word *should* is meant to recommend one particular course of action over several other possibilities, however without mentioning or excluding these others.

The use of the word *may* is meant to indicate that a particular course of action is permitted.

The use of the word *can* is synonymous with is able to – it is meant to indicate a capability or a possibility.

All floating-point values have been rounded to 3 decimal places.

6 Abbreviations and acronyms

AC	Access Category
ACK	Acknowledgment
AES	Advanced Encryption Standard
ASK	Amplitude Shift Keying
ASIE	Application-Specific Information Element
AWGN	Additive White Gaussian Noise
B-ACK	Block Acknowledgment

BcstAddr	Broadcast Device Address
BIFS	Beacon Inter-Frame Space
BP	Beacon Period
BPOIE	Beacon Period Occupancy Information Element
BPSK	Binary Phase Shift Keying
BPST	Beacon Period Start Time
CBC-MAC	Cipher Block Chaining-Message Authentication Code
CC	Convolutional Code
CCA	Clear Channel Assessment
CCM	Counter Mode Encryption and Cipher Block Chaining Message Authentication Code
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTR	Channel Time Reservation
CTT	Clear To Train
DAC	Digital-to-Analogue Converter
DAMI	Dual Alternate Mark Inversion
DBPSK	Differential Binary Phase Shift Keying
DCA	Distributed Contention Access
DestAddr	Destination Device Address
DevAddr	Device Address
DME	Device Management Entity
DQPSK	Differential Quadrature Phase Shift Keying
DRP	Distributed Reservation Protocol
EO	Encryption Offset
EUI	Extended Unique Identifier
FEC	Forward Error Correction
FER	Frame Error Rate
FFT	Fast Fourier Transform
FZ	Frank-Zadoff Sequence
GF	Galois Field
Gbps	Gigabits per second
Gsps	Gigasymbols per second
GTK	Group Temporal Key
HCS	Header Check Sequence
ID	Identifier
IDFT	Inverse Discrete Fourier Transform

IE	Information Element
IFFT	Inverse Fast Fourier Transform
IFS	Inter-Frame Space
Imm-ACK	Immediate Acknowledgment
KCK	Key Confirmation Key
LIFS	Long Inter-frame space
LQE	Link Quality Estimator
LQI	Link Quality Indicator
LSB	Least-Significant Bit
MAC	Medium Access Control
MAS	Medium Access Slot
MCDU	MAC Command Data Unit
Mbps	Megabits per second
McstAddr	Multicast Device Address
MIB	Management Information Base
MIC	Message Integrity Code
MIFS	Minimum Interframe Spacing
MKID	Master Key Identifier
MLME	MAC Sublayer Management Entity
MPDU	MAC Protocol Data Unit
MSB	Most-Significant Bit
MSC	Message Sequence Chart
MSPr	Master-Slave Pair
MSP	Master-Slave Period
MSDU	MAC Service Data Unit
No-ACK	No Acknowledgement
OFDM	Orthogonal Frequency Division Modulation
OOB	Out of Band
OOK	On-Off Keying
OUI	Organizationally Unique Identifier
PAA	Phased Array Antenna
PAN	Personal Area Network
PAL	Protocol Adaptation Layer
PDU	Protocol Data Unit
PER	Packet Error Rate
PHY	Physical (layer)
PHY-SAP	Physical Layer Service Access Point

PLCP	Physical Layer Convergence Protocol
PLME	Physical Layer Management Entity
PMK	Pair-wise Master Key
PPDU	PLCP Protocol Data Unit
PPM	Parts Per Million
PRBS	Pseudo-Random Binary Sequence
PRF	Pseudo-Random Function
PSD	Power Spectral Density
PSDU	PHY Service Data Unit
PT	Preamble Type
PTK	Pair-wise Temporal Key
QPSK	Quadrature Phase Shift Keying
RF	Radio Frequency
RMS	Root Mean Square
RS	Reed-Solomon
RTT	Request To Train
RX	Receive or Receiver
SAP	Service Access Point
SCS	Segment Check Sequence
SFC	Secure Frame Counter
SFN	Secure Frame Number
SIFS	Short Interframe Spacing
SrcAddr	Source Device Address
TCM	Trellis Coded Modulation
TDSF	Time Domain Spreading Factor
TKID	Temporal Key Identifier
TX	Transmit or Transmitter
TXOP	Transmission Opportunity
WPAN	Wireless Personal Area Network

7 General description (informative)

7.1 PHY general description

This Ecma Standard specifies a physical layer (PHY) for a Wireless Personal Area Network (WPAN), utilizing the unlicensed 60 GHz frequency band. Three types of devices are defined: Type A, Type B and Type C. All three device types coexist and interoperate with other device types. Furthermore, all device types can operate independently. That is, neither device type requires the presence of another type for operation.

Type A devices operate at an SCBT mandatory mode (A0) at 0.397 Gbps with other optional SCBT modes at data rates 0.794 Gbps to 6.350 Gbps (without channel bonding) and optional

OFDM modes at data rates 1.008 Gbps to 4.032 Gbps. Type B devices operate using DBPSK at data rates of 0.794 Gbps to 1.588 Gbps (without channel bonding); with optional modes of DQPSK, UEP-QPSK and Dual-AMI at data rates of 3.175 Gbps. Type C devices operate using OOK at data rates of 0.800 Gbps, and 1.600 Gbps; with optional mode of 4ASK at data rate of 3.200 Gbps.

Type A devices also support directional antennas via sector antennas or adaptive arrays. This standard specifies the necessary training and tracking waveforms and protocols.

There are multiple channels specified in this standard. Multiple adjacent channels may be bonded together for increased data rate for Type A (SCBT) and Type B devices. With bonded channels the data rates for these device types increase by a factor proportional to the number of bonded channels. One channel has been designated as the discovery channel. The discovery channel has a lower quality of service but supports the interference prone and time consuming antenna training process. The data channels are optimized for high throughput and spatial reuse.

7.2 MAC general description

7.2.1 General description of the architecture

As illustrated in Figure 2, the MAC is a sublayer of the Data Link Layer defined in the OSI basic reference model [Annex C.3]. The MAC service is provided by means of the MAC service access point (MAC SAP) to a single MAC service client, usually a higher layer protocol or adaptation layer. In this Standard the MAC sublayer is represented by a device address.

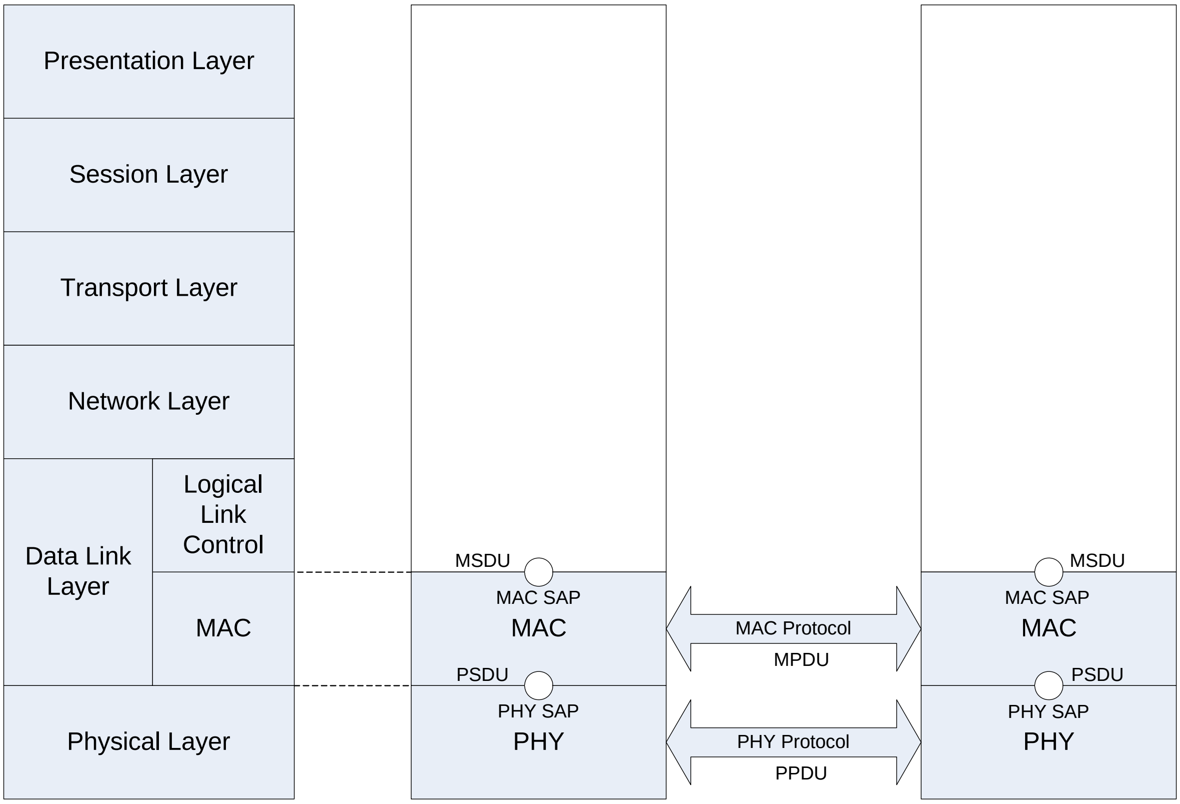


Figure 2 - Architectural reference model

The MAC sublayer in turn relies on the service provided by the PHY layer via the PHY service access point (PHY SAP). The MAC protocol applies between MAC sublayer peers.

7.2.2 Device address

Individual MAC sublayers are addressed via an EUI-48 [Annex C.1], and are associated with a volatile abbreviated address called a DevAddr. Unicast frames carry a destination DevAddr that identifies a single MAC sublayer.

DevAddrs are 16-bit values, generated locally, without central coordination. Consequently, it is possible for a single value to ambiguously identify two or more MAC entities. This Standard provides mechanisms for resolving ambiguous DevAddrs.

The MAC addressing scheme includes multicast and broadcast address values. A multicast address identifies a group of MAC entities. The broadcast address identifies all MAC entities.

7.2.3 Features assumed from the PHY

A MAC entity is associated with a single PHY entity via that entity's PHY SAP. The MAC sublayer requires the following features provided by the PHY:

- Frame transmission and reception;
- PLCP header error indication for both PHY and MAC header structures;
- Clear channel assessment for estimation of medium activity.

Frames are transmitted by the PHY from the source device and delivered to the destination device in identical bit order. Throughout this specification reference to the start of a frame refers to the leading edge of the first symbol of the PHY frame at the local antenna and end of a frame refers to the trailing edge of the last symbol of the PHY frame.

Frame transmission and reception are supported by the exchange of parameters between the MAC sublayer and the PHY layer. These parameters are included in the PLCP header and allow the MAC entity to control, and be informed of, the MAC and PHY related parameters.

Depending on antennas used by the PHY, the MAC sublayer may use the following features provided by the PHY for directional frame transmission and reception:

- Antenna beam switching among different sectors
- Antenna beam steering toward a desired direction

7.2.4 Overview of MAC service functionality

The MAC service defined in this standard provides:

- A reservation-based channel access mechanism;
- A contention-based channel access mechanism for antenna training in the discovery channel;
- A synchronization facility among cooperating MAC entities;
- Coexistence and interoperability among Type A, B and C devices;
- Device power management by scheduling of frame transmission and reception;
- Secure communication with data authentication and encryption using cryptographic algorithms;

Each device provides required MAC functions based on the device type, and optional functions as determined by the application.

Coordination of devices within radio range is achieved by the transmission and reception of beacon and control frames. Periodic beacon transmission provides the basic timing for the network, supports dynamic network organization, and carries reservation and scheduling information for accessing the medium. Exchange of control frames enables antenna training among cooperating devices, and device discovery of Type A, B and C devices.

Coordination among devices that send periodic beacon frames (referred to as beaconing devices) is fully distributed. Coordination among beaconing devices and devices that do not send beacon frames is achieved by the beaconing devices acting as controllers. Coordination among devices that do not send beacon frames is not specified in this standard.

7.2.4.1 Logical groups

Logical groups are formed around each beaconing device to facilitate contention-free frame exchanges while exploring medium reuse over different spatial regions. In this standard, these logical groups are a beacon group and an extended beacon group. Both groups are determined with respect to an individual beaconing device, which has its own individual neighbourhood.

7.2.4.2 Device discovery

The MAC sublayer defined in this standard enables device discovery through one or more of the following mechanisms:

- Transmission and reception of discovery beacon frames in the discovery channel;
- Exchange of antenna training control frames in the discovery channel;
- Exchange of interoperability control frames between Type A and Type B devices, between Type A and Type C devices, or between Type B and Type C devices.

7.2.4.3 Channel selection

Once a device discovers another device with which it intends to communicate with, the pair of the devices use explicit channel selection procedure, as described in 15.4.1 to scan one or more channels and to select a channel for frame exchange in a coordinated manner.

If no beacons are detected in the selected channel, the device creates its beacon period (BP) by sending a beacon. If one or more beacons are detected in the selected channel, the device synchronizes its BP to existing beacons in the selected channel. The device exchanges data with members of its beacon group using the same channel the device selected for beacons.

Each device operates in a dynamic environment and under unlicensed operation rules. Thus, it is subject to interference from other networks, and other unlicensed wireless entities in its channel. To enable the device to continue operation in this type of environment, each device has the capability to dynamically change the channel in which it operates without requiring disruption of links with its peers.

If at any time a device determines that the current channel is unsuitable, it uses the implicit channel selection procedure, as described in 15.4.2, to move to a new channel.

7.2.4.4 The superframe

Once a device finds its communication partner and selects a channel, the basic timing structure for frame exchange is a superframe. The superframe duration is specified as mSuperframeLength. The superframe is composed of 256 medium access slots (MASs), where each MAS duration is mMASLength.

Each superframe starts with a BP, which extends over one or more contiguous MASs. The start of the first MAS in the BP, and the superframe, is called the beacon period start time (BPST).

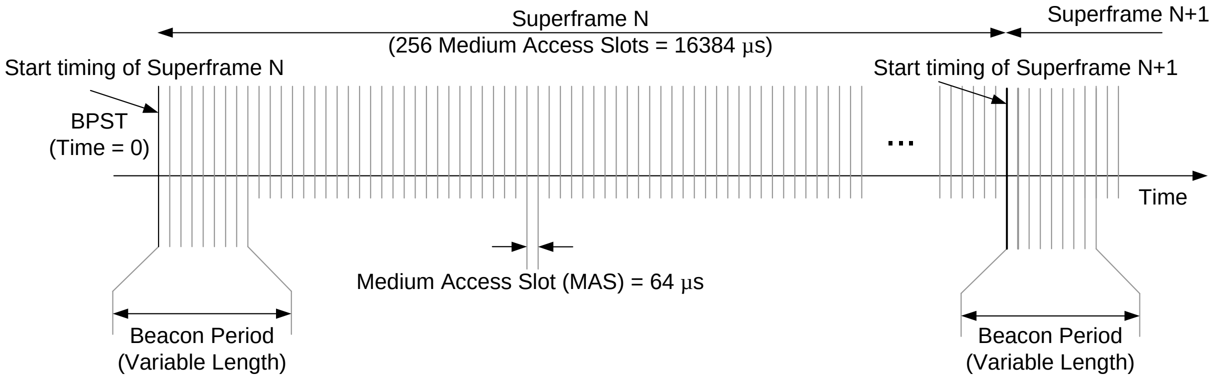


Figure 3 - MAC superframe structure

7.2.4.5 Beacon period protection

No transmissions other than beacons are attempted during the BP of any device.

A device may protect an alien BP, detected by reception of a beacon frame unaligned with the device's own BP, by announcing a reservation covering the alien BP in its beacon.

7.2.4.6 Medium access

The medium is accessed in one of three ways:

- During device discovery (15.3) and antenna training, devices send beacon and control frames in discovery channel using contention based access according to the rules specified in 15.2.
- During the BP, devices send only beacon frames, according to the rules specified in 15.5.
- During reservations, devices participating in the reservation send frames according to rules specified in 15.6.

7.2.4.7 Data communication between devices

Data is passed between the MAC entity and its client in MSDUs qualified by certain parameters. MSDUs are transported between devices in data frames. To reduce the frame error rate of a marginal link, data frames can be fragmented and reassembled, as described in 15.9. Fragments are numbered with an MSDU sequence number and a fragment number.

If the source device wishes to verify the delivery of a frame, then one of the acknowledgement policies is used, as described in 15.12. This standard provides for three types of acknowledgements to enable different applications. The No-ACK policy, described in 15.12.1, is appropriate for frames that do not require guaranteed delivery, or are delay sensitive and a retransmitted frame would arrive too late. The Imm-ACK policy, described in 15.12.2, provides an acknowledgement process in which each frame is individually acknowledged following the reception of the frame. The B-ACK policy, described in 15.12.3, lets the source send multiple frames without intervening ACK frames. Instead, the acknowledgements of the individual frames are grouped into a single response frame that is sent when requested by the source device. The B-ACK process decreases the overhead of the Imm-ACK process while allowing the source device to verify the delivery of frames to the destination.

If the source device does not receive the requested acknowledgement, then it may retransmit the frame, as described in 15.1.5 and 15.19.10, or it may discard the frame. The decision to retransmit or discard the frame depends on the type of data or command that is being sent, the number of times that the source device has attempted to send the frame, the length of time it has attempted to send the frame, and other implementation-dependent factors.

7.2.4.8 MAC frame data rates

This standard specifies three common PHY modes for the three types of devices respectively. The frame payloads of MAC beacon frames are transmitted using one of the common PHY modes corresponding to its device type, and hence at the rate of the common mode. In device discovery or antenna training, MAC frames are transmitted using one of the discovery modes in the discovery channel. Payloads of other frames may be transmitted at higher data rates if possible.

7.2.4.9 Security

Wireless networks present unique security challenges due to the loss of protection provided by wires and shielding. Distributed wireless networks present additional challenges due to the wide range of applications and use models that they must support. To name a few, eavesdroppers can overhear data exchanges not intended for them, whereas imposters can send forged data not using its own identity, can replay previously transmitted data, and can transmit modified data captured from a previous transmission.

This Standard defines two levels of security (Clause 16): no security and strong security protection. Security protection includes data encryption, message integrity, and replay attack protection. Secure frames are used to provide security protection to data and aggregated data frames as well as selected control and command frames.

Three security modes are defined to control the level of security for devices in their communications. This Standard allows for a device to use one of the two security levels or a combination of them in communicating with other devices by selecting the appropriate security mode (see 16.2).

This Standard further specifies a 4-way handshake mechanism to enable two devices to derive their pair-wise temporal keys (PTKs) while authenticating their identity to each other. A secure relationship is established following a successful 4-way handshake between two devices (see 16.3.1). A 4-way handshake between two devices is conducted based on a shared master key. How two devices obtain their shared master keys is outside the scope of this Standard.

In addition, this Standard provides means for the solicitation and distribution of group temporal keys (GTKs). While PTKs are used for protecting unicast frames exchanged between two devices, GTKs are employed for protecting multicast and broadcast frames transmitted from a source device to a multicast or broadcast group of recipient devices (see 16.3.2).

A pseudo-random function is defined based on the MIC generation by CCM using AES-128 (see 16.3.3). It can be made available to entities outside the MAC sublayer for random number generation.

Secure frame counters and replay counters are set up on a per-temporal key basis to guarantee message freshness (see 16.4). No specific mechanisms are created in this Standard to address denial of service attacks given the open nature of the wireless medium.

In this Standard, 128-bit symmetric temporal keys are employed based on AES-128 with CCM to provide payload encryption and message integrity code (MIC) generation (see 16.5).

In general, this Standard specifies security mechanisms, not security policies.

7.2.4.10 Information discovery

The protocols and facilities of this Standard are supported by the exchange of information between devices. Information can be broadcast in beacon frames or requested in Probe commands. For each type of information, an Information Element (IE) is defined. IEs can be included by a device in its beacon at any time and may optionally be requested or provided using the Probe command.

A device uses the MAC Capabilities IE and PHY Capabilities IE to announce information about its support of variable or optional facilities. Declaration of capabilities is especially useful when a device detects changes in its immediate neighbourhood.

7.2.4.11 Transmit rate and power adaptation

This standard provides transmit rate and power control mechanisms to select the optimal combination of transmit rate and power to increase throughput and/or reduce the frame error rate (FER).

A recipient device may recommend a transmit rate and/or power level change be used by a source device using explicit or implicit link feedback mechanisms. In addition, a source device may request a recipient device provide feedback on the quality of the link, base on which the source determines optimal transmit rate and power.

The transmit power and rate control mechanisms are described in 15.4 and 15.5.

7.2.4.12 Power management

An important goal of this standard is to enable long operation time for battery powered devices. An effective method to extend battery life is to enable devices to turn off completely or reduce power for long periods of time, where a long period is relative to the superframe duration.

This standard provides two power management modes in which a device can operate: active and hibernation. Devices in active mode transmit and receive beacons in every superframe. Devices in hibernation mode hibernate for multiple superframes and do not transmit or receive in those superframes.

In addition, this standard provides facilities to support devices that sleep for portions of each superframe in order to save power.

Power management mechanisms are described in 15.16.

7.2.5 MAC policies

It is desirable to allow and facilitate equitable and efficient coexistence of devices with varying medium access requirements. For this purpose, Annex A specifies policies governing channel selection and sharing of bandwidth. These policies impose, among other things, certain restrictions on the number and configuration of MASs in DRP reservations, on the location of reserved MASs within a superframe, and on channel selection order.

7.2.6 Support for higher-layer timer synchronization

Some applications, for example, the transport and rendering of audio or video streams, require synchronization of timers located at different devices. Greater accuracy (in terms of jitter bounds) or finer timer granularity than that provided by the synchronization mechanism described in 15.8 may be an additional requirement. In support of such applications, this Standard defines an optional MAC facility in Annex AD that enables layers above the MAC sublayer to accurately synchronize timers located in different devices. The facility is usable by more than one application at a time.

7.3 MUX general description

In order to enable the coexistence of concurrently active higher layer protocols within a single device, a multiplexing sublayer is defined. This sublayer routes outgoing and incoming MSDUs to and from their corresponding higher layers. The mandatory MUX sublayer is described in Annex A.

7.4 HDMI PAL description

This standard includes an HDMI pass-through PAL, Clause 17, which preserves the HDMI content protection scheme. The HDMI PAL interfaces with the wired HDMI interface’s data channels, clock channel, display data channel and CE control. The HDMI PAL removes the data channels’ TMDS encoding prior to wireless transmission and reinstates the TMDS coding prior to forwarding to the HDMI sink.

8 PHY layer (informative)

The PHY contains two functional entities: the PHY convergence function and the PHY layer management function. The PHY service is provided to the MAC through the PHY SAP.

9 Description of signal

9.1 Mathematical framework for SCBT, OFDM, DBPSK, DQPSK, UEP-QPSK, OOK and 4ASK

The transmitted RF signal can be written in terms of the complex baseband signal as follows:

$$s_{RF}(t) = \mathrm{Re}\left\{\sum_{n=0}^{N_{frame}-1} s_n\left(t - nT_{sym}\right)\exp\left(j2\pi f_c t\right)\right\} \tag{1}$$

where $\mathrm{Re}\{.\}$ represents the real part of the signal, T_{sym} is the symbol length, N_{frame} is the number of symbols in the frame, f_c is the centre frequency, and $s_n(t)$ is the complex baseband signal representation for the n^{th} symbol. The exact structure of the n^{th} symbol depends on its location within the frame:

$$s_n(t) = \begin{cases} s_{preamble,n}(t) & 0 \le n < N_{preamble} \\ s_{header,n-N_{preamble}}(t) & N_{preamble} \le n < N_{preamble} + N_{header} \\ s_{payload,n-N_{preamble}-N_{header}}(t) & N_{preamble} + N_{header} \le n < N_{preamble} + N_{header} + N_{payload} \\ s_{ATS,n-N_{preamble}-N_{header}-N_{payload}}(t) & N_{preamble} + N_{header} + N_{payload} \le n < N_{preamble} + N_{header} + N_{payload} + N_{ATS} \end{cases} \tag{2}$$

where $s_{preamble,n}(t)$ describes the n^{th} symbol of the preamble, $s_{header,n}(t)$ describes the n^{th} symbol of the header, $s_{payload}(t)$ describes the n^{th} symbol of the PPDU, $s_{ATS,n}(t)$ describes the n^{th} symbol of the antenna training sequence (ATS), $N_{preamble}$ is the number of symbols in the preamble, N_{header} is the number of symbols contained in the header, $N_{payload}$ is the number of symbols contained in the frame body, N_{ATS} is the number of symbols in the ATS, and $N_{frame} = N_{preamble} + N_{header} + N_{payload} + N_{ATS}$ is the number of symbols in the frame. The exact values of $N_{preamble}$, N_{header}, $N_{payload}$, N_{ATS}, and N_{frame} will be described in more detail in Clause 10.

The potentially complex time-domain signal $s_n(t)$ shall be created by passing the real and imaginary components of the discrete-time signal $s_n[k]$ through digital-to-analogue converters (DACs) and reconstruction filters as shown in Figure 4. When the discrete-time signal $s_n[k]$ is real, only the real digital-to-analogue converter and reconstruction filter need to be used. Clause 10 describes how to generate $s_n[k]$.

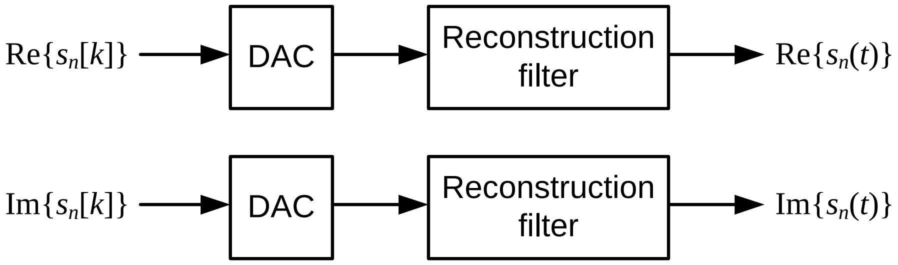


Figure 4 - Conversion from discrete-time signals to continuous-time signals

9.2 Mathematical framework for the narrow band section of the discovery mode preamble

The transmitted RF signal can be written in terms of the complex baseband signal as follows:

$$s_{RF}(t) = \mathrm{Re}\left\{\sum_{n=0}^{N_{NB}-1} s_{NB,n}(t-nT_{sym})\left[\exp(j2\pi f_c t)+\exp(j2\pi[f_c-f_0]t)+\exp(j2\pi[f_c+f_0]t)\right]\right\} \quad (3)$$

where $\mathrm{Re}\{.\}$ represents the real part of the signal, T_{sym} is the symbol length, $N_{NB} = 163839$ is the number of symbols in the narrow band preamble, f_c is the centre frequency, f_0=720 MHz is the offset frequency and $s_{NB,n}(t)$ describes the n^{th} symbol of the narrow band preamble.

The complex time-domain signal $s_{NB,n}(t)$ shall be created by passing the real and imaginary components of the discrete-time signal $S_{NB}[k]$ through digital-to-analogue converters (DACs) and reconstruction filters as shown in Figure 4. The sequence $S_{NB}[k]$ is described in Clause 10.

9.3 Mathematical framework for DAMI

The transmitted RF signal for a DAMI device is a single sideband (SSB) modulated signal accompanied by two pilot tones. The SSB signal can be written as:

$$s_{\mathrm{SBB}}(t) = s(t)\cos(2\pi f_{\mathrm{c}} t) + \hat{s}(t)\sin(2\pi f_{\mathrm{c}} t) \quad (4)$$

where f_{c} is the centre frequency and $\hat{s}(t)$ is the Hilbert transform of $s(t)$. The baseband signal $s(t)$ can be represented by

$$s(t) = \sum_{k=0}^{N_{\mathrm{frame}}-1} d[k]g(t-kT_{\mathrm{sym}}) \quad (5)$$

where N_{frame} is the number of symbols in the frame, T_{sym} is the symbol duration, $d[k]$ is the k^{th} symbol of the modulated data ($d[k] = -1, 0,$ or 1), and $g(t)$ is the baseband pulse shape. The exact value of the modulated data $d[k]$ depends on its location within the frame:

$$d[k] = \begin{cases} d_{\text{preamble}}[k] & 0 \le k < N_{\text{preamble}} \\ d_{\text{header}}[k - N_{\text{preamble}}] & N_{\text{preamble}} \le k < N_{\text{preamble}} + N_{\text{header}} \\ d_{\text{payload}}[k - N_{\text{preamble}} - N_{\text{header}}] & N_{\text{preamble}} + N_{\text{header}} \le k < N_{\text{preamble}} + N_{\text{header}} + N_{\text{payload}} \end{cases} \tag{6}$$

where $d_{preamble}[k]$ is the k^{th} symbol of the preamble, $d_{header}[k]$ is the k^{th} symbol of the header, and $d_{payload}[k]$ is the k^{th} symbol of the PPDU payload, $N_{preamble}$ is the number of symbols in the preamble, N_{header} is the number of symbols in the header, $N_{payload}$ is the number of symbols in the PPDU payload, and $N_{frame} = N_{preamble} + N_{header} + N_{payload}$ is the total number of symbols in the frame. Clause 10.3.3 describes how to generate $d[k]$.

The baseband pulse $g(t)$ shall be chosen such that the power spectral density (PSD) of the baseband signals complies with the transmit PSD mask given in 12.1.

The two pilot tones shall have frequencies f_c and $f_c - 1/(2T_{sym})$, respectively. Both of them shall be phase synchronized with the SSB signal. Their amplitudes shall be chosen such that the total power of each pilot is 25 dB (with ±1 dB tolerance) below the total power of the SSB signal.

10 PLCP sublayer

10.1 General PPDU frame format

This Clause provides a method for converting a PSDU (PHY layer SDU) into a PPDU (PHY layer PDU). Figure 5 shows the general format for the PPDU, which may be composed of four major components: the PLCP preamble, the PLCP header, the PPDU payload, and the Antenna Training Sequence (ATS). All device types shall follow this general frame format. However, particular parameters or components of the frame format may be different for different device types.

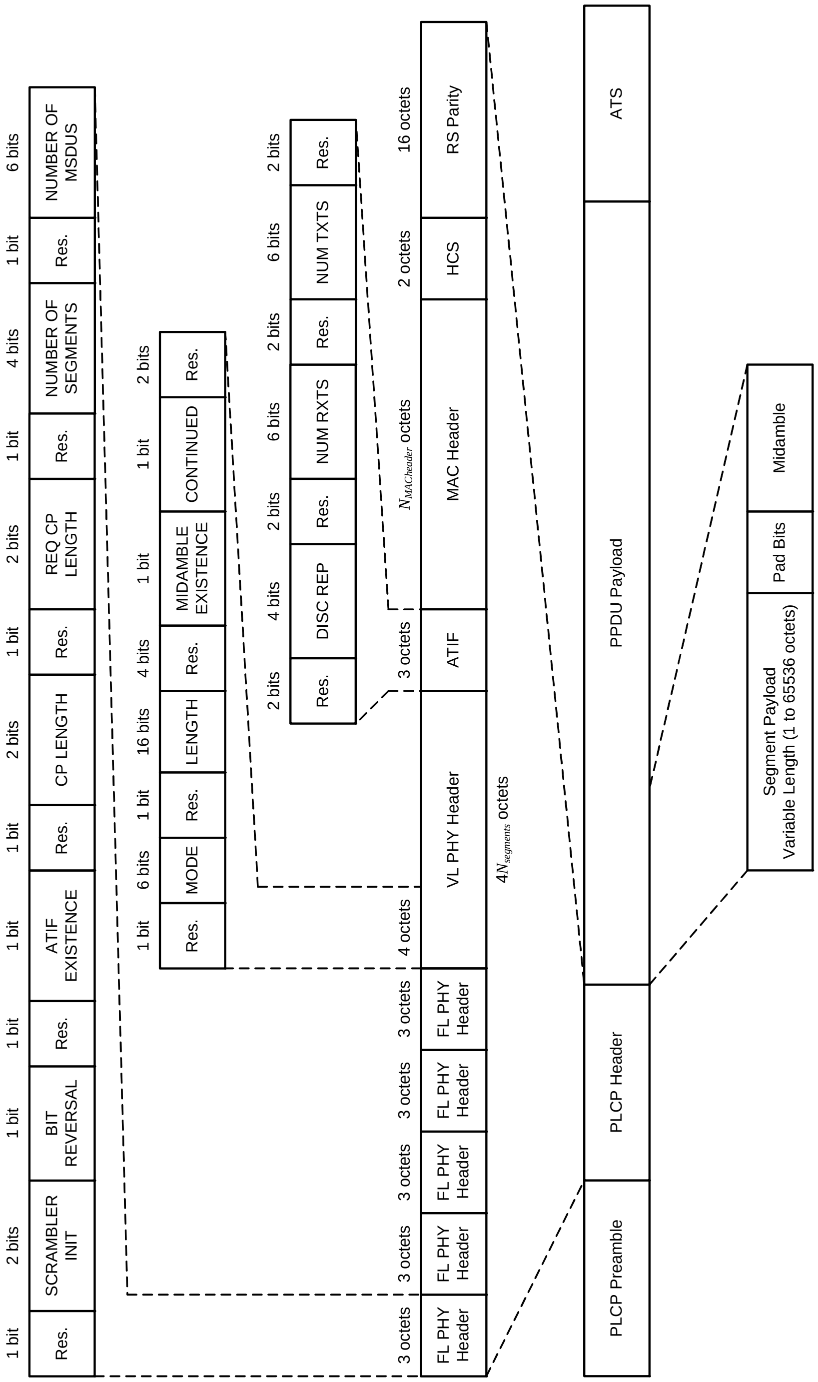


Figure 5 - General PPDU frame format

The components are listed in the order of transmission. The PLCP preamble shall be the first component of the PPDU. It consists of a frame synchronization sequence, and a channel estimation sequence (see 10.1.1). The goal of the PLCP preamble is to aid the receiver in timing synchronization, carrier offset recovery, and channel estimation.

The PLCP header shall be the second major component of the PPDU. The goal of this component is to convey necessary information about both the PHY and the MAC to aid in decoding of the PSDU payload at the receiver. The PLCP header may be further decomposed into PHY header, MAC header, HCS, and Reed-Solomon parity bits (see 10.1.2).

The PPDU payload may be the third major component of the PPDU. The PPDU may contain one or more segments as described in 10.1.3.

The ATS may be the last major component of the PPDU. This sequence is used to train an antenna array (see 10.1.4).

When transmitting the PPDU frame, the PLCP preamble shall be sent first, followed by the PLCP header, potentially the PPDU payload, and potentially the ATS.

10.1.1 PLCP preamble

A PLCP preamble shall be added prior to the PLCP header to aid the receiver in timing synchronization, carrier-offset recovery, and channel estimation.

The PLCP preamble for Type A (SCBT and OFDM), Type B (Single Carrier and DAMI) and Type C are different. The details of the PLCP preamble in each case can be found in 10.2.2.3, 10.2.3.3, 10.3.2.3, 10.3.3.3, and 10.4.4.

10.1.2 PLCP header

A PLCP header shall be added after the PLCP preamble to convey information about both the PHY and the MAC that is needed at the receiver in order to successfully decode the PPDU payload. The scrambled and Reed-Solomon encoded PLCP header shall be formed as shown in Figure 6:

1. Format the PHY header based on information provided by the MAC (See 10.1.2.1).
2. Calculate the HCS value (2 octets) over the combined PHY header and MAC header (See 10.1.2.2).
3. The resulting HCS value is appended to the MAC header. The resulting combination (MAC Header + HCS) is scrambled according to 10.2.2.5.1.1.
4. Apply a shortened Reed-Solomon code to the concatenation of variable length PHY header, ATIF, scrambled MAC header and HCS. The shortened Reed-Solomon code shall be as specified in 10.2.2.5.1.2.
5. Prepend five repetitions of the fixed length PHY header and append the 16 parity octets at the end to form the scrambled and shortened Reed-Solomon encoded PLCP header.

Further encoding and modulation for the header is different for Type A (SCBT and OFDM), Type B (Single Carrier and DAMI) and Type C. The details of the further encoding and modulation for the header for each case can be found in 10.2.2.4, 10.2.3.4, 10.3.2.4, 10.3.3.4, and 10.4.5.

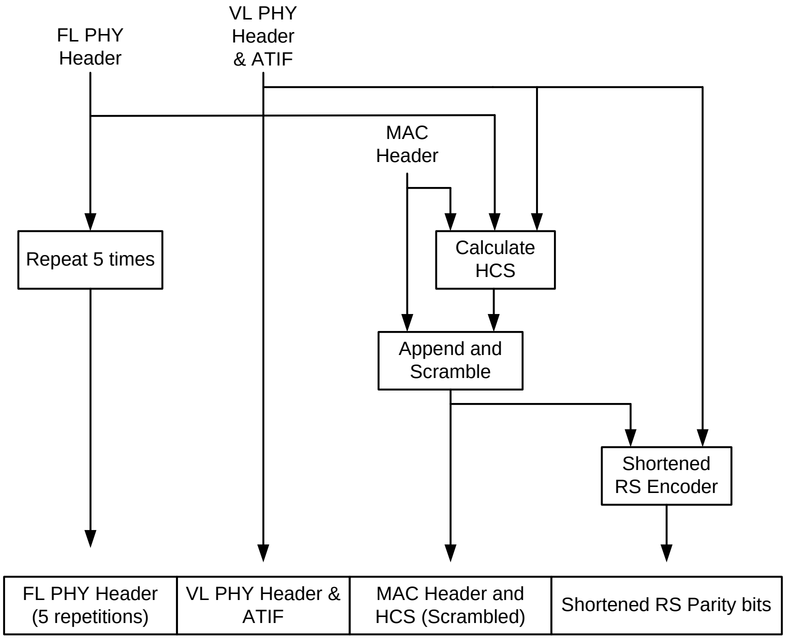


Figure 6 - Formation of the header

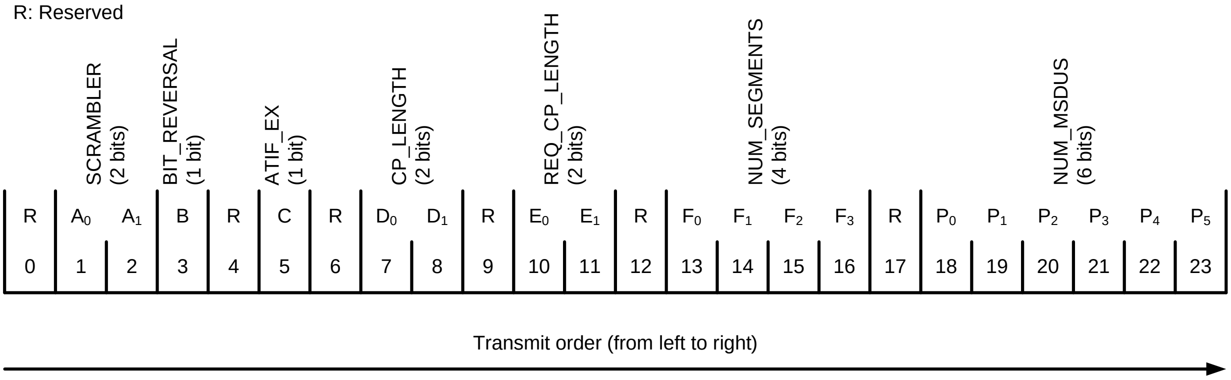
10.1.2.1 PHY header

The PHY header shall be formed by concatenating five repetitions of the fixed length PHY header, $N_{segments}$ sections of the variable length PHY Header and potentially an antenna training indicator field (ATIF). The overall length of the PHY header is therefore equal to $15+4N_{segments}+3I_{ATIF}$ octets, where I_{ATIF} is equal to zero when the header does not include the ATIF and is equal to one when the header includes the ATIF.

10.1.2.1.1 Fixed length PHY header

The fixed length PHY header contains information about the seed identifier for the data scrambler, the bit reversal state for PPDU payload, the existence of the ATIF, the CP length for the current frame, the requested CP length for the following frame, the number of segments in the frame, and the number of MSDUs in the frame.

The fixed length PHY header field shall be composed of 24 bits, numbered from 0 to 23 as illustrated in Figure 7. Bits 1-2 shall encode the seed value for the initial state of the scrambler, which is used to synchronize the descrambler of the receiver. Bit 3 shall encode whether or not all of the information data bits are inverted. Bit 5 shall encode whether or not the header includes an ATIF. Bits 7-8 shall encode the CP length for the current frame and bits 10-11 shall encode the requested CP length for the following frame. Bits 13-16 shall encode the number of segments in the frame. Bits 18-23 shall encode the number of MSDUs in the frame. All other bits which are not defined in this Clause shall be understood to be reserved for future use and shall be set to zero. The values of the defined fields are described in 10.2.2.4.1.1, 10.2.3.4.1.1, 10.3.2.4.1.1, 10.3.3.4.1.1 and 10.4.5.1.1.



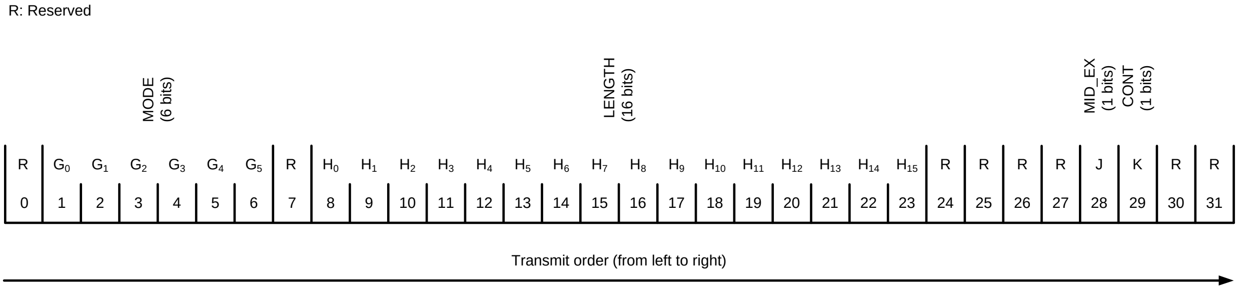
R	A_0	A_1	B	R	C	R	D_0	D_1	R	E_0	E_1	R	F_0	F_1	F_2	F_3	R	P_0	P_1	P_2	P_3	P_4	P_5
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

Figure 7 - Fixed length PHY header bit assignment

10.1.2.1.2 Variable length PHY header

Each section of the variable length PHY header contains information about the data rate of the corresponding segment, the length of the payload in the corresponding segment, whether the corresponding segment includes a midamble, and whether the corresponding segment is partitioned at the transmitter and should be reassembled at the receiver.

The variable length PHY header field shall be composed of $32N_{segments}$ bits and shall be constructed by concatenating $N_{segments}$ sections, each for one segment. Each section shall be composed of 32 bits, numbered from 0 to 31, as illustrated in Figure 8. Bits 1-6 shall encode the MODE field, which conveys the information about the type of modulation, the coding rate, and the spreading factor used in the segment. Bits 8-23 shall encode the LENGTH field, with the least-significant bit being transmitted first. Bit 28 shall encode whether the segment is appended by a midamble. Bit 29 shall encode whether the segment has partitioned at the transmitter, and should be reassembled at the receiver. All other bits which are not defined in this Clause shall be understood to be reserved for future use and shall be set to zero. The values of the defined fields are described in 10.2.2.4.1.2, 10.2.3.4.1.2, 10.3.2.4.1.2, 10.3.3.4.1.2 and 10.4.5.1.2.



R	G_0	G_1	G_2	G_3	G_4	G_5	R	H_0	H_1	H_2	H_3	H_4	H_5	H_6	H_7	H_8	H_9	H_{10}	H_{11}	H_{12}	H_{13}	H_{14}	H_{15}	R	R	R	R	J	K	R	R
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	30	31

Figure 8 - Variable length PHY header bit assignment

10.1.2.1.3 Antenna training indicator field

The Antenna Training Indicator Field (ATIF) shall follow the variable length PHY header if the ATIF_EXISTENCE bit in the fixed length PHY header is set to 1 (see 10.1.2.1.1). The ATIF contains information about the number of block repetitions in the discovery mode, the number of training symbols for receive antenna training, and the number of training symbols for the transmit antenna training.

The ATIF shall be composed of 24 bits, numbered from 0 to 23, as illustrated in Figure 9. Bits 2-5 shall encode the DISC_REP field which conveys the number of repetitions in the discovery mode. Bits 8-13 shall encode the field NUM_RXTS which conveys the number of training symbols for receive antenna training. Bits 16-21 shall encode the field NUM_TXTS which conveys the number of training symbols for transmit antenna training. All other bits which are not defined in this Clause shall be understood to be reserved for future use and shall be set to zero. The values of the defined fields are described in 10.2.2.4.1.3, 10.2.3.4.1.3, 10.3.2.4.1.3, 10.3.3.4.1.3, 10.4.5.1.3.

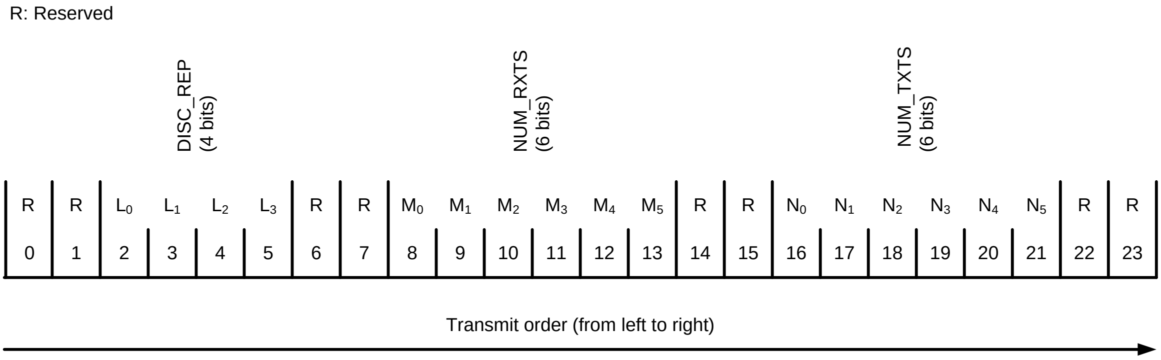


Figure 9 - Antenna Training Indicator Field bit assignment

10.1.2.2 Header check sequence (HCS)

The combination of PHY header and the MAC header shall be protected with a 2 octet CCITT CRC-16 header check sequence (HCS). The CCITT CRC-16 HCS shall be the ones complement of the remainder generated by the modulo-2 division of the combined PHY and MAC headers by the polynomial: $x^{16} + x^{12} + x^5 + 1$. The HCS bits shall be processed in the transmit order. All HCS calculations shall be made prior to data scrambling. A schematic of the processing order is shown in Figure 10. The registers shall be initialized to all ones.

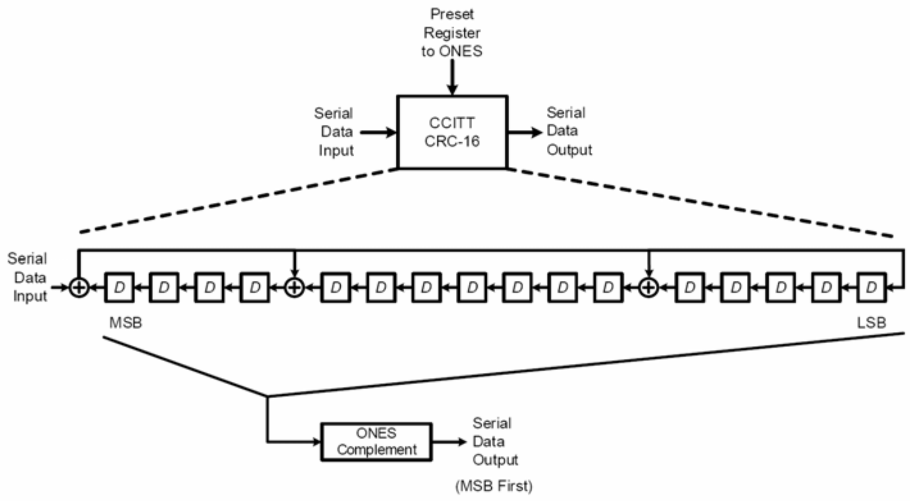


Figure 10 - CCITT CRC-16 block diagram

10.1.3 PPDU payload

The PSDU shall be fragmented into the MAC header and one or more MSDUs. Each MSDU may be further fragmented into two or more data bit blocks. A number of consecutive MSDUs that use the same transmission mode may be combined into one data bit block. The resulting data bit blocks shall be encoded and mapped according to the modulation and coding scheme of each device type to generate a transmit symbol block and may be appended by a midamble to form a segment. The details of each modulation and coding scheme is described in 10.2.2.5, 10.2.3.5, 10.3.2.5, 10.3.3.5 and 10.4.6. The resulting segments shall be joined consecutively to form the PPDU payload. Figure 11 depicts this operation. Each segment of the PPDU payload may be sent using a different data rate mode. The least-significant bit (LSB) of an octet shall be the first bit transmitted. Type A devices shall be capable of transmission and reception of multi-segment frames. Type B devices may be capable of transmission and reception of multi-segment frames. Type C frames shall not have more than one segment.

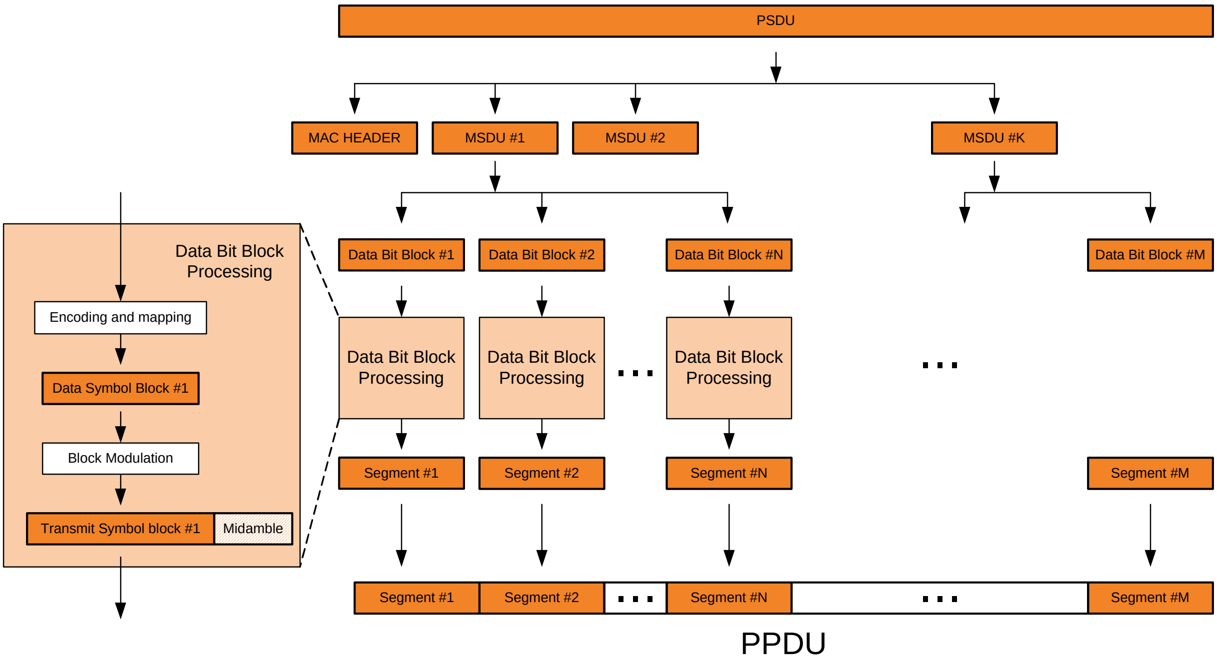


Figure 11 - Formation of the PPDU payload from the PSDU

10.1.3.1 Midamble

The midamble for Type A (SCBT and OFDM) and Type B (SC) are different. The details of the midamble for each case can be found in 10.2.2.5.4, 10.2.3.5.2, and 10.3.2.5.5. Type B DAMI and Type C frames shall not have a midamble.

10.1.4 Antenna training sequence

The Type A frames shall include the ATS, upon request. The Type B frames may include the ATS upon request. Type C frames shall not include the ATS. When included, the ATS shall be the concatenation of $N_{TXTS} + N_{RXTS}$ FZ sequences of length 256 (parameter $A_{FZ} = 16$). The FZ sequence is defined in 10.2.2.3, where N_{TXTS} and N_{RXTS} are the number of training sequences for transmitter and receiver antenna training, respectively (see Figure 12).

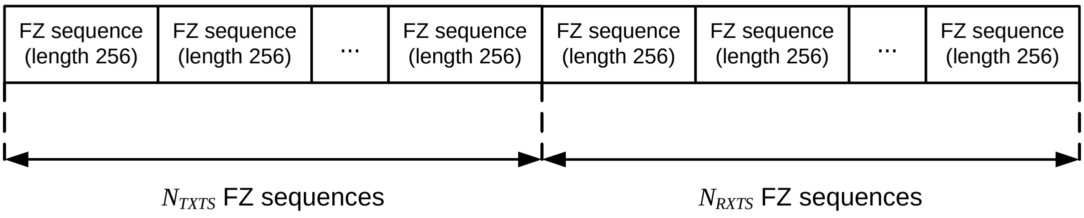


Figure 12 - Antenna training sequence

10.2 Type A PPDU

10.2.1 Mode dependent parameters

The PSDU mode dependent modulation parameters are listed in Table 1.

Table 1- PSDU mode dependent Parameters

Mode	Base Data Rate[1] (Gbps)				Modulation	Constellation	Encoding	CC Code Rate (R)	TDSF (N_{TDS})	Tail bits (N_{tail})
	No channel bonding	2 bonded channels	3 bonded channels	4 bonded channels						
A0	0.397	0.794	1.191	1.588	SCBT	BPSK	RS & CC	1/2	2	4
A1	0.794	1.588	2.381	3.175	SCBT	BPSK	RS & CC	1/2	1	4
A2	1.588	3.175	4.763	6.350	SCBT	BPSK	RS	1	1	0
A3	1.588	3.175	4.763	6.350	SCBT	QPSK	RS & CC	1/2	1	4
A4	2.722	5.443	8.165	10.886	SCBT	QPSK	RS & CC	6/7	1	6
A5	3.175	6.350	9.526	12.701	SCBT	QPSK	RS	1	1	0
A6	4.234	8.467	13.701	16.934	SCBT	NS8QAM	RS & TCM	5/6	1	8
A7	4.763	9.526	14.288	19.051	SCBT	NS8QAM	RS	1	1	0
A8	4.763	9.526	14.288	19.051	SCBT	TCM-16QAM	RS & TCM	2/3	1	6
A9	6.350	12.701	19.051	25.402	SCBT	16QAM	RS	1	1	0
A10[2]	1.588	3.175	4.763	6.350	SCBT	QPSK	RS & UEP-CC	R_{MSB}: 1/2	1	4
A11	4.234	8.467	12.701	16.934	SCBT	16QAM	RS & UEP-CC	R_{MSB}: 4/7, R_{LSB}: 4/5	1	4
A12	2.117	4.234	6.350	8.467	SCBT	UEP-QPSK	RS & CC	2/3	1	4
A13	4.234	8.467	12.701	16.934	SCBT	UEP-16QAM	RS & CC	2/3	1	4
A14	1.008	N/A	N/A	N/A	OFDM	QPSK	RS & CC	1/3	1	6
A15	2.016	N/A	N/A	N/A	OFDM	QPSK	RS & CC	2/3	1	6
A16	4.032	N/A	N/A	N/A	OFDM	16QAM	RS & CC	2/3	1	6
A17	2.016	N/A	N/A	N/A	OFDM	QPSK	RS & UEP-CC	R_{MSB}: 4/7, R_{LSB}: 4/5	1	6
A18	4.032	N/A	N/A	N/A	OFDM	16QAM	RS & UEP-CC	R_{MSB}: 4/7, R_{LSB}: 4/5	1	6
A19	2.016	N/A	N/A	N/A	OFDM	UEP-QPSK	RS & CC	2/3	1	6
A20	4.032	N/A	N/A	N/A	OFDM	UEP-16QAM	RS & CC	2/3	1	6
A21[2]	2.016	N/A	N/A	N/A	OFDM	QPSK	RS & CC	R_{MSB}: 2/3	1	6

NOTE

[1] *Base data rates assume a cyclic prefix length of zero.*

[2] *In modes A10 and A21 only the four MSB bits of each octet shall be transmitted, while the four LSB bits of each octet are discarded by the transmitter.*

All Type A devices shall support mode A0 (without channel bonding), and may support mode A0 with channel bonding or modes A1 through A21. In addition all Type A devices shall support mode B0 (without channel bonding) and mode C0. Type A devices may support modes B0 (with channel bonding), B1, B2, or B3 (with or without channel bonding), C1 or C2. Type A devices shall not support mode B4. See 10.3 and 10.4 for Type B and Type C modes, respectively.

10.2.2 SCBT

10.2.2.1 Timing related parameters

The timing parameters associated with the SCBT PHY are listed in Table 2.

Table 2 - Timing related parameters

Parameter	Description	Value
f_{sym}	Symbol frequency	1.728 Gsps
T_{sym}	Symbol duration	0.5787 ns
N_{SCBTB}	Number of symbols per SCBT block	256
N_D	Number of data symbols per SCBT block	252
N_P	Number of pilot symbols per SCBT block	4
N_{CP}	Number of symbols in the CP	0 32 64 96
N_{SCBTS}	Number of symbols per SCBT block	256 288 320 352
T_{SCBTB}	SCBT block interval	148.148 ns
T_{CP}	CP interval	0 18.5185 ns 37.037 ns 55.5556 ns
T_{SCBTS}	SCBT symbol interval	148.148 ns 166.667 ns 185.185 ns 203.707 ns

10.2.2.2 Frame related parameters

The frame related parameters associated with the PHY are listed in Table 3.

Table 3 - Frame Related Parameters

Parameter	Description	Value
N_{sync}	Number of symbols in the frame synchronization sequence	2048
T_{sync}	Duration of the frame synchronization sequence	1185.19 ns
N_{CE}	Number of symbols in the channel estimation sequence	768

Table 3 - Frame Related Parameters (concluded)

Parameter	Description	Value
T_{CE}	Duration of the channel estimation sequence	444.444 ns
$N_{preamble}$	Number of symbols in the PLCP preamble	2816
$T_{preamble}$	Duration of the PLCP preamble	1629.63 ns
N_{ATS}	Number of symbols in the ATS	$256(N_{TXTS}+N_{RXTS})N_{DISCREP}$
T_{ATS}	Duration of the ATS	$N_{ATS}T_{sym}$
N_{frame}	Number of symbols in the frame	$N_{sync}+N_{header}+N_{payload}+N_{ATS}$
T_{frame}	Duration of the frame	$(N_{sync}+N_{header}+N_{payload}+N_{ATS})T_{sym}$

10.2.2.3 PLCP preamble

The preamble for Type A SCBT frames can be subdivided into two distinct portions: a frame synchronization sequence, $S_{sync,A,SCBT}[.]$, and a channel estimation sequence, $S_{CE,A,SCBT}[.]$. Figure 13 shows the structure of the PLCP preamble for Type A SCBT frames.

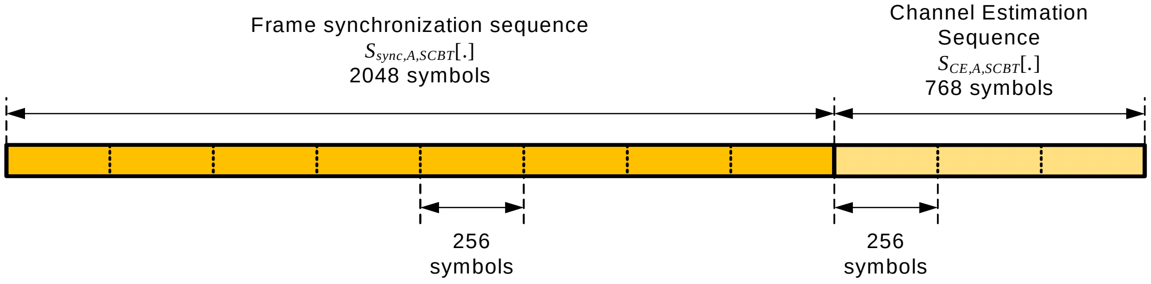


Figure 13 - Structure of the Type A PLCP preamble for Type A SCBT frames

Both the frame synchronization sequence and the channel estimation sequence are constructed based on the Frank-Zadoff (FZ) sequences. A Frank-Zadoff sequence with parameter A_{FZ}, which has length $N_{FZ}=A_{FZ}^2$, is defined as:

$$S_{FZ,A_{FZ}}[n] = \exp\left(j\frac{2\pi pq}{A_{FZ}}\right) \qquad (7)$$

where $p=(n \bmod A_{FZ})+1$, and $q=\left\lfloor \frac{n}{A_{FZ}} \right\rfloor+1$, for $n=0,\ldots, A_{FZ}^2-1$, and $\lfloor . \rfloor$ denotes the floor function, which returns the largest integer value smaller than or equal to its argument.

10.2.2.3.1 Frame synchronization sequence

The frame synchronization sequence for Type A SCBT frames consists of eight repetitions of a hierarchical sequence $S_h[.]$, covered by sequence $S_{cover}[.]$, as described in equation (10). The hierarchical sequence, $S_h[.]$, is of length 256, and is defined as the Kronecker product of the Frank-Zadoff sequence of length 16, $s_{FZ,4}[.]$, with itself. That is,

$$s_h[n] = s_{FZ,4}[n \bmod 16]\, s_{FZ,4}\left[\left\lfloor \frac{n}{16} \right\rfloor\right] \tag{8}$$

for $n=0,...,255$. The cover sequence is of length eight and is defined as

$$s_{cover}[n] = \begin{cases} 1 & n = 0,...,6 \\ -1 & n = 7 \end{cases} \tag{9}$$

The frame synchronization sequence is defined as the Kronecker product of the hierarchical sequence $s_h[.]$ and the cover sequence $s_{cover}[.]$. That is,

$$s_{sync,A,SCBT}[n] = s_{cover}\left[\left\lfloor \frac{n}{256} \right\rfloor\right] s_h[n \bmod 256] \tag{10}$$

for $n=0,...,2047$.

10.2.2.3.2 Channel estimation sequence

The channel estimation sequence consists of three repetitions of the Frank-Zadoff sequence of length 256, $s_{FZ,16}[.]$. That is, the channel estimation sequence is given by:

$$s_{CE,A,SCBT}[n] = s_{FZ,16}[n \bmod 256] \tag{11}$$

for $n=0,...,767$.

NOTE
The frame synchronization sequence can be used for frame acquisition and detection, coarse carrier frequency estimation, coarse symbol timing, and for synchronization within the preamble. Whereas, the channel estimation sequence can be used for estimation of the channel frequency response, fine carrier frequency estimation, and fine symbol timing.

10.2.2.4 PLCP header

For Type A SCBT frames, the formed PLCP header (as described in 10.1.2) shall be first padded with N_{padhdr} bits where

$$N_{padhdr} = 192\left\lceil \frac{N_{hdrbits}}{192} \right\rceil - N_{hdrbits} \tag{12}$$

where $N_{hdrbits}$ is the number of bits in the formed PLCP header (as described in 10.1.2). The padded header shall be encoded and modulated as described in 10.2.2.5, starting with the demultiplexing. Encoding and modulation parameters identical to mode A0 shall be used. The resulting data symbols shall be block modulated (see 10.2.2.5.3) in order to create the baseband signal.

10.2.2.4.1 PHY header

10.2.2.4.1.1 Fixed length PHY header

10.2.2.4.1.1.1 PLCP scrambler field (SCRAMBLER)

The bits A_0 and A_1 shall be set according to the scrambler seed identifier value. This two-bit value corresponds to the seed value chosen for the data scrambler (see 10.2.2.5.1.1).

10.2.2.4.1.1.2 Bit reversal field (BIT_REVERSAL)

The value of the BIT_REVERSAL bit shall be set to 0_B for all Type A SCBT frames.

10.2.2.4.1.1.3 ATIF existence field (ATIF_EX)

The ATIF_EX field determines whether or not an ATIF field exists at the end of the PHY header. The mapping between this field and existence of the ATIF is given in Table 4.

Table 4 - Mapping between the value of the ATIF_EX bit and existence of ATIF

ATIF_EX (C)	ATIF Existence
0	ATIF does not exist
1	ATIF does exist

10.2.2.4.1.1.4 CP length field (CP_LENGTH)

The mapping between the value of the CP_LENGTH field and N_{CP} for the current frame is given in Table 5. The default value of N_{CP} for the first frame, or for beacons or the discovery mode frames is 96.

Table 5 - Mapping between the value of the CP_LENGTH field and N_{CP}

CP_LENGTH (D_1D_0)	N_{CP}
00	0
01	32
10	64
11	96

10.2.2.4.1.1.5 Requested CP length field (REQ_CP_LENGTH)

The mapping between the value of the REQ_CP_LENGTH field and N_{CP} for the following frames is given in Table 6.

Table 6 - Mapping between the value of the REQ_CP_LENGTH field and N_{CP}

REQ_CP_LENGTH (E_1E_0)	N_{CP}
00	0
01	32

Table 6 - Mapping between the value of the REQ_CP_LENGTH field and N_{CP} (concluded)

REQ_CP_LENGTH (E_1E_0)	N_{CP}
10	64
11	96

10.2.2.4.1.1.6 Number of segments field (NUM_SEGMENTS)

Depending on the number of segments (NUM_SEGMENTS), bits F_0 to F_3 shall be set according to the values in Table 7. When a frame does not have a payload, this field shall be set to 0000_B.

Table 7 - Mapping between the value of the NUM_SEGMENTS field and number of segments

NUM_SEGMENTS ($F_3F_2F_1F_0$)	Number of segments ($N_{segments}$)
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

NOTE
Discovery mode frames and ACK frames may have zero segments (i.e. no payload).

10.2.2.4.1.1.7 Number of MSDUs field (NUM_MSDUS)

Depending on the number of MSDUs (NUM_MSDUS), bits P_0 to P_5 shall be set according to the values in Table 8. When a frame does not have a payload, this field shall be set to 000000_B.

Table 8 - Mapping between the value of the NUM_MSDUS field and number of MSDUs

NUM_MSDUS ($P_5P_4P_3P_2P_1P_0$)	Number of MSDUs (N_{MSDUs})	NUM_MSDUS ($P_5P_4P_3P_2P_1P_0$)	Number of MSDUs (N_{MSDUs})
000000	0	100000	32
000001	1	100001	33
000010	2	100010	34
000011	3	100011	35
000100	4	100100	36
000101	5	100101	37
000110	6	100110	38
000111	7	100111	39
001000	8	101000	40
001001	9	101001	41
001010	10	101010	42
001011	11	101011	43
001100	12	101100	44
001101	13	101101	45
001110	14	101110	46
001111	15	101111	47
010000	16	110000	48
010001	17	110001	49
010010	18	110010	50
010011	19	110011	51
010100	20	110100	52
010101	21	110101	53
010110	22	110110	54
010111	23	110111	55
011000	24	111000	56
011001	25	111001	57
011010	26	111010	58
011011	27	111011	59

Table 8 - Mapping between the value of the NUM_MSDUS field and number of MSDUs (concluded)

NUM_MSDUS ($P_5P_4P_3P_2P_1P_0$)	Number of MSDUs (N_{MSDUs})	NUM_MSDUS ($P_5P_4P_3P_2P_1P_0$)	Number of MSDUs (N_{MSDUs})
011100	28	111100	60
011101	29	111101	61
011110	30	111110	62
011111	31	111111	63

10.2.2.4.1.2 Variable length PHY header

10.2.2.4.1.2.1 Mode field (MODE)

Depending on the mode (MODE), bits G_0 to G_5 shall be set according to the values in Table 9.

Table 9 - MODE field

Mode	MODE ($G_5...G_0$)	Mode	MODE ($G_5...G_0$)	Mode	MODE ($G_5...G_0$)	Mode	MODE ($G_5...G_0$)
A0	000000	A16	010000	Reserved	100000	Reserved	110000
A1	000001	A17	010001	Reserved	100001	Reserved	110001
A2	000010	A18	010010	Reserved	100010	Reserved	110010
A3	000011	A19	010011	Reserved	100011	Reserved	110011
A4	000100	A20	010100	Reserved	100100	Reserved	110100
A5	000101	A21	010101	Reserved	100101	Reserved	110101
A6	000110	Reserved	010110	Reserved	100110	Reserved	110110
A7	000111	Reserved	010111	Reserved	100111	Reserved	110111
A8	001000	Reserved	011000	Reserved	100111	C0	111000
A9	001001	Reserved	011001	B0	101001	C1	111001
A10	001010	Reserved	011010	B1	101010	C2	111010
A11	001011	Reserved	011011	B2	101011	Reserved	111011
A12	001100	Reserved	011100	B3	101100	Reserved	111100
A13	001101	Reserved	011101	B4	101101	Reserved	111101
A14	001110	Reserved	011110	Reserved	101110	Reserved	111110
A15	001111	Reserved	011111	Reserved	101111	Reserved	111111

10.2.2.4.1.2.2 Segment length field (LENGTH)

The segment length field shall be an unsigned 16-bit integer that indicates the number of octets in the segment payload (which does not include the tail bits, or the pad bits).

10.2.2.4.1.2.3 Midamble existence field (MID_EX)

The MID_EX field determines whether or not the segment is appended by a midamble. The mapping between this bit and existence of the midamble is given in Table 10.

Table 10 - Mapping between the value of the MID_EX bit and existence of midamble

MID_EX (J)	Midamble existence
0	Midamble does not exist
1	Midamble exists

10.2.2.4.1.2.4 Segment continued field (CONT)

The CONT field determines whether or not this segment is a fragment of the received PSDU, and should be reassembled at the receiver. The value of this bit shall be according to Table 11.

Table 11 - Value of the CONT bit

CONT (K)	Fragmentation
0	The segment is not a fragment or is the last fragment
1	The segment is a fragment

10.2.2.4.1.3 Antenna training indicator field

10.2.2.4.1.3.1 Discovery mode repetition field (DISC_REP)

The discovery mode repetition field represents the number of repetitions in the discovery mode. The mapping between the value of DISC_REP and $N_{DISCREP}$ is given in Table 12. For all non-discovery mode frames the value of DISC_REP shall be set to $L_3L_2L_1L_0 = 0000_B$.

Table 12 - Mapping between the value of the DISC_REP field and number of discovery mode repetitions

DISC_REP ($L_3L_2L_1L_0$)	Number of repetitions in discovery mode ($N_{DISCREP}$)
0000	1
0001	2
0010	4
0011	8
0100	16
0101	32
0110	64
0111	128
1000	Reserved

Table 12 - Mapping between the value of the DISC_REP field and number of discovery mode repetitions (concluded)

DISC_REP ($L_3L_2L_1L_0$)	Number of repetitions in discovery mode ($N_{DISCREP}$)
1001	Reserved
1010	Reserved
1011	Reserved
1100	Reserved
1101	Reserved
1110	Reserved
1111	Reserved

10.2.2.4.1.3.2 Number of training symbols for RX training field (NUM_RXTS)

The number of training symbols for RX training field shall be an unsigned 6-bit integer. Its value shall indicate the number of training symbols for RX training in the ATS.

10.2.2.4.1.3.3 Number of training symbols for TX training field (NUM_TXTS)

The number of training symbols for TX training field shall be an unsigned 6-bit integer. Its value shall indicate the number of training symbols for TX training in the frame.

10.2.2.5 PPDU payload

The PPDU payload consists of one or more segments as described in 10.1. The segments are formed as described in 10.1.3. Each data bit block shall be processed as follows:

1. The resulting bits shall be encoded and mapped as described in 10.2.2.5.1 or 10.2.2.5.2.
2. The resulting symbols shall be modulated into SCBT symbols as described in 10.2.2.5.3.

10.2.2.5.1 FEC and mapping (Equal error protection)

Figure 14 depicts the general overview of the encoding and mapping scheme for modes A0 through A9. For the payload, first, the data bits shall be scrambled as specified in 10.2.2.5.1.1. Then, $N_{padbits}$ zero pad bits shall be appended to the end of the data bit block, where

$$N_{padbits} = 1792 \left\lceil \frac{N_{bits}}{1792} \right\rceil - N_{bits} \qquad (13)$$

and N_{bits} is the number of bits in the data bit block. The padded data bit block shall be encoded according to the Reed-Solomon code, as specified in 10.2.2.5.1.2.

The resulting RS coded payload bits or the formed header bits shall be demultiplexed into 4 groups, as described in 10.2.2.5.1.3. Depending on the data rate mode, each group of bits shall then be interleaved using the bit interleaver described in 10.2.2.5.1.4. The resulting bits shall be appended with N_{tail} tail bits (in each branch).

They then shall be encoded using the convolutional code or trellis coded modulation described in 10.2.2.5.1.5 and 10.2.2.5.1.6. The resulting encoded bits from the 4 branches shall then be multiplexed together into one group of symbols as specified in 10.2.2.5.1.7. The multiplexed bits are then mapped to constellations as specified in 10.2.2.5.1.8. The data symbols shall be spread in time domain as described in 10.2.2.5.1.9. The resulting data symbols shall then be appended with N_{padsym} zero symbols, where

$$N_{padsyms} = 2N_D \left\lceil \frac{N_{sym}}{2N_D} \right\rceil - N_{sym} \tag{14}$$

The resulting data symbols shall be interleaved with the Dual Helical Scan (DHS) symbol interleaver as described in 10.2.2.5.1.10.

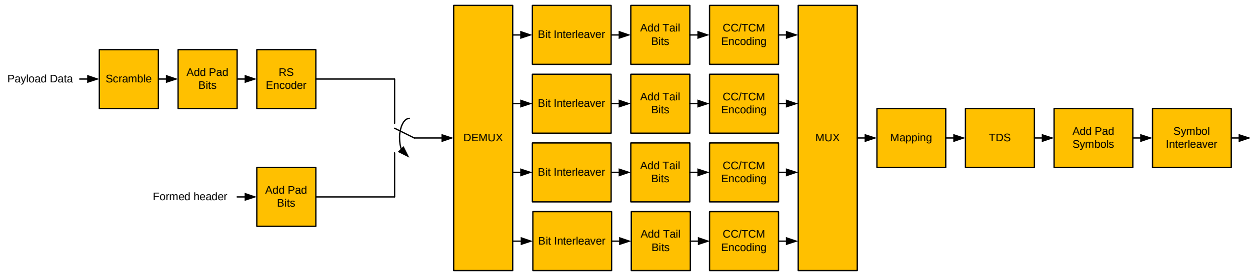


Figure 14 - General view of the encoding procedure

10.2.2.5.1.1 Data scrambler

A side-stream scrambler shall be used to whiten only portions of the PLCP header, i.e. the MAC header and HCS, and the entirety of each segment. In addition, the scrambler shall be initialized to a seed value specified by the MAC at the beginning of the MAC header and then re-initialized to the same seed value at the beginning of each segment.

The polynomial generator, $g(D)$, for the pseudo-random binary sequence (PRBS) generator shall be: $g(D) = 1 + D^{14} + D^{15}$, where D is a single bit delay element. Using this generator polynomial, the corresponding PRBS, $x[n]$, is generated as

$$x[n] = x[n-14] \oplus x[n-15],\ n = 0,1,2,\ldots \tag{15}$$

where $\oplus$ denotes modulo-2 addition. The following sequence defines the initialization vector, x_{init}, which is specified by the parameter “seed value” in:

$$\mathbf{x}_{init} = \left[x_i[-1] \quad x_i[-2] \quad \ldots \quad x_i[-14] \quad x_i[-15] \right] \tag{16}$$

where $x_i[-k]$ represents the binary initial value at the output of the k^{th} delay element. The scrambled data bits, v_m, are obtained as show in Figure 15.

$$v[n] = s[m] \oplus x[m],\ m = 0,1,2,\ldots \tag{17}$$

where $s[m]$ represents the non-scrambled data bits.

The 15-bit initialization vector or seed value shall correspond to the seed identifier as shown in Table 13. The seed identifier value shall be set to 00 when the PHY is initialized and this value shall be incremented in a 2-bit rollover counter for *each* frame sent by the PHY except for retransmitted Type C frames. At retransmission of Type C frames, the seed value shall be incremented in a 2-bit rollover for every two frames retransmitted by the PHY, as described in 10.4.6.1.3.

Table 13 - Scrambler seed selection

Seed Identifier (A_1A_0)	Seed Value $x_{init} = x_i[-1]\ x_i[-2] \ldots x_i[-15]$	PRBS Output First 16 bits $x[0]\ x[1] \ldots x[15]$
00	0011 1111 1111 111	0000 0000 0000 1000
01	0111 1111 1111 111	0000 0000 0000 0100
10	1011 1111 1111 111	0000 0000 0000 1110
11	1111 1111 1111 111	0000 0000 0000 0010

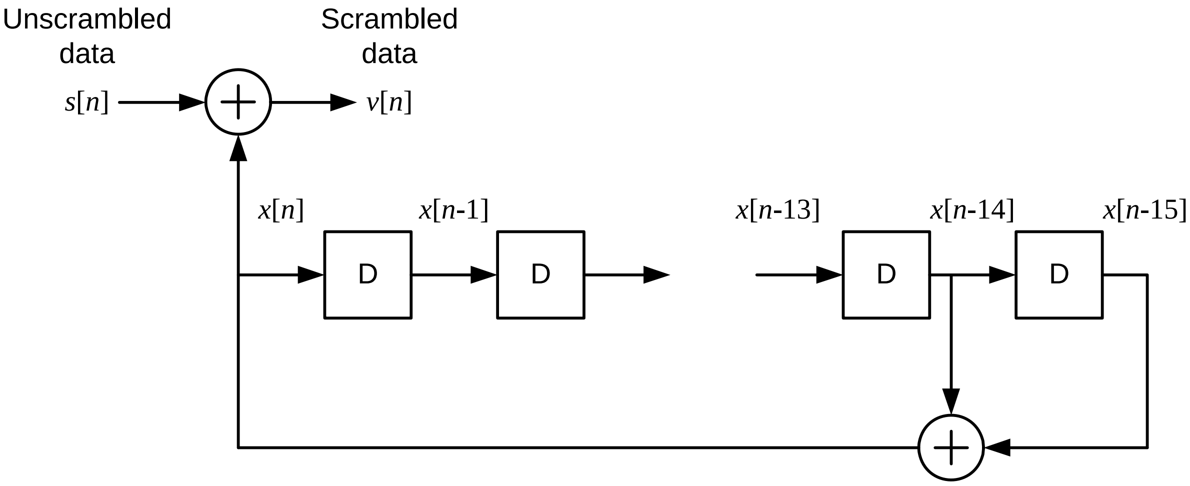


Figure 15 - Block diagram of the side-stream scramble

10.2.2.5.1.2 Reed-Solomon code

A systematic (N=255, K=239, T=8) Reed-Solomon code is shortened to different lengths to be used for the header and the payload. The RS(255,239) shall be shortened to RS($255\text{-}N_{RSpad}$,$239\text{-}N_{RSpad}$) by first padding a block of $239\text{-}N_{RSpad}$ data octets with N_{RSpad} zero octets. The resulting block shall be coded using the RS(255,239) Reed-Solomon code described below. The resulting 16 parity octets r_{15}, ..., r_0 shall be appended to the data block to form the coded data block of length $255\text{-}N_{RSpad}$.

For the header, the N_{RSpad} shall be equal to $222\text{-}4N_{segments}\text{-}3I_{ATIF}\text{-}N_{MACheader}$, where $N_{MACheader}$ is the length of the MAC header (refer to 14.2.1 and 14.2.2). For the payload, the N_{RSpad} shall be equal to 15.

The RS(255,239) code is defined over $GF(2^8)$ with a primitive polynomial $p(z) = z^8 + z^4 + z^3 + z^2 + 1$. As notation, the element $M = b_7z^7 + b_6z^6 + b_5z^5 + b_4z^4 + b_3z^3 + b_2z^2 + b_1z + b_0$, where $M \in GF(2^8)$, has the following binary representation $b_7b_6b_5b_4b_3b_2b_1b_0$, where b_7 is the MSB and b_0 is the LSB.

The code is specified by the generator polynomial $g(x)$ defined over $GF(2^8)$.

$$g(x) = \prod_{i=1}^{2T-1} (x - \alpha^i) \tag{18}$$

where $\alpha = 02_H$ is the root of the polynomial $p(z)$.

The mapping of the information octets $\boldsymbol{m} = (m_{238}, m_{237}, \ldots, m_0)$ to codeword octets $\boldsymbol{c} = (m_{238}, m_{237}, \ldots, m_0, r_{15}, r_{14}, \ldots, r_0)$ is achieved by computing the remainder polynomial $r(x)$

$$r(x) = \prod_{i=0}^{15} r_i x^i = x^{16} m(x) \bmod g(x) \tag{19}$$

where $m(x)$ is the information polynomial

$$m(x) = \prod_{i=0}^{238} m_i x^i \tag{20}$$

and r_i, $i = 0, \ldots, 15$, and m_i, $i = 0, \ldots, 238$, are elements of $GF(2^8)$.

10.2.2.5.1.3 Demultiplexer

The Reed-Solomon encoded bits $b[.]$ shall be demultiplexed to create four bit streams $c_m[.]$, $m=0,\ldots,3$. The $c_m[.]$ shall be obtained from $b[.]$ from $c_m[k] = b[4k+m]$.

10.2.2.5.1.4 Bit interleaver

The bit streams shall be interleaved by a bit interleaver of length 48. The interleaved bit $d_m[l]$ shall be equal to $c_m[k]$ by

$$l = \left[6 \left\lfloor \frac{k}{6} \right\rfloor + 7\big(k \bmod 6\big) \right] \bmod 48 \tag{21}$$

10.2.2.5.1.5 Convolutional code

The convolutional encoder shall use the rate $R=1/2$ code with generator polynomials, $g_0 = 23_O$, and $g_1 = 35_O$, as shown in Figure 16. The bit denoted as "A" shall be the first bit generated by the encoder, followed by the bit denoted as "B". Additional coding rates are derived from the "mother" rate $R=1/2$ convolutional code by employing "puncturing". Puncturing is a procedure for omitting some of the encoded bits at the transmitter (thus reducing the number of transmitted bits and increasing the coding rate) and inserting a dummy "zero" metric into the decoder at the receiver in place of the omitted bits. The puncturing patterns are illustrated in Figure 17, Figure 18, Figure 19, Figure 20, and Figure 21. In each of these cases, the tables shall be filled in with encoder output bits from left to right. For the last block of bits, the process shall be stopped at the point at which encoder output bits are exhausted, and the puncturing pattern applied to the partially filled block.

The encoder shall start from the all-zero state. After the encoding process for the PLCP header has been completed, the encoder shall be reset to the all-zero state before the encoding starts for each segment; in other words, the encoding of each segment shall also start from the all-zero state. The PSDU shall be encoded using the appropriate coding rate of $R = 4/7$, $2/3$, $4/5$, $5/6$, or $6/7$.

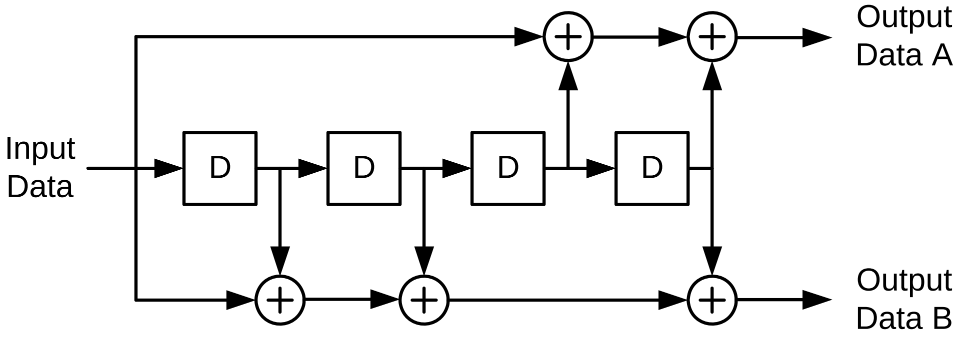


Figure 16 - Convolutional encoder: rate $R=1/2$, constraint length $K=5$

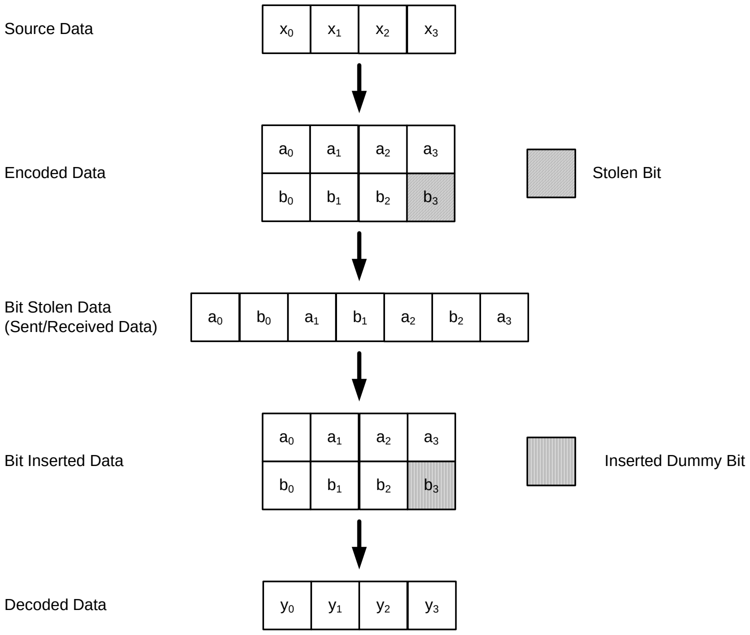


Figure 17 - An example of bit stealing and bit insertion for $R=4/7$ code

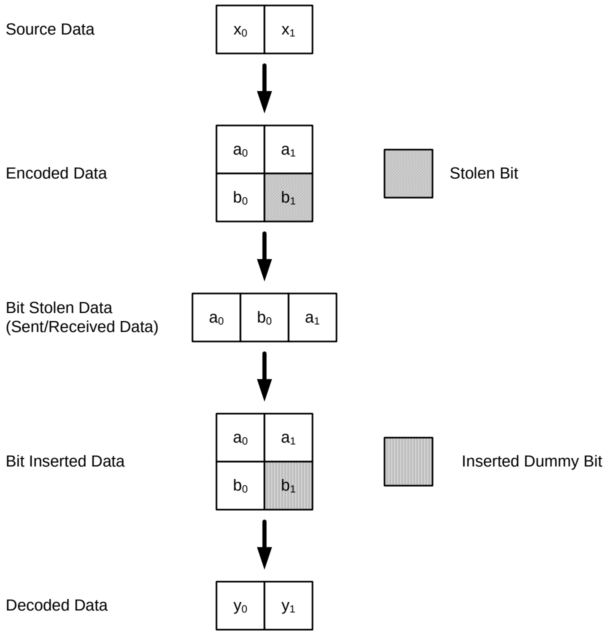


Figure 18 - An example of bit-stealing and bit-insertion for R=2/3 code

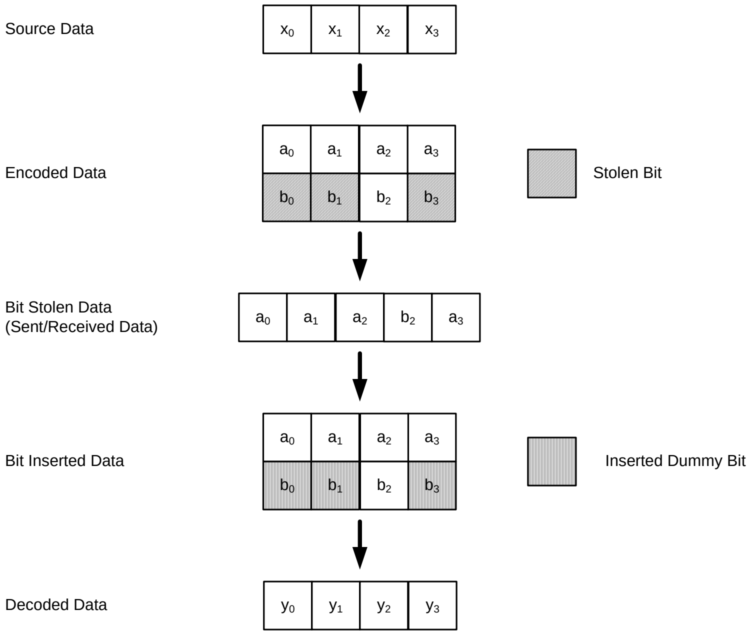


Figure 19 - An example of bit-stealing and bit-insertion for R=4/5 code

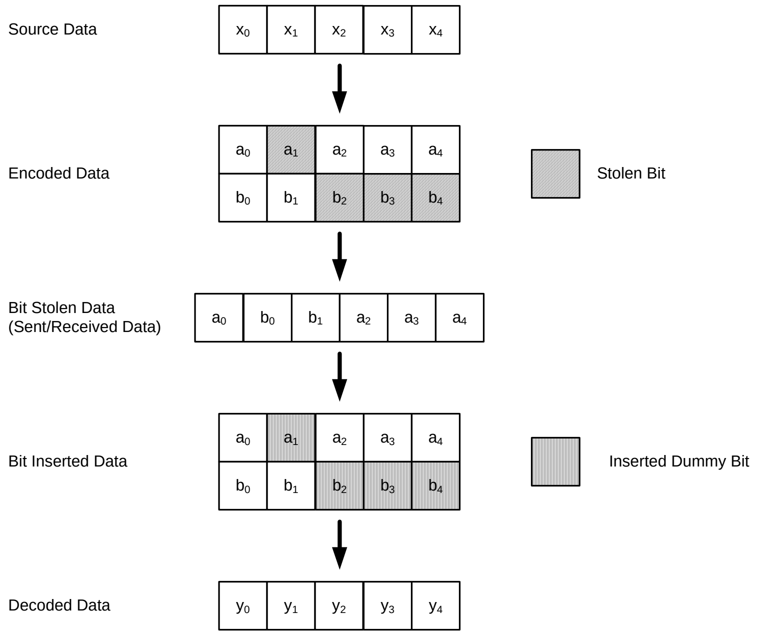


Figure 20 - An example of bit-stealing and bit-insertion for R=5/6 code

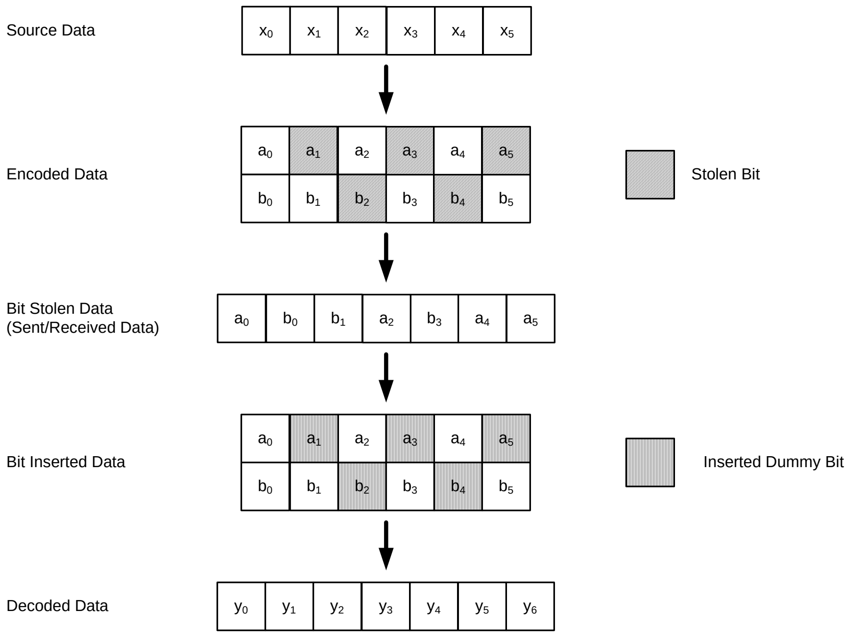


Figure 21 - An example of bit-stealing and bit-insertion for R=6/7 code

NOTE
Decoding by the Viterbi algorithm is recommended

10.2.2.5.1.6 Trellis coded modulation

For NS8QAM and 16QAM constellations, a "pragmatic" trellis coded modulation (TCM) is employed by using the convolutional code and the puncturing patterns as described in 10.2.2.5.1.5.

Before the convolutional encoding, the bits $e_m[.]$ shall be demultiplexed into two groups $e_{c,m}[.]$ and $e_{u,m}[.]$, where

$$e_{u,m}[i] = e_m\left[\left\lfloor \frac{i}{U} \right\rfloor (U+C) + i \bmod U\right] \tag{22}$$

and

$$e_{c,m}[i] = e_m\left[\left\lfloor \frac{i}{C} \right\rfloor (U+C) + i \bmod C + U\right] \tag{23}$$

The bits $e_{c,m}[.]$ shall be encoded and punctured to produce the bits $p_m[.]$. The resulting bits, $p_m[.]$, shall then be multiplexed with bits $e_{u,m}[.]$ to generate $f_m[.]$, where

$$f_m[i] = \begin{cases} e_{u,m}[i/P] & i \bmod P = 0 \\ e_{c,m}\left[\lfloor i/P \rfloor (P-1) + i \bmod P - 1\right] & i \bmod P \neq 0 \end{cases} \tag{24}$$

For a given data rate mode, the convolutional code rate (including encoding and puncturing) and C, P and U shall be as described in Table 14.

Table 14 - Parameters for trellis coded modulation

Mode	R	U	C	P
A6	5/6	3	5	3
A8	2/3	1	2	4

An example of this operation for data rate mode A6 is depicted in Figure 22.

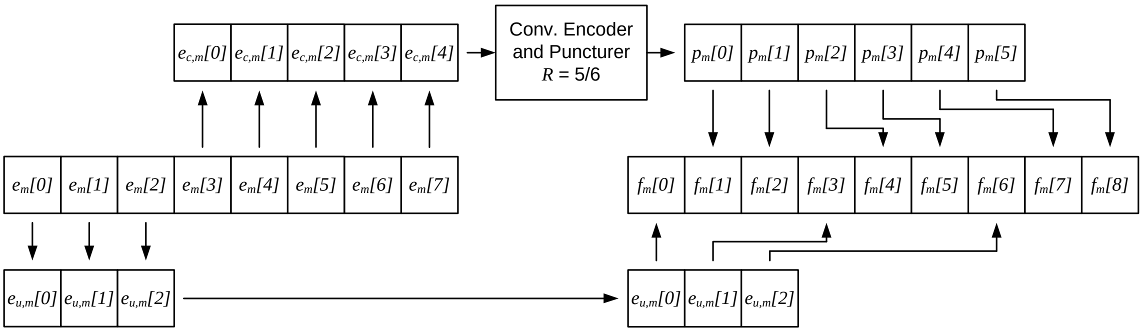


Figure 22 - An example of encoding for trellis coded modulation for mode A6

10.2.2.5.1.7 Multiplexer

The coded bits from the four branches $f_m[.]$, $m=0,\dots,3$ shall be multiplexed to create a single stream $g[.]$. The $g[.]$ shall be obtained from $f_m[.]$ from

$$g[k] = f_{\lfloor k/k_{MUX} \rfloor \bmod 4}\left[k \bmod k_{MUX} + k_{MUX}\left\lfloor \frac{k}{4k_{MUX}} \right\rfloor\right] \tag{25}$$

10.2.2.5.1.8 Mapping

The data bits shall be mapped to the constellations according to the data rate mode. The constellation mappings are given in 10.2.4.

10.2.2.5.1.9 Time domain spreading (TDS)

The data symbols shall be consecutively repeated N_{TDS} times, that is the output of the spread symbols $s[.]$ shall be derived from to the data symbols $v[.]$ by

$$s[n] = v\left[\left\lfloor \frac{n}{N_{TDS}} \right\rfloor\right] \tag{26}$$

where N_{TDS} is the time domain spreading factor (TDSF) defined for each mode in Table 1.

10.2.2.5.1.10 Symbol interleaver

The data symbols, $u[.]$, shall be interleaved using a 21 by 24 dual helical scan (DHS) interleaver. The DHS interleaver interleaves the data symbols by writing and reading the data symbols in a memory block with a helical scan pattern. Each block of 504 data symbols, $u[0],\ldots,u[503]$, shall be interleaved separately to form the interleaved symbols $w[0],\ldots,w[503]$.

The interleaving operation is described by defining the auxiliary symbols $r[m,n]$. The $r[m,n]$ are then given by $r[m,n] = u[k]$ where

$$m = \left[(k \bmod 21) + \left\lfloor \frac{k}{21} \right\rfloor\right] \bmod 24 \tag{27}$$

and

$$n = k \bmod 21 \tag{28}$$

and $w[k]$ are given by $w[k]=r[m,n]$, where

$$m = 23 - \left[(k \bmod 21) + \left\lfloor \frac{k}{21} \right\rfloor\right] \bmod 24 \tag{29}$$

and

$$n = k \bmod 21 \tag{30}$$

10.2.2.5.2 FEC and mapping (Unequal error protection)

Figure 23 depicts the general overview of the encoding and mapping scheme for modes A10 through A13. For the payload, first, the MSB and LSB bits of the each octet is separated as described in 10.2.2.5.2.1. Then, for each group, the data bits shall be scrambled as specified in 10.2.2.5.2.2. Then, $N_{padbits}$ shall be appended to the end of the data bit block, where

$$N_{padbits} = 1792 \left\lceil \frac{N_{bits}}{1792} \right\rceil - N_{bits} \tag{31}$$

and N_{bits} is the number of bits in the data bit block. The padded data bit block shall be encoded according to the Reed-Solomon code, as specified in 10.2.2.5.2.3.

The resulting RS coded payload bits or the formed header bits shall be demultiplexed into 4 groups, as described in 10.2.2.5.2.4. Depending on the data rate mode, each group of bits shall then be interleaved using the bit interleaver described in 10.2.2.5.2.5. The resulting bits shall be appended with N_{tail} tail bits (in each branch). Then they shall be encoded using the convolutional code described in 10.2.2.5.2.6. The resulting encoded bits from all 8 branches shall then be multiplexed together into one group of bits as specified in 10.2.2.5.2.7. The multiplexed bits are then mapped to constellations as specified in 10.2.2.5.2.8. The resulting data symbols shall then be appended with N_{padsym} zero symbols, where

$$N_{padsym} = 2N_D \left\lceil \frac{N_{sym}}{2N_D} \right\rceil - N_{sym} \tag{32}$$

The resulting data symbols shall be interleaved with the Dual Helical Scan (DHS) symbol interleaver as described in 10.2.2.5.2.9.

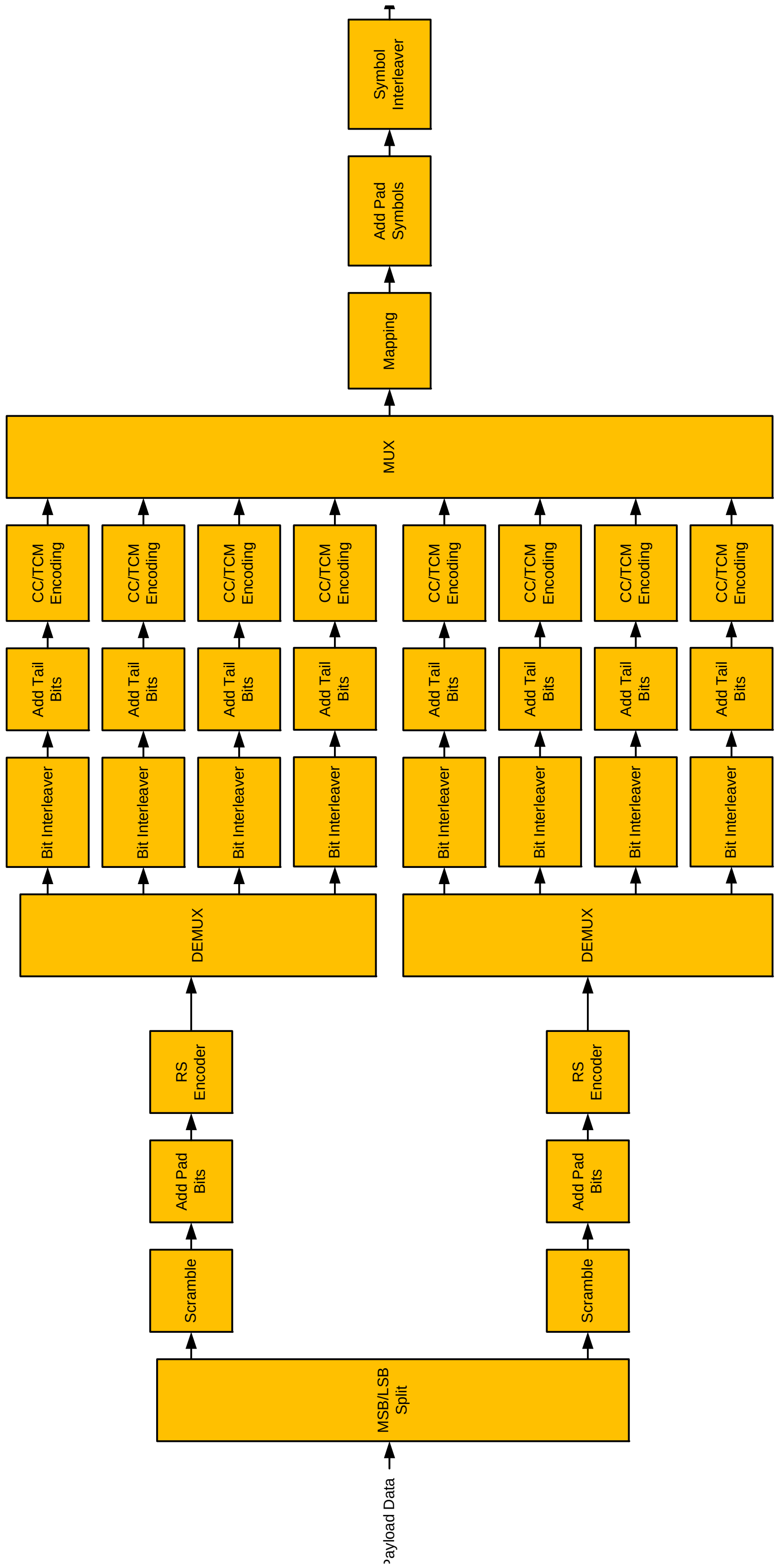


Figure 23 - General view of the encoding procedure

10.2.2.5.2.1 MSB/LSB separation

The incoming bit stream, $a[.]$, shall be demultiplexed into two streams, such that the MSBs and LSBs of each octet are separated. That is, the two bit streams $a_{LSB}[.]$ and $a_{MSB}[.]$ are created from the incoming bit stream $a[.]$ by

$$a_{LSB}[k] = a\left[n \bmod 4 + 8\left\lfloor \frac{k}{4} \right\rfloor \right] \tag{33}$$

and

$$a_{MSB}[k] = a\left[n \bmod 4 + 8\left\lfloor \frac{k}{4} \right\rfloor + 4 \right] \tag{34}$$

10.2.2.5.2.2 Data scrambler

Refer to 10.2.2.5.1.1.

10.2.2.5.2.3 Reed-Solomon code

Refer to 10.2.2.5.1.2.

10.2.2.5.2.4 Demultiplexer

Refer to 10.2.2.5.1.3.

10.2.2.5.2.5 Bit interleaver

Refer to 10.2.2.5.1.4

10.2.2.5.2.6 Convolutional code

The convolutional encoding in each branch shall be as described in 10.2.2.5.1.5. For modes A10 and A11 the code rate for the branches 0 through 3 (MSB branches) shall be R_{MSB}, and the code rate for the branches 4 through 7 (LSB branches) shall be R_{LSB}, as given in Table 1. For mode A10 only MSBs shall be encoded, modulated and transmitted. The LSBs shall be discarded.

10.2.2.5.2.7 Multiplexer

For mode A10, the bits from the eight branches $f_m[.]$, $m=0,...,7$ shall be multiplexed to create a single stream as described in 10.2.2.5.1.7.

For mode A11, the bits from the eight branches $f_m[.]$, $m=0,...,7$ shall be multiplexed to create a single stream $g[.]$. The $g[.]$ shall be obtained from $f_m[.]$ from

$$g[k] = f_m\left[n\left\lfloor \frac{k}{48} \right\rfloor + p \right] \tag{35}$$

where m and n are given in Table 15 as a function of $k \bmod 48$.

Table 15 - Multiplexing parameters for mode A11

$k \bmod 48$	m	n	p	$k \bmod 48$	m	n	p	$k \bmod 48$	m	n	p
0	0	7	0	16	2	7	2	32	4	5	4
1	0	7	1	17	2	7	3	33	5	5	0
2	0	7	2	18	2	7	4	34	5	5	1
3	0	7	3	19	2	7	5	35	5	5	2

Table 15 - Multiplexing parameters for mode A11 (concluded)

k mod 48	m	n	p	k mod 48	m	n	p	k mod 48	m	n	p
4	0	7	4	20	2	7	6	36	5	5	3
5	0	7	5	21	3	7	0	37	5	5	4
6	0	7	6	22	3	7	1	38	6	5	0
7	1	7	0	23	3	7	2	39	6	5	1
8	1	7	1	24	3	7	3	40	6	5	2
9	1	7	2	25	3	7	4	41	6	5	3
10	1	7	3	26	3	7	5	42	6	5	4
11	1	7	4	27	3	7	6	43	7	5	0
12	1	7	5	28	4	5	0	44	7	5	1
13	1	7	6	29	4	5	1	45	7	5	2
14	2	7	0	30	4	5	2	46	7	5	3
15	2	7	1	31	4	5	3	47	7	5	4

For modes A12 and A13, the symbols from the eight branches $f_m[.]$, $m=0,...,7$ shall be multiplexed to create a single stream $g[.]$. The $g[.]$ shall be obtained from $f_m[.]$ from

$$g[k] = f_{\lfloor k/k_{MUX} \rfloor \bmod 4 + 4(\lfloor 2k/k_{MUX} \rfloor \bmod 2)} \left[k \bmod \frac{k_{MUX}}{2} + \frac{k_{MUX}}{2} \left\lfloor \frac{k}{4k_{MUX}} \right\rfloor \right] \qquad (36)$$

where $k_{mux} = 2$ for mode A12, and $k_{mux} = 4$ for mode A13.

10.2.2.5.2.8 Mapping

Refer to 10.2.2.5.1.8.

10.2.2.5.2.9 Symbol interleaver

Refer to 10.2.2.5.1.10.

10.2.2.5.3 Block modulation

The multiplexed transmit symbols, $w[.]$ shall be formed into SCBT symbols. First the transmit data symbols shall be divided into blocks of length N_D=252. This transmit symbol block shall be appended with pilot symbol. The pilots symbols consist of a sequence of length N_P=4, $S_{pilot}[n]=(-1)^{n+1}$, $n=0,1,...,3$, to form an SCBT block. The resulting SCBT block shall be pre-appended with a cyclic prefix. This is done by prefixing the SCBT block with the last N_{CP} transmit symbols of the SCBT symbol. The cyclic prefix length, N_{CP}, shall be set based on the REQ_CP_LENGTH field of the last received frame from the current receiver. The N_{CP} may take the values 0, 32, 64 or 96. The N_{CP} for the first data frame or for data frames that have multiple destinations, e.g. beacon frames, is set to 96. Figure 24 depicts an example of the SCBT symbol formation.

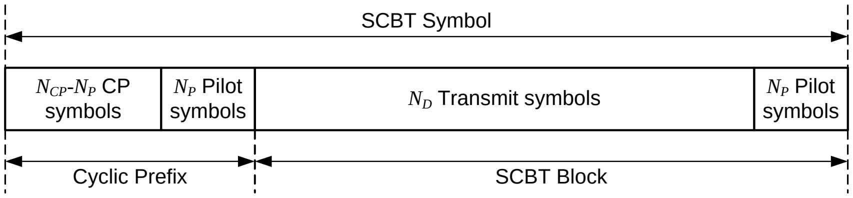


Figure 24 - Formation of the SCBT symbol

10.2.2.5.4 Midamble

The midamble sequence shall be identical to the channel estimation sequence, $s_{CE,A,SCBT}[.]$, as described in 10.2.2.3.2.

NOTE
The midamble sequence can be used to update estimation of the channel frequency response, fine carrier frequency estimation, and fine symbol timing.

10.2.2.6 Channel bonding

Two, three or four adjacent channels may be bonded together. In this case, the formation of frames shall be as specified in 10.2.2.1 through 10.2.2.5. The symbol rate shall be increased according to Table 16.

Table 16 - Symbol rates for channel bonding

Number of bonded channels	f_{sym} (Gsps)
2	3.456
3	5.184
4	6.912

10.2.3 OFDM

10.2.3.1 Timing related parameters

The OFDM PHY timing related parameters are listed in Table 17.

Table 17 - OFDM timing related parameter

Parameters	Description	Value
f_{sym}	Symbol rate	2.592 Gsps
T_{sym}	Symbol time	0.386 ns
N_{FFT}	Number of subcarriers	512
T_{FFT}	FFT period	197.53 ns
N_D	Number of data carriers	360
N_{DC}	Number of DC carriers	3
N_P	Number of pilot carriers	16
N_N	Number of null carriers	133
N_{CP}	Cyclic prefix length	64
T_{CP}	Cyclic prefix duration	24.70 ns
$T_{sym,OFDM}$	OFDM symbol Duration	222.23 ns

10.2.3.2 Frame related parameters

The OFDM PHY frame related parameters are listed in Table 18.

Table 18 - OFDM frame related parameters

Parameters	Description	Value
N_{sync}	Number of symbols in frame synchronization sequence	1792
T_{sync}	Duration of the frame synchronization sequence	691.7 ns
N_{CE}	Number of symbols in the channel estimation sequence	1088
T_{CE}	Duration of the channel estimation sequence	419.97 ns
$N_{preamble}$	Number of symbols in the PLCP preamble	2880
$T_{preamble}$	Duration of the frame preamble	1111.68 ns
N_{ATS}	Number of symbols in the ATS	$256(N_{\mathrm{TXTS}}+N_{\mathrm{RXTS}})N_{DISCREP}$
T_{ATS}	Duration of the ATS	$N_{ATS}T_{sym}$
N_{frame}	Number of Symbols in the frame	$N_{preamble}+N_{header}+N_{payload}+N_{ATS}$
T_{frame}	Duration of the frame	$N_{frame}T_{sym}$

10.2.3.3 PLCP preamble

The preamble includes 7 short training sequences and 2 long training sequences, which are used for frame synchronization and channel estimation, respectively. The short training sequence $S_{short,T}[.]$ is 256 symbols long, while the last sequence is rotated by 180 degree to indicate the end of short sequences and beginning of long sequences. The long training sequence $S_{long,T}[.]$ is 512 symbols long, with a cyclic prefix of length 64. The total length of the preamble is 5 OFDM symbols long. The preamble is normalized to unity.

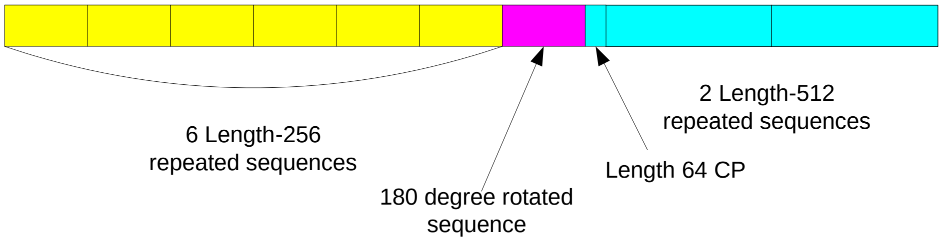


Figure 25 - PLCP preamble

The length-256 short training sequence shall be defined in the frequency domain using QPSK signal with inserted zeros. The frequency domain representation $S_{short,F}[.]$ is illustrated in Table 19.

Table 19 - Short training sequence in the frequency domain

n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$
0	0	0	128	1	-1	256	0	0	384	-1	-1
1	0	0	129	0	0	257	0	0	385	0	0
2	-1	1	130	-1	-1	258	0	0	386	1	-1
3	0	0	131	0	0	259	0	0	387	0	0
4	1	1	132	-1	1	260	0	0	388	-1	1
5	0	0	133	0	0	261	0	0	389	0	0
6	1	-1	134	-1	1	262	0	0	390	-1	-1
7	0	0	135	0	0	263	0	0	391	0	0
8	1	-1	136	1	-1	264	0	0	392	-1	1
9	0	0	137	0	0	265	0	0	393	0	0
10	1	-1	138	-1	1	266	0	0	394	1	-1
11	0	0	139	0	0	267	0	0	395	0	0
12	-1	1	140	1	1	268	0	0	396	-1	1
13	0	0	141	0	0	269	0	0	397	0	0
14	1	-1	142	-1	1	270	0	0	398	-1	-1
15	0	0	143	0	0	271	0	0	399	0	0
16	1	-1	144	1	1	272	0	0	400	-1	-1
17	0	0	145	0	0	273	0	0	401	0	0
18	-1	1	146	1	-1	274	0	0	402	1	1
19	0	0	147	0	0	275	0	0	403	0	0
20	-1	1	148	1	-1	276	0	0	404	-1	-1
21	0	0	149	0	0	277	0	0	405	0	0
22	-1	-1	150	1	1	278	0	0	406	1	1
23	0	0	151	0	0	279	0	0	407	0	0
24	1	1	152	1	-1	280	0	0	408	-1	1
25	0	0	153	0	0	281	0	0	409	0	0
26	-1	1	154	1	1	282	0	0	410	-1	1
27	0	0	155	0	0	283	0	0	411	0	0
28	1	-1	156	-1	1	284	0	0	412	1	1
29	0	0	157	0	0	285	0	0	413	0	0
30	-1	-1	158	-1	1	286	0	0	414	-1	1
31	0	0	159	0	0	287	0	0	415	0	0
32	-1	-1	160	1	-1	288	0	0	416	-1	-1

Table 19 - Short training sequence in the frequency domain (continued)

n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$
33	0	0	161	0	0	289	0	0	417	0	0
34	1	1	162	-1	-1	290	0	0	418	-1	-1
35	0	0	163	0	0	291	0	0	419	0	0
36	-1	-1	164	1	1	292	0	0	420	-1	-1
37	0	0	165	0	0	293	0	0	421	0	0
38	1	-1	166	1	-1	294	0	0	422	1	1
39	0	0	167	0	0	295	0	0	423	0	0
40	1	1	168	-1	1	296	0	0	424	1	-1
41	0	0	169	0	0	297	0	0	425	0	0
42	1	1	170	1	1	298	0	0	426	-1	1
43	0	0	171	0	0	299	0	0	427	0	0
44	-1	1	172	-1	1	300	0	0	428	1	-1
45	0	0	173	0	0	301	0	0	429	0	0
46	1	-1	174	-1	-1	302	0	0	430	1	1
47	0	0	175	0	0	303	0	0	431	0	0
48	-1	-1	176	1	-1	304	0	0	432	1	-1
49	0	0	177	0	0	305	0	0	433	0	0
50	1	-1	178	1	1	306	0	0	434	-1	1
51	0	0	179	0	0	307	0	0	435	0	0
52	1	1	180	0	0	308	0	0	436	-1	1
53	0	0	181	0	0	309	0	0	437	0	0
54	-1	-1	182	0	0	310	0	0	438	-1	-1
55	0	0	183	0	0	311	0	0	439	0	0
56	-1	-1	184	0	0	312	0	0	440	-1	-1
57	0	0	185	0	0	313	0	0	441	0	0
58	1	-1	186	0	0	314	0	0	442	-1	-1
59	0	0	187	0	0	315	0	0	443	0	0
60	1	-1	188	0	0	316	0	0	444	-1	-1
61	0	0	189	0	0	317	0	0	445	0	0
62	-1	1	190	0	0	318	0	0	446	1	-1
63	0	0	191	0	0	319	0	0	447	0	0
64	1	1	192	0	0	320	0	0	448	-1	1
65	0	0	193	0	0	321	0	0	449	0	0
66	-1	-1	194	0	0	322	0	0	450	1	1
67	0	0	195	0	0	323	0	0	451	0	0
68	1	-1	196	0	0	324	0	0	452	-1	1

Table 19 - Short training sequence in the frequency domain (continued)

n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$
69	0	0	197	0	0	325	0	0	453	0	0
70	1	-1	198	0	0	326	0	0	454	1	1
71	0	0	199	0	0	327	0	0	455	0	0
72	-1	-1	200	0	0	328	0	0	456	-1	1
73	0	0	201	0	0	329	0	0	457	0	0
74	-1	-1	202	0	0	330	0	0	458	1	-1
75	0	0	203	0	0	331	0	0	459	0	0
76	1	-1	204	0	0	332	0	0	460	-1	1
77	0	0	205	0	0	333	0	0	461	0	0
78	1	1	206	0	0	334	1	1	462	1	1
79	0	0	207	0	0	335	0	0	463	0	0
80	1	-1	208	0	0	336	1	1	464	1	-1
81	0	0	209	0	0	337	0	0	465	0	0
82	1	-1	210	0	0	338	-1	-1	466	1	1
83	0	0	211	0	0	339	0	0	467	0	0
84	1	-1	212	0	0	340	1	-1	468	-1	-1
85	0	0	213	0	0	341	0	0	469	0	0
86	1	-1	214	0	0	342	1	1	470	1	-1
87	0	0	215	0	0	343	0	0	471	0	0
88	-1	-1	216	0	0	344	-1	1	472	1	-1
89	0	0	217	0	0	345	0	0	473	0	0
90	1	1	218	0	0	346	-1	-1	474	1	-1
91	0	0	219	0	0	347	0	0	475	0	0
92	1	1	220	0	0	348	-1	1	476	-1	1
93	0	0	221	0	0	349	0	0	477	0	0
94	-1	-1	222	0	0	350	1	1	478	1	-1
95	0	0	223	0	0	351	0	0	479	0	0
96	-1	-1	224	0	0	352	1	-1	480	1	-1
97	0	0	225	0	0	353	0	0	481	0	0
98	1	-1	226	0	0	354	1	-1	482	1	-1
99	0	0	227	0	0	355	0	0	483	0	0
100	-1	-1	228	0	0	356	-1	-1	484	-1	-1
101	0	0	229	0	0	357	0	0	485	0	0
102	-1	-1	230	0	0	358	-1	-1	486	1	1
103	0	0	231	0	0	359	0	0	487	0	0
104	1	-1	232	0	0	360	1	1	488	1	-1

Table 19 - Short training sequence in the frequency domain (concluded)

n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$	n	$\mathrm{Re}\{S_{short,F}[n]\}$	$\mathrm{Im}\{S_{short,F}[n]\}$
105	0	0	233	0	0	361	0	0	489	0	0
106	1	-1	234	0	0	362	-1	-1	490	1	1
107	0	0	235	0	0	363	0	0	491	0	0
108	-1	1	236	0	0	364	-1	-1	492	1	1
109	0	0	237	0	0	365	0	0	493	0	0
110	-1	1	238	0	0	366	1	-1	494	1	-1
111	0	0	239	0	0	367	0	0	495	0	0
112	-1	-1	240	0	0	368	-1	1	496	-1	1
113	0	0	241	0	0	369	0	0	497	0	0
114	1	1	242	0	0	370	1	1	498	-1	1
115	0	0	243	0	0	371	0	0	499	0	0
116	-1	-1	244	0	0	372	1	1	500	1	-1
117	0	0	245	0	0	373	0	0	501	0	0
118	1	1	246	0	0	374	-1	1	502	1	-1
119	0	0	247	0	0	375	0	0	503	0	0
120	-1	1	248	0	0	376	-1	-1	504	-1	-1
121	0	0	249	0	0	377	0	0	505	0	0
122	1	-1	250	0	0	378	-1	1	506	1	-1
123	0	0	251	0	0	379	0	0	507	0	0
124	1	-1	252	0	0	380	1	-1	508	-1	-1
125	0	0	253	0	0	381	0	0	509	0	0
126	-1	1	254	0	0	382	1	1	510	-1	-1
127	0	0	255	0	0	383	0	0	511	0	0

The short training sequence, $S_{short,T}[.]$ is then determined by the inverse FFT of the $S_{short,F}[.]$. Specifically, let

$$S_{short\ ,T}[m] = \sum_{k=0}^{511} \frac{S_{short\ ,F}[k]}{\sqrt{356}} e^{\frac{j2\pi}{512}km} \tag{37}$$

where $S_{short,T}[m]$ represents the m^{th} element of the sequence $S_{short,T}[m]$ and $S_{short,F}[m]$ represents the k^{th} element of the sequence $S_{short,F}[m]$. Note that the $\sqrt{356}$ in (37) is the normalization factor due to the 356 non-zero QPSK symbols in the frequency representation.

The overall short training sequence shall be of length 1792 and be given by

$$S_{sync,A,OFDM}[n] = \begin{cases} T_{short,T}\left[(n \bmod 256) + 1\right] & n = 0,\ldots,1535 \\ -T_{short,T}\left[(n \bmod 256) + 1\right] & n = 1536,\ldots,1791 \end{cases} \quad (38)$$

The length 512 long training sequence is generated also in frequency domain using BPSK signal. The frequency domain representation $S_{long,F}[.]$ is illustrated in Table 20.

Table 20 - Frequency domain long training sequence

n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$
0	0	128	-1	256	0	384	-1
1	0	129	1	257	1	385	1
2	1	130	1	258	0	386	1
3	-1	131	-1	259	0	387	1
4	-1	132	1	260	0	388	-1
5	1	133	-1	261	0	389	1
6	1	134	-1	262	0	390	-1
7	1	135	-1	263	0	391	-1
8	1	136	1	264	0	392	-1
9	1	137	-1	265	0	393	1
10	1	138	-1	266	0	394	1
11	-1	139	-1	267	0	395	-1
12	-1	140	1	268	0	396	1
13	-1	141	1	269	0	397	1
14	-1	142	-1	270	0	398	1
15	1	143	-1	271	0	399	1
16	-1	144	-1	272	0	400	-1
17	1	145	1	273	0	401	-1
18	1	146	1	274	0	402	1
19	1	147	-1	275	0	403	-1
20	-1	148	1	276	0	404	-1
21	-1	149	-1	277	0	405	1
22	-1	150	1	278	0	406	-1
23	1	151	-1	279	0	407	1
24	1	152	-1	280	0	408	-1
25	1	153	-1	281	0	409	-1
26	1	154	1	282	0	410	-1
27	-1	155	-1	283	0	411	1
28	1	156	-1	284	0	412	-1
29	1	157	-1	285	0	413	-1
30	1	158	-1	286	0	414	1
31	1	159	1	287	0	415	-1

Table 20 - Frequency domain long training sequence (continued)

n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$
32	1	160	1	288	0	416	-1
33	-1	161	1	289	0	417	1
34	1	162	-1	290	0	418	-1
35	-1	163	-1	291	0	419	1
36	1	164	-1	292	0	420	-1
37	-1	165	1	293	0	421	1
38	1	166	-1	294	0	422	-1
39	1	167	1	295	0	423	1
40	1	168	1	296	0	424	-1
41	-1	169	-1	297	0	425	-1
42	-1	170	1	298	0	426	1
43	-1	171	1	299	0	427	-1
44	1	172	1	300	0	428	1
45	1	173	-1	301	0	429	1
46	1	174	1	302	0	430	1
47	-1	175	1	303	0	431	1
48	1	176	1	304	0	432	1
49	1	177	1	305	0	433	-1
50	1	178	-1	306	0	434	-1
51	-1	179	0	307	0	435	-1
52	-1	180	0	308	0	436	-1
53	1	181	0	309	0	437	1
54	1	182	0	310	0	438	-1
55	-1	183	0	311	0	439	-1
56	-1	184	0	312	0	440	-1
57	-1	185	0	313	0	441	1
58	1	186	0	314	0	442	-1
59	-1	187	0	315	0	443	1
60	1	188	0	316	0	444	1
61	-1	189	0	317	0	445	-1
62	-1	190	0	318	0	446	-1
63	-1	191	0	319	0	447	1
64	-1	192	0	320	0	448	1
65	1	193	0	321	0	449	-1
66	-1	194	0	322	0	450	-1
67	-1	195	0	323	0	451	-1
68	-1	196	0	324	0	452	1

Table 20 - Frequency domain long training sequence (continued)

n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$
69	-1	197	0	325	0	453	-1
70	1	198	0	326	0	454	-1
71	1	199	0	327	0	455	1
72	1	200	0	328	0	456	-1
73	-1	201	0	329	0	457	1
74	1	202	0	330	0	458	-1
75	-1	203	0	331	0	459	-1
76	1	204	0	332	0	460	-1
77	1	205	0	333	0	461	-1
78	1	206	0	334	1	462	1
79	-1	207	0	335	1	463	-1
80	-1	208	0	336	-1	464	-1
81	-1	209	0	337	-1	465	-
82	-1	210	0	338	-1	466	-1
83	1	211	0	339	1	467	1
84	-1	212	0	340	1	468	-1
85	1	213	0	341	1	469	1
86	-1	214	0	342	1	470	-1
87	1	215	0	343	1	471	-1
88	-1	216	0	344	-1	472	-1
89	-1	217	0	345	-1	473	1
90	1	218	0	346	1	474	-1
91	1	219	0	347	1	475	-1
92	-1	220	0	348	-1	476	1
93	-1	221	0	349	-1	477	1
94	1	222	0	350	-1	478	1
95	1	223	0	351	1	479	1
96	1	224	0	352	1	480	1
97	-1	225	0	353	1	481	-1
98	1	226	0	354	-1	482	-1
99	-1	227	0	355	-1	483	-1
100	1	228	0	356	-1	484	1
101	1	229	0	357	1	485	-1
102	-1	230	0	358	1	486	1
103	-1	231	0	359	-1	487	1
104	-1	232	0	360	1	488	1
105	-1	233	0	361	1	489	-1

Table 20 - Frequency domain long training sequence (concluded)

n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$	n	$S_{long,F}[n]$
106	1	234	0	362	1	490	1
107	-1	235	0	363	-1	491	-1
108	-1	236	0	364	-1	492	1
109	-1	237	0	365	1	493	-1
110	1	238	0	366	1	494	1
111	-1	239	0	367	1	495	-1
112	1	240	0	368	-1	496	1
113	1	241	0	369	-1	497	1
114	1	242	0	370	1	498	1
115	1	243	0	371	-1	499	1
116	-1	244	0	372	1	500	-1
117	-1	245	0	373	-1	501	-1
118	-1	246	0	374	1	502	-1
119	-1	247	0	375	1	503	1
120	-1	248	0	376	1	504	1
121	-1	249	0	377	-1	505	-1
122	1	250	0	378	-1	506	-1
123	-1	251	0	379	-1	507	-1
124	1	252	0	380	-1	508	1
125	1	253	0	381	-1	509	-1
126	1	254	0	382	-1	510	1
127	1	255	0	383	1	511	0

The long training sequence is determined by the inverse FFT of the $S_{long,F}$ given by

$$S_{long,T}[m] = \sum_{k=0}^{511} \frac{S_{long,F}[k]}{\sqrt{354}} e^{\frac{j2\pi}{512}km} \qquad (39)$$

where $S_{long,T}[m]$ represents the m^{th} element of the sequence $S_{long,T}[.]$ and $S_{long,F}[.]$ represents the k^{th} element of the sequence $S_{long,F}[.]$. Note that the $\sqrt{354}$ in equation (39) is the normalization factor due to the 354 non-zero QPSK symbols in the frequency representation.

10.2.3.4 PLCP header

For Type A OFDM frames, the formed PLCP header has a length of N_{header}. The PLCP header shall be first padded with zeros where the number of zero bits is 24. The padded header shall be demultiplexed and then shall be encoded with four encoders of rate 2/3 convolutional coding (labelled A through D) in the MSB branch before QPSK modulation (see 10.2.3.5.1.10), OFDM symbol padding (see 10.2.3.5.1.11), tone interleaver (see 10.2.3.5.1.13) and OFDM modulation (see 10.2.3.5.1.12), as illustrated in Figure 26. Let A(0), A(1), …, A(n), … be the bit stream at the output of encoder A with A(0) being the first bit, B(0), B(1), …, B(n), … be the bit stream at the output of encoder B with B(0) being the first bit, let C(0), C(1), …, C(n), … be the bit streams at the output of encoder C with C(0) being the first bit, D(0), D(1), …, D(n), … be the bit stream at the output of encoder D with D(0) being the first bit, the multiplexer in Figure 26 shall be implemented such that the output bit stream of the multiplexer is A(0), B(0), C(0), D(0), A(1), B(1), C(1), D(1), …, A(n), B(n), C(n), D(n),… with A(0) being the first bit.

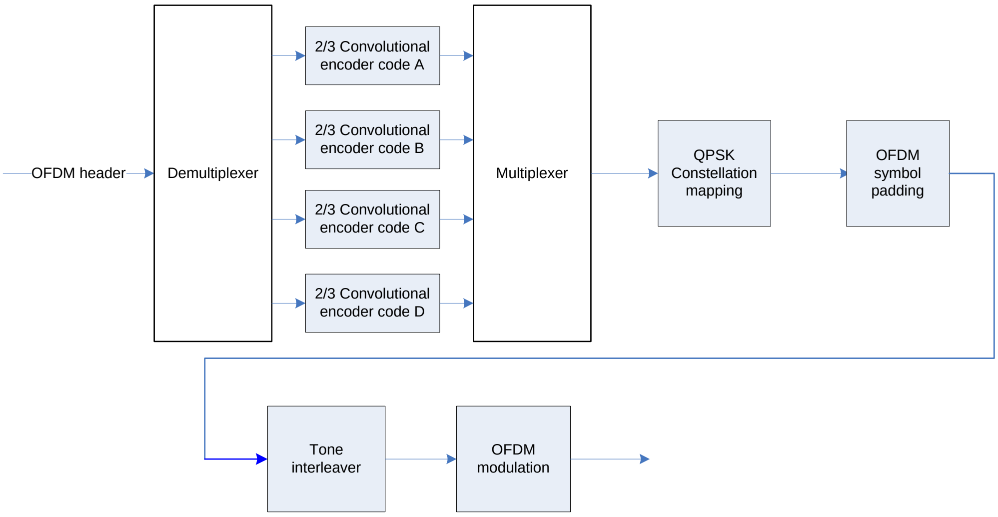


Figure 26 - OFDM block diagram for header operation

10.2.3.4.1 PHY header

10.2.3.4.1.1 Fixed length PHY header

10.2.3.4.1.1.1 PLCP SCRAMBLER FIELD (SCRAMBLER)

Refer to 10.2.2.4.1.1.1.

10.2.3.4.1.1.2 Bit reversal field (BIT_REVERSAL)

The BIT_REVERSAL bit shall be set to zero for all Type A OFDM frames.

10.2.3.4.1.1.3 ATIF existence field (ATIF_EX)

Refer to 10.2.2.4.1.1.3.

10.2.3.4.1.1.4 CP length field (CP_LENGTH)

The CP_LENGTH field D_1D_0 shall be set to 10_B, which corresponds to the cyclic prefix length of 64 for Type A OFDM.

10.2.3.4.1.1.5 Requested CP length field (REQ_CP_LENGTH)

The REQ_CP_LENGTH field D_1D_0 shall be set to 10_B.

10.2.3.4.1.1.6 Number of segments (NUM_SEGMENTS)

Refer to 10.2.2.4.1.1.6.

10.2.3.4.1.1.7 Number of MSDUs (NUM_MSDUS)

Refer to 10.2.2.4.1.1.7.

10.2.3.4.1.2 Variable length PHY header

10.2.3.4.1.2.1 Mode field (MODE)

Refer to 10.2.2.4.1.2.1.

10.2.3.4.1.2.2 Segment length field (LENGTH)

Refer to 10.2.2.4.1.2.2.

10.2.3.4.1.2.3 Midamble existence field (MID_EX)

Refer to 10.2.2.4.1.2.3.

10.2.3.4.1.2.4 Segment continued field (CONT)

Refer to 10.2.2.4.1.2.4.

10.2.3.4.1.3 Antenna training indicator field

10.2.3.4.1.3.1 Discovery mode repetition field (DISC_REP)

Refer to 10.2.2.4.1.3.1.

10.2.3.4.1.3.2 Number of training symbols for RX training field (NUM_RXTS)

Refer to 10.2.2.4.1.3.2.

10.2.3.4.1.3.3 Number of training symbols for TX training field (NUM_TXTS)

Refer to 10.2.2.4.1.3.3.

10.2.3.5 PPDU payload

The PPDU payload consists of one or more segments as described in 10.1. The segments are formed as described in 10.1.3. Each data bit block shall be processed as follows:

1. The bits shall be encoded and mapped as described in 10.2.3.5.1.
2. The resulting symbols shall be modulated into OFDM symbol as described in 10.2.3.5.1.12.

10.2.3.5.1 FEC and mapping

The block diagram of the OFDM PHY for modes A14 to A21 is shown in Figure 27. The information bits are first split into two streams as described in 10.2.3.5.1.1. Then the bits shall be scrambled as described in 10.2.3.5.1.2. Then, $N_{padbits,OFDM}$ zero pad bits shall be appended to the end of the data bit block as described in 10.2.3.5.1.3. The padded data bit block shall be encoded according to the Reed-Solomon code as described in 10.2.3.5.1.4.

The resulting RS coded payload bits shall be interleaved by the outer interleaver as described in 10.2.3.5.1.5. The resulting bits shall be encoded using eight parallel convolutional encoders as described in 10.2.3.5.1.7. Depending on the data rate mode, puncturing may be performed as described in 10.2.3.5.1.8.

For EEP modes (A14 to A16), the encoded bits from the 8 branches shall be multiplexed into a single data stream as described in 10.2.3.5.1.9.1 and interleaved as described in 10.2.3.5.1.9.2. For UEP coding (A17 and A18), the encoded bits from the 8 branches shall be multiplexed into a single data stream as described in 10.2.3.5.1.9.3 and interleaved as described in 10.2.3.5.1.9.4. For UEP mapping (A19 and A20), the encoded bits from the 8 branches shall be multiplexed into a single data stream as described in 10.2.3.5.1.9.5 and interleaved as described in 10.2.3.5.1.9.6. For MSB-only transmission (A21), the upper branch that consists of MSB RS encoder, outer interleaver, demultiplexer and convolutional encoders A through D as shown in Figure 27 shall be used. The data MUX shall only use the bits from the upper branch.

The resulting bits shall then be mapped to the appropriate constellation as described in 10.2.3.5.1.10 prior to the bit reversal tone interleaver as described in 10.2.3.5.1.13.

The QPSK and 16 QAM symbols shall be modulated with OFDM modulation as described in 10.2.3.5.1.12.

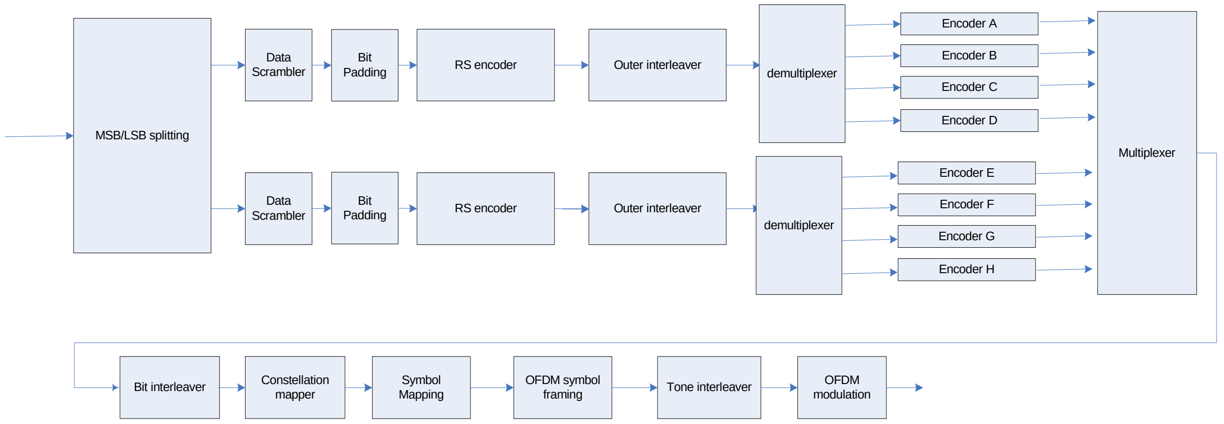


Figure 27 - Block diagram of the OFDM PHY baseband

10.2.3.5.1.1 MSB/LSB splitting

Refer to 10.2.2.5.2.1.

10.2.3.5.1.2 Data scrambler

Refer to 10.2.2.5.1.1.

10.2.3.5.1.3 Bit padding

$N_{padbits,OFDM}$ zero pad bits shall be appended to the end of the data bit block, where

$$N_{padbits,OFDM} = 1792\left\lceil \frac{N_{bits,OFDM}}{1792} \right\rceil - N_{bits,OFDM} \tag{40}$$

and $N_{bits,OFDM}$ is the number of bits in the data bit block. The padded data bit block shall be encoded according to the Reed-Solomon code, as specified in 10.2.3.5.1.4.

10.2.3.5.1.4 Reed-Solomon encoder

Refer to 10.2.2.5.1.2.

10.2.3.5.1.5 Outer interleaver

Let N_{OPi} be the interleaver size in bits. Interleaving encoders shall have an interleaving depth 4, leading to an interleaver size of $N_{OPi} = 32N_{RS} = 8064$ bits, where N_{RS} is number of bytes in a RS codeword.

Let $X = 0,...,N_{OPi}\text{-}1$ be the ordering of the original bits entering the interleaver. Let $P = 2^{\lceil \log_2 N_{OPi} \rceil}$ be the smallest power of 2 that is greater than or equal to the interleaver size and let $S = 0,...,P\text{-}1$. The ordering of bits leaving the interleaver shall be defined as the permutation

$$Y = prune\left(\left(\frac{S(S+1)}{2}\right) \bmod P\right) \tag{41}$$

where $prune(.)$ is the operation that removes all elements that are larger than N_{OPi}-1 from the input sequence without changing the order of the remaining elements.

10.2.3.5.1.6 Demultiplexing

For the first outer interleaver which accepts inputs from the first RS encoder, the output bit streams shall be demultiplexed to encoders A, B, C, D (see 10.2.3.5.1.7) in a round robin manner. Let bit 0 be the first incoming bit in the bit stream. With i=0, 1…,n, …, the bit streams at the input of encoder A shall be $b[3]$, $b[7]$, …, $b[4n+3]$ where $b[3]$ is the first incoming bit at the input of encoder A. The bit streams at the input of encoder B shall be $b[2]$, $b[6]$, …, $b[4n+2]$ where $b[2]$ is the first incoming bit at the input of encoder B. The bit streams at the input of encoder C shall be $b[1]$, $b[5]$, …, $b[4n+1]$ where $b[1]$ is the first incoming bit at the input of encoder C. The bit streams at the input of encoder D shall be $b[0]$, $b[4]$, …, $b[4n]$ where $b[0]$ is the first incoming bit at the input of encoder D.

For the second outer interleaver which accepts inputs from the second RS encoder, the output bit streams shall be demultiplexed to encoders E, F, G, H (see 10.2.3.5.1.7) in a round robin manner. Let bit 0 be the first incoming bit in the bit stream. With i=0, 1…,n, …, the bit streams at the input of encoder E shall be $b[3]$, $b[7]$, …, $b[4n+3]$ where $b[3]$ is the first incoming bit at the input of encoder E. The bit streams at the input of encoder F shall be $b[2]$, $b[6]$, …, $b[4n+2]$ where $b[2]$ is the first incoming bit at the input of encoder F. The bit streams at the input of encoder G shall be $b[1]$, $b[5]$, …, $b[4n+1]$ where $b[1]$ is the first incoming bit at the input of encoder G. The bit streams at the input of encoder H shall be $b[0]$, $b[4]$, …, $b[4n]$ where $b[0]$ is the first incoming bit at the input of encoder H.

10.2.3.5.1.7 Parallel convolutional codes

A parallel of 8 convolutional encoders shall be used, as depicted in Figure 28. The OFDM PHY transmitter uses eight parallel convolutional encoders, labelled A through H. The first four encoders, labelled A through D, is for the first outer Reed-Solomon coding branch and the last four encoders, labelled E through H is for the second outer Reed-Solomon coding branch.

For each of the 8 parallel convolutional codes, a mother code of rate 1/3 shall be used. The convolutional encoder shall use constraint length K = 7, delay memory 6, generator polynomial $g_0 = 133_O$, $g_1 = 171_O$, $g_2 = 165_O$, mother code rate 1/3. A detailed schematic diagram of the convolutional encoder is shown in Figure 29. The initial value of the delay register shall be zero at the beginning of every OFDM frame.

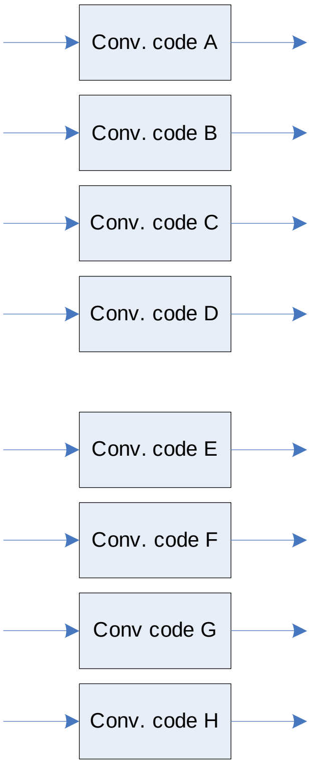


Figure 28 - Illustration of parallel convolutional codes

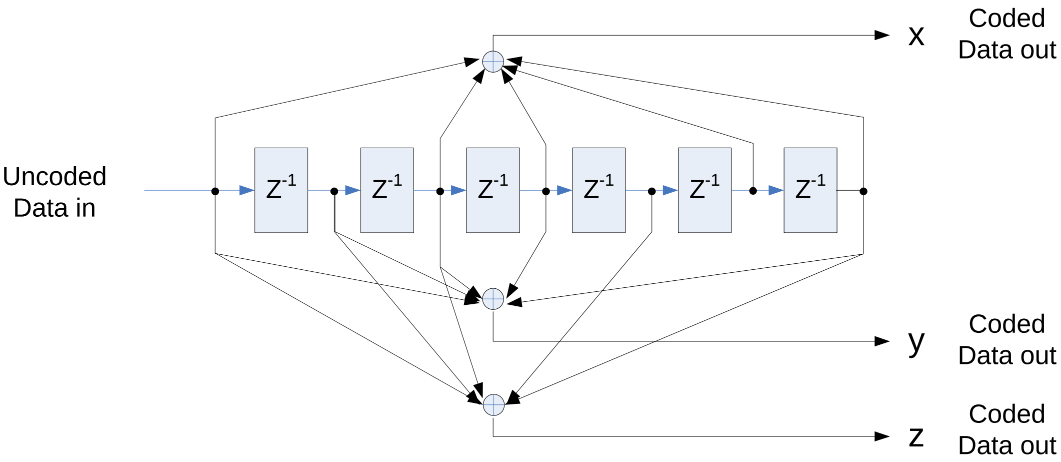


Figure 29 - Mother convolutional code generator

10.2.3.5.1.8 Code puncturing

Convolutional encoded data is punctured to generate code rates 4/7, 2/3, and 4/5. The puncturing patterns are illustrated in Figure 30, Figure 31, and Figure 32. In each of these cases, the tables shall be filled in with encoder output bits from left to right. For the last block of bits, the process shall be stopped at the point at which encoder output bits are exhausted and the puncturing pattern is applied to the partially filled block.

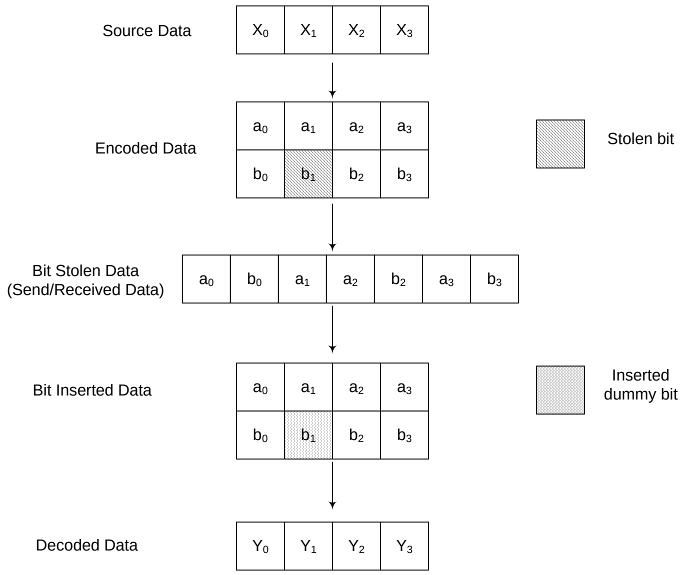


Figure 30 - An example of bit stealing and bit insertion for R=4/7 code

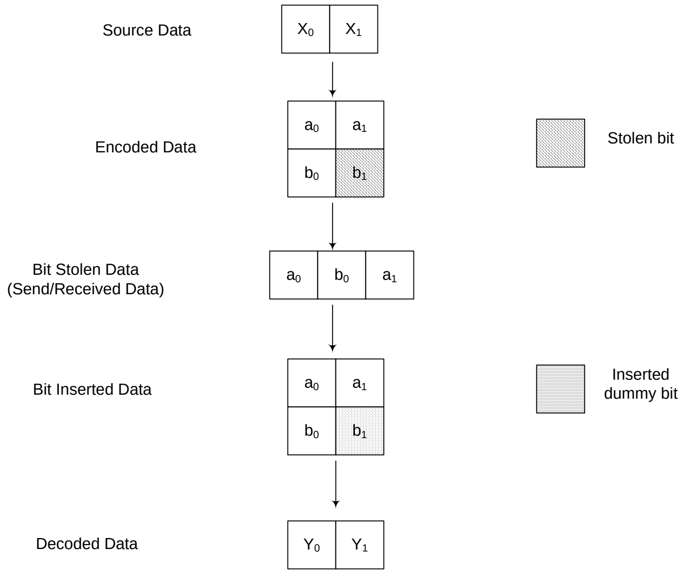


Figure 31 - An example of bit stealing and bit insertion for R=2/3 code

Figure 32 - An example of bit stealing and bit insertion for R=4/5 code

10.2.3.5.1.9 Data multiplexer and bit interleaver

The output of the 8 encoders, labelled A through H, shall be multiplexed to a single data stream prior to the bit interleaver, as illustrated in Figure 33. The method used to multiplex the encoded bits is dependent on the type of OFDM PHY mode.

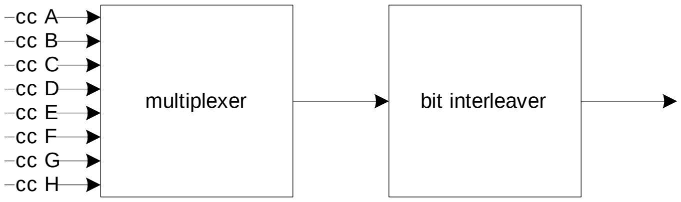


Figure 33 - Illustration of the multiplexer and bit interleaver

10.2.3.5.1.9.1 Multiplexer in EEP modes

In the EEP modes (A14 to A16), all the 8 encoders shall use the same code rate. The encoded bits shall be multiplexed and bit-interleaved every 48 bits. During the length 48 multiplexing/interleaving cycle, a group multiplexer shall be used first with fixed group size 6 for all eight encoders. Use A(1), A(2), A(3), A(4), A(5), A(6) to label the 6 encoded bits (in increasing order in time) from encoder A, and similarly B(1) through B(6), C(1) through C(6), D(1) through D(6), E(1) through E(6), F(1) through F(6), G(1) through G(6), and H(1) through H(6) from encoders B, C, D, E, F, G, and H, respectively. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Table 21, where the numbering for each bit represents the position of the bit in the bit stream after the multiplexer and bit 0 is the first bit at the output of the multiplexer.

Table 21 - Input pattern for the EEP modes (A14 to A16)

Numbering	0	1	2	3	4	5	6	7	8	9	10	11
Labelling	A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	B(1)	B(2)	B(3)	B(4)	B(5)	B(6)

Table 21 - Input pattern for the EEP modes (A14 to A16) (concluded)

Numbering	12	13	14	15	16	17	18	19	20	21	22	23
Labelling	C(1)	C(2)	C(3)	C(4)	C(5)	C(6)	D(1)	D(2)	D(3)	D(4)	D(5)	D(6)
Numbering	24	25	26	27	28	29	30	31	32	33	34	35
Labelling	E(1)	E(2)	E(3)	E(4)	E(5)	E(6)	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
Numbering	36	37	38	39	40	41	42	43	44	45	46	47
Labelling	G(1)	G(2)	G(3)	G(4)	G(5)	G(6)	H(1)	H(2)	H(3)	H(4)	H(5)	H(6)

10.2.3.5.1.9.2 Bit interleaver in EEP modes

The multiplexed 48 bits are then sent to the bit interleaver. Let $x = 0, \ldots, 47$ and $y = 0, \ldots, 47$ be the index at the input and output of the bit interleaver, respectively. The bit interleaver in the EEP mode shall implement the following relation:

$$y = \left[6\left\lfloor \frac{x}{6} \right\rfloor - 5(x \bmod 6) \right] \bmod 48 \qquad (42)$$

10.2.3.5.1.9.3 Multiplexer in UEP coding modes

In the UEP coding mode, top 4 encoders (or encoders A, B, C, D) shall use rate 4/7 convolutional codes, and bottom 4 encoders (or encoders E, F, G, H) shall use rate 4/5 convolutional codes. The encoded bits shall be multiplexed and bit interleaved every 48 bits. A group multiplexer with group size 7, 7, 7, 7, 5, 5, 5, 5 for all eight encoders, A through G, respectively, shall be used. Use A(1), A(2), A(3), A(4), A(5), A(6), A(7) to label the 7 encoded bits (in increasing order in time) from encoder A, and similarly B(1) through B(7), C(1) through C(7), D(1) through D(7), E(1) through E(5), F(1) through F(5), G(1) through G(5), and H(1) through H(5) from encoders B, C, D, E, F, G, and H, respectively. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Table 22, where the numbering for each bit represents the position of the bit in the bit stream after the multiplexer and bit 0 is the first bit at the output of the multiplexer.

Table 22 - Input pattern for the UEP coding modes

Numbering	0	1	2	3	4	5	6	7	8	9	10	11
Labelling	A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	A(7)	B(1)	B(2)	B(3)	B(4)	B(5)
Numbering	12	13	14	15	16	17	18	19	20	21	22	23
Labelling	B(6)	B(7)	C(1)	C(2)	C(3)	C(4)	C(5)	C(6)	C(7)	D(1)	D(2)	D(3)
Numbering	24	25	26	27	28	29	30	31	32	33	34	35
Labelling	D(4)	D(5)	D(6)	D(7)	E(1)	E(2)	E(3)	E(4)	E(5)	F(1)	F(2)	F(3)
Numbering	36	37	38	39	40	41	42	43	44	45	46	47
Labelling	F(4)	F(5)	G(1)	G(2)	G(3)	G(4)	G(5)	H(1)	H(2)	H(3)	H(4)	H(5)

10.2.3.5.1.9.4 Bit interleaver in UEP coding modes

The multiplexed 48 bits are then sent to the bit interleaver. Let $x = 0, \ldots, 47$ be the index at the input of the bit interleaver, and $y = 0, \ldots, 47$ be the index at the output of the bit interleaver. The bit interleaver in the first half cycle of the UEP coding mode shall implement the following relation:

$$y = \left[6 \left\lfloor \frac{x}{6} \right\rfloor - 5(x \bmod 6) \right] \bmod 48 \tag{43}$$

10.2.3.5.1.9.5 Multiplexer in UEP mapping modes

In the UEP mapping mode, all eight encoders shall use the same coding rate, but encoded bits from the top 4 encoders (or encoders A, B, C, D) shall be mapped to the I branch, and encoded bits from bottom 4 encoders (or encoders E, F, G, H) shall be mapped to the Q branch. In this case, the encoded bits shall be multiplexed and bit interleaved every 48 bits. During the length 48 multiplexing/interleaving cycle, a group multiplexer shall be used with fixed group size 6 for all eight encoders. Use A(1), A(2), A(3), A(4), A(5), A(6) to label the 6 encoded bits (in increasing order in time) from encoder A, and similarly B(1) through B(6), C(1) through C(6), D(1) through D(6), E(1) through E(6), F(1) through F(6), G(1) through G(6), and H(1) through H(6) from encoders B, C, D, E, F, G, and H, respectively. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Table 23, where the numbering for each bit represents the position of the bit in the bit stream after the multiplexer and bit 0 is the first bit at the output of the multiplexer.

Table 23 - Input pattern for the UEP mapping modes

Numbering	0	1	2	3	4	5	6	7	8	9	10	11
Labelling	A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	B(1)	B(2)	B(3)	B(4)	B(5)	B(6)
Numbering	12	13	14	15	16	17	18	19	20	21	22	23
Labelling	C(1)	C(2)	C(3)	C(4)	C(5)	C(6)	D(1)	D(2)	D(3)	D(4)	D(5)	D(6)
Numbering	24	25	26	27	28	29	30	31	32	33	34	35
Labelling	E(1)	E(2)	E(3)	E(4)	E(5)	E(6)	F(1)	F(2)	F(3)	F(4)	F(5)	F(6)
Numbering	36	37	38	39	40	41	42	43	44	45	46	47
Labelling	G(1)	G(2)	G(3)	G(4)	G(5)	G(6)	H(1)	H(2)	H(3)	H(4)	H(5)	H(6)

10.2.3.5.1.9.6 Bit interleaver in UEP mapping modes

The multiplexed 48 bits are then sent to the bit interleaver. Let $x = 0, \ldots, 47$ and $y = 0, \ldots, 47$ be the index at the input and at the output of the bit interleaver. The bit interleaver in the UEP mapping mode shall implement the following relation

$$y = \begin{cases} 2\left(\left[6 \left\lfloor \frac{x}{6} \right\rfloor - 5(x \bmod 6) \right] \bmod 24 \right) & 0 \le x \le 23 \\ 2\left(\left[6 \left\lfloor \frac{x}{6} \right\rfloor - 5(x \bmod 6) \right] \bmod 24 \right) + 1 & 24 \le x \le 47 \end{cases} \tag{44}$$

10.2.3.5.1.10 Mapping

Refer to 10.2.4.

10.2.3.5.1.11 Symbol padding

The resulting data symbols shall then be appended with $N_{padsym,OFDM}$ zero symbols, where

$$N_{padsym} = 360\left\lceil \frac{N_{sym}}{360} \right\rceil - N_{sym} \tag{45}$$

10.2.3.5.1.12 OFDM PHY modulation

The subcarriers are numbered from -256 to 255. The subcarriers shall be arranged as indicated in Table 24.

Table 24 - OFDM subcarrier allocation

Subcarrier type	Subcarrier number, k
Null	k= -256,...,-190 and k=190,...,255
Pilots	k= -189, -164, -139, -114, -89, -64, -39, -14, 14, 39, 64, 89, 114, 139, 164, 189
DC	k= -1,0,1
Data	All other k

The stream of complex symbols from the modulation mapping is divided into groups of *360* complex numbers, numbered from $n = 0$ to $n = 359$ where $n = 0$ corresponds to the first complex number received in time. Each of the complex numbers are mapped sequentially to the subcarriers, skipping the pilots and DC subcarriers, beginning with $n = 0$ mapped to $k = -188$ and $n = 359$ mapped to $k = 188$.

10.2.3.5.1.13 Bit reversal tone interleaver

All modulated QPSK or QAM symbols shall be interleaved by a block interleaver with a block size corresponding to the size of FFT in a single OFDM symbol. The interleaver ensures that the adjacent data symbols are mapped onto separate subcarriers.

At the transmitter side, the interleaver permutation shall be defined as follows: Let k be the index of the tones (including data tones, pilot tones, DC tones and null tones) before permutation ranging between 0 and 511, and i be the index of the interleaved tones also ranging between 0 and 511 (including data tones, pilot tones, DC tones and null tones) after permutation. Let

$$k = \sum_{j=0}^{8} a_j 2^j \tag{46}$$

with $a_8 a_7 \ldots a_0$ being the binary representation of integer k. Then the binary representation of integer *i* can be written as $a_0 a_1 \ldots a_8$, i.e.,

$$i = \sum_{j=0}^{8} a_j 2^{n-1-j} \tag{47}$$

DC, null, and pilot tones shall be inserted in the bit-reversal position before the tone interleaver.

10.2.3.5.2 Midamble

The midamble sequence shall be identical to the channel estimation sequence, $S_{CE,A,OFDM}[.]$, which consists of the two long training sequence, with a cyclic prefix of length 64 to be added in front of the long training sequence, as described in 10.2.3.3.

10.2.4 Constellation mapping

This Clause describes the techniques for mapping the coded and interleaved binary data sequence onto a complex constellation. The constellation mapping shall be chosen according to Table 1.

10.2.4.1 BPSK

The coded and interleaved binary serial input data, $g[i]$ where $i=0,1,2,...$, shall be converted into a complex number representing one of the two BPSK constellation points. The output values, $v[k]$ where $k=0,1,2,...$, are formed by:

$$v[k] = K_{const}(I[k] + jQ[k]) \tag{48}$$

where $I[k]$ and $Q[k]$ are given by Table 25. The resulting constellation is illustrated in Figure 34. The normalization factor $K_{const}=1$ for a BPSK constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements.

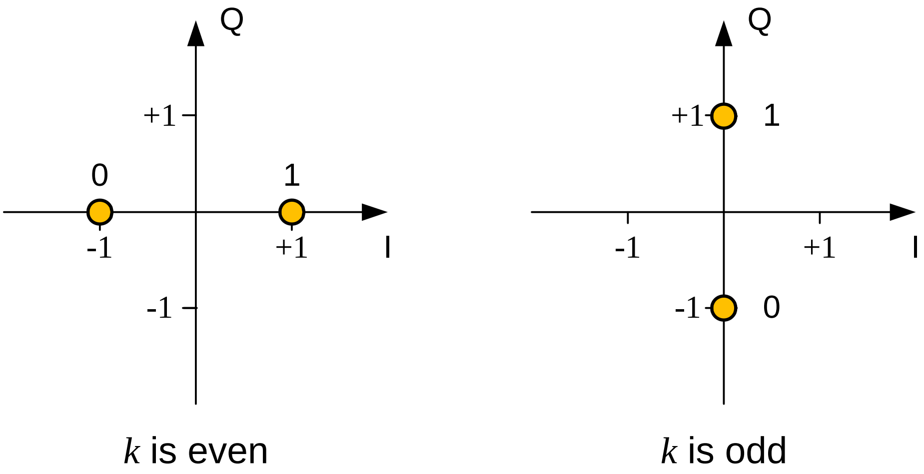


Figure 34 - BPSK constellation bit encoding

Table 25 - BPSK encoding table

	k is even		k is odd	
Input bit $g[k]$	$I[k]$	$Q[k]$	$I[k]$	$Q[k]$
0	-1	0	0	-1
1	1	0	0	1

10.2.4.2 QPSK

The coded and interleaved binary serial input data, $g[i]$ where $i=0,1,2,...$, shall be divided into groups of two bits and converted into a complex number representing one of the four QPSK constellation points. The conversion shall be performed according to the Gray-coded constellation mapping, illustrated in Figure 35 and described in Table 26, with the input bit, $g[2k]$ where $k=0,1,2,...$, being the earliest of the two in the stream. The output values, $v[k]$ where $k=0,1,2,...$, are formed by:

$$v[k] = K_{const}(I[k] + jQ[k]) \tag{49}$$

where $I[k]$ and $Q[k]$ are given by Table 26. The resulting constellation is illustrated in Figure 35. The normalization factor $K_{const} = 1/\sqrt{2}$ for a QPSK constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements. For QPSK, $g[2k]$ determines the I value, and $g[2k+1]$ determines the Q value, as illustrated in Table 26.

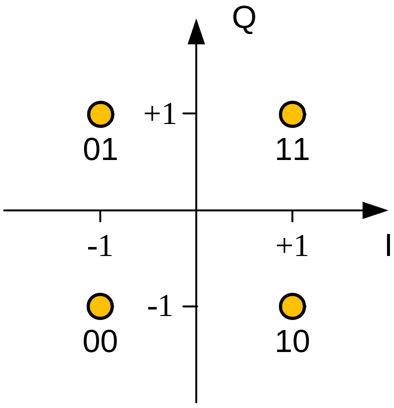


Figure 35 - QPSK constellation bit encoding

Table 26 - QSPK encoding table

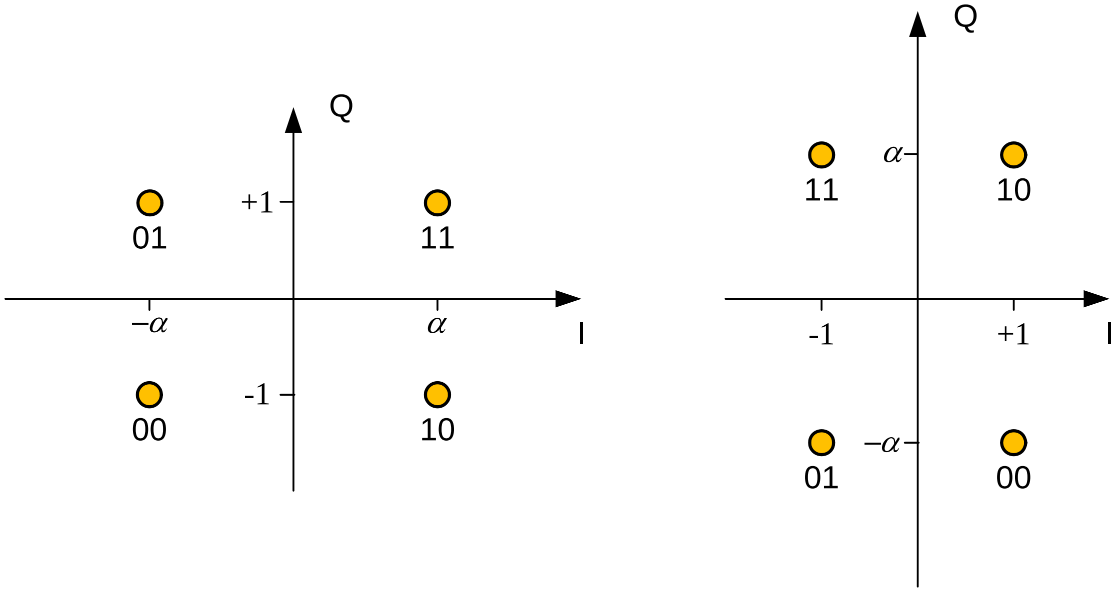
Input bits ($g[2k]$,$g[2k+1]$)	$I[k]$	$Q[k]$
00	-1	-1
01	-1	1
10	1	-1
11	1	1

10.2.4.3 UEP-QPSK

The coded and interleaved binary serial input data, $g[i]$ where $i=0,1,2,...$, shall be divided into groups of two bits and converted into a complex number representing one of the four UEP-QPSK constellation points. The conversion shall be performed according to the Gray-coded constellation mapping, illustrated in Figure 36 and described in Table 27, with the input bit, $g[2k]$ where $k=0,1,2,...$, being the earliest of the two in the stream. The output values, $v[k]$ where $k=0,1,2,...$, are formed by:

$$v[k]=K_{const}\left(I[k]+jQ[k]\right) \tag{50}$$

where $I[k]$ and $Q[k]$ are given by Table 27. The resulting constellation is illustrated in Figure 36. The value of α is 1.25. The normalization factor $K_{const}=1/\sqrt{1+\alpha^2}$ for the UEP-QPSK constellation. An approximate value of the normalization factor may be use, as long as the device conforms to the modulation accuracy requirements. For UEP-QPSK, $g[2k]$ determines the I value, and $g[2k+1]$ determines the Q value, as illustrated in Table 27.



k is even

k is odd

Figure 36 - UEP-QPSK constellation bit encoding

Table 27 - UEP-QPSK encoding table

	k is even		k is odd	
Input bits ($g[2k],g[2k+1]$)	$I[k]$	$Q[k]$	$I[k]$	$Q[k]$
00	-α	-1	1	-α
01	-α	1	-1	-α
10	α	-1	1	α
11	α	1	-1	α

10.2.4.4 NS8QAM

The coded and interleaved binary serial input data, $g[i]$ where $i=0,1,2,...$, shall be divided into groups of three bits and converted into a complex number representing one of the four NS8QAM constellation points. The conversion shall be performed according to constellation mapping illustrated in Figure 37 and described in Table 28, with the input bit, $g[3k]$ where $k= 0,1,2,...$, being the earliest of the three in the stream. The output values, $v[k]$ where $k= 0,1,2,...$, are performed by:

$$v[k] = K_{const}(I[k] + jQ[k]) \tag{51}$$

where $I[k]$ and $Q[k]$ are given by Table 28. The resulting constellation is illustrated in Figure 37. The normalization factor $K_{const} = 1/\sqrt{10}$ for a NS8QAM constellation.

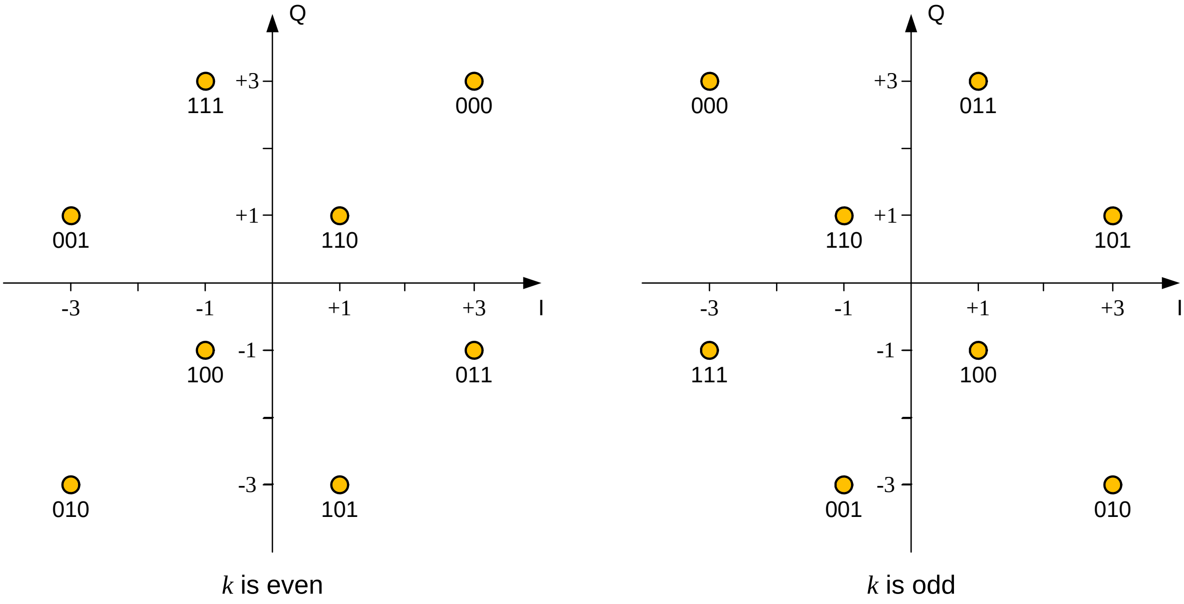


Figure 37 - NS8QAM constellation bit encoding

Table 28 - NSQAM encoding table

	k is even		*k* is odd	
Input bits ($g[3k]$, $g[3k+1]$, $g[3k+2]$)	$I[k]$	$Q[k]$	$I[k]$	$Q[k]$
000	3	3	-3	3
001	-3	1	-1	-3
010	-3	-3	3	-3
011	3	-1	1	3
100	-1	-1	1	-1
101	1	-3	3	1
110	1	1	-1	1
111	-1	3	-3	-1

10.2.4.5 16QAM

The coded and interleaved binary serial input data, $g[i]$ where $i=0,1,2,...$, shall be divided into groups of four bits and converted into a complex number representing one of the four 16QAM constellation points. The conversion shall be performed according to the constellation mapping, illustrated in Figure 38 and described in Table 29, with the input bit, $g[4k]$ where k=0,1,2,..., being the earliest of the four in the stream. The output values, $v[k]$ where k=0,1,2,..., are performed by:

$$v[k] = K_{const}(I[k] + jQ[k]) \tag{52}$$

where $I[k]$ and $Q[k]$ are given by Table 29. The resulting constellation is illustrated in Figure 38. The normalization factor $K_{const} = 1/\sqrt{10}$ for a 16QAM constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements. For 16QAM, $g[4k]$ determines the I value, and $g[4k+1]$ determines the Q value, as illustrated in Table 29.

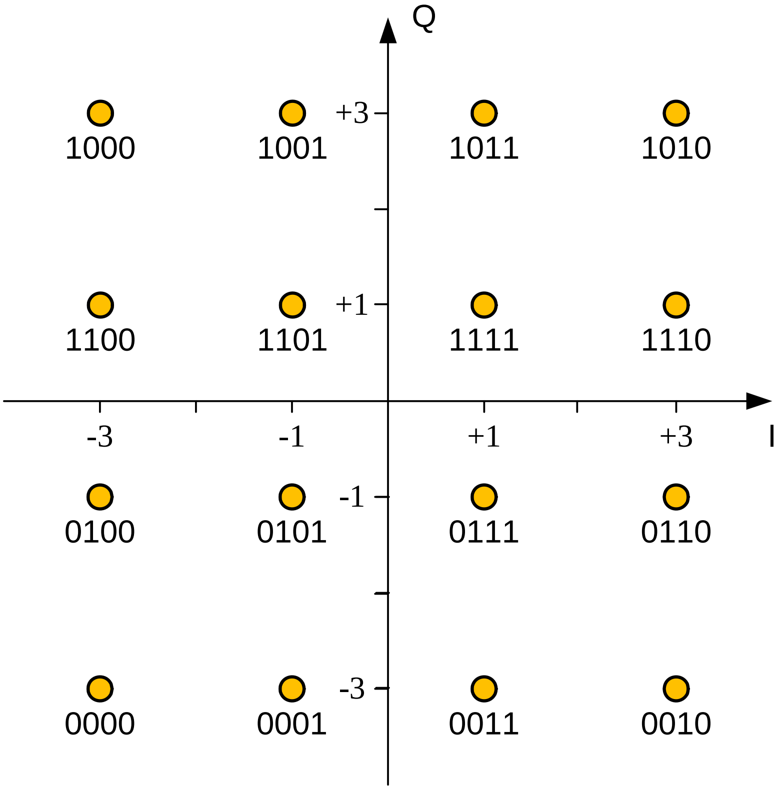


Figure 38 - 16QAM constellation bit encoding

Table 29 - 16QAM encoding table

Input bits ($g[4k]$,$g[4k+1]$,$g[4k+2]$,$g[4k+3]$)	$I[k]$	$Q[k]$
0000	-3	-3
0001	-1	-3
0010	3	-3
0011	1	-3
0100	-3	-1
0101	-1	-1

Table 29 - 16QAM encoding table (concluded)

Input bits ($g[4k]$,$g[4k+1]$,$g[4k+2]$,$g[4k+3]$)	$I[k]$	$Q[k]$
0110	3	-1
0111	1	-1
1000	-3	3
1001	-1	3
1010	3	3
1011	1	3
1100	-3	1
1101	-1	1
1110	3	1
1111	1	1

10.2.4.6 TCM-16QAM

The coded and interleaved binary serial input data, $g[i]$ where $i=0,1,2,...$, shall be divided into groups of four bits and converted into a complex number representing one of the four TCM-16QAM constellation points. The conversion shall be performed according to the constellation mapping, illustrated in Figure 39 and described in Table 30, with the input bit, $g[4k]$ where $k=0,1,2,...$, being the earliest of the four in the stream. The output values, $v[k]$ where $k=0,1,2,...$, are performed by:

$$v[k] = K_{const}(I[k] + jQ[k]) \tag{53}$$

where $I[k]$ and $Q[k]$ are given by Table 29. The resulting constellation is illustrated in Figure 39. The normalization factor $K_{const} = 1/\sqrt{10}$ for a TCM-16QAM constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements. For TCM-16QAM, $g[4k]$ determines the I value, and $g[4k+1]$ determines the Q value, as illustrated in Table 30.

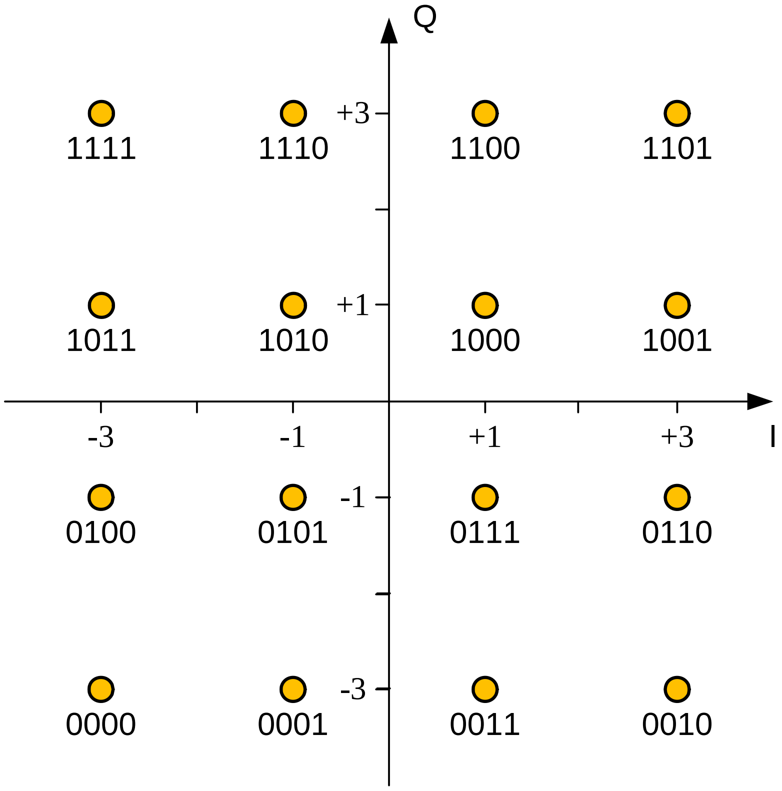


Figure 39 - TCM-16QAM constellation bit encoding

Table 30 - TCM-16QAM encoding table

Input bits $(g[4k],g[4k+1],g[4k+2],g[4k+3])$	$I[k]$	$Q[k]$
0000	-3	-3
0001	-1	-3
0010	3	-3
0011	1	-3
0100	-3	-1
0101	-1	-1
0110	3	-1
0111	1	-1
1000	1	1
1001	3	1
1010	-1	1
1011	-3	1
1100	1	3
1101	3	3
1110	-1	3

Table 30 - TCM-16QAM encoding table (concluded)

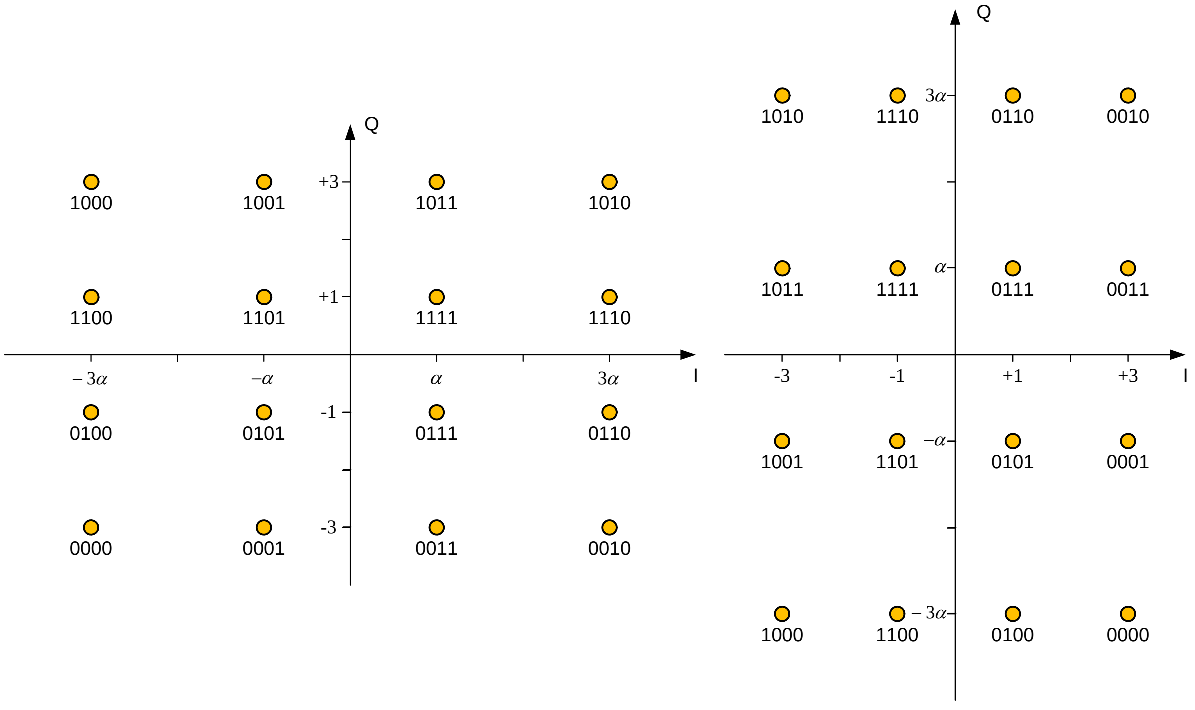
Input bits $(g[4k],g[4k+1],g[4k+2],g[4k+3])$	$I[k]$	$Q[k]$
1111	-3	3

10.2.4.7 UEP-16QAM

The coded and interleaved binary serial input data, $g[i]$ where $i=0,1,2,...$, shall be divided into groups of four bits and converted into a complex number representing one of the four UEP-16QAM constellation points. The conversion shall be performed according to the Gray-coded constellation mapping, specified in Table 31 and illustrated in Figure 40, with the input bit, $g[4k]$ where $k=0,1,2,...$, being the earliest of the four in the stream. The output values, $v[k]$ where $k=0,1,2,...$, are formed by:

$$v[k]=K_{const}(I[k]+jQ[k]) \tag{54}$$

where $I[k]$ and $Q[k]$ are given by Table 31. The resulting constellation is illustrated in Figure 40. The value of α is 1.25. The normalization factor is $K_{const}=1/\sqrt{5+5\alpha^2}$ for a UEP-16QAM constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements. For UEP-16QAM, $g[4k]$ determines the I value, and $g[4k+1]$ determines the Q value, as illustrated in Table 31.



k is even

k is odd

Figure 40 - UEP-16QAM constellation bit encoding

Table 31 - UEP-16QAM encoding table

	k is even		k is odd	
Input bits $(g[4k],g[4k+1],g[4k+2],g[4k+3])$	$I[k]$	$Q[k]$	$I[k]$	$Q[k]$
0000	-3α	-3	3	-3α
0001	$-\alpha$	-3	3	$-\alpha$
0010	3α	-3	3	3α
0011	α	-3	3	α
0100	$-\alpha$	-1	1	$-\alpha$
0101	$-\alpha$	-1	1	$-\alpha$
0110	3α	-1	1	3α
0111	α	-1	1	α
1000	-3α	3	-3	-3α
1001	$-\alpha$	3	-3	$-\alpha$
1010	3α	3	-3	3α
1011	α	3	-3	α
1100	-3α	1	-1	-3α
1101	$-\alpha$	1	-1	$-\alpha$
1110	3α	1	-1	3α
1111	α	1	-1	α

10.2.5 Discovery mode

The discovery mode is used for communications prior to training of antenna arrays. To compensate for the lack of array gain prior to training, the discovery mode increases the time-bandwidth product by repetition. The frame format for discovery mode shall follow the general frame format described in 10.1. Eight modes, D0 through D7, are defined. These modes are identical except for the number of repetitions (see Table 32).

Table 32 - Discovery modes

Mode	$N_{DISCREP}$	Data Rate (Mbps)
D0	128	2.255
D1	64	4.510
D2	32	9.020
D3	16	18.041
D4	8	36.082
D5	4	72.164
D6	2	144.327
D7	1	288.655

The value of the $N_{DISCREP}$ shall be set based on the requested value by the receiver. The initial value of $N_{DISCREP}$ shall be equal to 128 (mode D0).

10.2.5.1 Discovery mode PLCP preamble

The discovery mode preamble shall consist of the concatenation of a narrowband preamble, $S_{NB}[.]$, and a wideband preamble, $S_{WB}[.]$, as shown in Figure 41.

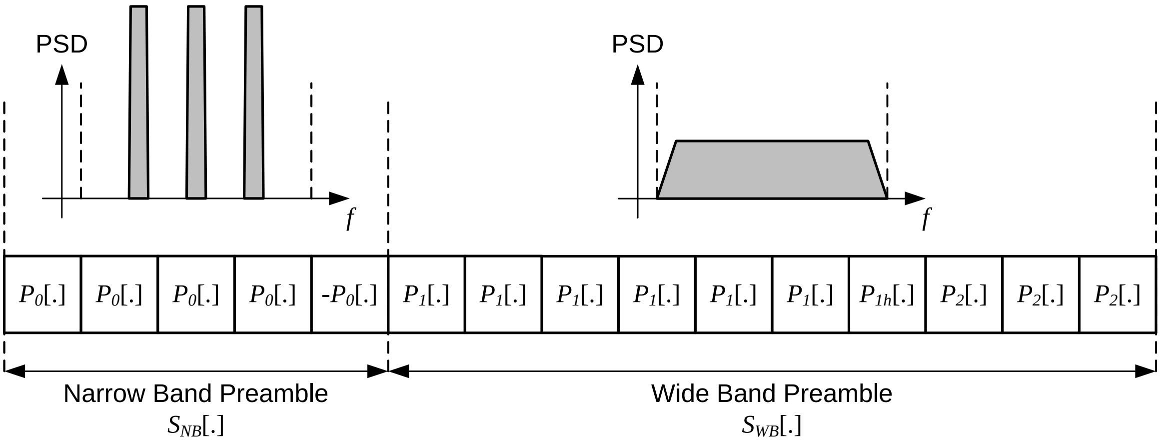


Figure 41 - Discovery mode preamble

The narrowband preamble, $S_{NB}[.]$, is defined as the Kronecker product of the sequence $S_0[.]$ and the a cover sequence $S_{cover,disc}[.]$. That is,

$$S_{NB}[n] = S_{cover,disc}\left[\left\lfloor \frac{n}{32768} \right\rfloor\right] S_0\left[n \bmod 32768\right] \qquad (55)$$

for n = 0,...,163839, where the cover sequence is of length five and is defined as

$$S_{cover,disc}[n] = \begin{cases} 1 & n = 0,\ldots,3 \\ -1 & n = 4 \end{cases} \qquad (56)$$

and $S_0[.]$ is a narrowband sequence of length 32768 and shall be obtained from $S_h[.]$ (defined in 10.2.2.3.1) by

$$S_0[n] = S_h\left[\left\lfloor \frac{n}{128} \right\rfloor\right] \qquad (57)$$

The wideband preamble, $S_{WB}[.]$, is the concatenation of six copies of $S_1[.]$, followed by one copy of sequence $S_{1h}[.]$, followed by three copies of sequence $S_2[.]$. That is,

$$S_{WB}[n] = \begin{cases} S_1[n \bmod 32768] & 0 \le n < 196608 \\ S_{1h}[n \bmod 32768] & 196608 \le n < 229376 \\ S_2[n \bmod 32768] & 229376 \le n < 327680 \end{cases} \qquad (58)$$

for n = 0,...,327679, where $S_1[.]$ is a wideband sequence of length 32768 and shall be obtained from $S_h[.]$ (defined in 10.2.2.3.1) by

$$S_1[n] = S_h[n \bmod 256] \tag{59}$$

$S_{1h}[.]$ is a wideband sequence of length 65536 and shall be obtained from $S_h[.]$ and the cover sequence $S_{h2}[.]$ by

$$S_{1h}[n] = S_h[n \bmod 256] S_{hFZ}\left[\left\lfloor \frac{n}{256} \right\rfloor\right] \tag{60}$$

where $S_{hFZ}[.]$ is a differentially encoded FZ sequence of length 256 (parameter A_{FZ}=16) defined by

$$S_{hFZ}[0] = S_{FZ,16}[0] \tag{61}$$

and

$$S_{h2}[n] = S_{h2}[n-1] S_{FZ,16}[n] \tag{62}$$

for n=1,...,255, and $S_2[.]$ is a wideband sequence of length 32768 and shall be obtained from $S_{FZ,16}[.]$ by

$$S_2[n] = S_{FZ,16}[n \bmod 256] \tag{63}$$

10.2.5.1.1 Three carrier modulation

The narrowband preamble, $S_{NB}[.]$, shall be modulated using three carriers, f_c, f_c+f_0, and f_c-f_0, as described in 9.2. The wideband preamble shall be modulated using one carrier, f_c, as described in 9.1.

10.2.5.2 Discovery mode PLCP header

The header for a discovery mode frame shall be formed based on the header of the Type A SCBT frames, as described in 10.2.2.4. After the formation of the header, each SCBT symbol of the header shall be repeated consecutively 128 times.

10.2.5.3 Discovery mode PPDU payload

The payload for a discovery mode frame is based on the payload of the Type A SCBT mode A0 frames, as described in 10.2.2.5. After the formation of the payload, each SCBT symbol of the payload shall be repeated consecutively $N_{DISCREP}$ times.

10.2.5.4 Discovery mode ATS

The ATS for a discovery mode frame is based on the ATS of the Type A frames, as described in 10.1.4. After the formation of the ATS, each block (of length 256) of the ATS shall be repeated consecutively $N_{DISCREP}$ times.

10.3 Type B PPDU

10.3.1 Mode dependent parameters

The Type B PSDU data rate-dependent modulation parameters are listed in Table 33.

Table 33 - PSDU mode dependent parameters

Mode	Base Data Rate (Gbps)				Modulation	Constellation	Encoding	TDSF (N_{TDS})
	no channel bonding	2 bonded channels	3 bonded channels	4 bonded channels				
B0	0.794	1.588	2.381	3.175	SC	DBPSK	RS & Diff	2
B1	1.588	3.175	4.763	6.350	SC	DBPSK	RS & Diff	1
B2	3.175	6.350	9.526	12.701	SC	DQPSK	RS & Diff	1
B3	3.175	6.350	9.526	12.701	SC	UEP-QPSK	RS	1
B4	3.175	6.350	9.526	12.701	DAMI	N/A	RS	1

All Type B devices shall support mode B0 (without channel bonding). Type B devices may support modes B0 (with channel bonding) or modes B1 through B4. In addition, all Type B devices shall support modes C0 and the transmission of mode A0 (without channel bonding). Type B devices may support modes C1 and C2. See 10.2 and 10.4 for Type A and Type C modes, respectively.

10.3.2 Single carrier (DBPSK, DQPSK, UEP-QPSK)

10.3.2.1 Timing related parameters

The timing parameters associated with the Type B PHY are listed in Table 34.

Table 34 - Timing related parameters

Parameter	Description	Value
f_{sym}	Symbol frequency	1.728 Gsps
T_{sym}	Symbol duration	0.5787 ns
N_B	Number of symbols per SC block	256
N_D	Number of data symbols per SC block	252
N_P	Number of pilot symbols per SC block	4
N_{CP}	Number of symbols in CP	0
T_{CP}	CP interval	0

10.3.2.2 Frame related parameters

The frame related parameters associated with the PHY are listed in Table 35.

Table 35 - Frame related parameters

Parameter	Description	Value
N_{sync}	Number of symbols in the frame synchronization sequence	2048
T_{sync}	Duration of the frame synchronization sequence	1185.19 ns

Table 35 - Frame related parameters (concluded)

Parameter	Description	Value
N_{CE}	Number of symbols in the channel estimation sequence	768
T_{CE}	Duration of the channel estimation sequence	444.444 ns
$N_{preamble}$	Number of symbols in the PLCP preamble	2816
$T_{preamble}$	Duration of the PLCP preamble	1629.63 ns
N_{ATS}	Number of symbols in the ATS	$256N_{RXTS}$
T_{ATS}	Duration of the ATS	$N_{ATS}T_{sym}$
N_{frame}	Number of symbols in the frame	$N_{sync}+N_{header}+N_{payload}+N_{ATS}$
T_{frame}	Duration of the frame	$(N_{sync}+N_{header}+N_{payload}+N_{ATS})T_{sym}$

10.3.2.3 PLCP preamble

The preamble for Type B SC frames can be subdivided into two distinct portions: a frame synchronization sequence, $s_{sync,B,SC}[.]$, and a channel estimation sequence, $s_{CE,B,SC}[.]$. Figure 42 shows the structure of the PLCP preamble for Type B SC frames.

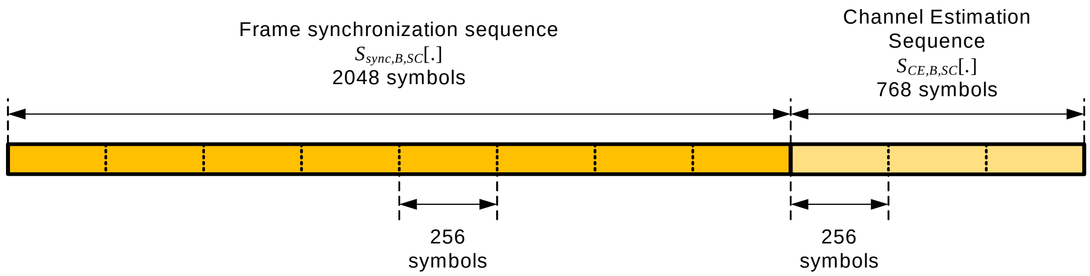


Figure 42 - Structure of the Type B PLCP preamble for Type B SC frames

Both the frame synchronization sequence and the channel estimation sequence are constructed based on the Frank-Zadoff sequences as defined in 10.2.2.3.

10.3.2.3.1 Frame synchronization sequence

The frame synchronization sequence for Type B is a modified hierarchical sequence $s_{h,B}[.]$, based on the hierarchical sequence $s_h[.]$ defined in 10.2.2.3.1.

The modified synchronization sequence $s_{h,B}[.]$ for Type B is obtained by adding the real and imaginary part of each term of sequence $s_h[.]$ as follows

$$s_{h,B}[n] = \mathrm{Re}\{s_h[n]\} + \mathrm{Im}\{s_h[n]\} \tag{64}$$

for $n=0,\ldots,255$.

The frame synchronization sequence for Type B SC frames is defined as the Kronecker product of the hierarchical sequence $s_{h,B}[.]$ and the cover sequence $s_{cover}[.]$ defined in 10.2.2.3.1. That is,

$$s_{sync,B,SC}[n] = s_{cover}\left[\left\lfloor \tfrac{n}{256} \right\rfloor\right] s_{h,B}[n \bmod 256] \tag{65}$$

for $n=0,\ldots,2047$.

10.3.2.3.2 Channel estimation sequence

The channel estimation sequence for Type B SC frames is a modified channel estimation sequence $S_{CE.B,SC}[.]$, based on the channel estimation sequence $S_{SC,A,SCBT}[.]$ defined in 10.2.2.3.2.

The channel estimation sequence for Type B SC frames, $S_{CE,B,SC}[n]$ is defined as:

$$S_{CE,B,SC}[n]=\begin{cases}1 & \mathrm{Re}\{S_{CE,A,SCBT}[n]\}>\mathrm{Im}\{S_{CE,A,SCBT}[n]\}\mathrm{or}\,\mathrm{Re}\{S_{CE,A,SCBT}[n]\}=\mathrm{Im}\{S_{CE,A,SCBT}[n]\}>0\\ -1 & \mathrm{Re}\{S_{CE,A,SCBT}[n]\}<\mathrm{Im}\{S_{CE,A,SCBT}[n]\}\mathrm{or}\,\mathrm{Re}\{S_{CE,A,SCBT}[n]\}=\mathrm{Im}\{S_{CE,A,SCBT}[n]\}<0\end{cases} \qquad (66)$$

for $n=0,...,767$.

NOTE
The frame synchronization sequence can be used for frame acquisition and detection, coarse carrier frequency estimation, coarse symbol timing, and for synchronization within the preamble. Whereas, the channel estimation sequence can be used for estimation of the channel frequency response, fine carrier frequency estimation, and fine symbol timing.

10.3.2.4 PLCP header

For Type B SC frames, the formed PLCP header (as described in 10.1.2) shall be first padded with N_{padhdr} bits where

$$N_{padhdr}=192\left\lfloor\frac{N_{hdrbits}}{192}\right\rfloor-N_{hdrbits} \qquad (67)$$

where $N_{hdrbits}$ is the number of bits in the formed PLCP header (as described in 10.1.2). The padded header shall be encoded and modulated as described in 10.3.2.5, starting with the demultiplexing. Encoding and modulation parameters identical to mode B0 shall be used. The resulting data symbols shall be block modulated (as described in 10.3.2.5.4) in order to create the baseband signal.

10.3.2.4.1 PHY header

10.3.2.4.1.1 Fixed length PHY header

10.3.2.4.1.1.1 PLCP scrambler field (SCRAMBLER)

Refer to 10.2.2.4.1.1.1.

10.3.2.4.1.1.2 Bit reversal field (BIT_REVERSAL)

The value of the BIT_REVERSAL bit shall be set to 0_B for all Type B SC frames.

10.3.2.4.1.1.3 ATIF existence field (ATIF_EX)

Refer 10.2.2.4.1.1.3.

10.3.2.4.1.1.4 CP length field (CP_LENGTH)

The value of the CP_LENGTH field shall be set to 00_B for all Type B SC frames.

10.3.2.4.1.1.5 Requested CP length field (REQ_CP_LENGTH)

The value of the REQ_CP_LENGTH field shall be set to 00_B for all Type B SC frames.

10.3.2.4.1.1.6 Number of segments field (NUM_SEGMENTS)

Refer to 10.2.2.4.1.1.6.

10.3.2.4.1.1.7 Number of MSDUs field (NUM_MSDUS)

Refer to 10.2.2.4.1.1.7.

10.3.2.4.1.2 Variable length PHY header

10.3.2.4.1.2.1 Mode field (MODE)

Refer to 10.2.2.4.1.2.1.

10.3.2.4.1.2.2 Segment length field (LENGTH)

Refer to 10.2.2.4.1.2.2.

10.3.2.4.1.2.3 Midamble existence field (MID_EX)

Refer to 10.2.2.4.1.2.3.

10.3.2.4.1.2.4 Segment continued field (CONT)

Refer to 10.2.2.4.1.2.4.

10.3.2.4.1.3 Antenna training indicator field

10.3.2.4.1.3.1 Discovery mode repetition field (DISC_REP)

Refer to 10.2.2.4.1.3.1.

10.3.2.4.1.3.2 Number of training symbols for RX training field (NUM_RXTS)

The value of the NUM_RXTS field for all Type B frames transmitted by a Type A device shall be set to 000000_B. For the value of the NUM_RXTS field for all Type B frames transmitted by a Type B device refer to 10.2.2.4.1.3.2.

10.3.2.4.1.3.3 Number of training symbols for TX training field (NUM_TXTS)

The value of the NUM_TXTS field for all Type B frames transmitted by a Type B device shall be set to 000000_B. For the value of the NUM_RXTS field for all Type A frames transmitted by a Type A device refer to 10.2.2.4.1.3.3.

10.3.2.5 PPDU payload

The PPDU payload consists of one or more segments as described in 10.1.3. The each segment is formed as described in 10.1.3. Each data bit block shall be processed as follows:

1. The bits shall be mapped as described in 10.3.2.5.1 or 10.3.2.5.2.
2. The resulting symbols shall be modulated into SC blocks as described in 10.3.2.5.4.

10.3.2.5.1 FEC and mapping (Equal error protection)

Figure 43 depicts the general overview of the encoding and mapping scheme for modes B0, B1, and B2. For the payload, first, the data bits shall be scrambled as specified in 10.3.2.5.1.1. Then, $N_{padbits}$ zero pad bits shall be appended to the end of the data bit block, where

$$N_{padbits} = 1972\left\lceil \frac{N_{bits}}{1792} \right\rceil - N_{bits} \tag{68}$$

and N_{bits} is the number of bits in the data bit block. The padded data bit block shall be encoded according to the Reed-Solomon code, as specified in 10.3.2.5.1.2.

The resulting RS coded payload bits or the formed header bits shall be demultiplexed into 4 groups, as described in 10.3.2.5.1.3. Each group of bits shall then be interleaved using the bit interleaver described in 10.3.2.5.1.4. The resulting bits from the 4 branches shall then be multiplexed together into one group of symbols as specified in 10.3.2.5.1.5. The multiplexed bits are then mapped to constellations as specified in 10.3.2.5.1.6. The data symbols shall be spread in time domain as described in 10.3.2.5.1.7. The resulting data symbols shall then be appended with N_{padsym} zero symbols, where

$$N_{padsym} = 2N_D \left\lceil \frac{N_{sym}}{2N_D} \right\rceil - N_{sym} \tag{69}$$

The resulting data symbols shall be interleaved with the dual helical scan (DHS) symbol interleaver as described in 10.3.2.5.1.8. The differential encoding shall then be applied as described in 10.3.2.5.1.9.

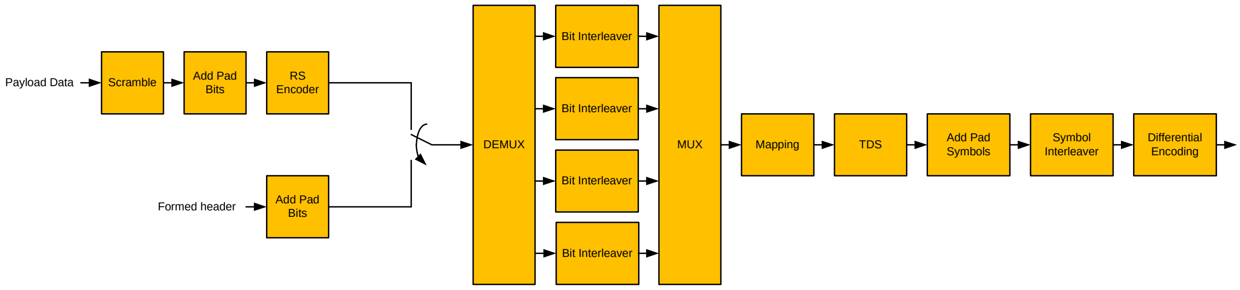


Figure 43 - Type B encoding procedure

10.3.2.5.1.1 Data scrambler

Refer to 10.2.2.5.1.1.

10.3.2.5.1.2 Reed-Solomon code

Refer to 10.2.2.5.1.2.

10.3.2.5.1.3 Demultiplexer

Refer to 10.2.2.5.1.3.

10.3.2.5.1.4 Bit interleaver

Refer to 10.2.2.5.1.4.

10.3.2.5.1.5 Multiplexer

Refer to 10.2.2.5.1.7.

10.3.2.5.1.6 Mapping

Refer to 10.2.2.5.1.8.

10.3.2.5.1.7 Time domain spreading (TDS)

Refer to 10.2.2.5.1.9.

10.3.2.5.1.8 Symbol interleaver

Refer to 10.2.2.5.1.10.

10.3.2.5.1.9 Differential encoding

For modes B0, B1 and B2, the padded data symbols shall be differentially encoded. The differently encoded data symbols $t[n]$ are formed from data symbols $v[n]$ by

$$t[n] = \begin{cases} v[n] & n \bmod N_D = 0 \\ t[n-1]v[n]/\left|v[n-1]\right| & n \bmod N_D > 0 \end{cases} \tag{70}$$

NOTE: As equation (70) specifies, the differential encoding shall be reset for every SC block. The last symbol of the channel estimation sequence or the last symbol of the pilot symbol sequence may be used to initialize the differential decoder.

10.3.2.5.2 FEC and mapping (Unequal error protection)

Figure 44 depicts the general overview of the encoding and mapping scheme for mode B3. For the payload, first, the MSB and LSB bits of the each octet is separated as described in 10.3.2.5.2.1. Then, for each group, the data bits shall be scrambled as specified in 10.3.2.5.2.2. Then, $N_{padbits}$ zero pad bits shall be appended to the end of the data bit block, where

$$N_{padbits} = 1792\left\lceil \frac{N_{bits}}{1792} \right\rceil - N_{bits} \tag{71}$$

and N_{bits} is the number of bits in the data bit block. The padded data bit block shall be encoded according to the Reed-Solomon code, as specified in 10.3.2.5.2.3.

The resulting RS coded payload bits or the formed header bits shall be demultiplexed in 4 groups, as described in 10.3.2.5.2.4. Depending on the data rate mode, each group of bits may then be interleaved using the bit interleaver described in 10.3.2.5.2.5. The resulting bits from all 8 branches shall then be multiplexed together into one group of symbols as specified in 10.3.2.5.2.6. The multiplexed bits are then mapped to constellations as specified in 10.3.2.5.3. The resulting data symbols shall then be appended with N_{padsym} zero symbols, where

$$N_{padsym} = 2N_D\left\lceil \frac{N_{sym}}{2N_D} \right\rceil - N_{sym} \tag{72}$$

The resulting data symbols shall be interleaved with the dual helical scan (DHS) symbol interleaver as described in 10.3.2.5.2.7.

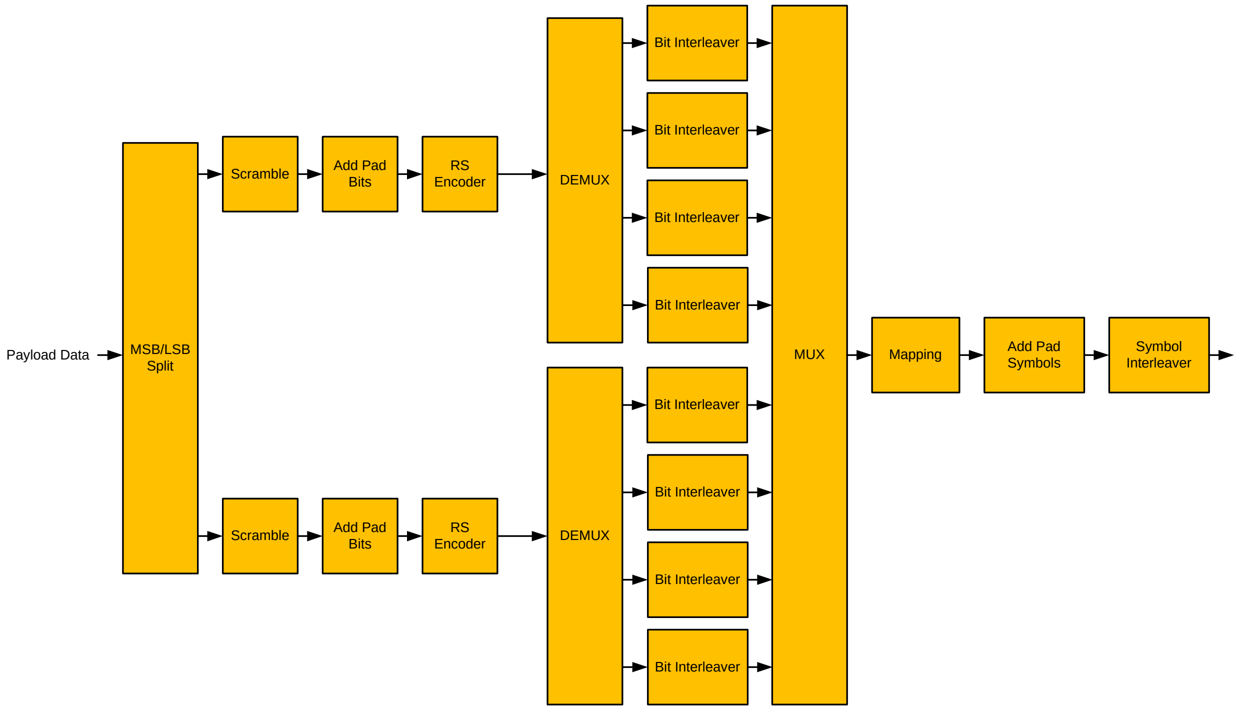


Figure 44 - General view of the encoding procedure

10.3.2.5.2.1 MSB/LSB separation

Refer to 10.2.2.5.2.1.

10.3.2.5.2.2 Data scrambler

Refer to 10.2.2.5.1.1.

10.3.2.5.2.3 Reed-Solomon code

Refer to 10.2.2.5.1.2.

10.3.2.5.2.4 Demultiplexer

Refer to 10.2.2.5.1.3.

10.3.2.5.2.5 Bit interleaver

Refer to 10.2.2.5.1.4.

10.3.2.5.2.6 Multiplexer

The multiplexer shall be identical to that described for mode A12 in 10.2.2.5.2.7.

10.3.2.5.2.7 Symbol interleaver

Refer to 10.2.2.5.1.8.

10.3.2.5.3 Constellation mapping

The constellation mapping for mode B0, B1, B2 and B3 shall be chosen according to Table 33.

10.3.2.5.3.1 DBPSK

The coded and interleaved binary serial input data, $g[i]$ where $i=1,2,\ldots$, shall be converted into a complex number representing one of the two DBPSK constellation points. The output values, $v[k]$ where $k=0,1,2,...$ are formed by:

$$v[k] = K_{const}\, I[k] \tag{73}$$

where $I[k]$ is given by Table 36. The resulting constellation is illustrated in Figure 45. The normalization factor $K_{const}=1$ for a DBPSK constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements. The differential encoding is described in 10.3.2.5.1.9 and is applied after the multiplexing and symbol padding as described in Table 36.

Table 36 DBPSK encoding table

Input bits $g[k]$	$I[k]$
0	-1
1	1

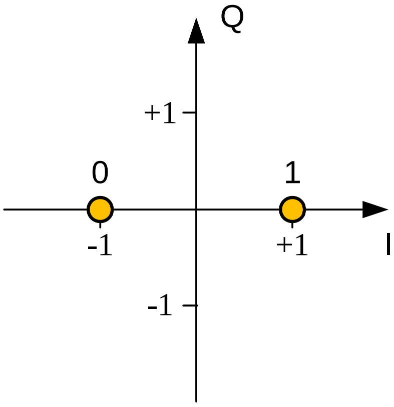


Figure 45 - DBPSK constellation bit encoding

10.3.2.5.3.2 DQPSK

The coded and interleaved binary serial input data, $g[i]$ where $i=1,2,\ldots$, shall be divided into groups of two bits and converted into a complex number representing one of the four DQPSK constellation points. The conversion shall be performed according to the Gray-coded constellation mapping, illustrated in Figure 46, with the input bit, $g[2k]$ where $k=1,2,\ldots$, being the earliest of the two in the stream. The output values, $v[k]$ where $k=1,2,\ldots$, are formed by:

$$v[k] = K_{const}\,(I[k] + jQ[k]) \tag{74}$$

where $I[k]$ and $Q[k]$ are given by Table 37. The resulting constellation is illustrated in Figure 46. The normalization factor $K_{const}=1$ for a DQPSK constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements. For DQPSK, $g[2k]$ determines the I value, and $g[2k+1]$ determines the Q value, as illustrated in Table 37.

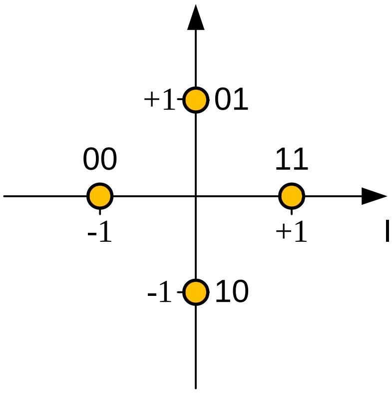


Figure 46 - DQPSK constellation bit encoding

Table 37 - DQPSK encoding table

Input bits ($g[2k]$, $g[2k+1]$)	$I[k]$	$Q[k]$
00	-1	0
01	0	1
10	0	-1
11	1	0

10.3.2.5.3.3 UEP-QPSK

Refer to 10.2.4.3.

10.3.2.5.4 Block modulation

The multiplexed transmit symbols, $t[.]$ shall be modulated formed into the SC block. First the transmit data symbols shall be divided into blocks of length N_D=252. This transmit symbol block shall be appended with pilot symbols. The pilot symbols consist of a sequence of length N_P=4, $S_{pilot}[n]=(-1)^{n+1}$, n=0,1,2,3. Figure 47 depicts an example of the SC block formation.

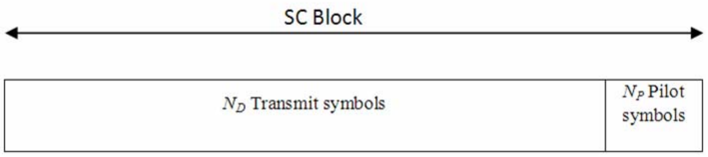


Figure 47 - Formation of the SC block

10.3.2.5.5 Midamble

The midamble sequence shall be identical to the channel estimation sequence, $S_{CE,B,SC}[.]$, as described in 10.3.2.3.2.

NOTE

The midamble sequence may be used to update estimation of the channel frequency response, fine carrier frequency estimation, and fine symbol timing.

10.3.3 Dual alternate mark inversion (DAMI)

The DAMI PPDU format shall follow the general PPDU described in 10.1 and it may consist of three major components, The PLCP preamble, the PLCP header, and the PPDU payload. DAMI frames shall not include the ATS. Furthermore, the payload may consist of zero or one segment. The segment shall not include a midamble.

10.3.3.1 Timing related parameters

The timing-related parameters associated with the DAMI PHY are listed in Table 38. It is noted that one symbol corresponds to one bit for DAMI devices, meaning that the symbol length is the same as the bit length.

Table 38 - Timing related parameters for DAMI PHY

Parameter	Description	Value
f_{sym}	Symbol frequency	3.456 Gsps
T_{sym}	Symbol duration	0.2894 ns

10.3.3.2 Frame related parameters

The frame-related parameters associated with the DAMI PHY are listed in Table 39.

Table 39 - Frame related parameters for DAMI PHY

Parameter	Description	Value
N_{sync}	Number of symbols in the frame synchronization sequence	2048
T_{sync}	Duration of the frame synchronization sequence	592.59 ns
N_{CE}	Number of symbols in the channel estimation sequence	768
T_{CE}	Duration of the channel estimation sequence	222.22 ns
$N_{preamble}$	Number of symbols in the PLCP preamble	2816
$T_{preamble}$	Duration of the PLCP preamble	814.81 ns
N_{frame}	Number of symbols in the frame	$N_{preamble}+N_{header}+N_{payload}$
T_{frame}	Duration of the frame	$N_{frame}T_{sym}$

10.3.3.3 PLCP preamble

10.3.3.3.1 Frame synchronization sequence

The frame synchronization sequence for DAMI consists of two hierarchical sequences, $s_{h,B,1}[.]$ and $s_{h,B,2}[.]$. Both are of length 256 and are defined based on the hierarchical sequence $s_h[.]$ defined in 10.2.2.3.1. That is,

$$
\begin{aligned}
&S_{\mathrm{h}}[n] = S_{\mathrm{FZ},4}[n \bmod 16]\, S_{\mathrm{FZ},4}\left[\left\lfloor \frac{n}{16} \right\rfloor\right] \\
&S_{\mathrm{h},1}[n] = \begin{cases} 1 & \mathrm{Re}\left(S_{\mathrm{h}}[n]\right) + \mathrm{Im}\left(S_{\mathrm{h}}[n]\right) \geq 0 \\ 0 & \mathrm{Re}\left(S_{\mathrm{h}}[n]\right) + \mathrm{Im}\left(S_{\mathrm{h}}[n]\right) < 0 \end{cases} \\
&\tilde{S}_{\mathrm{h},1}[n] = \tilde{S}_{\mathrm{h},1}[n-2] \oplus S_{\mathrm{h},1}[n] \text{ with } \tilde{S}_{\mathrm{h},1}[-2] = \tilde{S}_{\mathrm{h},1}[-1] = 0 \\
&S_{\mathrm{h,B},1}[n] = \tilde{S}_{\mathrm{h},1}[n] - \tilde{S}_{\mathrm{h},1}[n-2] \\
&S_{\mathrm{h},2}[n] = \begin{cases} 0 & \mathrm{Re}\left\{S_{\mathrm{h}}[n]\right\} + \mathrm{Im}\left\{S_{\mathrm{h}}[n]\right\} \geq 0 \\ 1 & \mathrm{Re}\left\{S_{\mathrm{h}}[n]\right\} + \mathrm{Im}\left\{S_{\mathrm{h}}[n]\right\} < 0 \end{cases} \\
&\tilde{S}_{\mathrm{h},2}[n] = \tilde{S}_{\mathrm{h},2}[n-2] \oplus S_{\mathrm{h},1}[n] \text{ with } \tilde{S}_{\mathrm{h},2}[-2] = \tilde{S}_{\mathrm{h},2}[-1] = 0 \\
&S_{\mathrm{h,B},2}[n] = \tilde{S}_{\mathrm{h},2}[n] - \tilde{S}_{\mathrm{h},2}[n-2]
\end{aligned} \tag{75}
$$

for $n = 0, 1, 2, \ldots, 255$, where $\oplus$ denotes the modulo-2 addition, and $\lfloor x \rfloor$ is the floor function which returns the largest integer smaller than x. The frame synchronization sequence, $S_{sync,B,DAMI}[.]$, is of length 2048, and is defined as the concatenation of seven repetitions of $S_{h,B,1}[.]$ with one repetition of $S_{h,B,2}[.]$. That is,

$$
S_{\mathrm{sync,B,DAMI}}[n] = \begin{cases} \sqrt{2} S_{\mathrm{h,B},1}[n \bmod 256] & \text{if } 0 \leq n \leq 1791 \\ \sqrt{2} S_{\mathrm{h,B},2}[n \bmod 256] & \text{if } 1792 \leq n \leq 2047 \end{cases}. \tag{76}
$$

10.3.3.3.2 Channel estimation sequence

The channel estimation sequence for DAMI, $S_{CE,B,DAMI}[.]$, consists of three repetitions of a modified Frank-Zadoff sequence, $\tilde{S}_{\mathrm{MFZ,\ B}}[\cdot]$, which is of length 256 and is defined as follows:

$$
\begin{aligned}
&S_{\mathrm{MFZ}}[n] = \begin{cases} 1 & \mathrm{Re}\left\{S_{\mathrm{FZ},16}[n]\right\} \geq \mathrm{Im}\left\{S_{\mathrm{FZ},16}[n]\right\} \\ 0 & \mathrm{Re}\left\{S_{\mathrm{FZ},16}[n]\right\} < \mathrm{Im}\left\{S_{\mathrm{FZ},16}[n]\right\} \end{cases} \\
&\tilde{S}_{\mathrm{MFZ}}[n] = \tilde{S}_{\mathrm{MFZ}}[n-2] \oplus S_{\mathrm{MFZ}}[n] \text{with } \tilde{S}_{\mathrm{MFZ}}[-2] = \tilde{S}_{\mathrm{MFZ}}[-1] = 0 \\
&S_{\mathrm{MFZ,\ B}} = \tilde{S}_{\mathrm{MFZ}}[n] - \tilde{S}_{\mathrm{MFZ}}[n-2]
\end{aligned} \tag{77}
$$

for $n = 0, 1, 2, \ldots, 255$. The channel estimation sequence, $S_{\mathrm{CE,B,DAMI}}[\cdot]$, is of length 768 and is defined as $S_{\mathrm{CE,B,DAMI}}[n] = \sqrt{2} S_{\mathrm{MFZ,B}}[n \bmod 256]$ for $n = 0,1,2,\ldots,767$.

10.3.3.4 PLCP header

10.3.3.4.1 PHY header

10.3.3.4.1.1 Fixed length PHY header

10.3.3.4.1.1.1 PLCP scrambler field (SRAMBLER)

Refer to 10.2.2.4.1.1.1.

10.3.3.4.1.1.2 Bit reversal field (BIT_REVERSAL)

The value of the BIT_REVERSAL field shall be set to 0_B for all Type B DAMI frames.

10.3.3.4.1.1.3 ATIF existence field (ATIF_EX)

The value of the ATIF existence field shall be set to 0_B for all Type B DAMI frames.

10.3.3.4.1.1.4 CP length field (CP_LENGTH)

The value of the CP_LENGTH field shall be set to 00_B for all Type B DAMI frames.

10.3.3.4.1.1.5 Requested CP length field (REQ_CP_LENGTH)

The value of the REQ_CP_LENGTH field shall be set to 00_B for all Type B DAMI frames.

10.3.3.4.1.1.6 Number of segments field (NUM_SEGMENTS)

The NUM_SEGMENTS field shall be set to $F_3F_2F_1F_0=0001_B$ for all Type B DAMI frames with payload or $F_3F_2F_1F_0=0000_B$ for Type B DAMI frames with no payload (e.g. ACK frames).

10.3.3.4.1.1.7 Number of MSDUs field (NUM_MSDUS)

The NUM_MSDUS field shall be set to $P_5P_4P_3P_2P_1P_0=000001_B$ for all Type B DAMI frames with payload or $P_5P_4P_3P_2P_1P_0=000000_B$ for Type B DAMI frames with no payload (e.g. ACK frames).

10.3.3.4.1.2 Variable length PHY header

10.3.3.4.1.2.1 Mode field (MODE)

The value of the mode field shall be set to $G_5G_4G_3G_2G_1G_0=101101_B$ for all Type B DAMI frames.

10.3.3.4.1.2.2 Segment length field (LENGTH)

Refer to 10.2.2.4.1.2.2.

10.3.3.4.1.2.3 Midamble existence field (MID_EX)

The value of the midamble existence field shall be set to 0_B for all Type B DAMI frames.

10.3.3.4.1.2.4 Segment continued field (CONT)

The value of the continued field shall be set to 0_B for all Type B DAMI frames.

10.3.3.4.1.3 Antenna training indicator field

The DAMI frames shall not include the ATIF.

10.3.3.5 PPDU payload

10.3.3.5.1 FEC and mapping

Figure 48 depicts the general overview of the encoding and mapping scheme for DAMI devices. First, N_{padbits} zero pad bits shall be appended to the end of the data block, where

$$N_{padbits} = 1792\left\lceil \frac{N_{bits}}{1792} \right\rceil - N_{bits} \tag{78}$$

where N_{bits} is the number of bits in the data block. The padded data block shall be encoded according to the Reed-Solomon (RS) code, as specified in 10.2.2.5.1.2. The resulting RS coded data block is then coded and mapped according the DAMI coding, as specified in 10.3.3.5.2.

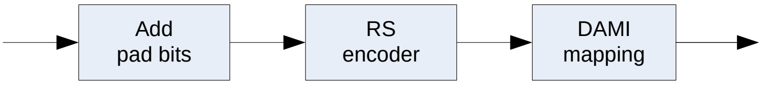


Figure 48 - General view of the encoding and mapping procedure for DAMI devices

10.3.3.5.2 DAMI coding

The coded binary serial input data, $b[k]$, where $k = 0, 1, 2, \ldots$, shall be first precoded to form an intermediate data $\hat{b}[k]$, defined as follows:

$\hat{b}[k] = \hat{b}[k-2] \oplus b[k]$ where $\oplus$ denotes the modulo-2 addition and two initial values $\hat{b}[-2] = \hat{b}[-1] = 0$ shall be used for the precoding. The output values, $d[k]$, are formed by $d[k] = K_{const} I[k]$ where $I[k]$ is given by Table 40. The resulting constellation is illustrated in Figure 49. The normalization factor $K_{const} = \sqrt{2}$ for DAMI constellation.

Table 40 - DAMI encoding table

Precoded input bits $\hat{b}[k-2], \hat{b}[k]$	$I[k]$
00	0
01	1
10	-1
11	0

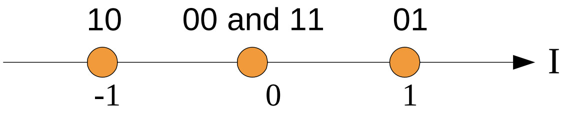


Figure 49 - DAMI constellation

10.3.3.5.3 Midamble

Type B DAMI frames shall not include a midamble.

10.3.4 Channel bonding

Type B devices may perform channel bonding. If a Type B device uses channel bonding, it shall be as described in 10.2.2.6.

NOTE

In Type B DAMI signal the pilots will shift as the T_{sym} changes according to 10.2.2.6.

10.3.5 Discovery mode

Type B devices shall not transmit frames in the discovery mode.

10.4 Type C PPDU

The Type C PPDU format shall follow the general PPDU described in 10.1 and it may consist of three major components, the PLCP preamble, the PLCP header, and the PPDU payload. Type C frames shall not include the ATS. Furthermore, the payload may consist of zero or one segment. The segment shall not include a midamble.

10.4.1 Mode dependent parameters

The PSDU data rate-dependent modulation parameters are listed in Table 41.

Table 41 - PSDU mode dependent parameters

Mode	Base Data Rate (Gbps)	Modulation	Constellation	Encoding	TDSF (N_{TDS})
C0	0.800	SC	OOK	RS	2
C1	1.600	SC	OOK	RS	1
C2	3.200	SC	4ASK	RS	1

All Type C devices shall support mode C0. Type C devices may support modes C1 and C2.

10.4.2 Timing related parameters

The timing parameters associated with the Type C PHY are listed in Table 42.

Table 42 - Timing related parameters

Parameter	Description	Value
f_{sym}	Symbol frequency	1.728 Gsps
T_{sym}	Symbol duration	0.5787 ns
N_D	Number of data symbols per SC block	$508N_{TDS}$
N_P	Number of pilot symbols per SC block	$4N_{TDS}$
T_{SCS}	SC block interval	$(N_D+N_P)T_{sym}$

10.4.3 Frame related parameters

The frame related parameters associated with the PHY are listed in Table 43.

Table 43 - Frame related parameters

Parameter	Description	Value
N_{sync}	Number of symbols in the frame synchronization sequence	4096
T_{sync}	Duration of the frame synchronization sequence	2370.37 ns
N_{CE}	Number of symbols in the channel estimation sequence	1536
T_{CE}	Duration of the channel estimation sequence	888.89 ns
$N_{preamble}$	Number of symbols in the PLCP preamble	5632
$T_{preamble}$	Duration of the PLCP preamble	3259.26 ns
N_{frame}	Number of symbols in the frame	$N_{preamble}+N_{header}+N_{payload}$

Table 43 - Frame related parameters (concluded)

Parameter	Description	Value
T_{frame}	Duration of the frame	$(N_{preamble}+N_{header}+N_{payload})T_{sym}$

10.4.4 PLCP preamble

The Type C PLCP preamble can be divided into two distinct portions: a frame synchronization sequence, $S_{sync,C}[.]$, and a channel estimation sequence, $S_{CE,C}[.]$. Structure of the Type C PLCP preamble shows the structure of the Type C PLCP preamble. Each symbol of the Type C PLCP preamble shall be consecutively repeated $N_{TDS}=2$ times as described in 10.4.6.1.5.

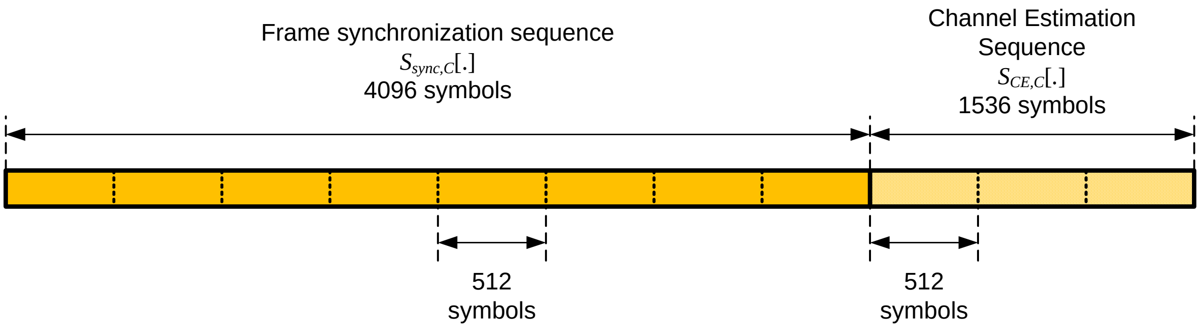


Figure 50 - Structure of the Type C PLCP preamble

Both $S_{sync,C}[.]$ and $S_{CE,C}[.]$ are constructed based on the frame synchronization sequence $S_{sync,A,SCBT}[.]$ and channel estimation sequence $S_{CE,A,SCBT}[.]$ described in 10.2.2.3.1 and 10.2.2.3.2. The preamble construction is given in 10.4.4.1 and 10.4.4.2.

10.4.4.1 Frame synchronization sequence

The Type C frame synchronization sequence consists of seven repetitions of a modified hierarchical sequence, $S_{h,C}[.]$, and a complementary sequence, $\bar{S}_{h,C}[.]$. The modified hierarchical sequence, $S_{h,C}[.]$, is of length 256, and is derived from the modified hierarchical sequence $S_{h,B}[.]$ as defined in 10.3.2.3.1. That is,

$$S_{h,C}[n]=\begin{cases}1 & S_{h,B}[n]<0\\ 0 & S_{h,B}[n]>0\end{cases} \qquad (79)$$

for $n=0,\ldots,255$. The complementary sequence $\bar{S}_{h,C}[.]$ is defined as

$$\bar{S}_{h,C}[n]=1-S_{h,C}[n] \qquad (80)$$

for $n=0,\ldots,255$. The unspread frame synchronization sequence $\tilde{S}_{sync,C}[.]$ is defined as

$$\tilde{S}_{sync,C}[n]=\begin{cases}S_{h,C}[n \bmod 256] & n=0,\ldots,1791\\ \bar{S}_{h,C}[n \bmod 256] & n=1792,\ldots,2047\end{cases} \qquad (81)$$

and the frame synchronization sequence $S_{sync,C}[.]$ is defined as

$$S_{sync,C}[n] = \tilde{S}_{sync,C}\left[\left\lfloor \frac{n}{N_{TDS}} \right\rfloor\right] \qquad n = 0, \dots, 4095 \tag{82}$$

where N_{TDS} is the Time Domain Spreading Factor (TDSF) defined for each mode in Table 41.

10.4.4.2 Channel estimation sequence

The Type C channel estimation sequence is constructed by from two sequences, $S_{C1}[.]$ and $S_{C2}[.]$. Both sequences $S_{C1}[.]$ and $S_{C2}[.]$ are of length 256 and are defined from the sequence $S_{CE,B,SC}[.]$ described in 10.3.2.3.2. That is

$$\begin{aligned} S_{C1}[n] &= \left(S_{CE,B,SC}[n] + 1\right)/2 \\ S_{C2}[n] &= \left(-S_{CE,B,SC}[n] + 1\right)/2 \end{aligned} \tag{83}$$

For $n=0,\dots,255$. The unspread channel estimation sequence $\tilde{S}_{CE,C}[.]$ is defined as:

$$\tilde{S}_{CE,C}[n] = \begin{cases} S_{C1}[n \bmod 256] & n = 0, \dots, 255 \text{ and } 512, \dots, 767 \\ S_{C2}[n \bmod 256] & n = 256, \dots, 511 \end{cases} \tag{84}$$

And the channel estimation sequence $S_{CE,C}[.]$ is defined as

$$S_{CE,C}[n] = \tilde{S}_{CE,C}\left[\left\lfloor \frac{n}{N_{TDS}} \right\rfloor\right] \qquad n = 0, \dots, 1535 \tag{85}$$

where N_{TDS} is the Time Domain Spreading Factor (TDSF) defined for each mode in Table 41.

NOTE
The frame synchronization sequence can be used for frame acquisition and detection, coarse carrier frequency estimation, coarse symbol timing, and for synchronization within the preamble. Whereas, the channel estimation sequence can be used for estimation of the channel frequency response, fine carrier frequency estimation, and fine symbol timing.

10.4.5 PLCP header

A PLCP header shall be added after the PLCP preamble to convey information about both the PHY and the MAC that is needed at the receiver in order to successfully decode the PSDU. The scrambled and Reed-Solomon encoded PLCP header shall be formed as described in 10.1.2.

For Type C frames, the formed PLCP header (as described in 10.1.2) shall be further encoded and modulated as described in 10.4.6. Encoding and modulation parameters identical to mode C0 shall be used. The resulting data symbols shall be block modulated (as described in 10.4.6.1.7) in order to create the baseband signal.

10.4.5.1 PHY header

10.4.5.1.1 Fixed length PHY header

10.4.5.1.1.1 PLCP scrambler field (SCRAMBLER)

Refer to 10.2.2.4.1.1.1.

10.4.5.1.1.2 Bit reversal field (BIT_REVERSAL)

The BIT_REVERSAL field determines whether or not all of the data bits of PPDU payload are reversed. The mapping between this bit and reversal of PPDU payload is given in Table 44.

Table 44 - Mapping between the value of the BIT_REVERSAL field and inverting of PPDU payload

BIT_REVERSAL (B)	Bit reversal state
0	All bits of PPDU payload are not reversed
1	All bits of PPDU payload are reversed

10.4.5.1.1.3 ATIF existence field (ATIF_EX)

The ATIF_EX field shall be set to 0_B for all Type C frames.

10.4.5.1.1.4 CP length field (CP_LENGTH)

The CP_LENGTH field shall be set to 00_B for all Type C frames.

10.4.5.1.1.5 Requested CP length field (REQ_CP_LENGTH)

The REQ_CP_LENGTH field shall be set to 00_B for all Type C frames.

10.4.5.1.1.6 Number of segments field (NUM_SEGMENTS)

The NUM_SEGMENTS field shall be set to $F_3F_2F_1F_0=0001_B$ for Type C frames with payload or $F_3F_2F_1F_0=0000_B$ for Type C frames with no payload (e.g. ACK frames).

10.4.5.1.1.7 Number of MSDUs field (NUM_MSDUS)

The NUM_MSDUS field shall be set to $P_5P_4P_3P_2P_1P_0=000001_B$ for Type C frames with payload or $P_5P_4P_3P_2P_1P_0=000000_B$ for Type C frames with no payload (e.g. ACK frames).

10.4.5.1.2 Variable length PHY header

10.4.5.1.2.1 Mode field (MODE)

Refer to 10.2.2.4.1.2.1.

10.4.5.1.2.2 Segment length field (LENGTH)

Refer to 10.2.2.4.1.2.2.

10.4.5.1.2.3 Midamble existence field (MID_EX)

The MID_EX field shall be set to 0_B for all Type C frames.

10.4.5.1.2.4 Segment continued field (CONT)

The CONT field shall be set to 0_B for all Type C frames.

10.4.5.1.3 Antenna training indicator field

Type C frames shall not include the antenna training indicator field (ATIF).

10.4.6 PPDU payload

The PPDU payload may consist of zero or one segment as described in 10.1. The segments are formed as described in 10.1.3. Each data bit block shall be processed as follows:

1. The bits shall be encoded as described in 10.4.6.1.

2. If mode C2 is selected, 4ASK training sequence shall be appended at the beginning of the PPDU payload as described in 10.4.6.1.6

3. The resulting symbols shall be modulated into SC blocks as described in 10.4.6.1.7.

10.4.6.1 FEC and mapping

Figure 51 depicts the general overview of the encoding and mapping scheme. For the payload, first, the data bits shall be scrambled as specified in 10.4.6.1.1. Then, $N_{padbits}$ zero pad bits shall be appended to the end of the data bit block where

$$N_{padbits} = 1792\left\lceil \frac{N_{bits}}{1792} \right\rceil - N_{bits} \tag{86}$$

And N_{bits} is the number of bits in the data block. The padded data bit block shall be encoded according to the Reed-Solomon code, as described in 10.4.6.1.2. The resulting RS coded payload bits are reversed as described in 10.4.6.1.3.

The resulting reversed payload bits or the formed header bits shall be mapped into symbols as described in 10.4.6.1.4. The data symbols shall be spread in time domain as described in 10.4.6.1.5. The resulting data symbols shall be appended with N_{padsym} zero symbols, where

$$N_{padsym} = N_D\left\lceil \frac{N_{sym}}{N_D} \right\rceil - N_{sym} \tag{87}$$

and N_{sym} is the number of symbols in the data symbol block

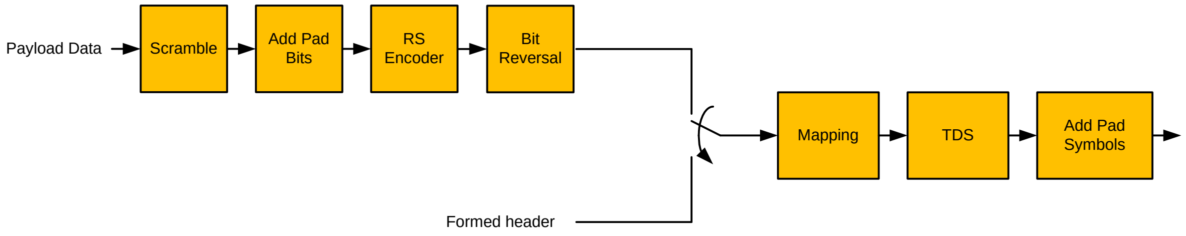


Figure 51 - General view of the encoding procedure

10.4.6.1.1 Data scrambler

Refer to 10.2.2.5.1.1.

10.4.6.1.2 Reed-Solomon code

Refer to 10.2.2.5.1.2.

10.4.6.1.3 Bit reversal

The bit reversal procedure shall be adopted at all the data bits of PPDU payload for all Type C frame retransmissions. The BIT_REVERSAL field will be set to “1” when bit reversal is applied or “0” when bit reversal is not applied. When the bit reversal is applied the output bits $g[.]$ shall be derived from the original data bits $b[.]$ by

$$g[n] = \mathrm{NOT}\left(b[n]\right) \tag{88}$$

where $\mathrm{NOT}(.)$ is a bitwise NOT operation.When the bit reversal is not applied the output bits $g[.]$ shall be identical to the original data bits $b[.]$.

Figure 52 illustrates the bit reversal procedure using an example where the scrambler is initialized with seed "00". During initial transmission of a frame, the frame to be transmitted is not a retransmission frame and therefore the data bits shall not be reversed. As such, the BIT_REVERSAL field shall be set to "0", and the SCRAMBLER field shall be set to a seed value as defined in 10.2.2.5.1.1.

If the initial transmission of a frame is determined to be unsuccessful, the frame may be retransmitted, as described in 15.19.10. At the r^{th} retransmission, where r is an odd number, the seed value of SCRAMBLER field shall remain unchanged and the data bits shall be reversed. Therefore, BIT_REVERSAL field shall be set to "1". At the r^{th} retransmission, where r is an even number, SCRAMBLER field shall be set to a seed value as defined in 10.2.2.5.1.1 and the data bits shall not be reversed. Therefore, BIT_REVERSAL field shall be set to "0".

NOTE: As described in 15.19.10, it is assumed that compliant implementation shall enable MAC to convey information to PHY such that the latter is to be able to determine the value of k, for example, via an explicit MAC-PHY interface primitive, via decoding of retry bit (described in 14.2.3.6) and tracking of retransmission times by the PHY, or through some other means which is beyond the scope of this standard.

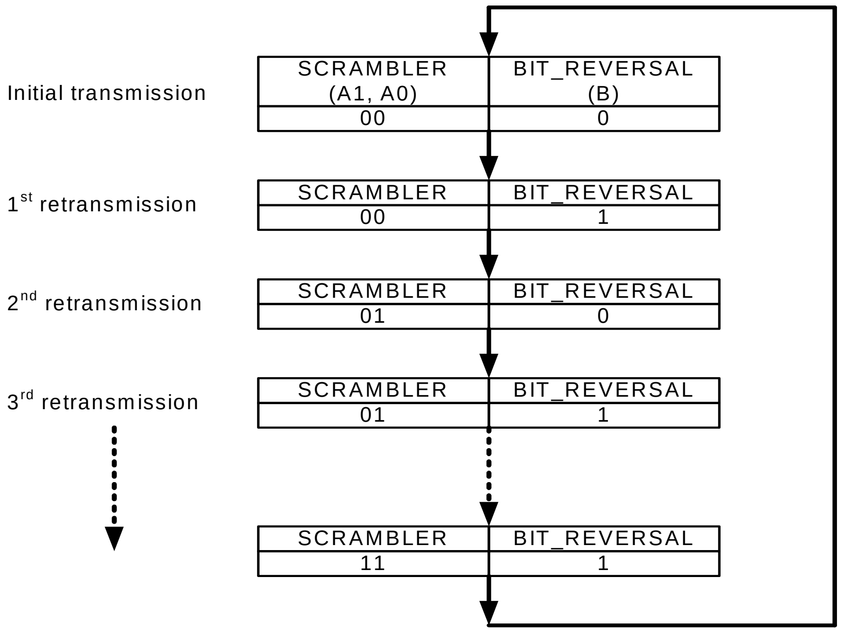


Figure 52 - Bit reversal procedure

10.4.6.1.4 Constellation mapping

10.4.6.1.4.1 OOK

The coded and repeated binary serial input data, $g[i]$ where i=0,1,2,..., shall be converted into a real number representing one of the two OOK constellation points. The output values, $v[k]$ where k=0,1,2,..., are formed by:

$$v[k] = K_{MOD}\; I[k] \tag{89}$$

where $I[k]$ is given by Table 45. The resulting constellation is illustrated in Figure 53. The normalization factor is $K_{const} = \sqrt{2}$ for the OOK constellation. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements.

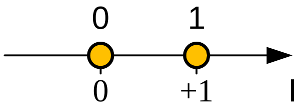


Figure 53 - OOK constellation bit encoding

Table 45 - OOK encoding table

Input bits $g[k]$	$I[k]$
0	0
1	1

10.4.6.1.4.2 4ASK

The coded and repeated binary serial input data, $g[i]$ where i=0,1,2,..., shall be divided into groups of two bits and converted into a real number representing one of the four levels of the ASK constellation. The conversion shall be performed according to the Gray-coded constellation mapping, illustrated in Figure 54, with the input bit, $g[2k]$ where k=0,1,2,..., being the earliest of the two in the stream. The output values, $v[k]$ where k=0,1,2,..., are formed by:

$$v[k] = K_{const}\; I[k] \tag{90}$$

where $I[k]$ is given by Table 46. The resulting constellation is illustrated in Figure 54.

The normalization factor is $K_{const} = \sqrt{\frac{2}{7}}$ for 4ASK.

Table 46 - 4ASK encoding table

Input bits ($g[2k]$, $g[2k+1]$)	$I[k]$
00	0
01	1
11	2
10	3

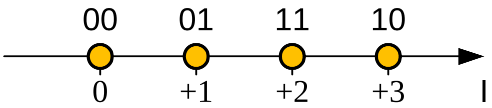


Figure 54 - 4ASK constellation bit encoding

10.4.6.1.5 Time domain spreading (TDS)

Refer to 10.2.2.5.1.9.

10.4.6.1.6 4ASK training sequence

When 4ASK modulation mode is selected at the PLCP header, a dedicated 4-levels training sequence ($S_{4ASK}[.]$) is appended prior the start of the PPDU payload to enable the proper threshold setting at the receiver side.

The training sequence is based on the first 64 elements of the sequence $S_{h,B}[.]$ defined in 10.3.2.3.1. Every two consecutive elements of $S_{h,B}[.]$ sequence is used to create the $S_{4ASK}[.]$ sequence

$$S_{4ASK}[n] = \begin{cases} 0 & S_{h,B}[2n] > 0 \text{ and } S_{h,B}[2n+1] > 0 \\ 1 & S_{h,B}[2n] > 0 \text{ and } S_{h,B}[2n+1] < 0 \\ 2 & S_{h,B}[2n] < 0 \text{ and } S_{h,B}[2n+1] < 0 \\ 3 & S_{h,B}[2n] < 0 \text{ and } S_{h,B}[2n+1] > 0 \end{cases} \qquad (91)$$

for n=0,...,31.

The 32 symbols of the $S_{4ASK}[.]$ sequence are transmitted with TDSF of 2 and with a normalization factor of $K_{const} = \sqrt{2/7}$.

10.4.6.1.7 Block modulation

The repeated transmit symbols shall be divided into block of length N_D=508N_{TDS}. This transmit symbol block shall be appended with pilot symbol as described in Figure 55. The pilot symbols consist of a sequence of length N_p=4N_{TDS}. The pilot symbols for mode C0, C1 and C2 shall be chosen according to Table 47.

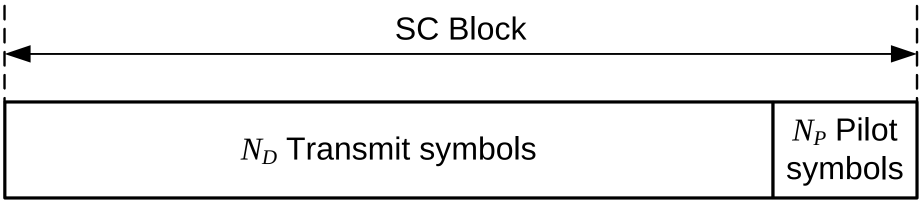


Figure 55 - Formation of the SC block

Table 47 - Pilot symbols

n	0	1	2	3	4	5	6	7
$S_{pilot,C0}[n]$	1	1	0	0	1	1	0	0
$S_{pilot,C1}[n]$	1	0	1	0	N/A	N/A	N/A	N/A
$S_{pilot,C2}[n]$	3	0	3	0	N/A	N/A	N/A	N/A

11 General requirements

11.1 Operating band frequencies

11.1.1 Operating frequency range

This PHY operates in the 57 - 66 GHz frequency band.

11.1.2 Channel numbering

The relationship between centre frequency, f_c, and BAND_ID number, n_b, is given in Table 48.

Table 48 - Band allocation

BAND_ID (n_b)	Channel Bonding	Lower Frequency (GHz)	Centre Frequency (GHz)	Upper Frequency (GHz)
1	No	57.240	58.320	59.400
2	No	59.400	60.480	61.560
3	No	61.560	62.640	63.720
4	No	63.720	64.800	65.880
5	Yes (1 & 2)	57.240	59.400	61.560
6	Yes (2 & 3)	59.400	61.560	63.720
7	Yes (3 & 4)	61.560	63.720	65.880
8	Yes (1, 2, & 3)	57.240	60.480	63.720
9	Yes (2, 3, & 4)	59.400	62.640	65.880
10	Yes (1, 2, 3, & 4)	57.240	61.560	65.880

This definition provides a unique numbering system for all channels within the band 57 - 66 GHz. The third channel (BAND_ID = 3) shall be used as the discovery channel. Figure 56 depicts the defined channels.

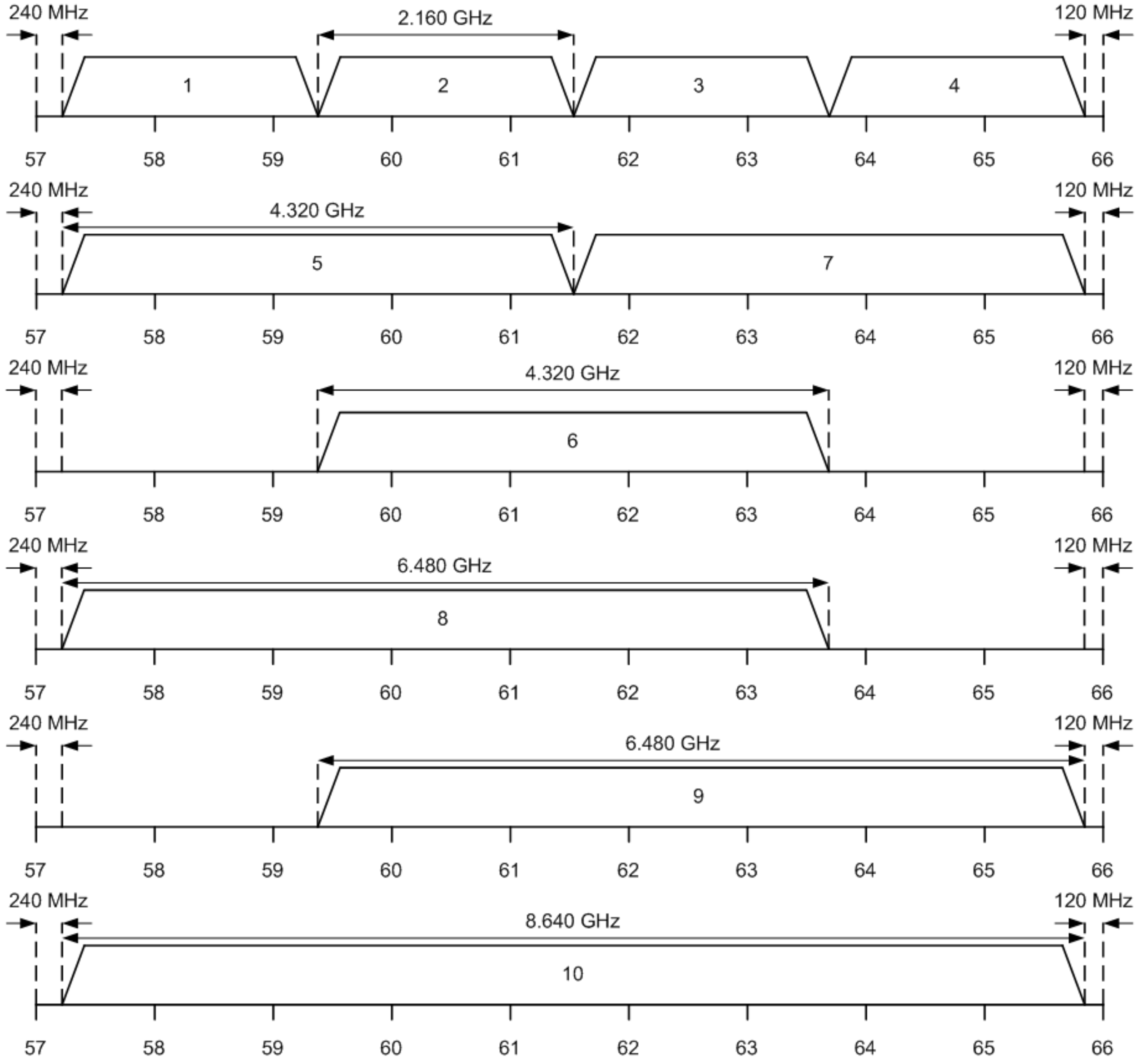


Figure 56 - Channel numbering

11.2 PHY layer timing

The values for the device Type A and Type B PHY layer timing parameters are defined in Table 49.

Table 49 - Type A and B PHY timing parameters

PHY Parameter	Value
pMIFS	6x256xT_{sym} = 888 ns
pSIFS	18x256xT_{sym} = 2666 ns
pCCADetectTime	5037 ns

The values for the device Type C PHY layer timing parameters are defined in Table 50.

Table 50 - Type C PHY layer timing parameters

PHY Parameter	Value
pMIFS_C	14x256xT_{sym} = 2074 ns
pSIFS_C	41x256xT_{sym} = 6074 ns
pCCADetectTime	5037 ns

11.2.1 Receive-to-transmit turnaround time

The RX-to-TX turnaround time shall not be greater than pSIFS_C. This turnaround time shall be measured at the air interface. The time elapsed from the trailing edge of the last received symbol to the leading edge of the first transmitted symbol of the PLCP preamble for the next frame shall not be greater than pSIFS_C.

11.2.2 Transmit-to-receive turnaround time

The TX-to-RX turnaround time shall not be greater than pSIFS_C. This turnaround time shall be measured at the air interface. The time elapsed from the trailing edge of the last transmitted symbol until the receiver is ready to begin the reception of the next PHY frame shall not be greater than pSIFS_C.

11.2.3 Time between successive transmissions

For uninterrupted successive transmissions by a device, the interframe spacing after the frame shall be pSIFS_C if Number of segments field is zero, and shall not be less than pMIFS_C if the Number of segments field is nonzero. The interframe spacing time shall be measured at the air interface. When the Number of segments field is zero, the time elapsed from the trailing edge of the last transmitted symbol to the leading edge of the first transmitted symbol of the PLCP preamble for the following packet shall be equal to pSIFS_C. When the PLCP length field is nonzero, the time elapsed from the trailing edge of the last transmitted symbol to the leading edge of the first transmitted symbol of the PLCP preamble for the following frame shall not be less than pMIFS_C.

12 Transmitter specifications

12.1 Transmit PSD mask

12.1.1 Transmit PSD: Type A and Type B

The transmit spectral mask shall conform to the values as indicated in the Figure 57 and Table 51.

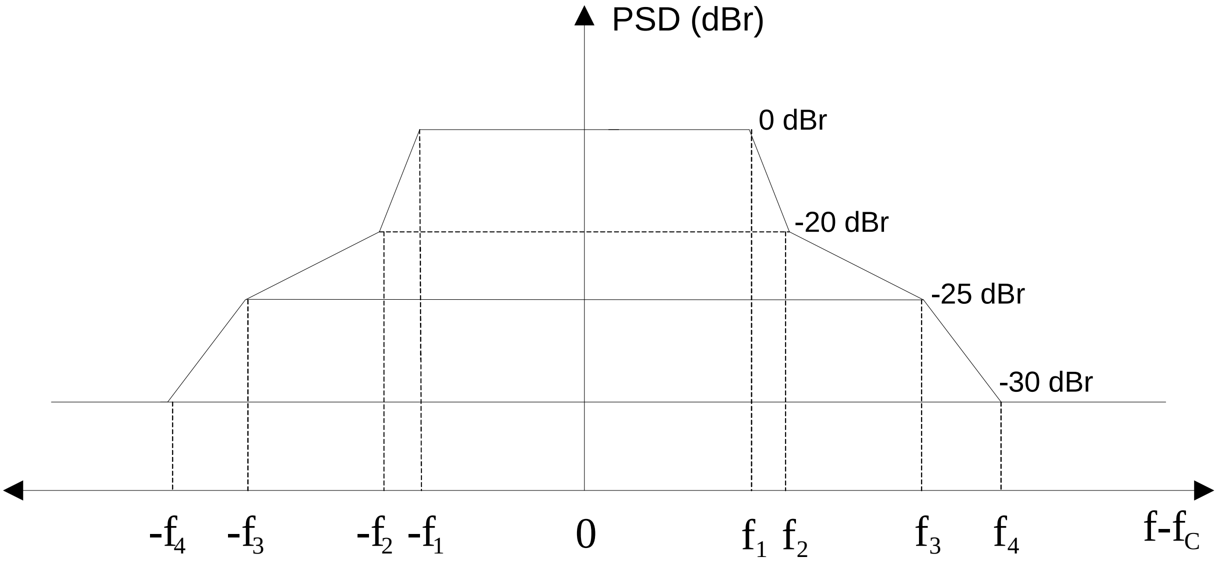


Figure 57 - Transmit spectral mask

Table 51 - Transmit spectral mask limit

Frequency	Relative Limit (dBr)
$\lvert f-f_c\rvert \le f_1$	0
$f_1 \le \lvert f-f_c\rvert \le f_2$	$-20(\lvert f-f_c\rvert-f_1)/(f_2-f_1)$
$f_2 \le \lvert f-f_c\rvert \le f_3$	$-20-5(\lvert f-f_c\rvert-f_2)/(f_3-f_2)$
$f_3 \le \lvert f-f_c\rvert \le f_4$	$-25-5(\lvert f-f_c\rvert-f_3)/(f_4-f_3)$
$\lvert f-f_c\rvert \ge f_4$	-30

where f_1, f_2, f_3 and f_4 for a single channel transmission, a two bonded channels transmission, a three bonded channels transmission and a four bonded channels transmission are given in Table 52.

Table 52 - Transmit spectral mask requirements

Channel Bonding	f_1 (MHz)	f_2 (MHz)	f_3 (MHz)	f_4 (MHz)
Single Channel Transmission	1050	1080	1500	2000
Two Bonded Channels Transmission	2100	2160	3000	4000
Three Bonded Channels Transmission	3150	3240	4500	6000
Four Bonded Channels Transmission	4200	4320	6000	8000

The transmit spectral mask shall be measured with 1 MHz resolution bandwidth. The transmit spectral mask requirement does not include any carrier leakage.

12.1.2 Transmit PSD: Type C

Type C devices, which generate a single line spectra at the carrier frequency fc, shall conform to the values as indicated in Figure 58 and Table 53. Type C devices do not support channel bonding transmission. The maximum total EIRP transmitted by a Type C device shall be limited to 10dBm. The transmit spectral mask shall be measured with 1 MHz resolution bandwidth.

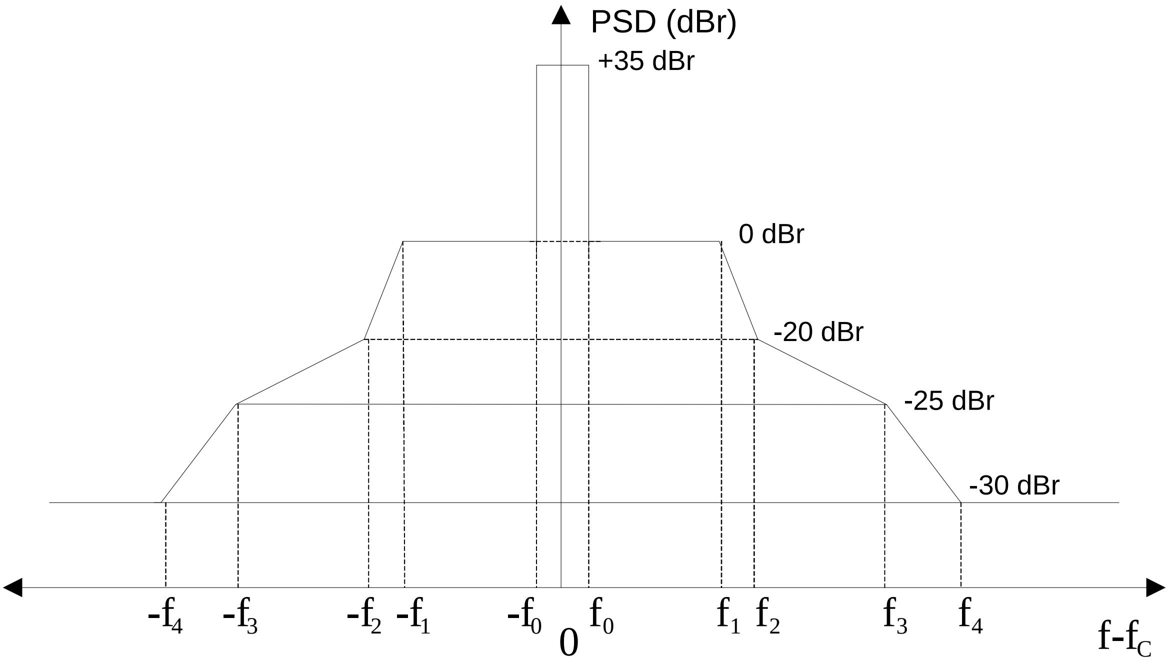


Figure 58 - Transmit spectral mask

Table 53 - Transmit spectral mask limit

Frequency	Relative Limit (dBr)
$\|f-f_c\| \le f_0$	35
$f_0 \le \|f-f_c\| \le f_1$	0
$f_1 \le \|f-f_c\| \le f_2$	$-20(\|f-f_c\|-f_1)/(f_2-f_1)$
$f_2 \le \|f-f_c\| \le f_3$	$-20-5(\|f-f_c\|-f_3)/(f_4-f_3)$
$f_3 \le \|f-f_c\| \le f_4$	$-25-5(\|f-f_c\|-f_3)/(f_4-f_3)$
$\|f-f_c\| \ge f_4$	-30

where f_0, f_1, f_2, f_3 and f_4 for a single channel transmission are given in the Table 54.

Table 54 - Transmit spectral mask requirement

	f_0 (MHz)	f_1 (MHz)	f_2 (MHz)	f_3 (MHz)	f_4 (MHz)
Single Channel Transmission	4	1050	1080	1500	2000

12.2 Transmit centre frequency tolerance

The transmitted centre frequency tolerance shall be ±20 ppm maximum.

12.3 Symbol clock frequency tolerance

The symbol clock frequency tolerance shall be ±20 ppm maximum.

12.4 Clock synchronization

The transmit centre frequencies and the symbol clock frequency shall be derived from the same reference oscillator.

12.5 Transmit power control

A device should provide support for transmit power control to optimize its power consumption and minimize interferences to other existing links, while still providing a reliable link for the transfer of information. The transmitter should change its transmission power at the receiver's request.

Transmit power shall be changed with a step size granularity of 2dB. The relative accuracy of change in transmit power shall be the maximum of ±1 dB or ±20% of the change (in the dB scale). As an example, for a change of 4 dB and a change of 8 dB, the allowed relative accuracy is ±1.0 dB and ±1.6 dB, respectively.

12.6 Transmitter EVM

12.6.1 Type A

12.6.1.1 SCBT

The Error Vector Magnitude (EVM) defines the average constellation error power with reference to the power of a highest constellation point of the modulation scheme.

Figure 59 illustrates EVM.

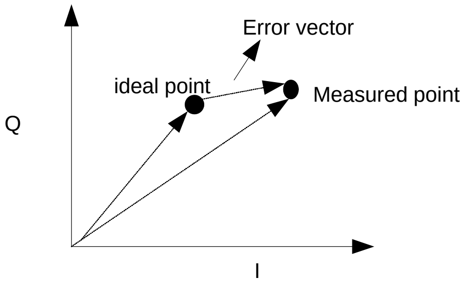


Figure 59 - Illustrative diagram for EVM for one constellation point

This EVM is defined in percentage as

$$EVM = \frac{\sqrt{\frac{1}{N}\cdot\sum_{1}^{N}(\Delta I_i^2 + \Delta Q_i^2)}}{R_{max}} \quad . \tag{92}$$

where N is the number of symbols used in measurement, $\Delta I_i^2 + \Delta Q_i^2$ is the error vector, and R_{max} is the magnitude of the maximum constellation point. The EVM shall be measured over the payload with identical modulation scheme.

Considering the error vector as noise added to thermal and channel noise, the required EVM can be estimated from the transmitter implementation margin. The transmitter implementation margin is the excess power needed at the transmitter to negate the effect of transmitter imperfections so that the received SNR of a real transmitter will be identical to that of an ideal transmitter.

The measurement of the EVM assumes the use of a raised cosine filter with 25% excess bandwidth at the transmitter and a near "ideal" receiver corresponding to an AWGN channel.

The EVM measurement includes imperfections of the transmitter due to transmitter filter inaccuracy, D/A converter, I/Q imbalances, phase noise, and non-linearity of amplifiers.

Based on the assumption on the required margin at transmitter shown in the following table, the EVM is computed using the required SNR and transmitter margin. The results are shown in Table 55. The requirement for EVM shall be as shown in this table based on a near ideal receiver and AWGN channel.

Table 55 - Maximum allowable EVM values for Type A SCBT modes

Mode	Max. Allowed EVM (dB)	Max. Allowed EVM (%)
A0	-4.8	33.4
A1	-6.3	23.7
A2	-9.5	11.2
A3	-6.4	23.1
A4	-8.4	14.4
A5	-9.6	10.9
A6	-10.9	8.1
A7	-12.0	6.3
A8	-11.1	7.7
A9	-12.5	5.6
A10	-6.4	23.1
A11	-9.5	11.1
A12	-7.4	18.2
A13	-11.1	7.7

12.6.1.1.1 RMS error measurement and calculation

The relative constellation RMS error calculation shall be performed using a device capable of converting the transmitted signal into a stream of samples at 1.728 Gsps or more, with sufficient accuracy in the DC offset, phase noise, etc. The sampled signal shall then be processed in a manner similar to that of an ideal receiver. The necessary steps for receiver processing are listed below.

1. Detect the start of the packet and frame boundary.

2. Estimate the correct sampling time. Correct as needed.

3. Estimate the channel impulse response.

4. Equalize the sampled signal with the estimated channel impulse response.

5. For each of the sampled signal, find the closest constellation point and compute the Euclidean distance.

6. Compute the RMS error, averaged over all data symbols and over all frames, as follows:

$$EVM = \frac{1}{N_f}\sqrt{\frac{\sum_n |R[n]-C[n]|^2}{N_{datasyms}P_0}} \qquad (93)$$

where N_f is the number of packets under test, $N_{datasyms}$ is the number of data symbols in the segment (excluding the pilot symbols), P_0 is the average power over all payload data symbols, $C[n]$ is the transmitted n^{th} data symbol, and $R[n]$ is the observed n^{th} data symbol, and the sum is calculated over all the data symbols in the segments.

The RMS error shall be computed over one segment of the frame only. P_0 is re-computed for each frame. The test shall be performed over a minimum of N_f = 100 frames, where the PSDU of each packet is at least 16384 symbols in length and is generated from random data.

12.6.1.2 OFDM

The relative constellation RMS error, averaged over all data and pilot subcarriers of the OFDM symbols and over all of the frames shall not exceed the values given in Table 56.

Table 56 - Permissible relative constellation error

Mode	Modulation	Multipath Margin (dB)	Accepted degradation due to inaccuracies in the constellation points (dB)	Relative constellation error (dB)
A14	QPSK (1/2)	1.5	2	-7.3
A15	QPSK (2/3)	1.5	3	-12.3
A16	16 QAM (2/3)	3	4	-19.6
A17	UEP-QPSK (Coding)	1.5	3	-13.1
A18	UEP-16 QAM (Coding)	3	4	-20.9
A19	UEP-QPSK (Mapping) 2/3	1.5	3	-13.1
A20	UEP-16 QAM (Mapping) 2/3	3	4	-20.9
A21	QPSK MSB-only 2/3	1.5	3	-12.3

The relative constellation error values for each of the data rates and modulations used are based on a multi-path margin listed in Table 56 with an implementation loss of 3 dB.

In addition, it is assumed that the degradation due to the relative constellation error can be no more than 2.0 dB, 3.0 dB and 4.0 dB for data rates at 1.008 Gbps, 2.016 and 4.032 Gbps, respectively.

The relative constellation RMS error calculation shall be performed using a device capable of converting the transmitted signal into a stream of complex samples at 2.592 Gsps or more, with sufficient accuracy in the I/Q imbalance, DC offset, phase noise, etc. The sampled signal shall then be processed in a manner similar to that of an ideal receiver. The necessary steps necessary for receiver processing are listed below:

1. Detect the start of the packet and frame boundary.

2. Estimate the correct sampling time and frequency offset. Correct as needed.

3. Estimate the channel frequency response and equalize the channel.

4. For each of the data and pilot subcarriers, find the closest constellation point and compute the Euclidean distance.

5. Compute the RMS error, averaged over all the data and pilot subcarriers, and over all frames as given by

$$E_{RMS} = \frac{1}{N_f}\sum_{i=1}^{N_f}\sqrt{\sum_{n=N_{preamble}+N_{header}}^{N_{frame}}\left(\frac{\sum_{k=1}^{N_D}\left|R_{D,n}[k]-C_{D,n}[k]\right|^2+\sum_{k=1}^{N_P}\left|R_{P,n}[k]-C_{P,n}[k]\right|^2}{(N_D+N_P)N_{frame}P_0}\right)} \qquad (94)$$

where N_f is the number of packets under test, N_{packet} is the number of symbols in the packet, $N_{preamble}$ is the number of symbols in the PLCP preamble, N_{header} is the number of symbols in the PLCP header, N_{frame} = N_{packet} - $N_{preamble}$ - N_{header} is the number of symbols in the PSDU, N_D is the number of data subcarriers, N_P is the number of pilot subcarriers, P_0 is the average power of the data and pilot constellations, $C_{D,n}[k]$ and $C_{P,n}[k]$ are the transmitted k^{th} data subcarrier and k^{th} pilot subcarrier for the n^{th} OFDM symbol, respectively, and $R_{D,n}[k]$ and $R_{P,n}[k]$ are the observed k^{th} data subcarrier and k^{th} pilot subcarrier for the n^{th} OFDM symbol, respectively. The values for N_D and N_P are defined in Table 16.

12.6.2 Type B

The relative constellation RMS error, averaged over all data and over all of the frames, shall not exceed the values given in Table 57 when using the method of 12.6.1.1.1.

The measurement of the EVM assumes the use of a raised cosine filter with 25% excess bandwidth at the transmitter and a near ideal receiver corresponding to an AWGN channel.

The EVM measurement includes imperfections of the transmitter due to transmitter filter inaccuracy, D/A converter, I/Q imbalances, phase noise, and non-linearity of amplifiers.

Table 57 - Permissible relative constellation error

Modes	Relative Constellation Error (dB)	Relative Constellation Error (%)
B0	-7	20
B1	-7	20

Table 57 - Permissible relative constellation error (concluded)

Modes	Relative Constellation Error (dB)	Relative Constellation Error (%)
B2	-8.2	15
B3	-8.2	15
B4	-9.2	12

12.6.3 Type C

The relative constellation RMS error, averaged over all data and over all of the frames, shall not exceed the values given in Table 58. The relative constellation error values are based on a multi-path margin of 3 dB and implementation loss of 2 dB. In addition, it is assumed that the degradation due to the relative constellation error can be no more than 0.5 dB for a data rate of 0.864 Gbps and 1.0 dB for a data rates of 1.728 Gbps and 1.5 dB for a data rate of 3.456 Gbps.

Table 58 - Permissible relative constellation error

Mode	Relative Constellation Error (dB)	Relative Constellation Error (%)
C0	-5.5	28.2
C1	-7.0	20
C2	-11.8	6.6

The relative constellation RMS error calculation shall be performed using a device capable of converting the transmitted signal into a stream of samples at 1.728 Gsps or more, with sufficient accuracy in the DC offset, phase noise, etc. The sampled signal shall then be processed in a manner similar to that of an ideal receiver. An example of the minimum steps necessary for receiver processing is listed below.

1. Detect the start of the packet and frame boundary.

2. Estimate the correct sampling time. Correct as needed.

3. Estimate the channel impulse response.

4. Equalize the sampled signal with the estimated channel impulse response.

5. For each of the sampled signal, find the closest constellation point and compute the Euclidean distance.

6. Compute the RMS error, averaged over all data symbols and over all frames, as follows:

$$E_{RMS} = \frac{1}{N_f}\sum_{i=1}^{N_f}\sqrt{\frac{\sum_{n=0}^{N_{frame}-N_{sync}-N_{header}-1}\left|R[n]-C[n]\right|^2}{N_{payload}\,P_0}} \qquad (95)$$

where N_f is the number of packets under test, N_{packet} is the number of symbols in the packet, N_{sync} is the number of symbols in the PLCP preamble, N_{header} is the number of symbols in the PLCP header, $N_{payload} = N_{frame} - N_{sync} - N_{header}$ is the number of symbols in the PSDU, P_0 is the average power over all payload symbols of the data constellations,

$C[n]$ is the transmitted n^{th} data symbol, and $R[n]$ is the observed n^{th} data symbol. The values for N_{sync}, N_{header}, $N_{payload}$, and N_{frame} are defined in Table 43. The RMS error shall be computed over the payload portion of the packet only. P_0 is re-computed for each packet.

The test shall be performed over a minimum of N_f = 100 frames, where the PSDU of each packet is at least 16384 symbols in length and is generated from random data. The RMS error shall be measured without any tone-nulling applied.

13 Receiver specification

13.1 Type A device

13.1.1 SCBT receiver sensitivity

For a packet error rate (PER) less than 8% with payload length of 2048 octets, the minimum receiver sensitivity numbers (expressed in EIR$_x$P) in AWGN for modes A0 through A13 are listed in Table 59.

Table 59 - Minimum receiver sensitivity for different SCBT modes

Mode	Minimum Receiver Sensitivity (dBm)
A0	-60.0
A1	-57.0
A2	-50.5
A3	-53.9
A4	-49.8
A5	-47.4
A6	-45.7
A7	-43.5
A8	-43.5
A9	-40.7
A10	-53.9
A11	-43.5
A12	-52.3
A13	-43.5

13.1.2 OFDM receiver sensitivity

For a packet error rate (PER) less than 8% with payload length of 2048 octets, the minimum receiver sensitivity numbers (expressed in EIR$_x$P) in AWGN for modes A14 up to and including A21 are listed in Table 60.

Table 60 - Minimum receiver sensitivities for the different modes

Mode	Minimum Receiver Sensitivity (dBm)
A14	-60.6

Table 60 - Minimum receiver sensitivities for the different modes (concluded)

Mode	Minimum Receiver Sensitivity (dBm)
A15	-54.3
A16	-50.2
A17	-56.0
A18	-52.0
A19	-54.3
A20	-50.2
A21	-54.3

13.2 Type B device receiver sensitivity

For a packet error rate (PER) less than 8% with payload length of 2048 octets, the minimum receiver sensitivity numbers (expressed in EIR$_x$P) in AWGN for modes B0 up to and including B4 are listed in Table 62.

Table 61 - Minimum receiver sensitivity for different SC modes

Mode	Minimum Receiver Sensitivity (dBm)
B0	-60.7
B1	-57.7
B2	-54.6
B3	-54.6
B4	-58.6

13.3 Type C device receiver sensitivity

For a packet error rate (PER) less than 8% with payload length of 2048 octets, the minimum receiver sensitivity numbers (expressed in EIR$_x$P) in AWGN for modes C0 through C2 are listed in Table 62.

Table 62 - Minimum receiver sensitivities

Mode	Minimum Receiver Sensitivity (dBm)
C0	-62.2
C1	-57.0
C2	-53.5

13.4 Receiver CCA performance

Transmissions at a receiver at an EIR$_x$P a receiver level equal to or greater than -85.0 dBm shall cause CCA to indicate the channel is busy with a probability greater than 90% within pCCADetectTime.

14 MAC frame formats

This Clause specifies the format of MAC frames. An overview of the MAC frame with descriptions of common fields is followed by Clauses for each frame type and subtype. The final Clause contains a list of information elements that may appear in beacon frames and some command frames. Octets of the EUI (Annex C.1) are passed to the PHY SAP in ascending index-value order.

14.1 Frame format conventions

The following conventions and definitions apply throughout this Clause.

14.1.1 Figures

MAC frames are described as a sequence of fields in a specific order. Figures in Clause 14 depict fields in the order they are delivered to the PHY SAP, from left to right, where the left-most field is transmitted first in time. In field figures, bits within the field are numbered from the least-significant bit on the right to the most-significant bit on the left.

An example sequence of fields is specified in Figure 60.

octets: 2	1	…	4
First field transmitted (2 octets)	Second field transmitted (1 octet)	…	Last field transmitted (4 octets)

Figure 60 - Example sequence of fields

14.1.2 Octet order

Unless otherwise noted, fields longer than a single octet are delivered to the PHY SAP in order from the octet containing the least-significant bits to the octet containing the most-significant bits.

An example of a bitmap specification for a two-octet field is specified in Figure 61.

bits: b15-b13	b12-b8	b7-b0
Most-significant bits of second octet transmitted	Least-significant bits of second octet transmitted	First octet transmitted

Figure 61 - Example bitmap specification for a field

14.1.3 Encoding

Values specified in decimal are encoded in unsigned binary unless otherwise stated.

A bitmap is a sequence of bits, labelled as bit[0] through bit[N-1]. A bitmap is encoded in a field such that bit[0] corresponds to the least-significant bit of the field and subsequent bitmap elements correspond to subsequent significant bits of the field. Octets of the field are presented to the PHY SAP in order from least-significant octet to most-significant octet.

Reserved fields and subfields are set to zero on transmission and ignored on reception.

14.2 General MAC frame format

Frame transactions that do not employ aggregation shall use unaggregated MAC frames, while frame transactions that employ aggregation shall use aggregated MAC frame format.

14.2.1 Unaggregated MAC frame

An unaggregated MAC frame consists of a fixed-length MAC Header and an optional variable-length MAC Frame Body.

The Fixed-Length MAC header is specified in Figure 62.

octets: 2	**2**	**2**	**2**	**2**
Frame Control	DestAddr	SrcAddr	Access Information	Sequence Control

Figure 62 - Fixed-length MAC Header format

The overall length of the MAC Header of an unaggregated MAC frame is 10 octets.

The MAC frame body, when present, contains a Frame Payload and Frame Check Sequence (FCS) as shown in Figure 63.

octets: L	**4**
Frame Payload	FCS

Figure 63 - MAC frame body format

In secure frames the frame payload includes security fields as shown in Figure 64. The left-most four fields in Figure 64 are collectively referred to as the security header.

octets: 3	**1**	**2**	**6**	**P**	**8**
Temporal Key Identifier (TKID)	Security Reserved	Encryption Offset (EO)	Secure Frame Number (SFN)	Secure Payload	Message Integrity Code (MIC)

Figure 64 - Frame payload field format for secure frames

The Frame Payload length ranges from zero to mMaxFramePayloadSize. If the Frame Payload length is zero, the FCS field is not included, and there is no MAC Frame Body. The Frame Payload length includes the length of the security fields for a secure frame.

In this Clause, a reference to the payload of a frame indicates the Frame Payload field of a non-secure frame, or the Secure Payload field of a secure frame. The payload is a sequence of octets labelled as payload[0] through payload[P-1]. Octets are passed to the PHY SAP in ascending index-value order.

14.2.2 Aggregated MAC frame

An aggregated MAC frame consists of a variable-length MAC header and a variable-length MAC Frame Body. The MAC header of an aggregated MAC frame is specified in Figure 65.

octets: 2	2	2	2	2+4N
Frame Control	DestAddr	SrcAddr	Access Information	Aggregation Header

Figure 65 - Variable-length MAC header of an aggregated MAC frame

The aggregation header field is specified in Figure 66.

octets: 2	2	2	...	2	2
MSDU Count	Sequence Control of MSDU 1	Length of MSDU 1	...	Sequence Control of MSDU N	Length of MSDU N

Figure 66 - Aggregation header

The MSDU Count field contains the number of MSDUs included in the aggregated frame.

The Length fields in the Aggregation Header field indicate the length in octets of the corresponding MSDUs. The lengths do not include the Pad octets.

The overall length of the MAC Header of an aggregated MAC frame is 10+N*4 octets.

The MAC Frame Body of an aggregated MAC frame contains multiple MSDUs, each aligned to a 4-octet boundary, and their corresponding FCS. The aggregated MAC frame body is specified in Figure 67.

octets: M_1	4	0-3	M_2	4	...	0-3	Mn	4
MSDU 1	FCS of MSDU 1	Pad to 4-octet boundary	MSDU 2	FCS of MSDU 2	...	Pad to 4-octet boundary	MSDU N	FCS of MSDU N

Figure 67 - Frame body for aggregated MAC frames

The Frame Payload size for aggregated data frames is subject to the same maximum size as any Frame Payload.

14.2.3 Frame control

The Frame Control field is specified in Figure 68.

bits: b15-b14	b13	b12-b9	b8-b6	b5-b4	b3	b2-b0
Reserved	Retry	Frame Subtype / Delivery ID	Frame Type	ACK Policy	Secure	Protocol Version

Figure 68 - Frame control field format

14.2.3.1 Protocol version

The Protocol Version field is invariant in size and placement across all revisions of this Standard. For this revision of the Standard, Protocol Version is set to zero. All other values are reserved.

14.2.3.2 Secure

The Secure bit is set to ONE in a secure frame, which is protected using the temporal key specified by the Temporal Key Identifier (TKID). The Secure bit is set to ZERO otherwise. Frames with the Secure bit set to ONE use the Frame Payload format for secure frames as shown in Figure 68. Valid settings for the Secure bit in each frame type are listed in Table 114 in 16.2.

14.2.3.3 ACK policy

The ACK Policy field is set to the type of acknowledgement requested by the transmitter. Acknowledgement procedures are described in 15.12. The allowed values for the ACK Policy field are defined in Table 63.

Table 63 - ACK policy field encoding

Value	ACK policy type	Description
0	No-ACK	The recipient(s) do not acknowledge the transmission, and the sender treats the transmission as successful without regard for the actual result. The use of this policy is defined in 15.12.1.
1	Imm-ACK	The addressed recipient returns an Imm-ACK frame after correct reception, according to the procedures defined in 15.12.2.
2	B-ACK	The addressed recipient keeps track of the frames received with this policy until requested to respond with a B-ACK frame, according to the procedures defined in 15.12.3.
3	B-ACK Request	The addressed recipient returns a B-ACK frame after reception, according to the procedures defined in 15.12.3.

14.2.3.4 Frame type

The Frame Type field is set to the type of frame that is being sent. Table 64 lists the valid frame type values, descriptions, and the Clauses that describe the format and use of each of the individual frame types.

Table 64 - Frame type field encoding

Value	Frame type	Clause
0	Beacon frame	14.3
1	Control frame	14.4
2	Command frame	14.5
3	Data frame	14.6
4	Aggregated MAC frame	14.2.2
5	Master-Slave frame	14.8
6 - 7	Reserved	

If the Frame type field is set to Aggregated data frame, a variable-length MAC header that is described in 15.10 shall be used.

14.2.3.5 Frame subtype / delivery ID

The Frame Subtype / Delivery ID field is used to assist a receiver in the proper processing of received frames. In control or command frames, this field is used as Frame Subtype, as defined in Table 68 in 14.4 and Table 69 in 14.5. In data frames and aggregated data frames, this field is used as Delivery ID and set to Stream Index. This field is reserved in all other frame types.

14.2.3.6 Retry

The Retry bit is set to ONE in any data, aggregated data, or command frame that is a retransmission of an earlier frame. It is reserved in all other frame types.

14.2.4 DestAddr

The DestAddr field is set to the DevAddr of the intended recipient(s) of the frame. The DevAddr specifies a single device for a unicast frame, a group of devices for a multicast frame, or all devices for a broadcast frame. DevAddr values are described in 15.1.1.

14.2.5 SrcAddr

The SrcAddr field is set to the DevAddr of the transmitter of the frame.

14.2.6 Sequence control

The Sequence Control field identifies the order of MSDUs/MCDUs and their fragments. The Sequence Control field is specified in Figure 69. The Sequence Control field is reserved in control frames.

bits: b15	b14	b13-b3	b2-b0
Reserved	More Fragments	Sequence Number	Fragment Number

Figure 69 - Sequence control field format

14.2.6.1 Fragment number

The Fragment Number field is set to the number of the fragment within the MSDU or MCDU. The fragment number is zero in the first or only fragment of an MSDU or MCDU and is incremented by one for each successive fragment of that MSDU or MCDU.

14.2.6.2 Sequence number

The Sequence Number field is set to the sequence number of the MSDU or MCDU, as defined in 15.1.9.3.

The Sequence Number field is used for duplicate frame detection, as described in 15.1.7, and to preserve frame order when using the B-ACK mechanism, as described in 15.12.3.

The Sequence Number field is reserved in control frames.

14.2.6.3 More fragments

The More Fragments field is set to ZERO to indicate that the current fragment is the sole or final fragment of the current MSDU or MCDU; otherwise, the field is set to ONE.

14.2.7 Access information

The Access information field is specified in Figure 70.

bits: b15	b14	b13-b0
Access Method	More Frames	Duration

Figure 70 - Access information field format

14.2.7.1 Duration

The Duration field is 14 bits in length and is set to an expected medium busy interval after the end of the PLCP header of the current frame in units of microseconds. The duration value is set as defined in 15.1.9.1.

14.2.7.2 More frames

In frames sent with the Access Method bit set to ONE, the More Frames bit is set to ZERO if the transmitter will not send further frames to the same recipient during the current reservation block; otherwise it is set to ONE.

In frames sent with the Access Method bit set to ZERO, the More Frames bit is set to ZERO if the transmitter will not send further frames to the same recipient during the current superframe; otherwise it is set to ONE.

The More Frames bit is reserved in beacon and control frames. Additional rules regarding the More Frames field are specified in 15.1.9.2.

14.2.7.3 Access method

The Access Method bit is set to ONE in all frames transmitted within a hard or private DRP reservation block by the reservation owner or target prior to the release of the reservation block.

The Access Method bit may be set to ONE in frames transmitted within a Soft DRP reservation block without backoff by the reservation owner.

The Access Method bit in an Imm-ACK, B-ACK or CTT control frame is set to the same value as the Access Method bit in the corresponding received frame.

The Access Method bit is set to ZERO in discovery beacon frames and Type B and Type C Poll frames sent in discovery poll blocks.

The Access Method bit is reserved in other beacon frames and Type C Poll frames sent by a Type C Master and Type C Poll response frames sent by a Type C slave.

The Access Method bit is set to ZERO in frames transmitted at all other times.

14.2.8 Frame payload

The Frame Payload field is a variable length field that carries the information that is to be transferred to a device or group of devices. In a secure frame, it includes the required security fields as shown in Figure 64 and defined below.

14.2.8.1 Temporal key identifier (TKID)

The TKID field is an identifier for the temporal key used to protect the frame. The TKID uniquely identifies this key from any other temporal keys held by the sender and the recipient(s) of the frame. It does not need to uniquely identify the key for devices not holding the key.

14.2.8.2 Security reserved

The Security Reserved field is reserved, but included in authentication of the frame.

14.2.8.3 Encryption offset (EO)

The Encryption Offset field indicates where encryption starts, in octets, relative to the beginning of the Secure Payload, as shown in Figure 64. A value of zero indicates that the entire Secure Payload is encrypted. A non-zero value in this field indicates that the first EO octets of the Security Payload are not encrypted. Regardless of the value of this field, the entire Secure Payload, along with other appropriate fields, is authenticated by the MIC.

14.2.8.4 Secure frame number (SFN)

The SFN field provides message freshness as a defence against replay attacks. The SFN field in a secure frame is set to the next value of the sender's secure frame counter (SFC) for the temporal key used by this frame. SFC setting and replay protection are described in 16.4.2.

14.2.8.5 Secure payload

The Secure Payload field in secure frames is the counterpart of the Frame Payload field in non-secure frames. It contains the information specific to individual frame types and protected by the symmetric key identified in the TKID field of the same frame.

14.2.8.6 Message integrity code (MIC)

The MIC field contains an 8-octet cryptographic checksum used to protect the integrity of the MAC Header and Frame Payload.

14.2.9 FCS

The FCS field contains a 32-bit value that represents a CRC polynomial of degree 31.

The CRC is calculated over a calculation field, which is the entire Frame Payload field for this specification. The calculation field is mapped to a message polynomial M(x) of degree k-1, where k is the number of bits in the calculation field. The least-significant bit of the first octet presented to the PHY SAP is the coefficient of the x^{k-1} term, and the most-significant bit of the last octet transmitted is the coefficient of the x^0 term.

The CRC is calculated using the following Standard generator polynomial of degree 32:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1 \tag{96}$$

The CRC polynomial is the one's complement of the modulo 2 sum of the following remainders:

- The remainder resulting from $x^k \times (x^{31} + x^{30} + \ldots + x + 1)$ divided (modulo 2) by G(x).
- The remainder resulting from $x^{32} \times M(x)$, divided (modulo 2) by G(x).

The FCS field value is derived from the CRC polynomial such that the least-significant bit is the coefficient of the x^{31} term and the most-significant bit is the coefficient of the x^0 term. Figure 71 illustrates the encoding of the FCS field for the CRC polynomial:

$$a_{31}x^{31} + a_{30}x^{30} + a_{29}x^{29} + \ldots + a_2x^2 + a_1x + a_0 \tag{97}$$

bits: b31	**b30**	**b29**	**…**	**b2**	**b1**	**b0**
a_0	a_1	a_2	…	a_{29}	a_{30}	a_{31}

Figure 71 - FCS field encoding

In a common implementation, at the transmitter, the initial remainder of the division is preset to all ONEs and is then modified via division of the calculation field by the generator polynomial G(x). The one’s complement of this remainder is the FCS field. At the receiver, the initial remainder is preset to all ONEs. The serial incoming bits of the calculation field and FCS, when divided by G(x) in the absence of transmission errors, results in a unique non-zero remainder value. The unique remainder value is the polynomial:

$$x^{31} + x^{30} + x^{26} + x^{25} + x^{24} + x^{18} + x^{15} + x^{14} + x^{12} + x^{11} + x^{10} + x^{8} + x^{6} + x^{5} + x^{4} + x^{3} + x + 1 \tag{98}$$

14.3 Beacon frames

MAC header field settings for beacon frames are described in Table 65. Beacon frames are also referred to as beacons throughout this specification.

Table 65 - MAC header field values for beacon frames

Header field	Value
Protocol Version	0
Secure	0
ACK Policy	0 (No-ACK)
Frame Type	0 (beacon frame)
Frame Subtype / Delivery ID	Reserved
Retry	Reserved
DestAddr	BcstAddr
SrcAddr	DevAddr of the transmitter
Sequence Control	As defined in 14.2.6 and 15.1.9.3
Duration	As defined in 14.2.7.1 and 15.1.9.1
More Frames	Reserved
Access Method	Reserved

The beacon frame payload is specified in Figure 72.

octets: 8	**L_1**	**…**	**L_N**
Beacon Parameters	Information Element 1	…	Information Element N

Figure 72 - Payload format for beacon frames

The information elements (IEs) that may be included in a beacon frame are listed in Table 83 in 14.7. IEs are included in order of increasing Element ID, except for ASIEs. ASIEs do not appear prior to any IE with Element ID zero through seven, but may appear anywhere after those IEs. DRP IEs that have the same Target DevAddr and Stream Index are adjacent to each other in the beacon.

The beacon parameters field is specified in Figure 73.

octets: 6	**1**	**1**
Device Identifier	Beacon Slot Number	Device Control

Figure 73 - Beacon parameters field format

The device identifier field is set to the EUI-48 (Annex C.1) of the device sending the beacon. A device may use a NULL EUI-48 value (all bits set to ONE) to indicate it does not have a unique EUI-48 value.The EUI is a sequence of 6 octets, labelled as eui[0] through eui[5]. The first three octets (eui[0] through eui[2]) are the manufacturer's OUI, and the last three octets (eui[3] through eui[5]) are the manufacturer-selected extension identifier.

The Beacon Slot Number field is set to the number of the beacon slot where the beacon is sent within the beacon period (BP), in the range of [0, mMaxBPLength-1], except in beacons sent in signalling slots. In signalling slots it is set to the number of the device's non-signalling beacon slot.

The Device Control field is specified in Figure 74.

bits: b7-b6	b5	b4-b2	b1	b0
Security Mode	Reserved	Status	Signalling Slot	Movable

Figure 74 - Device control field format

The Movable bit is set to ONE if the beacon is movable according to 15.5.5.10 (BP contraction), and is set to ZERO otherwise.

The Signalling Slot bit is set to ONE if the beacon is sent in a signalling beacon slot according to 15.5.5.6**,** and is set to ZERO otherwise.

The status field is set to the following values to indicate the status or function of the beacon frame.

Table 66 - Status field meaning

Value	Meaning
0	Ready
1	Discovery
2	Response
3	Preemptive
4	Dual

The security mode field is set to the security mode at which the device is currently operating.

14.4 Control frames

Default MAC Header field settings for control frames are listed in Table 67. Specific MAC Header field settings and payload descriptions for each of the control frames are shown in the following Clauses.

Table 67 - MAC header field values for control frames

Header field	Value
Protocol Version	0
Secure	As defined in 14.2.3.2
ACK Policy	0 (No-ACK)
Frame Type	1 (control frame)
Frame Subtype	Value from Table 68
Retry	Reserved
DestAddr	DevAddr of the recipient
SrcAddr	DevAddr of the transmitter
Sequence Control	Reserved
Duration	As described in 14.2.7.1 and 15.1.9.1
More Frames	Reserved
Access Method	As described in 14.2.7.3

Table 68 lists valid values for the frame subtype field for control frames.

Table 68 - Frame subtype field encoding for control frames

Value	Control frame subtype	Description
0	Imm-ACK	Acknowledges correct receipt of the previously-received frame
1	B-ACK	Acknowledges correct or incorrect receipt of one or more preceding frames
2	RTT	Announces to a recipient device that the sender is ready to initiate antenna training and requests confirmation of the ability to perform antenna training
3	CTT	Responds to a RTT control frame that the recipient is able to perform antenna training
4	TRN	Provides antenna training status and/or feedback and requests to change training parameters
5	B-Poll	Provides Master services to Type B slave device
6	B-Poll Response	Acknowledges B-Poll sent by a Type A Master device
7-13	Reserved	Reserved
14	Application-specific	At discretion of application owner
15	Reserved	Reserved

14.4.1 Immediate acknowledgement (Imm-ACK)

In Imm-ACK frames, the DestAddr field is set to the SrcAddr of the received frame that is acknowledged. Imm-ACK frames have no frame payload.

14.4.2 Block acknowledgement (B-ACK)

In B-ACK frames, the DestAddr field is set to the SrcAddr of the frame that requested the B-ACK.

The B-ACK frame acknowledges correct or incorrect receipt of the previous sequence of frames and provides information for the transmission of the next sequence of frames as described in 15.12.3. The B-ACK frame payload is specified in Figure 75.

octets: 2	**1**	**1**	**2**	**0-n**
Buffer Size	Frame Count	Reserved	Sequence Control	Frame Bitmap

Figure 75 - Payload format for B-ACK frames

The Buffer Size field specifies the maximum number of octets in the sum of the frame payloads of all frames in the next B-ACK sequence.

The Frame Count field specifies the maximum number of frames in the next B-ACK sequence.

The sequence control and frame bitmap fields together specify an acknowledgement window of MSDU fragments and their reception status. The Sequence Control field specifies the Sequence Number and Fragment Number that start the acknowledgement window.

bits: b15-14	b13-b3	b2-b0
Reserved	Sequence Number	Fragment Number

Figure 76 - Sequence control field format

The Frame Bitmap field varies in length. A zero-length Frame Bitmap field indicates an acknowledgement window of length zero. Otherwise, the least-significant octet of the Frame Bitmap field corresponds to the MSDU indicated by the Sequence Control field, and each bit of the octet corresponds to a fragment of that MSDU. The least-significant bit in each octet corresponds to the first fragment and successive bits correspond to successive fragments. Successive octets present in the Frame Bitmap field correspond to successive MSDUs, and each bit corresponds to a fragment of the MSDU. The acknowledgement window ends at fragment seven of the MSDU that corresponds to the most-significant octet in the Frame Bitmap.

For all bits within the Frame Bitmap, a value of ONE indicates that the corresponding fragment was received in either the current sequence or an earlier one. A value of ZERO indicates that the corresponding fragment was not received in the current sequence (although it may have been received in an earlier one). Bits of the least-significant octet of the Frame Bitmap field corresponding to fragments prior to the start of the acknowledgement window are undefined. Frames with a Sequence Number earlier than the Sequence Number indicated in the Sequence Control field were not received in the last B-ACK sequence. Such frames were previously received or are no longer expected.

14.4.3 Application-specific

The payload format for Application-specific control frames is specified in Figure 77.

octets: 2	N
Specifier ID	Data

Figure 77 - Payload format for application-specific control frames

The Specifier ID field is set to a 16-bit value that identifies a company or organization, as listed in Annex C.4. The owner of the Specifier ID defines the format and use of the Data field.

14.4.4 B-Poll

A B-Poll Frame shall be used by a Type A device to start Master-Slave operation with a Type B slave device and to direct the Type B Slave when to send its dual-beacon.

The payload of a B-Poll frame is specified in Figure 78.

octets: 4	L_1	...	L_N
Directed Beacon Parameters	Information Element 1	...	Information Element N

Figure 78 - Payload format for B-poll frames

The information elements (IEs) that may be included in a beacon frame are listed in Table 83 in 14.7. IEs are included in order of increasing Element ID, except for ASIEs. ASIEs do not appear prior to any IE with Element ID zero through seven, but may appear anywhere after those IEs. DRP IEs that have the same Target DevAddr and Stream Index are adjacent to each other in the beacon. In a B-Poll frame, the Type A master shall only include the IEs that need to included in the directed dual-beacon of the Type B slave.

The Directed Beacon Parameters field is specified in Figure 79.

octets: 2	1	1
Directed Beacon Offset	Beacon Slot Number	Device Control

Figure 79 - Directed Beacon Parameters field format

The Directed Beacon Offset field is set to the time when Type B is directed to send its first beacon of its dual-beacon, measured from the end of the PLCP header of the B-Poll frame in units of microseconds.

The Beacon Slot Number field is set to the number of the beacon slot where the Type B slave is directed to send its first beacon of its dual-beacon within the beacon period (BP), in the range of [mSignalSlotCount, mMaxBPLength-1].

The Device Control field is specified in Figure 80.

bits: b7-b6	b5-b2	b1	b0
Directed Beacon Offset	Beacon Slot Number	Signalling Slot	Movable

Figure 80 - Device Control field format

The Movable bit is set to ONE if the beacon is movable according to 15.5.5.10, and is set to ZERO otherwise.

The Signalling Slot bit is set to ONE if the Type B master is directed to send an additional beacon in a signalling beacon slot according to 15.5.5.6, and is set to ZERO otherwise.

14.4.5 B-Poll response frame

In B-Poll response frame, the DestAddr field is set to the SrcAddr of the received B-Poll frame. The payload of B-Poll response frames may carry a number of IEs, which is specified in Figure 81.

Octets: L_1	**...**	**L_N**
Information Element 1	...	Information Element N

Figure 81 - payload format for B-Poll response frame

14.4.6 ATTP control frames

RTT and CTT control frames shall be used to initiate and negotiate antenna training between source and destination devices. RTT is transmitted by the source device whenever it wants to initiate an antenna training procedure with a destination device. Upon receipt of a RTT frame, the addressed device shall respond with a CTT frame provided it is available to engage in the antenna training procedure. The antenna training procedure is performed by exchanging TRN frames.

14.4.6.1 Request to train (RTT)

In RTT frames, the DestAddr field is set to the DevAddr of the device to which antenna training is desired. RTT frames shall carry in their frame payload the ATIE with the transmitter device settings.

14.4.6.2 Clear to train (CTT)

In CTT frames, the DestAddr field is set to the SrcAddr of the received RTT frame. CTT frames shall carry in their frame payload the ATIE with the receiver device settings.

14.4.6.3 Training control (TRN)

A TRN frame contains an ATFIE and/or an AFIE in its payload. In antenna tracking, a TRN frame may be optionally piggybacked with some data payload. When there is data payload present, the aggregated frame format shall be used and the ATFIE and/or AFIE shall be the last MSDU in the aggregated frame.

The payload of a TRN frame without data payload is specified in Figure 82.

Octets: 4, 40, or 44
ATFIE and/or AFIE

Figure 82 - Payload format of a TRN frame with non-zero length payload

The MAC header of an aggregated TRN frame shall be set according to 14.2.2. The MAC Frame body of an aggregated TRN frame is specified in Figure 83.

octets: M_1	4	0-3	...	0-3	M_n	4
MSDU 1	FCS of MSDU 1	Pad to 4-octet boundary	...	Pad to 4-octet boundary	ATFIE and/or AFIE	FCS of MSDU N

Figure 83 - Payload format of an aggregated TRN frame

The ATFIE and AFIE are included in a TRN frame in order of increasing element ID.

14.5 Command frames

Default MAC header settings for command frames are shown in Table 69.

Table 69 - Default MAC header field values for command frames

Header field	Value
Protocol Version	0
Secure	As defined in 14.2.3.2
ACK Policy	0 (No-ACK) or 1 (Imm-ACK)

Table 69 - Default MAC header field values for command frames (concluded)

Header field	Value
Frame Type	2 (command frame)
Frame Subtype	Value from Table 70
Retry	As defined in 14.2.3.6
DestAddr	DevAddr of the recipient
SrcAddr	DevAddr of the transmitter
Sequence Control	As defined in 14.2.6
Duration	As defined in 14.2.7.1 and 15.1.9.1
More Frames	As defined in 14.2.7.2
Access Method	As defined in 14.2.7.3

Table 70 specifies the values for the frame subtype field for command frames.

Table 70 - Frame subtype field encoding for command frames

Value	Command frame subtype	Description
0	DRP Reservation Request	Used to request creation or modification of a DRP reservation
1	DRP Reservation Response	Used to respond to a DRP reservation request command
2	Probe	Used to request for, or respond with, information elements
3	Pair-wise Temporal Key (PTK)	Used to derive a PTK via a 4-way handshake between two devices
4	Group Temporal Key (GTK)	Used to solicit or distribute a GTK within a secure relationship
5	Link Feedback	Used to exchange link feedback information and/or to recommend transmit power/rate change
6	Relay	Used to request for or respond with information elements
7	Transmit Switched Diversity (TSD) Set Request	Used to initiate or to terminate the TSD operation. In addition, used to acquire channel state information
8	Transmit Switched Diversity (TSD) Set Response	Used to respond to the TSD Set Request command
9	FUCA	Informs a recipient device of FUCA map information on the next frame
10	TSD Switch	Requests to switch the transmit antenna to an arbitrary antenna during the TSD operation

Table 70 - Frame subtype field encoding for command frames (concluded)

Value	Command frame subtype	Description
11	Channel Selection	Used to select initial channel to send data beacons, i.e. mode-A0 or mode B0 beacons
12 - 13	Reserved	Reserved
14	Application-specific	At discretion of application owner
15	Reserved	Reserved

14.5.1 DRP reservation request

The DRP Reservation Request command frame is used to create or modify a DRP reservation. The DRP Reservation Request command frame payload is specified in Figure 84.

octets: M_1	**M_2**	**…**	**M_N**
DRP IE-1	DRP IE-2	…	DRP IE-N

Figure 84 - Payload format for DRP reservation request command frames

Each DRP IE field included in the command frame corresponds to a reservation request identified by the Target/Owner DevAddr, Stream Index, and Reservation Type in the IE. The DRP IE is defined in 14.7.8.

14.5.2 DRP reservation response

The DRP Reservation Response command frame is used to respond to a DRP Reservation Request command frame. The DRP Reservation Response command frame payload is specified in Figure 85.

octets: M_1	**M_2**	**…**	**M_N**	**2 to 34**
DRP IE-1	DRP IE-2	…	DRP IE-N	DRP Availability IE

Figure 85 - Payload format for DRP Reservation Response command frames

The DRP Reservation Response command frame includes all the DRP IEs from the reservation request. The DRP Availability IE is included according to the rules defined in 15.6.

14.5.3 Channel selection

The set of channel selection command frames are used to request channel scanning, to respond with channel scanning results and to select channel to exchange MSDUs.

The ACK policy field in the MAC header of channel selection command frame shall always be set to Imm-ACK.

The payload of channel selection command frame is specified in Figure 86.

octets: 2	L_1	...	L_N
Channel Selection Control	Information Element 1	...	Information Element N

Figure 86 - Payload format for Channel selection command frames

The Channel Selection Control field is specified in Figure 87.

bits: b15-b14	b13-b12	b11-b10	b9-b0
Command ID	Reserved	Reason Code	Channel Bitmap

Figure 87 - Channel Selection Control field format

The Command ID field is set to the value as listed in Table 71 that identifies the Channel selection command.

Table 71 - Channel selection command ID

Command ID	Channel selection command
0	Channel Scanning Request
1	Channel Scanning Response
2	Channel Change Request
3	Channel Change Response

The encoding of Channel Bitmap field is specified in Figure 88.

Bit	Band ID	Description
0	1	Channel 1
1	2	Channel 2
2	3	Channel 3
3	4	Channel 4
4	5	Channel 5
5	6	Channel 6
6	7	Channel 7
7	8	Channel 8
8	9	Channel 9
9	10	Channel 10

Figure 88 - Channel Bitmap Field Encoding

14.5.3.1 Channel Scanning Request

The Channel Bitmap field is set such that the bits corresponding to the channels requested to be scanned are set to ONE.

The Reason Code field in a Channel Scanning Request is reserved.

The first Information Elements field shall include a Scan Timing IE to indicate the time when the device returns back to Discovery Channel to listen for scanning response to the scanning request.

14.5.3.2 Channel Scanning Response

The Channel Bitmap field is set to the same as in the Channel Scanning Request.

The Reason Code field in a Channel Scanning Response is reserved.

The Information Elements fields shall include a number of DRP available IEs. The number of DRP available IEs shall be the same as the number of channels requested to be scanned.

14.5.3.3 Channel Change Request

The first Information Elements field shall include a Channel Change IE.

The Reason Code and Channel Bitmap fields in a Channel Change Request are reserved.

14.5.3.4 Channel Change Response

The first Information Element field shall include a Channel Change IE that is the same as the one in the Channel Change Request to which it is responding. The Reason Code field shall be set appropriately as listed in Table 72.

Table 72 - Reason code field encoding

Value	Code	Meaning
0	Accepted	The channel change request is agreed upon
1	Unavailable	The channel change request is rejected because the channel in concern does not have enough MASs
2	Conflict	The channel change request is rejected because the channel change request conflicts with the scanning response sent.
3	Invalid	The device does not support channel

The Channel Bitmap field is reserved in the Channel Change Response.

14.5.4 Link feedback

The set of Link Feedback command frames are used to exchange feedback information pertaining to power control, rate control and certain link quality metrics between two devices. The payload of the Link Feedback command is specified in Figure 89.

octets: 1	**0 or N**
Feedback Control	Link Feedback

Figure 89 - Payload format for link feedback command frames

The Feedback Control field is specified in Figure 90.

bits: b7-b6	b5-b3	b2-b0
Command ID	Reserved	Request Bitmap

Figure 90 - Feedback Control field format

The command ID field is set to the value as listed in Table 73 that identifies the link feedback command.

Table 73 - Link feedback command ID

Command ID	Link feedback command	Description
0	Transmit power and rate control	Recommends transmit power and rate change to a source device.
1	Link Feedback Response	Provides, in response to a link feedback request, link quality metrics and/or recommendation of rate and transmission power change.
2	Link Feedback Request	Requests a sink device to provide link feedback.

The Request Bitmap field is set according to Table 74.

Table 74 - Request bitmap

Bit	Meaning	Description
0	TPRC	Bit set to ONE to indicate requesting for, or the existence of sink's recommendation on transmit rate and power change
1	LQI	Bit set to ONE to indicate requesting for, or the existence of feedback on LQI of the link
2	FER	Bit set to ONE to indicate requesting for, or the existence of feedback on frame loss/error rate of the link

A link feed back command may contain 0-3 link feedback fields depending on setting of the Feedback Control field.

14.5.4.1 Link feedback request

There shall be none Link Feedback field in the Link feedback request frame.

14.5.4.2 Transmit power and rate control (TPRC)

The Feedback Control field of the TPRC is specified in Figure 91.

Bits: b7-b6	b5-b3	b2-b0
=0	Reserved	=0

Figure 91 - Setting of Feedback Control field of TPRCs

The Link Feedback field of the TPRC is specified in Figure 92.

bits: b7-b4	B3-b0
Data rate	Transmit Power Level Change

Figure 92 - Link Feedback field format of TPRCs

The Transmit Power Level Change sub field is set to the change in transmit power level that the sink device recommends according to Table 92.

The Data Rate sub field is set to the data rate that the sink device recommends according to Table 93.

14.5.4.3 Link feedback response

The Request Bitmap field of a link feedback response is set to the same as in the link feedback request to which the device is responding.

The Link Feedback field of a link feedback response is specified in Figure 93.

octets: 1	2	7
TPRC	LQI	FER

Figure 93 - Link feedback field format

The existence of TPRC, LQI and FER sub fields in the Link Feedback field is determined by the Request Bitmap field. A bit set to ONE in the Request Bitmap field indicates that the corresponding sub field exists in the Link Feedback field. The order the sub fields appear is specified in Figure 93.

The TPRC sub field is set according to Figure 92.

The LQI sub field is set to the Link Quality Indicator (LQI) of the link between the source and sink devices.

The format of the FER sub field is specified in Figure 94.

octets: 1	1	2	2	1
Recipient Frame Loss Count	Recipient Frame Error Count	Recipient Frame Count	Source Frame Count	Measurement Window Size

Figure 94 - FER field format

The Measurement Window Size field is set to the amount of time in milliseconds during which the measurements as included in FER field were taken.

The Source Frame Count field is set to the total number of frames, including retransmissions that were transmitted by the sink device to the source device.

The Recipient Frame Count field is set to the total number of frames that were correctly received by the sink device from the source device.

The Recipient Frame Error Count field contains the total number of frames that were received with FCS errors but not with HCS errors by the sink device from the source device.

The Recipient Frame Loss Count field is set to the number of frames as observed by the sink device to have been lost.

14.5.5 Probe

The Probe command frame is used to request information from a device or respond to a Probe request. The payload format is specified in Figure 95.

Octets: M_1	**M_2**	**…**	**M_N**
Information Element 1	Information Element 2	…	Information Element N

Figure 95 - Payload format for probe command frames

If the payload includes a Probe IE, the command requests information from the recipient. Each Information Element field contains one information element.

14.5.6 Pairwise temporal key (PTK)

The PTK command frame is used in a 4-way handshake by a pair of devices, as described in 16.3.1, to authenticate each other and to derive a shared symmetric PTK for securing certain unicast traffic between the two devices. The PTK command frame is specified in Figure 96.

octets: 1	**1**	**3**	**11**	**16**	**16**	**8**
Message Number	Status Code	PTKID	Reserved	MKID	I-Nonce / R-Nonce	PTK MIC

Figure 96 - Payload format for PTK command frames

The Message Number is set to 1, 2, 3, or 4, respectively, in the PTK command containing the first, second, third, or fourth message of the 4-way handshake. The other values of this field are reserved.

The Status Code in a PTK command indicates the current status of the 4-way handshake at the device sending this command. It is encoded as shown in Table 75.

Table 75 - Status code field encoding in PTK commands

Value	Meaning
0	Normal-the 4-way handshake proceeds
1	Aborted-the 4-way handshake is aborted per security policy
2	Aborted-the 4-way handshake is aborted in order to yield to a concurrent 4-way handshake using the same master key
3	PTKID not accepted-it is the TKID of a PTK or GTK being possessed by this device
4 - 255	Reserved

The PTKID is set to a non-zero number as the TKID of the PTK to be derived from this 4-way handshake procedure. The initiator of the 4-way handshake chooses this value after determining that this value is different from the TKID of the PTK, if any, that is to be replaced by the new PTK, and the TKID of any PTK or GTK it currently possesses.

The MKID identifies the master key used in this 4-way handshake as described in 16.3.1.

The I-Nonce/R-Nonce is a random number generated by the initiator or responder for this 4-way handshake. This field is set to I-Nonce, the random number generated by the initiator in the command containing a Message Number of 1 or 3, and is set to R-Nonce, the random number generated by the responder in the command containing a Message Number of 2 or 4.

The PTK MIC in the PTK command containing a Message Number of 1 is set to zero on transmission and is ignored on reception.

The PTK MIC in the PTK command containing a Message Number of 2, 3, or 4 is set to the MIC that protects the fields in the payload of this command using the KCK generated from the first two messages of the 4-way handshake as specified in 16.3.1.

The MAC Header for the PTK command frame is set as indicated in Table 110, with the ACK Policy set to Imm-ACK.

14.5.7 Group temporal key (GTK)

The GTK command frame is used to solicit or distribute a GTK following a PTK update. The GTK is used to secure certain multicast traffic from a sending device to a group of recipient devices, and is chosen by the sending device. The GTK command frame is always in secure form, and the Secure Payload field is specified in Figure 97.

octets: 1	**1**	**3**	**3**	**2**	**6**	**16**
Message Number	Status Code	GTKID	Reserved	GroupAddr	GTK SFC	GTK

Figure 97 - Payload format for GTK command frames

The Message Number is set to 0 in the GTK command transmitted by a multicast recipient device to solicit a new GTK from a multicast sender. The Message Number is set to 1 in the GTK command transmitted by a multicast sender to distribute a new GTK to a multicast recipient. The Message Number is set to 2 in the GTK command transmitted by a multicast recipient device to respond to the distribution of a new GTK command.

The Status Code in a GTK command indicates the current status of the GTK solicitation or distribution at the device sending this command. It is encoded as shown in Table 76.

Table 76 - Status Code field encoding in GTK commands

Value	Meaning
0	Normal-GTK solicitation or distribution proceeds
1	Rejected-GTK solicitation or distribution is rejected per security policy
2	GTKID not accepted-it is the TKID of a PTK or GTK being possessed by this device
3 - 255	Reserved

The GTKID in the GTK command containing a Message Number of 0 is set to the TKID of the GTK being solicited. It is set to zero if the soliciting device does not know the TKID of the GTK it is soliciting.

The GTKID in the GTK command containing a Message Number of 1 is set to a non-zero number as the TKID of the GTK being distributed. The distributor chooses this value after determining that this value is different from the TKID of the GTK, if any, that is to be replaced by the new GTK, and the TKID of any PTK or GTK the distributor or recipient currently possesses.

The GTKID in the GTK command containing a Message Number of 2 is set to the GTKID in the last received GTK command containing a Message Number of 1.

The GroupAddr is set to the McstAddr or BcstAddr for which the GTK is being solicited or distributed. It is set to 0x0001 if the GTK is applied to all broadcast and multicast traffic from the device distributing this GTK.

The GTK SFC in the GTK command containing a Message Number of 0 is set to zero on transmission and ignored on reception.

The GTK SFC in the GTK command containing a Message Number of 1 is set to the current value of the secure frame counter set up for the GTK being distributed.

The GTK SFC in the GTK command containing a Message Number of 2 is set to the GTK SFC in the last received GTK command containing a Message Number of 1.

The GTK is the GTK distributed by the multicast sender for the McstAddr. In a GTK command soliciting a GTK, the GTK is set to zero prior to encryption.

The MAC Header for the GTK command frame is set as indicated in 14.5.7, with the ACK Policy set to Imm-ACK.

14.5.8 Application-specific

The payload format for Application-specific command frames is specified in Figure 98.

octets: 2	**N**
Specifier ID	Data

Figure 98 - Payload format for application-specific command frame

The Specifier ID field is set to a 16-bit value that identifies a company or organization, as listed in Annex C.5. The owner of the Specifier ID defines the format and use of the Data field.

14.5.9 Relay

The Relay command frame is used to request information from a device or respond to a Relay request. The payload is the Relay IE as specified in 14.7.22.

14.5.10 Transmit switched diversity (TSD) request

The TSD Set Request command frame is used to initiate the TSD operation, to scan channels or to terminate the TSD operation by acquiring the information on the destination's capability of the TSD operation. This command frame also indicates whether the TSD operation is in scan mode or not. The payload format for TSD Set Request command frame payload is specified in Table 77.

Table 77 - Payload format for TSD set request command frames

Octet: 1	2
TSDStatus	Reserved

The TSDStatus field indicates whether the TSD set operation begins or ends. This field also indicates whether the TSD set operation is in scan mode or not. The TSDStatus field is specified in Table 78.

Table 78 - TSD status field format

bits: b7-b1	b0
Reserved	TSD Status indicator

The TSD status indicator field is set to 0b00 to indicate that this TSD set procedure is to begin the TSD operation, to 0b01 to indicate that this TSD set procedure is to end the TSD operation or to 0b10 to indicate that this TSD set procedure is to request channel status measured from this command frame. The value of 0b11 is reserved. The TSD status indicator field is encoded as Table 79.

Table 79 - TSD status indicator field encoding

TSD Status indicator	Message
0b00	BEGIN
0b01	END
0b10	SCAN
0b11	Reserved

14.5.11 Transmit switched diversity (TSD) set response

The TSD Set Response command frame is used to respond TSD Set Request command frame. The TSD Set Request command frame payload is specified in Figure 99 .

Octet: 1	1	1
TSDStatus	TSDPermit	ChannelStatus

Figure 99 - Payload format for TSD set response command frames

The TSDPermit field indicates whether the device supports the TSD operation or not. The TSDPermit field is specified in Figure 100.

bits: b7-b1	b0
Reserved	TSD support indicator

Figure 100 - TSD permit field format

The TSDPermit field is set to one to indicate that the device supports the TSD operation or to zero to indicate that the device does not support the TSD operation. The TSD support indicator field is encoded as Table 80.

Table 80 - TSD support indicator field encoding

TSD support indicator	Message
0	NO_PERMIT
1	PERMIT

The ChannelStatus field indicates the link quality of the channel, where the link quality shall be defined as an estimate of the received SNR. When reporting the estimates of the link quality, the device shall quantize these values to the nearest values in Table 81 and report them as the link quality estimates. The standard encoding, summarized in Table 81, is used to report the estimates in the range from 0 dB (0b0000 001) to +25 dB (0b0001 1010). The all-ZERO bit implies that reporting of the estimates is not supported by the device, or that the estimate is too small to be measured accurately. Additionally, the range from 0b0001 to 0b1111 1111 are reserved for further use.

Table 81 - Channel status field format

Bits: b7-b0	Description
0b0000 0000	Link quality estimate is too low to be measured
0b0000 0001 - 0b0001 1010	Estimate of the link quality = ChannelStatus - 1 dB
0b0001 1011 - 0b1111 1111	Reserved

14.5.12 Transmit switched diversity (TSD) switch

This frame is used to feedback the transmit antenna switching information for the TSD operation. The TSD Switch control frame commands the recipient of the frame to switch the transmit antenna. This frame is sent from the destination to the source when the source is required to switch the TX antenna. A payload of the TSD Switch control frame is specified in Figure 101.

Octet: 1
TSD Switch

Figure 101 - TSD frame payload

The TSD Switch field specifies the antenna switching information for the recipient of the control frame. The TSD Switch field is specified in Figure 102.

bits: b7-b1	b0
Reserved	Antenna Switch

Figure 102 - TSD switch field format

The Antenna Switch field indicates that the source switches the transmit antenna. The Antenna Switch field is encoded as shown in Table 82.

Table 82 - Antenna switch field encoding

Antenna Switch	Message
0	No Switch

Table 82 - Antenna switch field encoding (concluded)

Antenna Switch	Message
1	Switch

14.5.13 Fast uplink channel allocation (FUCA)

The FUCA command frame is used to inform the reservation target the duration of the next data frame, after which the target may send a short frame back to the reservation owner in the current private DRP reservation.

The Next Frame Duration field is set to the duration of the next data frame, in unit of microsecond, which is sent by reservation owner following the FUCA command frame.

The Sequence Number field is set to the sequence number of the next data frame to be transmitted by reservation owner. FUCA map field contains the information generated by HDMI PAL in accordance with the next data frame.

The FUCA frame payload is shown in Figure 103.

Octets: 2	2	5
Next Frame Duration	Sequence Number	FUCA Map

Figure 103 - Payload format for FUCA frames

14.6 Data frames

MAC Header and Frame Payload fields in data frames are set as described in 14.2.1.

14.7 Information elements

This Clause defines the information elements (IEs) that can appear in beacons and certain control or command frames.

The general format of all IEs is specified in Figure 104.

octets: 1	**1**	**N**
Element ID	Length (=N)	IE-specific fields

Figure 104 - General IE format

The Element ID field is set to the value as listed in Table 83 that identifies the information element.

The Length field is set to the length, in octets, of the IE-specific fields that follow.

The IE-specific fields contain information specific to the IE.

Table 83 contains a list of IEs defined in this Standard.

Table 83 - Information elements

Element ID	Information element	Description
0	Scan Timing IE	Announces the start and duration of a scanning period
1	Beacon Period Occupancy IE (BPOIE)	Provides information on neighbours' BP occupancy in the previous superframe

Table 83 - Information elements (continued)

Element ID	Information element	Description
2	Channel Bonding IE (CBOIE)	Negotiates and indicate the use of channel bonding
3 - 7	Reserved	
8	DRP Availability IE	Indicates a device's availability for new DRP reservations
9	Distributed Reservation Protocol (DRP) IE	Indicates a reservation with another device
10	Hibernation Mode IE	Indicates the device will go to hibernation mode for one or more superframes but intends to wake at a specified time in the future
11	Reserved	Reserved
12	MAC Capabilities IE	Indicates which MAC capabilities a device supports
13	PHY Capabilities IE	Indicates which PHY capabilities a device supports
14	Probe IE	Indicates a device is requesting one or more IEs from another device or/and responding with requested IEs
15	Application-specific Probe IE	Indicates a device is requesting an Application-specific IE from another device
16	Link Feedback IE	Provides data rate and power control feedback
17	Hibernation Anchor IE	Provides information on devices in hibernation mode
18	Channel Change IE	Indicates a device will change to another channel
19	Identification IE	Provides identifying information about the device, including a name string
20	Master Key Identifier (MKID) IE	Identifies some or all of the master keys held by the transmitting device
21	Relinquish Request IE	Indicates that a neighbour requests that a device release one or more MASs from its reservations
22	Multicast Address Binding (MAB) IE	Indicates an address binding between a multicast EUI-48 and a McstAddr
23	Relay IE	Indicates a device requesting relay information element from another device or/and responding with requested IE
24	ACIE	This IE shall be transmitted during the discovery process, in beacons and during association. It contains information about the RX and TX antenna types and feedback capabilities.
25	ATIE	This IE shall be transmitted in the RTT and CTT control frames. It is used to negotiate the parameters of the training protocol.
26	ATFIE	This IE shall be transmitted in antenna training frames. It is used to convey information about the iterative training process and to request for parameters.
27	AFIE	This IE shall be transmitted in a training frame. It conveys the feedback for the transmitter.

Table 83 - Information elements (concluded)

Element ID	Information element	Description
28	UEP IE	Provide information on supported UEP types and UEP MCS modes
29	MSP Timing IE	This IE is used in a Master-Slave operation, to convey the timing information of the current Master-Slave operation.
30	CTR Description IE	Used in a Master-Slave operation to describe an allocated CTR
31	MSP Interval Change IE	Used in a Master-Slave operation to announce that the Master device is preparing to change its MSP Interval.
32	Scan Countdown IE	Used in a Master-Slave operation to announce that the Master and Slave devices are preparing to perform aperiodic channel scanning.
33 - 254	Reserved	Reserved
255	Application-Specific IE (ASIE)	Use varies depending on the application

14.7.1 Application-specific IE (ASIE)

The ASIE is specified in Figure 105.

octets: 1	**1**	**2**	**N**
Element ID	Length (=2+N)	ASIE Specifier ID	Application-specific Data

Figure 105 - ASIE format

The ASIE Specifier ID field is set to a 16-bit value that identifies a company or organization, as listed in Annex C.4.

The owner of the ASIE Specifier ID defines the format and use of the Application-specific Data field.

14.7.2 Application-specific probe IE

The Application-specific Probe IE is used to request an application-specific IE from a device. It is specified in Figure 106.

octets: 1	**1**	**2**	**2**	**N**
Element ID	Length (=4+N)	Target DevAddr	ASIE Specifier ID	Application-specific Request Information

Figure 106 - Application-specific probe IE format

The Target DevAddr field is set to the DevAddr of the device from which an ASIE is requested.

The ASIE Specifier ID is set to a 16-bit value that identifies a company or organization, as listed in Annex C.4.

The owner of the ASIE Specifier ID defines the format and use of the Application-specific Request Information field.

14.7.3 ATTP IE

Four IEs shall be employed to support antenna training:

- Antenna Capability IE (ACIE)
- Antenna Training IE (ATIE)
- Antenna Training Frame IE (ATFIE)
- Antenna Feedback IE (AFIE)

14.7.3.1 ACIE

The ACIE is shown in Figure 107. This IE shall be transmitted by Type A and Type B devices during the discovery process, in beacons and during association. It contains information about the device type and the training and feedback capabilities of the RX and TX antenna of the transmitting device. The ACIE fields describe the following:

bits: 8	8	16
Element ID	Length	Antenna Capabilities

Figure 107 - ACIE

The 16 bits of the Antenna Capabilites field have the following meaning:

bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7	bits 8-15
Type A (0) Type B (1)	Type B that can transmit sequence? (yes:1, no:0)	Type A with trainable TX phased array antenna? (yes:1, no:0)	Type A with trainable RX phased array antenna? (yes:1, no:0)	supports giving index feedback? (yes:1, no:0)	supports giving Fourier codebook feedback? (yes:1, no:0)	supports giving Walsh codebook feedback? (yes:1, no:0)	supports giving quantised weights feedback? (yes:1, no:0)	reserved

Figure 108 - Antenna Capabilities field

14.7.3.2 ATIE

The ATIE is depicted in Figure 109. This IE is used to negotiate and configure the antenna training procedure. The ATIE shall be transmitted in the RTT and CTT control frames used in the antenna training process. The ATIE fields describe the following:

- Antenna Capabilities: (16 bits) set to the same value as in the ACIE (see 14.7.3.1)
- Number of Elements of TX Antenna (6 bits). This number shall lie between 1 and 36 (inclusive).
- Training Block Size: number of training sequences used per block.
- Continuation (1 bit): set to 0 if new training, set to 1 if continuation of a previous training.
- Training Desired (1 bit): if set to 1, indicates that training is desired. If set to 0, training is not desired. This field has an implication in the subsequent training sequence exchange.
- Feedback Needed (1 bit): indicates whether implicit (set to 0) or explicit (set to 1) feedback is required from the receiver. This field has an implication in the subsequent training sequence exchange.

- Type of Feedback Wanted (4 bits). Indicates what kind of explicit feedback is wanted. This field shall be ignored by the receiver if Feedback Needed is set to 0. Bits 2-3 are reserved, bits 0-1 have the following meaning:

Table 84 - Bits 0 and 1 definition

Bits 0-1	**Type of feedback**
00	index
01	Fourier codebook
10	Walsh codebook
11	quantised weights

- Status: indicates the status of the negotiation, and shall be set by the transmitter and receiver devices involved in the antenna training procedure. Possible values are: 0 = request; 1 = accepted; 2 = not supported; 3 = rejected.

bits: 8	8	16	6	6	6	1	1	1	4	2	6
Element ID	Length	Antenna Capabilities	Num of TX Antenna Elements	Num of RX Antenna Elements	Training Block Size	Continuation	Training Desired	Feed-back Needed	Type of Feed-back Wanted	Status	Reserved

Figure 109 - ATIE

14.7.3.3 ATFIE

The Antenna Training Frame IE (ATFIE) is shown in Figure 110. The ATFIE is transmitted in TRN control frames to convey information about the iterative training process. And to request the spreading factor and number of training sequences for the next TRN control frame in the reverse direction.

bits: 8	8	2	1	4	6	3
Element ID	Length	Iteration	Training/Tracking	Requested DISC_REP	Requested NUM_RXTS	Reserved

Figure 110 - ATFIE

- Iteration counts the iteration number of the TRN control frame. Possible values are 0,1,2,3.
- Requested DISC_REP is set to the number of repetitions in the discovery mode, encoded the same as the DISC_REP field in the ATIF (10.1.2.1.3).
- Requested number of training symbols for receiver training, encoded just like NUM_RXTS in the ATIF (10.1.2.1.3).

14.7.3.4 AFIE

Antenna Feedback IE (AFIE) is shown in Figure 111. The AFIE, if present as indicated by the ACK Needed field negotiated through the ATIE, provides the feedback to the transmitter on the status of the antenna training procedure and on the transmitter weights. The AFIE shall be included in a TRN Frame. The AFIE fields describe the following:

- Status: indicates the status of the antenna training process, as taken from Table 85.
- Type of feedback: encodes the type of feedback given in this AFIE, according to Table 84.

- Selected Index: indicates the index of the best received training symbol, or the selected codeword. Only valid if the Type of feedback == 00, 01 or 11.
- Quantised Transmitter Weights: indicates quantised transmitter weights. Only valid if Type of feedback == 11.

bits: 8	8	1	4	3	8	288
Element ID	Length	Status	Type of Feedback	Reserved	Selected Index	Quantised Transmitter Weights

Figure 111 - AFIE

Table 85 - Status field

Status	Value
Done	0
Continue	1

14.7.4 Beacon period occupancy IE (BPOIE)

The BPOIE provides information on the BP observed by the device sending the IE. The BPOIE is specified in Figure 112.

octets: 1	**1**	**1**	**K**	**2**	**…**	**2**
Element ID	Length (=1+K+2×N)	BP Length	Beacon Slot Info Bitmap	DevAddr 1	…	DevAddr N

Figure 112 - BPOIE format

The BP Length field is set to the length of the BP, measured in beacon slots, as defined in 15.5.5.3.

The Beacon Slot Info Bitmap field consists of K octets of 2-bit elements to indicate the beacon slot occupancy and movability in the BP, where K = Ceiling (BP_Length/4). Each element n, numbered from 0 to 4×K-1, corresponds to beacon slot n and is encoded as shown in Table 86. Element zero is the least-significant two bits of the field. Unused elements, if any, are set to zero.

Table 86 - Beacon slot info bitmap element encoding

Element value	Beacon slot status
0	Unoccupied (non-movable) No PHY indication of medium activity was received in the corresponding beacon slot in the last superframe, or any frame header received with a valid HCS was not a beacon frame.
1	Occupied & non-movable A beacon frame was received with a valid HCS and FCS in the corresponding beacon slot in the last superframe, and the Movable bit in that beacon was set to ZERO, or a beacon frame was received in the corresponding beacon slot in a previous superframe that indicated a hibernation period that has not expired, as described in 15.16.
2	Occupied & movable A PHY indication of medium activity was received in the corresponding beacon slot in the last superframe, but did not result in reception of a frame with valid HCS and FCS.
3	Occupied & movable A beacon frame was received with a valid HCS and FCS in the corresponding beacon slot in the last superframe, and the Movable bit in that beacon was set to ONE.

The DevAddr fields correspond to beacon slots encoded as occupied in the Beacon Slot Info Bitmap. They are included in ascending beacon slot order. If a beacon was received with a valid HCS at a beacon slot in the last superframe, the corresponding DevAddr field is set to the SrcAddr in the MAC header of that received beacon. If a frame was received with an invalid HCS from a beacon slot in the last superframe, the corresponding DevAddr field is set to BcstAddr. If a neighbour of the device is in hibernation mode, the DevAddr field that corresponds to the hibernating neighbour's beacon slot is set to the DevAddr of that neighbour.

14.7.5 Channel bonding IE (CBOIE)

A CBOIE is used to negotiate bonding of two or more channels and to announce the channels that are involved in the channel bonding. The CBOIE is specified in Figure 113.

octets: 1	1	2	1
ElementID	Length (=3)	TargetDevAddr	Bonding Control

Figure 113 - Channel Bonding IE format

The Target DevAddr field is set to the DevAddr of the recipient device to which the device is sending the channel bonding request.

The Bonding Control field is specified in Figure 114.

bits: b7	b6	b5-b4	b3-b0
Bonding Status	Reserved	Reason Code	Bonded Channel Bit Map

Figure 114 - Bonding control field format

The Bonding Status bit indicates the status of the channel bonding negotiation process. The Bonding Status bit is set to ZERO in a CBOIE for a channel bonding request. It is set to ONE by a device granting or accepting the bonding request, which is then referred to as an established channel bonding.

The Reason code field is used by the targeted device to indicate whether it accepts a channel bonding request and is encoded as shown in Table 87.

Table 87 - Reason code field encoding

Value	Code	Meaning
0	Accepted	The channel bonding request is granted
1	Unavailable	The channel bonding request is rejected because one of the channels is not idle
2	Conflict	The channel bonding request is rejected because one of the channels is already bonded.
3	Invalid	The band ID is not bonded channel band ID

The Bonded Channel Bitmap field is specified in Figure 115.

Bit	Band ID	Description
0	1	Bonding channel 1
1	2	Bonding channel 2
2	3	Bonding channel 3
3	4	Bonding channel 4

Figure 115 - Bonded channel bitmap

14.7.6 Channel change IE

A Channel Change IE announces that a device is preparing to change to another channel.

The Channel Change IE is specified in Figure 116.

octets: 1	1	1	1
Element ID	Length (=2)	Channel Change Countdown	New Channel Number

Figure 116 - Channel change IE format

The Channel Change Countdown field is set to the number of superframes remaining until the device changes to the new channel. If this field is zero, the device will change to the new channel at the end of the current superframe.

The New Channel Number field is set to the channel number of the new channel to which the device will change.

The Channel Change Countdown field in a discovery beacon and channel selection command frames shall be set to Zero; and the New Channel Number field in a discovery beacon shall be set to the channel in which the device sets up its superframe structure.

14.7.7 CTR description IE

CTR Description IE is used in a Master-Slave operation, to describe an allocated CTR.

The CTR Description IE is specified in Figure 117.

octets: 1	1	2	2
Element ID	Control Information	CTR start time	CTR Duration

Figure 117 - CTR description IE format

The format of the Control Information field is specified in Figure 118.

bits: b7- b5	b4	b3- b0
Reserved	Target	CTR ID

Figure 118 - Control information format

The CTR ID field contains a unique 4-bit ID, assigned by a Master device, to identify an allocated CTR.

The Target field indicates the target device of the CTR. The possible values and its interpretations are:

0: The CTR is allocated for Slave device to transmit to Master device.

1: The CTR is allocated for Master device to transmit to Slave device.

The CTR start time field indicates the time at which the CTR starts. It is measured, in microseconds, from the start of the MSP.

The CTR Duration field indicates, in microseconds, the total duration of the CTR, including the trailing SIFS and Guard-time (if any).

14.7.8 Distributed reservation protocol (DRP) IE

A DRP IE is used to negotiate a reservation or part of a reservation for certain MASs and to announce the reserved MASs. The DRP IE is specified in Figure 119.

octets: 1	1	2	1	2	4		4
Element ID	Length (=4+4xN)	DRP Control	Antenna Index	Target/Owner DevAddr	DRP Allocation 1	...	DRP Allocation N

Figure 119 - DRP IE format

The DRP Control field is specified in Figure 120.

bits: b15-b14	b13	b12	b11	b10	b9	b8-b6	b5-b3	b2-b0
Reserved	Waveform	Unsafe	Conflict Tie-breaker	Owner	Reservation Status	Reason Code	Stream Index	Reservation Type

Figure 120 - DRP control field format

The Reservation Type field is set to the type of the reservation and is encoded as shown in Table 88.

Table 88 - Reservation type field encoding

Value	Reservation Type
0	Alien BP
1	Hard
2	Soft
3	Private
4	Absence
5 - 7	Reserved

The Stream Index field identifies the stream of data to be sent in the reservation. This field is reserved if the Reservation Type is Alien BP.

The Reason Code is used by a reservation target to indicate whether a DRP reservation request was successful and is encoded as shown in Table 89. The Reason Code is set to zero in a DRP IE sent during negotiation by a reservation owner and by a device maintaining an established reservation. The Reason Code is set to Modified by a device if some of the MASs claimed in the reservation have been removed or if DRP IEs have been combined, split or both. The field is reserved if the Reservation Type is Alien BP.

Table 89 - Reason code field encoding

Value	Code	Meaning
0	Accepted	The DRP reservation request is granted
1	Conflict	The DRP reservation request or existing reservation is in conflict with one or more existing DRP reservations
2	Pending	The DRP reservation request is being processed
3	Denied	The DRP reservation request is rejected or existing DRP reservation can no longer be accepted
4	Modified	The DRP reservation is still maintained but has been reduced in size or multiple DRP IEs for the same reservation have been combined
5 - 7	Reserved	Reserved

The Reservation Status bit indicates the status of the DRP negotiation process. The Reservation Status bit is set to ZERO in a DRP IE for a reservation that is under negotiation or in conflict. It is set to ONE by a device granting or maintaining a reservation, which is then referred to as an established reservation. The bit is set to ONE if Reservation Type is Alien BP.

The Owner bit is set to ONE if the device transmitting the DRP IE is the reservation owner, or to ZERO if the device transmitting the DRP IE is a reservation target. The bit is reserved if the Reservation Type is Alien BP.

The Conflict Tie-breaker bit is set to a random value of ZERO or ONE when a reservation request is made. The same value selected is used as long as the reservation is in effect. For all DRP IEs that represent the same reservation, the Conflict Tie-breaker bit is set to the same value.

The Target/Owner DevAddr field is set to the DevAddr of the reservation target if the device transmitting this DRP IE is the reservation owner. The reservation target may be a unicast or multicast DevAddr. The field is set to the DevAddr of the reservation owner if the device transmitting the DRP IE is a reservation target. The field is reserved if the Reservation Type is Alien BP.

The Unsafe bit is set to ONE if any of the MASs identified in the DRP Allocation fields is considered in excess of reservation limits.

The Waveform bit encoding for Hard and Soft reservations is specified in Table 90. This bit is reserved for reservations of other types.

Table 90 - Waveform field encoding

	Device Type	
Value	Type A	Type B
0	SCBT	SC
1	OFDM	DAMI

A DRP IE contains one or more DRP Allocation fields. Each DRP Allocation field is encoded using a zone structure. The superframe is split into 16 zones numbered from 0 to 15 starting from the BPST. Each zone contains 16 consecutive MASs, which are numbered from 0 to 15 within the zone.

The format of a DRP Allocation field is specified in Figure 121.

octets: 2	**2**
Zone Bitmap	MAS Bitmap

Figure 121 - DRP allocation field format

The Zone Bitmap field identifies the zones that contain reserved MASs. If a bit in the field is set to ONE, the corresponding zone contains reserved MASs, where bit zero corresponds to zone zero.

The MAS Bitmap specifies which MASs in the zones identified by the Zone Bitmap field are part of the reservation. If a bit in the field is set to ONE, the corresponding MAS within each zone identified by the Zone Bitmap is included in the reservation, where bit zero corresponds to MAS zero within the zone.

14.7.9 DRP availability IE

The DRP Availability IE is used by a device to indicate its view of the current utilization of MASs in the current superframe. The DRP Availability IE is specified in Figure 122.

octets: 1	1	N (0 to 32)
Element ID	Length (=N)	DRP Availability Bitmap

Figure 122 - DRP availability IE format

The DRP Availability Bitmap field is up to 256 bits long, one bit for each MAS in the superframe, where the least-significant bit of the field corresponds to the first MAS in the superframe and successive bits correspond to successive MASs. Each bit is set to ONE if the device is available for a DRP reservation in the corresponding MAS, or is set to ZERO otherwise. If the DRP Availability Bitmap field is smaller than 32 octets, the bits in octets not included at the end of the bitmap are treated as ZERO.

14.7.10 Hibernation anchor IE

The Hibernation Anchor IE is specified in Figure 123.

octets: 1	1	3	…	3
Element ID	Length (=3×N)	Hibernation Mode Device Information 1	…	Hibernation Mode Device Information N

Figure 123 - Hibernation anchor IE format

The Hibernation Mode Device Information field is specified in Figure 124.

octets: 2	1
Hibernation Mode Neighbour DevAddr	Wakeup Countdown

Figure 124 - Hibernation mode device information field format

The Hibernation Mode Neighbour DevAddr field is set to the DevAddr of the neighbour in hibernation mode.

The Wakeup Countdown field is set to the number of remaining superframes before the device in hibernation mode is expected to wake up. A value of zero indicates that the device is scheduled to be in active mode in the next superframe.

14.7.11 Hibernation mode IE

The Hibernation Mode IE is specified in Figure 125.

octets: 1	1	1	1
Element ID	Length (=2)	Hibernation Countdown	Hibernation Duration

Figure 125 - Hibernation mode IE format

The Hibernation Countdown field is set to the number of superframes remaining until the device begins hibernation. A value of zero indicates that the device will enter hibernation mode at the end of the current superframe.

The Hibernation Duration field is set to the number of superframes for which the device intends to hibernate.

14.7.12 Identification IE

The Identification IE provides identifying information about the device, including a name string. The Identification IE is specified in Figure 126.

octets:1	**1**	**M_1**	**…**	**M_N**
Element ID	Length (=M_1+…+ M_N)	Device Information 1	…	Device Information N

Figure 126 - Identification IE format

The general format of the Device Information field is specified in Figure 127.

octets: 1	**1**	**N**
Device Information Type	Device Information Length (=N)	Device Information Data

Figure 127 - Device information field format

The encoding for the Device Information Type field is shown in Table 91.

Table 91 - Device information type field encoding

Value	Device Information Data field contents
0	Vendor ID
1	Vendor Type
2	Name String
3 - 255	Reserved

The Device Information Length field indicates the length, in octets, of the Device Information Data Field that follows.

The Device Information Data field, if Device Information Type is Vendor ID, is specified in Figure 128.

octets: 3
Vendor ID

Figure 128 - Device information data field format for vendor ID

The Vendor ID is set to an OUI that indicates the vendor of the device. The OUI is a sequence of 3 octets, labelled as oui[0] through oui[2]. Octets of the OUI are passed to the PHY SAP in ascending index-value order.

The Device Information Data field, if Device Information Type is Vendor Type, is specified in Figure 129.

octets: 3	3
Vendor ID	Device Type ID

Figure 129 - Device information data field format for vendor type

The Vendor ID field is set to an OUI that indicates the entity that assigns the values used in the Device Type ID field. The Device Type ID field indicates the type of device.

The Device Information Data field, if Device Information Type is Name String, contains the name of the device encoded in Unicode UTF-16LE format, and is specified in Figure 130.

octets: 2	…	2
Name String Unicode Char 1	…	Name String Unicode Char N

Figure 130 - Device information data field format for name string

14.7.13 Link feedback IE

The Link Feedback IE contains information on the recommended change to the data rate and transmit power level by a recipient device for one or more source devices. The Link Feedback IE is specified in Figure 131.

octets: 1	1	3		3
Element ID	Length (=3xN)	Link 1	…	Link N

Figure 131 - Link feedback IE format

The Link field is specified in Figure 132.

bits: b23-b20	b19-b16	b15-b0
Data Rate	Transmit Power Level Change	DevAddr

Figure 132 - Link field format

The DevAddr field is set to the DevAddr of the source device for which the feedback is provided.

The Transmit Power Level Change field is set to the change in transmit power level that the recipient recommends to the source device. The Transmit Power Level Change field encoding is shown in Table 92.

Table 92 - Transmit power level change field encoding

Value	Power level change	Power level change (dB)
1000 - 1101	Reserved	Reserved
1110	-2	-4
1111	-1	-2
0000	no change	no change
0001	+1	+2 dB
0010	+2	+4 dB
0011 - 0111	Reserved	Reserved

The Data Rate field is set to the PHY mode of the data rate that the recipient device recommends that the source device use. The Data Rate field is encoded as shown in Table 93.

Table 93 - Data rate field encoding

Value	Type A SCBT modes	Type A OFDM modes	Type B modes	Type C modes
0	A0	A14	B0	C0
1	A1	A15	B1	C1
2	A2	A16	B2	C2
3	A3	A17	B3	Reserved
4	A4	A18	Reserved	Reserved
5	A5	A19	Reserved	Reserved
6	A6	A20	Reserved	Reserved
7	A7	A21	Reserved	Reserved
8	A8	Reserved	Reserved	Reserved
9	A9	Reserved	Reserved	Reserved
10	A10	Reserved	Reserved	Reserved
11	A11	Reserved	Reserved	Reserved
12	A12	Reserved	Reserved	Reserved
13	A13	Reserved	Reserved	Reserved
14-15	Reserved	Reserved	Reserved	Reserved

14.7.14 MAC capabilities IE

The MAC Capabilities IE is specified in Figure 133.

octets: 1	1	2	2
Element ID	Length (=4)	MAC Capability Bitmap	Reserved

Figure 133 - MAC capabilities IE format

The MAC Capability Bitmap field indicates capabilities supported by the MAC entity. A bit is set to ONE if the corresponding attribute is supported, or is set to ZERO otherwise. This field is encoded as described in Table 94. Subsequent octets are reserved and may or may not be present.

Table 94 - MAC capability bitmap

octet	Bit	Attribute	Description
0	0	Reserved	
	1	Hard DRP	Capable of being the owner and target of Hard DRP reservations
	2	Soft DRP	Capable of being the owner and target of Soft DRP reservations
	3	Block ACK	Capable of transmitting and acknowledging frames using the B-ACK mechanism
	4	Explicit DRP negotiation	Capable of negotiating a DRP reservation using command frames
	5	Hibernation anchor	Capable of acting as a hibernation anchor
	6	Probe	Capable of responding to Probe IEs received in command frames
	7	Reserved	Reserved
1	0	TPRC	Capable of receiver-driven transmit power and rate control
	1	Link Feedback	Capable of transmitter-driven link feedback
	2	Relay Capability	Capable of being a relay device
	3	Relay Support Capability	Capable of being a device that send or receive data via relay device
	4	TSD Capability	Capable of TSD (Transmit Switched Diversity) operation
	5	FUCA Capability	Capable of sending or receiving FUCA command frame
	6	OOB Capability	Capable of OOB operation
	7	Reserved	Reserved

14.7.15 Master key identifier (MKID) IE

The MKID IE is used to identify some or all of the master keys possessed by the device. The MKID IE is specified in Figure 134.

octets: 1	**1**	**16**	**…**	**16**
Element ID	Length (=16×N)	MKID 1	…	MKID N

Figure 134 - MKID IE format

Each MKID field is set to the identifier of a master key possessed by the device.

14.7.16 MSP interval change IE

A MSP Interval Change IE is used in a Master-Slave operation, to announce that the Master device is preparing to change its MSP Interval.

The MSP Interval Change IE is specified in Figure 135.

octets: 1	1	1	2
Element ID	Length (=3)	Change countdown	New MSP Interval

Figure 135 - MSP interval change IE format

The Change countdown is field is set to the number of MSP remaining until the Master device changes to the new MSP Interval. If this field is zero, the Master device will change to use the new MSP Interval at the end of the current MSP.

The New MSP Interval field is set to the length, in microseconds, of the new MSP Interval that the Master device is changing to.

14.7.17 MSP timing IE

MSP Timing IE is used in a Master-Slave operation, to convey the timing information of the current Master-Slave operation.

The MSP Timing IE is specified in Figure 136.

octets: 1	1	2	2
Element ID	Length (=4)	Next MSP start time	CEP Duration

Figure 136 - MSP timing IE format

The Next MSP Start time is the length of time, in microseconds, to the start of the next MSP. It shall be measured from the time the frame is being transmitted, to the determined next MSP start time.

CEP Duration indicates the length, in microseconds, of the CEP in the current MSP.

14.7.18 Multicast address binding (MAB) IE

Each device maps multicast EUI-48s to McstAddrs in the 16-bit DevAddr address range. The MAB IE declares the binding between a multicast EUI-48 and the McstAddr that the device will use when transmitting frames destined for that multicast EUI-48.

The format of the MAB IE is shown in Figure 137.

octets: 1	1	8	...	8
Element ID	Length (=8×N)	Multicast Address Binding Block 1	...	Multicast Address Binding Block N

Figure 137 - MAB IE format

The format of the Multicast Address Binding Block field is shown in Figure 138.

octets : 6	2
MEUI	MDevAddr

Figure 138 - Multicast address binding block format

The MEUI field is set to the multicast EUI-48 supplied by the MAC client at the MAC SAP.

The MDevAddr field is set to the multicast DevAddr bound to the MEUI field by the MAC entity from the McstAddr address range.

14.7.19 PHY capabilities IE

The PHY Capabilities IE pertaining to the PHY is specified in Figure 139.

octets: 1	1	9	2
Element ID	Length (=11)	PHY Capability Bitmap	Reserved

Figure 139 - PHY capabilities IE format

The PHY Capability Bitmap field indicates capabilities supported by the PHY, as defined in the Physical Layer Clauses of this specification (Clauses 7 - 15). A bit is set to ONE if the corresponding attribute is supported, or is set to ZERO otherwise. This field is encoded as described in Table 95. Subsequent octets are reserved and may or may not be present.

Table 95 - PHY capability bitmap

Octet	Bit	Attribute	Description	Type A	Type B	Type C
0	0	Type A	Bit set to one indicates that the device type is A	1	0	0
	1	Type B	Bit set to one indicates that the device type is B	0	1	0
	2	Type C	Bit set to one indicates that the device type is C	0	0	1
	3	Channel 1	Bit set to one indicates support for channel 1 (BAND ID=1)	0/1	0/1	0/1
	4	Channel 2	Bit set to one incates support for channel 2 (BAND ID=2)	0/1	0/1	0/1
	5	Channel 4	Bit set to one incates support for channel 4 (BAND ID=4)	0/1	0/1	0/1
	6	Channel 5	Bit set to one incates support for channel 5 (BAND ID=5)	0/1	0/1	0
	7	Channel 6	Bit set to one incates support for channel 6 (BAND ID=6)	0/1	0/1	0
1	0	Channel 7	Bit set to one incates support for channel 7 (BAND ID=7)	0/1	0/1	0
	1	Channel 8	Bit set to one incates support for channel 8 (BAND ID=8)	0/1	0/1	0
	2	Channel 9	Bit set to one incates support for channel 9 (BAND ID=9)	0/1	0/1	0
	3	Channel 10	Bit set to one incates support for channel 10 (BAND ID=10)	0/1	0/1	0
	4	A1 (TX)	Bit set to one indicates TX support for mode A1	0/1	0	0
	5	A1 (RX)	Bit set to one indicates RX support for mode A1	0/1	0	0
	6	A2 (TX)	Bit set to one indicates TX support for mode A2	0/1	0/1	0
	7	A2 (RX)	Bit set to one indicates RX support for mode A2	0/1	0	0
2	0	A3 (TX)	Bit set to one indicates TX support for mode A3	0/1	0	0
	1	A3 (RX)	Bit set to one indicates RX support for mode A3	0/1	0	0
	2	A4 (TX)	Bit set to one indicates TX support for mode A4	0/1	0	0
	3	A4 (RX)	Bit set to one indicates RX support for mode A4	0/1	0	0
	4	A5 (TX)	Bit set to one indicates TX support for mode A5	0/1	0/1	0
	5	A5 (RX)	Bit set to one indicates RX support for mode A5	0/1	0/1	0
	6	A6 (TX)	Bit set to one indicates TX support for mode A6	0/1	0	0
	7	A6 (RX)	Bit set to one indicates RX support for mode A6	0/1	0	0

Table 95 - PHY capability bitmap (continued)

Octet	Bit	Attribute	Description	Type A	Type B	Type C
3	0	A7 (TX)	Bit set to one indicates TX support for mode A13	0/1	0	0
	1	A7 (RX)	Bit set to one indicates RX support for mode A14	0/1	0	0
	2	A8	Bit set to one indicates TX support for mode A8	0/1	0	0
	3	A8	Bit set to one indicates RX support for mode A8	0/1	0	0
	4	A9 (TX)	Bit set to one indicates TX support for mode A9	0/1	0	0
	5	A9 (RX)	Bit set to one indicates RX support for mode A9	0/1	0	0
	6	A10 (TX)	Bit set to one indicates TX support for mode A10	0/1	0	0
	7	A10 (RX)	Bit set to one indicates RX support for mode A10	0/1	0	0
4	0	A11 (TX)	Bit set to one indicates TX support for mode A11	0/1	0	0
	1	A11 (RX)	Bit set to one indicates RX support for mode A11	0/1	0	0
	2	A12 (TX)	Bit set to one indicates TX support for mode A12	0/1	0	0
	3	A12 (RX)	Bit set to one indicates RX support for mode A12	0/1	0	0
	4	A13 (TX)	Bit set to one indicates TX support for mode A13	0/1	0	0
	5	A13 (RX)	Bit set to one indicates RX support for mode A13	0/1	0	0
	6	A14 (TX)	Bit set to one indicates TX support for mode A14	0/1	0	0
	7	A14 (RX)	Bit set to one indicates RX support for mode A14	0/1	0	0
5	0	A15 (TX)	Bit set to one indicates TX support for mode A15	0/1	0	0
	1	A15 (RX)	Bit set to one indicates RX support for mode A15	0/1	0	0
	2	A16 (TX)	Bit set to one indicates TX support for mode A16	0/1	0	0
	3	A16 (RX)	Bit set to one indicates RX support for mode A16	0/1	0	0
	4	A17 (TX)	Bit set to one indicates TX support for mode A17	0/1	0	0
	5	A17 (RX)	Bit set to one indicates RX support for mode A17	0/1	0	0
	6	A18 (TX)	Bit set to one indicates TX support for mode A18	0/1	0	0
	7	A18 (RX)	Bit set to one indicates RX support for mode A18	0/1	0	0
6	0	A19 (TX)	Bit set to one indicates TX support for mode A19	0/1	0	0
	1	A19 (RX)	Bit set to one indicates RX support for mode A19	0/1	0	0
	2	A20 (TX)	Bit set to one indicates TX support for mode A20	0/1	0	0
	3	A20 (RX)	Bit set to one indicates RX support for mode A20	0/1	0	0
	4	A21 (TX)	Bit set to one indicates TX support for mode A21	0/1	0	0
	5	A21 (RX)	Bit set to one indicates RX support for mode A21	0/1	0	0
	6	B1 (TX)	Bit set to one indicates TX support for mode B1	0/1	0/1	0
	7	B1 (RX)	Bit set to one indicates RX support for mode B1	0/1	0/1	0

Table 95 - PHY capability bitmap (concluded)

Octet	Bit	Attribute	Description	Type A	Type B	Type C
7	0	B2 (TX)	Bit set to one indicates TX support for mode B2	0/1	0/1	0
	1	B2 (RX)	Bit set to one indicates RX support for mode B2	0/1	0/1	0
	2	B3 (TX)	Bit set to one indicates TX support for mode B3	0/1	0/1	0
	3	B3 (RX)	Bit set to one indicates RX support for mode B3	0/1	0/1	0
	4	B4 (TX)	Bit set to one indicates TX support for mode B4	0/1	0/1	0
	5	B4 (RX)	Bit set to one indicates RX support for mode B4	0/1	0/1	0
	6	C1 (TX)	Bit set to one indicates TX support for mode C1	0/1	0/1	0/1
	7	C1 (RX)	Bit set to one indicates RX support for mode C1	0/1	0/1	0/1
8	0	C2 (TX)	Bit set to one indicates TX support for mode C2	0/1	0/1	0/1
	1	C2 (RX)	Bit set to one indicates RX support for mode C2	0/1	0/1	0/1
	2	Multi-Segment	Bit set to one indicates support for multi-segment PPDU	1	0/1	0
	3	Reserved				
	4	Reserved				
	5	Reserved				
	6	Reserved				
	7	Reserved				

14.7.20 Probe IE

The Probe IE is used to request information from a device. It is specified in Figure 140.

octets: 1	1	2	1	...	1
Element ID	Length (=2+N)	Target DevAddr	Requested Element ID 1	…	Requested Element ID N

Figure 140 - Probe IE format for standard IEs

The Target DevAddr field is set to the DevAddr of the device from which IEs are requested or the device that requests IEs.

Each Requested Element ID field is set to the element ID of a requested IE.

14.7.21 Relinquish request IE

The Relinquish Request IE is used to request that a device release one or more MASs from one or more existing reservations. It identifies the target device and the desired MASs, and is specified in Figure 141.

octets: 1	1	2	2	4		4
Element ID	Length (=4+4×N)	Relinquish Request Control	Target DevAddr	Allocation 1	…	Allocation N

Figure 141 - Relinquish request IE format

The Relinquish Request Control field is specified in Figure 142.

bits: b15-b4	**b3-b0**
Reserved	Reason Code

Figure 142 - Relinquish request control field format

The Reason Code field indicates the reason for the request, and is encoded as shown in Table 96.

Table 96 - Reason code field encoding

Value	Code	Meaning
0	Non-specific	No reason specified
1	Over-allocation	The target device holds more MASs than permitted by policy
2 - 15	Reserved	Reserved

The Target DevAddr field is set to the DevAddr of the device that is requested to release MASs.

A Relinquish Request IE contains one or more Allocation fields. Each Allocation field is encoded using a zone structure. The superframe is split into 16 zones numbered from 0 to 15 starting from the BPST. Each zone contains 16 consecutive MASs, which are numbered from 0 to 15 within the zone.

The general format of an Allocation field is specified in Figure 143.

octets: 2	**2**
Zone Bitmap	MAS Bitmap

Figure 143 - Allocation field format

The Zone Bitmap field identifies the zones that contain requested MASs. If a bit in the field is set to ONE, the corresponding zone contains requested MASs, where bit zero corresponds to zone zero.

The MAS Bitmap specifies which MASs in the zones identified by the Zone Bitmap field are part of the request. If a bit in the field is set to ONE, the corresponding MAS within each zone identified by the Zone Bitmap is included in the request, where bit zero corresponds to MAS zero within the zone.

14.7.22 Relay IE

Relay IE is used to request or respond for relay operation such as Relay Reservation, Relay Set, Relay Complete and Relay Switch or is also used for a device to inform a corresponding device or relay device of setting parameters for relay operation. Relay IE is shown in Figure 144.

octets: 1	1	1	1	1	1	1	2	2	2	2
Element ID	Length	Relay Command Type	Relay Control	R-D Link LQI	Relay Mode	Path Change Interval	Detour Start Duration	Relay DevAddr	Source DevAddr	Destination DevAddr

Figure 144 - Relay IE

The Relay Command Type field is specified in Table 97.

Table 97 - Relay command type field

Value	Relay Command Type
0	Relay Reservation Request
1	Relay Reservation Response
2	Relay Set Request
3	Relay Set Response
4	Relay Complete Request
5	Relay Complete Response
6	Relay Switch Request
7	Relay Switch Response

The Relay Control field is specified in Figure 145.

bits: b7-b6	b5	b4-b3	b2	b1-b0
Path ID	Result Code	Relay Reservation	Start Relay	Relay Antenna Complete

Figure 145 - Relay control field

The Path ID field indicates the ID of the path requested by the source device in Relay Switch Request command.

The Result Code field is set to ONE if the request command is successful and is set to ZERO if the request command is failed.

The Relay Reservation field indicates the reason codes of relay reservation request. The possible values and its interpretations are:

0: Relay reservation is accepted by a relay device.

1: Relay reservation is rejected because the relay has been already reserved by another device.

2: Relay reservation request is invalid because the requested device does not support relaying

3: Reserved.

The Start Relay field is set to ONE if a device received Relay Set Request command is able to do antenna training with source and destination device. Otherwise it is set to ZERO.

The Relay Antenna Complete field indicates the reason codes of Relay Complete request. The possible values and its interpretations are:

0: Relay Complete request is accepted because the antenna training between relay device and destination device is successful. In this case, a number of DRP available IEs which informs the source of the result of Channel Scanning Response shall be included in the Relay Complete Response frame.

1: Relay Complete request is rejected because the antenna training between relay device and destination device is failed.

2 - 3: Reserved.

The R-D Link LQI field indicates LQI of the link between a relay device and a destination device. It is set only for Relay Complete Response command frame in order for the destination device to inform the source of the LQI of the relay-destination link if antenna training of the link is successfully done.

The Relay Mode field is specified in Figure 146.

bits: b7-b5	b4-b1	b0
Reserved	Path Order	Transmission Mode

Figure 146 - Relay mode field

The Transmission Mode field indicates whether transmission mode in DRP is Normal or Alternation. The Transmission Mode bit is set to ZERO for the Normal mode and is set to ONE for the Alternation mode, as described in 19.2.2, respectively.

The Path Order field is set to the order of path usage when Alternation mode is set. The Path Order field encoding is shown in Table 98.

Table 98 - Path order field

Value	Path Order
0000	0 - 1
0001	0 - 2
0010	0 - 1 - 2
0011	0 - 2 - 1
0100 - 1111	Reserved

Each number in the path order specifies the Path's ID. 0 means direct path between source device and destination device. Successive numbers such 1, 2 are named as the relay paths in order of antenna training between relay and two devices. Direct path is used first and relay paths are used consecutively. After one cycle of path order is used, the path order is repeated.

The Path Change Interval field indicates the instant when the path of data transmission between source device and destination device is changed. From the start position of one reserved consecutive MASs, every instant of Path Change Interval can have an opportunity to change the path, Within one Path Change Interval, only one path is used for data transmission. The unit of this field is MAS.

The Detour Start Duration field indicates when data transmission via detour path starts in the current DRP. It is set only for Relay Switch Request/Response Command.

The Relay DevAddr field is set to the DevAddr of the device which is requested from a source device for relaying. This is used only for Relay Complete Request command frame in order to let destination device inform the address of a relay device.

Source DevAddr field is set to the DevAddr of the device which requests relay operation and Destination DevAddr is set to the DevAddr of the recipient device to which the device is sending the relay operation request. Two fields are used for informing to the relay device that these two devices intend to use it.

14.7.23 Scan Countdown IE

A Scan Countdown IE is used in a Master-Slave operation, to announce that the Master device is preparing to perform channel scan, as specified in 15.19.11.1.

The Scan Countdown IE is specified in Figure 147.

octets: 1	1	1
Element ID	Length (=1)	Scan countdown

Figure 147 - MSP interval change IE format

The Scan countdown field is set to the number of MSP remaining until the Master device performs channel scanning as specified in 15.19.11.1.

14.7.24 Scan Timing IE

The Scan Timing IE announces the start and duration of a period of time when a device listens for specific frame transmissions. The format of Scan Timing IE is specified in Figure 148.

octets: 1	1	5
ElementID	Length (=5)	Scan Timing

Figure 148 - Discovery scanning IE format

The Scan Timing field is set to the start time and duration of the scanning time period and is specified in Figure 149.

bits: b39-b20	b19-b0
Scan Duration	Scan Start-time

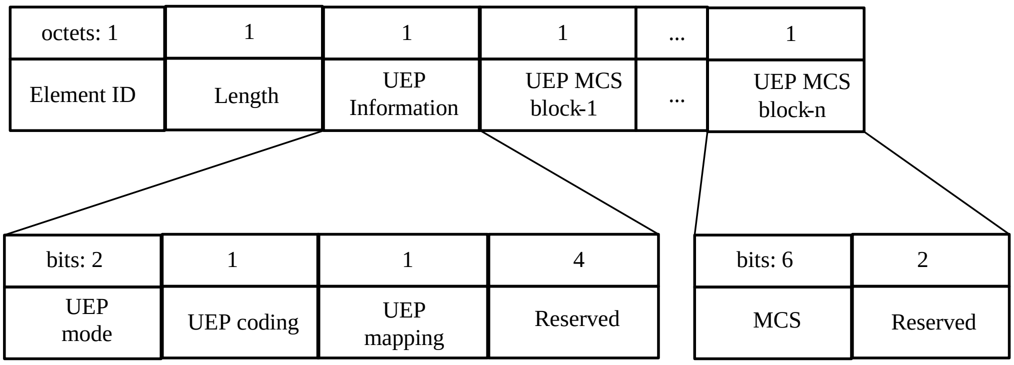
Figure 149 - Scanning timing format

Scan Start-time field is set to the starting time of scanning using the same antenna beam measured from the end of the PLCP header of the beacon frame in units of microseconds.

Scan Duration is set to the duration of the scanning in units of microseconds, using the same antenna beam.

14.7.25 UEP information IE

UEP information IE shall be used to indicate the supported UEP types and UEP MCS modes as specified in UEP information response command.



octets: 1	1	1	1	...	1
Element ID	Length	UEP Information	UEP MCS block-1	...	UEP MCS block-n

bits: 2	1	1	4
UEP mode	UEP coding	UEP mapping	Reserved

bits: 6	2
MCS	Reserved

Figure 150 - UEP information response command

UEP mode definition is specified in Table 99.

Table 99 - UEP mode

UEP Mode	Description
0b00	EEP
0b01	S-UEP
0b10	P-UEP
0b11	Reserved

UEP coding bit shall set to "1" if UEP coding is supported.

UEP mapping bit shall set to "1" if UEP mapping is supported.

Each UEP MCS block is used to indicate a supported UEP MCS mode.

14.8 Master-slave operation command

The Master-Slave Operation Command frame is used by Type C to send Type C device commands.

MAC Header field settings for Master-Slave Operation Frame are described in Table 100.

Table 100 - MAC header field values for master-slave operation frames

Header field	Value
Protocol	0
Secure	0
ACK Policy	0 (No-ACK) or 1 (Imm-ACK)
Frame Type	5 (Master-Slave Operation Frame)
Frame Subtype / Delivery ID	Value from Table 101
Retry	As defined in 15.1.5
DestAddr	BcstAddr or DevAddr of recipient
SrcAddr	DevAddr of the transmitter
Sequence Control	0
Duration	0
More Frames	0
Access Method	As specified in 14.2.7.3

Table 101 contains a list of Master-Slave Operation commands and its respective Command Type value.

Table 101 - Command type field encoding for master-slave operation command

Value	Command Type	Description
0	Type C Poll	Used by Master device to start and maintain a MSP

Table 101 - Command type field encoding for master-slave operation command (concluded)

Value	Command Type	Description
1	Type C Poll Response	Used by a Slave device to acknowledge a received Type C Poll frame from its associated Master device
2	Association Request command	Used by Slave device to request for association to a Master device
3	Association Response command	Used by Master device to respond to an Association request from a Slave device
4	MSPr Services command	Used by Master device to inform Slave device of the services that are available
5	Disassociation Request command	Used by a device to request a disassociation of the Master-Slave MSP
6	Channel Time Request command	Used to request channel time for transmission during the Data Exchange Period
7	Channel Time Response command	Used to response to a channel time request
8	Channel Time Termination command	Used to terminate an allocated channel time
9	Channel Time Modify Request command	Used to request modification to an allocated channel time
10	Channel Time Modify Response command	Used to respond to a modification request to an allocated channel time
11	Remote Scan Response command	Used by Master device to request Slave device to perform Remote Scan
12	Remote Scan Response command	Used by Slave device to respond to Master device following a Remote Scan Request
13	Channel Release Request command	Used by a Type A or Type B device to preempt a Type C device on the current channel
14-254	Reserved	Reserved
255	Application Specific command	Used by device to send application specific information

The Master-Slave operation frame payload format varies for different frame subtype as described in Table 101. The following subclause describes the Payload format for each of the Master-Slave operation frame subtype listed in Command Type field encoding for Master-Slave Operation command.

14.8.1 Type C poll

The Type C Poll frame shall be used by a Master device during a Master-Slave operation to start and maintain a Master-Slave pair, as described in 15.19.4 and 15.19.8.2. The Type C Poll frame shall also be used by device in non Master-Slave operation for the purpose of device discovery as described in 15.3.

The header fields of a Type C Poll shall be set as shown in Table 102.

Table 102 - Type C poll header fields

Ack Policy	Retry	DestAddr
0 (no ACK)	0	BcstAddr

The Type C Poll payload shall be formatted as specified in Figure 151.

Octets: 1	6	6	L_2	...	L_n
MSPr Control	Device Identifier	MSP Timing IE	Information Element$_2$	...	Information Element$_n$

Figure 151 - Type C poll payload format

MSPr Control contains control information required to maintain a MSPr and it shall be formatted as specified in Figure 152.

bits: b7-b2	b1	b0
Reserved	Status	Associated

Figure 152 - MSPr Control field format

The "Associated" bit indicates the association status of the Master device, as described in 15.19.8.2, and it shall be set to one of the following values:

0: Unassociated

1: Associated

The "Status" bit indicates the status of the Master device, as described in 15.19.8.2, and it shall be set to one of the following value:

0: Polling

1: Discovery

The "Device Identifier" field is set to the EUI-48 (as specified in Annex C.1) of the Master device. The EUI is a sequence of 6 octets, labelled as eui[0] through eui[5]. The first three octets (i.e. eui[0] through eui[2]) contains the manufacturer's OUI, and the last three octets (i.e. eui[3] through eui[5]) contains the manufacturer-selected extension identifier.

Type C Poll payload shall contain at least a MSP Timing Information Element, as described in 14.7.17. In addition, it may contain other Information Elements if necessary. For example, as described in 15.19.7.3.1, if one or more Channel Time Reservations (CTRs) are allocated, it shall contain the allocated CTR's CTR Description information Element (described in 14.7.7).

14.8.2 Type C poll response

The Type C Poll Response frame shall be used by a Slave device to acknowledge a received Type C Poll frame from its associated Master device as described in 15.19.8.3.

The Type C Poll Response payload shall be formatted as specified in Figure 153.

octets: 6
MSP Timing Information Element

Figure 153 - Type C poll response payload format

14.8.3 Association request command

The Association Request command payload shall be formatted as specified in Figure 154.

octets: 6	1	2
Device Indentifier	Slave Utility	ATP

Figure 154 - Association request command payload format

The "Device Identifier" field is set to the EUI-48 (as specified in Annex C.1) of the Master device. The EUI is a sequence of 6 octets, labelled as eui[0] through eui[5]. The first three octets (i.e. eui[0] through eui[2]) contains the manufacturer's OUI, and the last three octets (i.e. eui[3] through eui[5]) contains the manufacturer-selected extension identifier.

The Slave utility field shall be formatted as specified in Figure 155.

bits: b7-b1	b0
Reserved	MSP services inquiry

Figure 155 - Slave utility field format

The Master services inquiry bit shall be set to one if the associating Type C device wishes to receive a MSPr Services command from the Type C Master device after the association process is completed, as described in 15.19.6.2. Otherwise, the Master services inquiry bit shall be set to zero.

The Association Timeout Period (ATP) is the maximum amount of time in milliseconds that the target device will be given to respond to the request, as described in 15.19.6.1.

14.8.4 Association response command

The Association Response command payload shall be formatted as specified in Figure 156.

octets: 1
Reason Code

Figure 156 - Association response command payload format

The valid values of the Reason Code are:

0: Success

1: Association denied

2 - 255: Reserved

14.8.5 MSPr services command

The MSPr Services command is used to provide information about the available application layer capabilities of the Type C Master device. The MSPr Services command payload shall be formatted as specified in Figure 157.

octets: 2	0-127
ASIE Specifier ID	MSP Services

Figure 157 - MSPr services command payload format

The ASIE Specifier ID field is set to a 16-bit value that identifies a company or organization, as listed in Annex C.4. The following MSPr Services information shall be ignored if the ASIE Specifier ID can not be understood by a receiving Type C Slave device.

The MSPr Services field contains information describing the available application layer capabilities. The format of the content within the MSPr Services field is outside the scope of this standard.

14.8.6 Disassociation request command

The Disassociation Request command payload shall be formatted as specified in Figure 158.

octets: 1
Reason Code

Figure 158 - Disassociation request command payload format

The valid values of the Reason Code are:

0: Normal Completion

1: MSPr Termination

2: Channel Release

3 - 255: Reserved

14.8.7 Channel time request command

The Channel Time Request command payload shall be formatted as specified in Figure 159.

octets: 1	2	1	1	2
Reservation ID	Time Unit	Minimum number of TUs	Desired number of TUs	Request Timeout

Figure 159 - Channel time request command payload format

The Reservation ID is a unique ID that is used to reference to a Channel Time reservation within the Data Exchange Period. A Type C Master device shall generate a unique Reservation ID for this field if it is the sender of the Channel Time Request command. If a Type C Slave device is the sender of the Channel Time Request command, the Reservation ID shall be set to zero.

The Time Unit field indicates the unit of time that the requesting device is using. The desired and minimum amount of time requested is a one or multiple blocks of this time unit.

The resolution of Time Unit is $1\mu s$ and therefore has a range of [0 - 65535] μs.

The Minimum number of TUs field indicates the minimum number of Time Units requested in order to support a successful Channel Time request.

The Desired number of TUs field indicates the desired number of Time Units requested. It shall be least equal or more than the Minimum number of TUs.

The Request Timeout field indicates the maximum amount of time in milliseconds that the target device will be given to respond to the request, as described in 15.19.7.3.1.

14.8.8 Channel time response command

The Channel Time Response command payload shall be formatted as specified in Figure 160.

octets: 1	1	1
Reservation ID	Available number of TUs	Reason Code

Figure 160 - Channel time response command payload format

The Reservation ID is a unique ID that is used to reference to a Channel Time reservation within the Data Exchange Period. A Type C Master device shall generate a unique Reservation ID for this field if it is the sender of the Channel Time Response command. If a Type C Slave device is the sender of the Channel Time Response command, the Reservation ID shall be set to Reservation ID in the received Channel Time Request command.

The Available number of TUs field indicates the number of Time Units that has been allocated as a result of a Channel Time Request command.

The Reason Code field indicates the status of the Channel Time request. The codes assignable to this field are:

0: Success

1: Channel time unavailable

2: Requested denied

3: Wait until DEP modification

4 - 255: Reserved

14.8.9 Channel time termination command

The Channel Time Termination command payload shall be formatted as specified in Figure 161.

octets: 1	1
Reservation ID	Reason Code

Figure 161 - Channel time response command payload format

The Reservation ID field indicates the channel time reservation which is to be terminated.

The Reason Code field indicates the reason for the Channel Time termination. The codes assignable to this field are:

0: Normal completion

1: Channel time unavailable

2: Superframe overloading

3: MSP shutting down

4 - 255: Reserved

14.8.10 Channel time modify request

The Channel Time Modify Request command payload shall be formatted as specified in Figure 162.

octets: 1	2	1	1	2
Reservation ID	Time Unit	Minimum number of TUs	Desired number of TUs	Request Timeout

Figure 162 - Channel time modify request command payload format

The Reservation ID field indicates the channel time reservation which is to be modified.

The Time Unit field indicates the unit of time that the requesting device is using. The desired and minimum amount of time requested is a one or multiple blocks of this time unit.

The resolution of Time Unit is $1\mu s$ and therefore has a range of [0 - 65535] μs.

The Minimum number of TUs field indicates the minimum number of Time Units requested in order to support a successful Channel Time Modify Request.

The Desired number of TUs field indicates the desired number of Time Units requested. It shall be least equal or more than the Minimum number of TUs.

The Request Timeout field indicates the maximum amount of time in milliseconds that the target device will be given to respond to the request, as described in 15.19.7.3.3.

14.8.11 Channel time modification response command

The Channel Time Modification Response command payload shall be formatted as specified in Figure 163.

octets: 1	1	1
Reservation ID	Available number of TUs	Reason Code

Figure 163 - Channel time response command payload format

The Reservation ID field indicates the channel time reservation which is to be modified.

The Available number of TUs field indicates the number of Time Units that has been allocated as a result of a Channel Time Modification Request command.

The Reason Code field indicates the status of the Channel Time Modification request. The codes assignable to this field are:

0: Success

1: Channel time unavailable

2: Request denied

3: Wait until DEP modification

4 - 255: Reserved

14.8.12 Remote scan request command

The Remote Scan Request command payload shall be formatted as specified in Figure 164.

octets: 1	1	...	1	2
Num of Channels	Channel 1	...	Channel n	Request Timeout

Figure 164 - Remote scan request command payload format

The Num of Channels field indicates the number of channel to be scanned.

The Channel number fields indicate the channel number of the channel to be scanned. The mapping of the channel number is described in 11.1.2.

The Request Timeout field indicates the maximum amount of time in milliseconds that the target device will be given to respond to the request, as described in 15.19.11.2.

14.8.13 Remote scan response command

The Remote Scan Response command payload shall be formatted as specified in Figure 165.

octets: 1	1	1	7	...	7
Num of Channels	Channel rating list	Num of MSPs=n	MSP description 1	...	MSP description n

Figure 165 - Remote scan response command payload format

The Num of Channels field indicates the number of channels that were scanned.

The Channel rating list contains the channels scanned, ordered from the best (least interference) to worst (most interference). The Channel rating list field shall be formatted as specified in Figure 166.

octets: 1	...	1
Best Channel	...	Worst Channel

Figure 166 - Channel rating list format

The Num of MSPs indicates the total number of MSPs detected across all the channels the were scanned.

The MSP description field contains the information of each MSP that were detected across all the channels scanned. Each MSP description field shall be formatted as specified in Figure 167.

octets: 1	6
Channel	Master device ID

Figure 167 - MSP description field format

The Channel field indicates the channel in which the detected MSP operates within.

The Master device ID contains the EUI-48 identifier for the Type C Master device of the detected MSP.

14.8.14 Channel release request command

The Channel Release Request Command shall be used by a Type A or Type B device to preempt a Type C device on the current channel, as described in 15.19.13.2.

The Channel Release Request Command shall always be sent with the following fields set to the specified values:

- ACK Policy: 0 (no ACK).

- DestAddr: BcstAddr.

The Channel Release Request Command shall not contain any payload.

14.8.15 Application specific command

The Application Specific frame payload shall be formatted as specified in Figure 168.

octets: 2	L
ASIE Specifier ID	Application Specific data

Figure 168 - Application specific frame payload format

The ASIE Specifier ID field is set to a 16-bit value that identifies a company or organization, as listed in Annex C.4. The following Application Specific data shall be ignore if the ASIE Specifier ID can not be understood by a receiving Type C Slave device.

The Application Specific data field is defined by the application identified in the ASIE Specifier ID field. The format and its uses are beyond the scope of this standard.

15 MAC sublayer functional description

This Clause specifies MAC sublayer functionality. The rules for transmission and reception of MAC frames, including setting and processing MAC header fields and information elements, are specified in 15.1.

Channel time is divided into superframes, with each superframe composed of two major parts, the beacon period (BP) and the data period. Beacon transmission and reception in the BP and merging of BPs are specified in 15.5.5.

During the data period devices send and receive data using reservations established using the distributed reservation protocol (DRP). The DRP enables a device to gain scheduled access to the medium within a negotiated reservation, and is specified in 15.6.

Device synchronization is specified in 15.8. The fragmentation and reassembly of MSDUs is specified in 15.9. Aggregation of multiple MSDUs in a single frame is specified in 15.10. Acknowledgement mechanisms are specified in 15.12. Clauses 15.13 through 15.17 specify probe commands, channel selection, multi-rate support, transmit power control, power management mechanisms and use of ASIEs. Clause 15.21 specifies values for all MAC sublayer parameters.

15.1 Frame processing

This Clause provides rules on preparing MAC frames for transmission and processing them on reception. The rules cover MAC header fields and information elements.

15.1.1 Frame addresses

Frames are addressed using DevAddrs. There are four types of DevAddrs: Private, Generated, Multicast, and Broadcast. Table 103 shows the range for each type of DevAddr.

Table 103 - DevAddr types and ranges

Type	Range
Private	0x0000 - 0x00FF
Generated	0x0100 - 0xFEFF
Multicast (McstAddr)	0xFF00 - 0xFFFE
Broadcast (BcstAddr)	0xFFFF

A device shall associate a DevAddr of either type Private or type Generated with its local MAC sublayer. A device that uses a NULL EUI-48 shall use a Private DevAddr. If a device uses a Generated DevAddr, it shall select the DevAddr from the Generated DevAddr range at random with equal probability and should ensure that the generated value is unique among all devices in its extended beacon group. Private DevAddres are reserved for devices outside the scope of this specification. Selection and conflict resolution for Private DevAddrs is out of scope of this Standard.

In all frames transmitted, a device shall set the SrcAddr field to its own DevAddr. In unicast frames, the DestAddr field shall be set to the DevAddr of the recipient. In multicast frames, the DestAddr field shall be set to an address from the Multicast DevAddr range, as specified in 15.1.10.18. In broadcast frames, the DestAddr field shall be set to the Broadcast DevAddr.

A device shall not transmit frames addressed to a recipient with a Private DevAddr at any time outside a Private reservation. A device with a Private DevAddr shall not transmit non-beacon frames outside a Private reservation.

15.1.1.1 DevAddr conflicts

A device with a Generated DevAddr shall recognize that its DevAddr is in conflict if any of the following conditions occurs:

- It receives a MAC header in which the SrcAddr is the same as its own DevAddr; or

- It receives a beacon frame in which the BPOIE contains a DevAddr that is the same as its own but corresponds to a beacon slot in which the device did not transmit a beacon and was not in hibernation mode.

A device that recognizes that its DevAddr is in conflict shall generate a new DevAddr to resolve the DevAddr conflict.

15.1.2 Frame reception

Unless otherwise indicated, a frame is considered to be received by the device if it has a valid header check sequence (HCS) and segment check sequence (FCS) as defined in 14.2.9 and indicates a protocol version that is supported by the device. The HCS is validated by the PHY, which indicates whether or not a header error occurred.

A MAC header is considered to be received by the device if it has a valid HCS and indicates a protocol version supported by the device, regardless of the FCS validation.

15.1.3 Antenna training frame transaction

An antenna training frame transaction in the discovery channel consists of an RTT/CTT frame exchange, a sequence of training frames, and an acknowledgement frame if requested by the ACK policy.

Figure 169 shows a frame transaction example where both the source and recipient device have a configuration of 4 iterations in setting the Duration field for the RTT/CTT frames. This offers protection until the end of the entire antenna training procedure in case there are no frame transmission errors. In case additional iterations are required, the source and recipient devices shall set the Duration field to cover and protect these additional iterations. Other devices receiving these frames shall update their NAV as described in 15.2.2.

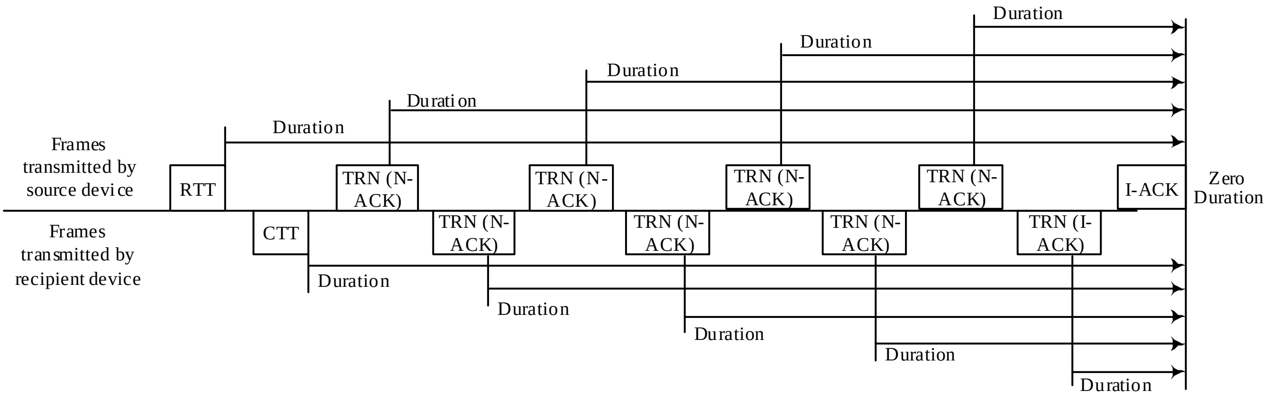


Figure 169 - Frame transaction of an antenna training example with 4 iterations

15.1.4 Frame transfer

A source device shall transmit MSDUs associated with the same Delivery ID and addressed to the same destination EUI-48 in the order in which they arrived at the local MAC SAP. The device shall treat each MSDU of length n as a sequence of octets, labelled MSDU[0] to MSDU[n-1], and shall place these octets in the payload field in ascending index-value order. The device shall transmit fragments of an MSDU or MCDU in order of increasing fragment number.

When using the B-ACK mechanism, a source device may retransmit some previously transmitted frames, causing the sequence numbers and fragment numbers of the retransmitted frames to be out of order with respect to previously transmitted frames.

A source device may reorder MSDUs for transmission if their associated Delivery IDs or destination EUI-48s are different.

A recipient device shall release MSDUs to the MAC client that were transmitted by the same source device with the same Delivery ID in order of increasing sequence number values.

A source device may fragment or aggregate MSDUs for transfer between peer MAC entities, but the recipient device shall deliver whole individual MSDUs through the MAC SAP to the MAC client.

15.1.5 Frame retry

A frame retry is a retransmission of a previously transmitted frame from the same source device to the same recipient device. In a frame that is retransmitted, the source device shall set the Retry bit to ONE.

Unless otherwise stated, in this specification "transmission" means transmission of a new frame or retransmission of a previously transmitted frame.

A device may retransmit a frame as needed, taking into consideration such factors as delay requirements, fairness policies, channel conditions, and medium availability. A device shall apply the medium access rules for new frame transmissions when retransmitting frames, unless stated otherwise.

15.1.6 Inter-frame space (IFS)

Four types of IFS are used in this Standard: the beacon inter-frame space (BIFS), the long inter-frame space (LIFS), the minimum inter-frame space (MIFS), the short inter-frame space (SIFS). The actual values of the MIFS, SIFS are PHY-dependent.

A device shall not start transmission of a frame on the medium with non-zero length payload earlier than MIFS, or with zero length payload earlier than SIFS, after the end of a frame it transmitted previously on the medium. A device shall not start transmission of a frame on the medium earlier than SIFS duration after the end of a previously received frame on the medium.

15.1.6.1 MIFS

The length of MIFS is given by the pMIFS parameter defined in Table 49.

15.1.6.2 SIFS

Within a frame transaction, all frames shall be separated by a SIFS interval. The length of SIFS is given by the pSIFS parameter defined in Table 49.

15.1.6.3 BIFS

When transmitting beacon frames in discovery channel, the discovery beacon and poll frames transmitted using one antenna block or beam shall be separated by a BIFS interval.

15.1.6.4 LIFS

When transmitting beacon frames in discovery channel, LIFS is the minimum time that a device defer access to the medium after it determines the medium to have become idle.

15.1.7 Duplicate detection

Because a device may not receive an Imm-ACK or B-ACK response for a frame it transmitted, it may send duplicate frames even though the intended recipient has already received and acknowledged the frame. A recipient device shall consider a received frame to be a duplicate if the Retry bit is set and the Sequence Control field has the same value as the previous frame received with the same SrcAddr, DestAddr, and Delivery ID field values. A recipient device shall not release a duplicate frame to the MAC client.

15.1.8 RTT/CTT use

An RTT/CTT exchange is used to initiate antenna training between a source and a destination device, and shall be used only on the discovery channel. With an appropriately set Duration field as specified in 15.1.9.1, the RTT and CTT frames prevent the neighbours of the source and recipient devices from accessing the medium while the source and recipient are exchanging the following frames.

If a device receives an RTT frame addressed to it in the discovery channel, it shall transmit a CTT frame SIFS after the end of the received frame if and only if its NAV is zero.

On receiving an expected CTT response, the source device shall transmit the antenna training frames for which it transmitted the preceding RTT frame SIFS after the end of the received CTT frame. If the source device does not receive the expected CTT frame SIFS plus the CTT frame transmission time after the end of the RTT frame transmission, and it transmitted the RTT in the discovery channel, it shall invoke a backoff as specified in 15.2.6.

15.1.9 MAC header fields

15.1.9.1 Duration

A device shall set the Duration field in beacon frames to one of the following:

— The time remaining in the BP measured from the end of the PLCP header of the beacon frame, as determined by the largest BP length announced by neighbours of the device in the previous superframe;

— The transmission time of the frame body of the beacon frame;

A device shall set the Duration field in RTT, command, data, or aggregated data frames to the sum of:

— The transmission time of the frame body of the current frame;

— The transmission time of the expected response frame for the current frame (CTT, Imm-ACK, or B-ACK frame), if any;

— The transmission time of subsequent frames, if any, to be sent to the same recipient up to and including (a) the next RTT frame or frame with ACK Policy set to Imm-ACK or B-ACK Request or (b) the last frame in the DCA TXOP or reservation block, whichever is earlier; or, alternatively, the transmission time of the next frame in the DCA TXOP or reservation block to be sent to the same recipient, if any; and

— All the IFSs separating the frames included in the Duration calculation.

A device shall round a fractional calculated value for Duration in microseconds up to the next integer.

A device may estimate the transmission time of a B-ACK frame body based on the expected length and data rate, or may assume a zero-length frame body.

A device shall set the Duration field in CTT, Imm-ACK and B-ACK frames to the larger of zero or a value equal to the duration value contained in the previous frame minus SIFS, minus the transmission time of the frame body of the received frame to which the CTT, Imm-ACK or B-ACK is responding, minus the transmission time up to the end of the PLCP header of this CTT, Imm-ACK or B-ACK frame.

The following exceptions to previous rules are allowed:

— For frames with ACK Policy set to B-ACK Request, a device may set the Duration to the sum of the transmission time of the frame body of the B-ACK Request frame plus a SIFS plus the estimated transmission time of the expected B-ACK response frame.

— A device may set the Duration for any frame sent in a Hard or Private reservation block to zero.

An example of Duration field values is specified in Figure 170.

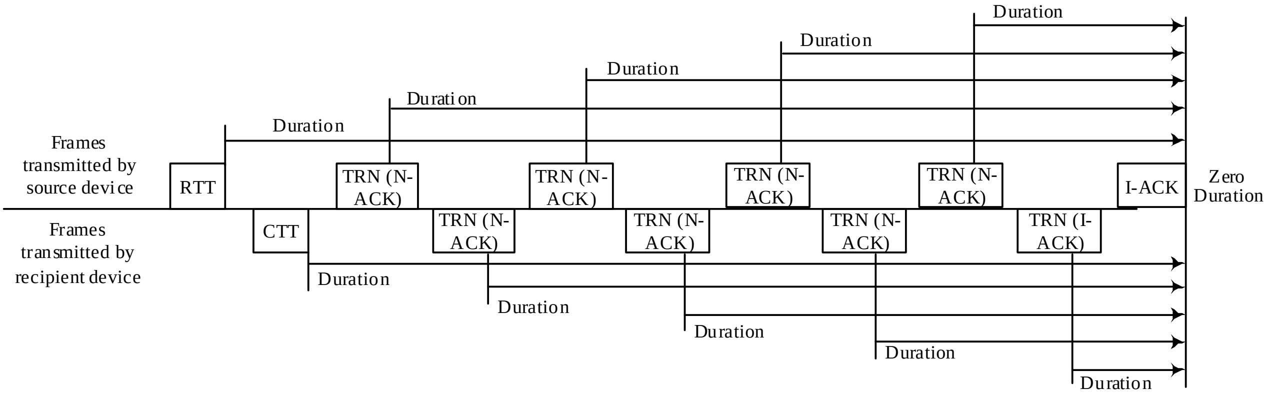


Figure 170 - Duration field values of an example of antenna training with 4 iterations

15.1.9.2 More frames

If a device sets the More Frames bit to ZERO in a frame sent with Access Method set to ONE, it shall not transmit additional frames to the same recipient(s) within the reservation block.

15.1.9.3 Sequence number

The Sequence Number field value is used for duplicate detection for frames sent using the Imm-ACK acknowledgement policy. It is used for both duplicate detection and reordering for frames sent using the B-ACK mechanism.

A device shall assign each MSDU or MCDU transmitted a sequence number from a modulo 2048 counter.

A device shall assign the same sequence number to each fragment of an MSDU or MCDU.

A device shall use a dedicated counter for MCDUs.

A device shall use a dedicated counter for each sequence of MSDUs addressed to the same DestAddr with the same Delivery ID using B-ACK acknowledgement policy.

A device may use one counter for all other MSDUs, or may use a dedicated counter for MSDUs with the same Delivery ID field value addressed to the same DestAddr.

In each beacon frame transmitted in a superframe, a device shall set the Sequence Number field from a dedicated counter that increments once per superframe, modulo 2048, or shall set it to zero.

15.1.10 Information elements

IEs are contained in beacon and command frames. They convey certain control and management information. IEs may be explicitly requested using Probe command frames.

A device shall include IEs in its beacon frame such that they apply to the superframe in which the beacon is transmitted. A device shall interpret IEs contained in beacons received in the current superframe to apply to that superframe.

The remainder of this Clause describes when each IE is generated.

15.1.10.1 Application-specific IE (ASIE)

A device may include an ASIE in its beacon for each of its applications which have made the request, as described in 15.17. The scope of the ASIE is dependent on the application that requested the inclusion of the ASIE.

15.1.10.2 Application-specific probe IE

A device may send an Application-specific Probe IE in order to request a specific ASIE. The scope and required response is dependent on the application that defines the ASIE.

15.1.10.3 ATTP IE

A device shall employ the following four IEs to support antenna training and tracking according to sub clause 15.18.

- Antenna Capability IE (ACIE)
- Antenna Training IE (ATIE)
- Antenna Training Frame IE (ATFIE)
- Antenna Feedback IE (AFIE)

15.1.10.4 Beacon period occupancy IE (BPOIE)

A device shall always include a BPOIE in its beacon. In the BPOIE the device shall reflect beacons received from neighbours in the previous superframe, as well as information retained based on hibernation mode rules.

15.1.10.5 Channel bonding IE (CBOIE)

A CBOIE is used to negotiate bonding of two or more channels and to announce the channels that are involved in the channel bonding.

When using channel bonding, a device shall include a CBOIE in its beacons transmitted in a designated beaconing channel to indicate the (proposed) bonded channels as defined in 15.11.

15.1.10.6 Channel change IE

A device should include a Channel Change IE in its beacon prior to changing to a different channel. A device that includes a Channel Change IE should change channels as indicated in the IE. A Type A device shall include a Channel Change IE in its discovery beacon to announce its home channel to potential newly discovered device when the device returns to Discovery Channel to discover additional device.

15.1.10.7 CTR description IE

The CTR description IE, specified in 14.7.7, is used in Master-Slave operation to describe an allocated CTR. For every allocated CTR, its corresponding CTR description IE shall be included in every Type C Poll frame as described in 15.19.7.3.1.

15.1.10.8 Distributed reservation protocol (DRP) IE

A device shall include DRP IEs in its beacon for all reservations in which it participates as a reservation owner or target, as described in 15.6.

15.1.10.9 DRP availability IE

A device shall include a DRP Availability IE in its beacon as required to support DRP reservation negotiation, as described in 15.6.

15.1.10.10 Hibernation anchor IE

A device that indicates it is capable of acting as a hibernation anchor should include a Hibernation Anchor IE in its beacon to provide information on neighbours that are currently in hibernation mode as described in 15.16.5.

15.1.10.11 Hibernation mode IE

A device shall include a Hibernation Mode IE in its beacon before entering hibernation mode, as specified in 15.16.3. A device that receives a Hibernation Mode IE shall report the beacon slot of the transmitter as occupied and non-movable in the BPOIE included in its beacons during the reported hibernation duration.

15.1.10.12 Identification IE

A device may include an Identification IE in its beacon to provide its own identifying information to neighbours.

15.1.10.13 Link feedback IE

A device may include a Link Feedback IE in its beacon to provide feedback on a link with a specific neighbour.

15.1.10.14 MAC capabilities IE

A device may include a MAC Capabilities IE in its beacon.

15.1.10.15 Master key identifier (MKID) IE

A device may include a MKID IE in its beacon to identify some or all of the master keys it possesses.

15.1.10.16 MSP interval change IE

The MSP interval change IE, specified in 14.7.16, is used in Master-Slave operation to announce that the master device is preparing to change its MSP Interval. The MSP interval change IE shall be included in Type C Poll frame during MSP Interval changing procedure as described in 15.19.12.1.

15.1.10.17 MSP timing IE

The MSP timing IE, specified in 14.7.17, is used in Master-Slave operation to convey the timing information of the current MSP. The MSP timing IE shall be included in every Type C Poll frame and Type C Poll Response frame as described in 15.19.8.2 and 15.19.8.3 respectively.

15.1.10.18 Multicast address bind (MAB) IE

A device may include a MAB IE for any active multicast bindings between multicast EUI-48s and McstAddrs. A device should include a MAB IE in its beacon for at least mMaxLostBeacons+1 superframes on registering a multicast address binding for transmission and upon detection of a change in the beacon group.

The MAC entity shall translate the multicast EUI-48 provided by the MAC client along with an MSDU to the bound multicast DevAddr for use in the transmission of the MSDU over the medium.

A device shall not transmit frames with a McstAddr destination address unless a binding to a multicast EUI-48 has been declared by inclusion of a corresponding MAB IE in its beacon.

On receipt of a MAB IE the MAC sublayer shall establish an association between the source of the MAB IE and the multicast DevAddr and multicast EUI-48 in each Multicast Address Binding Block, to be used in address translations for the bound multicast addresses.

The MAC entity shall deliver received MSDUs addressed to an activated multicast DevAddr to the MAC client on the multicast EUI-48 bound to that multicast DevAddr by the source device of the MSDU.

15.1.10.19 PHY capabilities IE

A device may include a PHY Capabilities IE in its beacon.

15.1.10.20 Probe IE

A device may include a Probe IE in its beacon to request certain IEs from another device.

A device may include a Relinquish Request IE in its beacon to request that a neighbour release one or more MASs from reservations.

If a reservation target receives a request to relinquish MASs included in the reservation, it shall include in its beacon a DRP Availability IE and a Relinquish Request IE identifying those MASs with the Target DevAddr field set to the DevAddr of the reservation owner.

15.1.10.21 Relay IE

If a device supports the relay procedure as specified in 19.1.2, 19.1.4, and 19.1.5; it shall include a Relay IE in its beacon or command frame to support the relay procedures.

If a device supports the relay procedure as specified in 19.2.7.1, it shall include a Relay IE in its beacon or command frame to detour to another reliable path when a current path is considered unavailable as described in 19.2.6.

15.1.10.22 Scan Countdown IE

The Scan Countdown IE, specified in 14.7.23, is used in Master-Slave operation to announce that the master device is preparing to perform aperiodic channel scanning. The Scan Countdown IE shall be included in Type C Poll frame during the aperiodic channel scanning procedure as described in 15.19.11.1.

15.1.10.23 Scan Timing IE

A device shall always include a Scan Timing IE in a discovery beacon frame, i.e. beacon with status field set to discovery. In the Scan Timing IE, the device shall indicate the time and duration the device will listen for response to the beacon using the same antenna beam. A device shall include Scan Timing IE in a Channel Scanning Request command frame to indicate the time when the device returns to Discovery Channel to listen for Channel Scanning Response command frame.

15.1.10.24 UEP IE

A device may include an UEP IE in its beacon to indicate the UEP types and UEP MCS modes that it can support.

15.2 Distributed contention access (DCA)

The DCA mechanism provides distributed contention access to the medium for the following types of frame exchanges in the discovery channel:

- Discovery beacons and poll frames sent by Type A, B devices
- RTT/CTT control frames
- Antenna training frames
- Channel selection command frames

A frame transferred over the wireless medium using DCA is referred to as a DCA frame.

Both Type A and Type B devices shall use both physical and virtual carrier sensing before accessing the medium, while Type C devices shall employ physical carrier sense only. The RTT and CTT frames shall be transmitted using PHY mode D0 (see 10.2.5). The training frames shall be transmitted using one of the discovery PHY modes with adaptive spreading factor.

15.2.1 DCA medium availability

A device shall consider the medium available for DCA at the following times:

- All the times in the discovery channel.
- Soft reservation blocks with Reservation Status set to ONE if the device is the reservation owner or a neighbour of the reservation owner.

At all other times, a device shall consider the medium unavailable for DCA.

15.2.2 NAV

A Type A or a Type B device that transmits and receives frames using DCA shall maintain a network allocation vector (NAV) that contains the remaining time that a neighbour device has indicated it will access the medium. The device that receives a MAC header not addressed to it shall update its NAV with the received Duration field if the new NAV value is greater than the current NAV value. A device shall consider the updated NAV value to start at the end of the PLCP header on the medium.

A device that receives a MAC header with invalid HCS outside shall update its NAV as if the frame were correctly received with Duration equal to mAccessDelay.

A device shall reduce its NAV as time elapses until it reaches zero. The NAV shall be maintained to at least mClockResolution.

15.2.3 Medium status

For DCA purposes, a device shall consider the medium to be busy for any of the following conditions:

- When its CCA mechanism indicates that the medium is busy;
- When the device is transmitting or receiving a frame on the medium;
- When the Duration announced in a previously transmitted frame has not yet expired;
- When the device’s NAV is greater than zero, if the device maintains a NAV.
- When the medium is unavailable for DCA.

At all other times a device shall consider the medium to be idle.

15.2.4 Obtaining a TXOP

A device shall consider itself to have obtained a TXOP if it meets the following conditions:

- The device has one or more newly generated DCA frames;
- The device had a backoff counter of zero value and had no DCA frames prior to the generation of the new DCA frames;
- The device determines that the medium has been idle for LIFS or longer;

The device shall start transmitting a frame, which may be an RTT frame, as soon as the above conditions are satisfied. The device shall treat the start of the frame transmission on the wireless medium as the start of the TXOP.

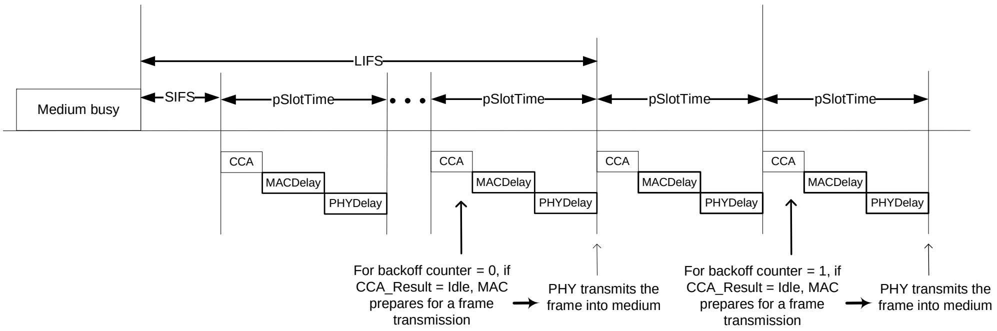
A device shall also consider itself to have obtained a TXOP if it meets the following conditions:

- The device has one or more DCA frames buffered for transmission, including retry;
- The device set the backoff counter to zero in the last backoff and determines that the medium has been idle for LIFS since that backoff at the end of the current DCA slot, or the device decrements its backoff counter from one to zero in the current DCA slot;

The TXOP shall start at the end of the current backoff slot, i.e., the start of the next backoff slot.

A device shall ensure that the TXOP it has obtained ends SIFS plus mGuardTime before the medium becomes unavailable for DCA.

Figure 171 illustrates the timing relationship in obtaining a TXOP.



CCA = pCCADetectTime
MACDelay = Time required by MAC for preparing a frame transmission and transfering the frame to PHY SAP
PHYDelay = Time required of PHY to transfer the frame from PHY-SAP to wireless medium

Figure 171 - DCA timing relationships

15.2.5 Using a TXOP

A device that has obtained a TXOP is referred to as a TXOP owner. A TXOP owner shall initiate one or more frame transactions without backoff, in the TXOP it has obtained, subject to the following criteria:

- Each transaction in the TXOP will be completed within the obtained TXOP; and
- The recipient device will be available to receive and respond during that frame transmission.

A recipient device shall not transmit a CTT frame in response to a received RTT frame if its NAV is greater than zero. A recipient device shall not transmit a CTT, Imm-ACK or B-ACK response to a received frame requiring such a response if the response will not be completed SIFS before the medium becomes unavailable for its DCA.

Under the rules stated above, the following timings apply to transmissions, including responses, in a TXOP (these timings are referenced with respect to transmission to or reception from the wireless medium):

- The TXOP owner shall transmit the first frame of the first or sole frame transaction in the TXOP at the start of the TXOP.
- After transmitting a discovery beacon frame or a Poll frame, the TXOP owner shall transmit a subsequent frame BIFS after the end of that transmitted frame.
- After transmitting a frame with the ACK Policy set to No-ACK or B-ACK, the TXOP owner shall transmit a subsequent frame pMIFS or pSIFS after the end of that transmitted frame.
- After receiving an RTT frame or a non-RTT frame with the ACK Policy set to Imm-ACK or B-ACK Request, the recipient device shall transmit a CTT frame or an Imm-ACK or B-ACK frame SIFS after the end of the received frame.
- After receiving an expected CTT, Imm-ACK or B-ACK response to the preceding frame it transmitted, the TXOP owner shall transmit the next frame, or retransmit a frame it transmitted earlier in the case of receiving a B-ACK, SIFS after the end of the received frame.
- After receiving a requested B-ACK frame with a valid HCS but an invalid FCS, the TXOP owner shall retransmit the last frame it transmitted, or transmit the next frame, SIFS after the end of the B-ACK frame.

If a device cannot transmit its next frame according to these timing requirements, it shall consider the TXOP ended.

15.2.6 Invoking a backoff procedure

A device shall maintain a backoff counter to transmit frames using DCA.

A device shall set the backoff counter to an integer sampled from a random variable uniformly distributed over the range [0, CW], inclusive, when it invokes a backoff. The device shall initialize CW to mCWmin before invoking any backoff, adjusting CW in the range [mCWmin, mCWmax], inclusive, in the course of performing DCA as described below.

The device shall set CW back to mCWmin after receiving a CTT or Imm-ACK frame or the MAC header of a B-ACK frame expected in response to the last transmitted frame, or upon transmitting a frame with ACK Policy set to No-ACK. A device shall also set CW back to mCWMin, but shall not select a new backoff counter value, after discarding a buffered DCA frame.

A device shall invoke a backoff procedure and draw a new backoff counter value as specified below.

1. A device shall invoke a backoff, with CW set to mCWmin, when it has a DCA frame arriving at its MAC SAP, or a DCA frame generated at the MAC sublayer entity under the following conditions:

   - The device had a backoff counter of zero value but is not in the middle of a DCA frame transaction; and the device determines that the medium is busy.

2. A device shall invoke a backoff, with CW set to mCWmin, at the end of transmitting a DCA frame with the ACK policy set to No-ACK or B-ACK, or at the end of receiving an expected Imm-ACK or B-ACK response to its last transmitted DCA frame, under the following condition:

   - The device has no other DCA frames for transmission in the current TXOP obtained.

3. A device shall invoke a backoff, with CW set to mCWmin, at the end of transmitting a DCA frame with the ACK policy set to No-ACK or B-ACK, or at the end of currently receiving the DCA frame header of an expected Imm-ACK or B-ACK response frame to its last transmitted DCA frame, under the following conditions:

   - The device has one or more DCA frames that need to be transferred over the wireless medium; and

   - The device finds that there is not enough time remaining in the current TXOP obtained to complete the next DCA frame transaction.

4. A device shall invoke a backoff, with CW (but not the backoff counter in general) kept to the same value, at the start of a TXOP obtained under the following condition:

   - The device finds that there is not enough time to complete a pending DCA frame transaction in the obtained TXOP.

5. A device shall invoke a backoff, with CW set to the smaller of mCWmax or 2×CW+1 (the latter CW being the last CW value), at SIFS plus the Imm-ACK frame transmission time after the end of the last DCA frame it transmitted, under the following condition:

   — The device does not receive an expected CTT or Imm-ACK frame, or does not correctly receive the MAC header of a requested B-ACK frame by this time.

15.2.7 Decrementing a backoff counter

Upon invoking a backoff, a device shall ensure that the medium is idle for LIFS before starting to decrement the backoff counter. To this end, a device shall define the first DCA slot to start at the time SIFS after the medium has been idle, as defined in Figure 172, with subsequent DCA slots following successively until the medium becomes busy. All DCA slots have a length of pSlotTime.

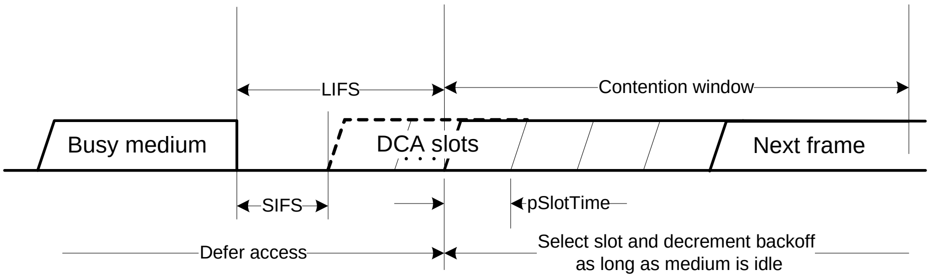


Figure 172 - Transmission of beacons in the discovery channel

A device shall treat the CCA result at pCCADetectTime after the start of a DCA slot to be the CCA result for that DCA slot. If the medium is idle in a DCA slot, and the medium has been idle for at least LIFS, the device shall decrement the backoff counter by one at that time. This procedure is also defined in Figure 172.

The device shall freeze the backoff counter when the medium becomes busy. The device shall treat the residual backoff counter value as if the value were set due to the invocation of a backoff, following the above procedure to resume decrementing the backoff counter.

15.3 Device discovery

After powering up, a device shall first discover another device with which it intends to exchange MPDUs following the process specified in this clause according to its device type.

15.3.1 Type A

A Type A device shall scan for mode-A0, mode-B0 beacons with Status set to Ready and mode-C0 poll frames with Status set to Polling in the discovery channel for at least one superframe after powering up. If such mode-A0 or mode-B0 beacons are received, the Type A device shall send the beacon in the same mode with Status set to Preemptive in a signalling slot that is randomly chosen in the BP indicated by the received beacon. In addition, the Type A device may also send the same preempting beacons in any slots in the BP indicated by the received beacon. If such mode-C0 Poll frames are received, the Type A device shall send a channel release request according to 15.19.13.2. If none of aforementioned frames are received during the scan, the Type A device shall start to send discovery beacon blocks that consist of mode-D0 discovery beacons, mode-B0 and mode-C0 Poll frames in the discovery channel as described in 15.5.1.1. In order to discover other devices after starting the transmission in a Data Channel, a Type A device shall switch back to the discovery channel and send discovery beacon blocks as described in 15.5.2.1. After sending a set of discovery beacon blocks, the Type A device shall scan for mode-D0 beacons as specified in 15.5.1.3. If a mode-D0 discovery beacon is received correctly in the discovery channel, the Type A device may start the antenna training with the device from which the mode-D0 beacon is received as described in 15.18.

15.3.2 Type B

A Type B device shall scan each channel for at least one superframe after powering up. After the scanning, the Type B device shall not transmit any frames unless the scanning indicates that one of the following conditions is true:

- If a channel is detected as busy and at least one mode-B0 beacon with Status set to Ready is received correctly, the Type B device may join the device from which the mode-B0 beacon is received by sending mode-A0 and mode-B0 beacons as described in 15.5.3.
- If a channel is detected as busy and at least one mode-B0 Poll frame is received correctly, the Type B device may join the device from which the mode-B0 Poll frame is received by transmitting a mode-B0 Poll frame with frame Status set to Response during the time period as indicated in the Scan Timing IE in the received mode-B0 Poll frame.
- If it has detected an idle channel during the scanning and has not joined another device by responding to a mode-B0 beacon or mode-B0 Poll frame, the Type B device shall start the discovery process specified below:

  A Type B device shall scan for mode-B0 beacons with Status set to Ready and mode-C0 poll frames with Status set to Polling in the Discovery channel for at least one superframe. If such mode-B0 beacons are received, the Type B device shall send a beacon in mode-B0 with Status set to Preemptive in a signalling slot that is randomly chosen in the BP indicated by the received beacon. In addition, the Type B device may also send the same preempting beacons in any slots in the BP indicated by the received beacon. If such mode-C0 Poll frames are received, the Type B device shall send a channel release request according to 15.19.13.2. If none of aforementioned frames are received during the scan, the Type B device shall start to send discovery beacon blocks that consist of mode-B0 discovery beacons and mode-C0 Poll frames in the discovery channel as described in 15.5.1.2. After sending a set of discovery beacon blocks, the Type B device shall scan for mode-B0 beacon or Poll frames as specified in 15.5.1.3. If a mode-B0 beacon or Poll frame is received correctly in the discovery channel, the Type B device may start an explicit channel selection process with the device from which the mode-B0 beacon or Poll frame is received, as described in 15.4.1.

15.3.3 Type C

A Type C device shall scan each channel for at least one superframe, after powering up. If it does not receive any mode-C0 Poll frame during the scan, the device shall scan the Discovery Channel for at least mDBPMax.

After the scanning, a Type C Slave device shall not transmit any frames unless the scanning indicates that one of the following conditions is true:

- If a channel is detected as busy and at least one mode-C0 Poll frame with Status set to Polling is received correctly, the Type C Slave device may associate with the device from which the mode-C0 Poll frame is received as described in 15.19.6.1.
- If a mode-C0 poll frame with Status set to Discovery is received correctly in the discovery channel, the Type C device may follow the explicit channel selection process with the Master device from which the mode-C0 Poll is received, as specified in 15.4.1.

After the scanning, a Type C Master device shall not transmit any frames unless it has detected an idle channel during the scanning. The device may start transmitting Type C Poll frame with Status set to Polling in the idle channel as specified in 15.5.4, if it is not associated with another device by accepting an association from another device.

15.4 Channel selection

A Type A or B device shall use the explicit channel selection process specified in 15.4.1 to select a channel to send its mode-A0 or mode-B0 data beacons, B-Poll or Type C Poll frames, before it sets up or joins a beacon group in that channel.

A device shall send channel selection command frames with a mode that its intended recipient supports. A device shall not send channel selection command frames with any discovery mode.

After Type A or Type B devices started or joined a beacon group by sending mode-A0 or mode-B0 beacons in a selected data channel, an implicit channel selection procedure described in 15.4.2 allows a group of devices in the beacon group to change channels in a coordinated manner.

15.4.1 Explicit channel selection

A device shall not start explicit channel selection unless the device has received a discovery beacon or poll frame in Discovery Channel from another device with which it intends to exchange MPDUs. The device shall transmit channel selection command frames only in Discovery Channel using DCA. In the explicit channel selection process, the device shall first coordinate explicit channel scanning with the newly discovered device as described in 15.4.1.1 followed by explicit channel switch as described in 15.4.1.2.

15.4.1.1 Explicit channel scan

A device shall request newly discovered device to perform channel scanning by sending a Channel Scanning Request. Once the device receives an ACK frame to its Channel Scanning Request, the device shall leave the Discovery Channel to scan the same channel(s) as specified in its Channel Scanning Request. At the time it specified in the Scan Timing IE in its Channel Scanning Request, the device shall return to Discovery Channel to listen for Channel Scanning Response.

Upon reception of such a Channel Scanning Request, a device shall respond with an ACK frame, after which it shall performed the requested scanning in the channels indicated in the Channel Scanning Request. The device shall return back to Discovery Channel at the time indicated in the Channel Scanning Request and send a Channel Scanning Response which includes DRP available IE(s) to indicate the scanning result.

The formats of Channel Scanning Request and Response are specified in 14.5.3.1 and 14.5.3.2 respectively.

15.4.1.2 Explicit channel switch

Upon reception of a Channel Scanning Response, a device shall send Channel Change Request to the device from which it received the Channel Scanning Response. After sending a Channel Change Request, the device shall listen for a Channel Change Response. Once it receives an Channel Change Response with Reason Code field set to Accepted, It shall wait for a period time that is randomly chosen over the range [0, mSuperframeLength] before switching to the accepted channel to set up a superframe structure as specified in 15.5.2 and 15.5.3 with the exception that the pair of devices will engage in a Master-Slave operation, as specified in 15.7.2, in the agreed channel. If the device receives a frame with Reason Code field set to a value other than Accepted, the device shall transmit a revised channel change request. The device shall not switch to a channel that is not accepted by the recipient.

Upon reception of a Channel Change Request, a device shall respond with a Channel Change Response frame with Reason Code field appropriately set as in Table 72. A device shall not switch to a channel until an Ack frame to its accepted Channel Change Response is received. Before it switches to an agreed channel to set up a superframe structure, the device shall wait for a period of time that is randomly chosen over the range [0, mSuperframeLength] with the exception that the pair of devices will engage in a Master-Slave operation, as specified in 15.7.2, in the agreed channel.

The formats of Channel Change Request and Response are specified in 14.5.3.3 and 14.5.3.4 respectively.

15.4.2 Implicit channel selection

A device may initiate implicit channel selection after it has performed a channel scan. If a device initiates implicit channel selection, it shall include a Channel Change IE in its beacon sent in the current channel, as described in 14.7.6.

In a Channel Change IE, the device shall set the New Channel Number field to the number of the new channel. It shall set the Change Channel Count field to the remaining number of superframes before the device will move to another channel. In successive superframes, the Change Channel Count field should be decremented.

If the value set in the Change Channel Count field is zero, the device shall move to the new channel at the end of the current superframe.

On reception of the Channel Change IE, a device that also intends to change channels in a coordinated manner should include a Channel Change IE with the same field values in its beacon.

15.5 Transmission and reception of beacons and poll frames

Devices transmit beacons in the discovery channel to discover other devices they intend to communicate with or to perform antenna training. Once a device finds another device with which it intends to communicate, it shall set up a superframe structure by transmitting a beacon as specified in 15.5.5.

15.5.1 Transmission of beacons in the discovery channel

A device shall set the status field of the beacon frames to Discovery when transmitting beacons in the discovery channel to discover neighbours. The beacons with Status field set to Discovery are referred to as discovery beacons. To discover devices of different types it supports, a device shall transmit beacons with status set to Discovery in sequence using different PHY modes corresponding to the types of the devices it intends to discover, as specified in the following sub clauses. A beacon frame transmitted by a Type A device using mode-B0 mode with Status set to Discovery is called a mode-B0 Poll frame. A beacon frame transmitted by a Type A/B device using mode-C0 mode with Status set to Discovery is called a mode-C0 Poll frame. DCA shall be used to transmit discovery beacons and mode-B0/mode-C0 Poll frames in discovery channel.

15.5.1.1 Discovery beacon blocks of a Type A device

A Type A device shall transmit mode-D0 discovery beacons only in the discovery channel.

After the completion of a mode-D0 discovery beacon transmission, the Type A device shall transmit one or more Poll frame blocks. The number of Poll frame blocks shall be equal to the number of sectors the device is capable of covering using a number of antenna blocks or beams. A Poll frame block shall consist of transmission of a mode-B0 Poll frame and a mode-C0 Poll frame in the same sector and scanning for mode-C0 and mode-B0 Poll responses in the specified order. The transmission of the mode-D0 discovery beacon and the Poll frames in series shall be separated by BIFS and completed in one TXOP obtained. The scanning shall start SIFS time after the end of mode-C0 Poll frame transmission. The Type A device shall first scan for responses to the transmitted mode-C0 Poll frames for a duration of C-SCAN, after which the Type A device shall scan for responses to the previously transmitted mode-B0 poll frame for a duration of B-SCAN. After scanning for the mode-B0 response, the Type A device shall scan for mode-D0 response for a duration of D-SCAN. If the Type A device has multiple antennas that cover multiple sectors, after the completion of the scanning in the first sector the Type A device shall switch to the next sector to transmit the next Poll frame block in which the transmission of the mode-B0 and mode-C0 Poll frames in series shall be separated by BIFS and completed in one TXOP obtained. The timing of Type A discovery blocks is depicted in Figure 173.

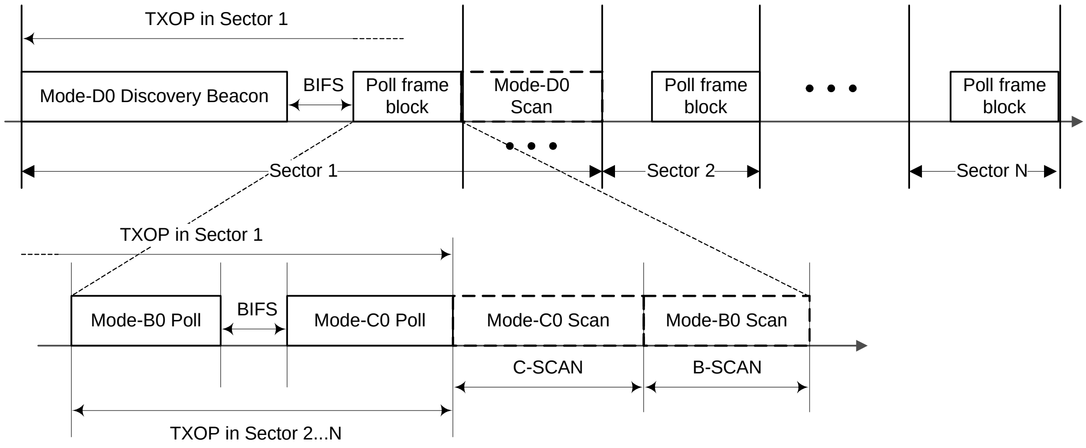


Figure 173 - Discovery beacon blocks of a Type A device

15.5.1.2 Discovery beacon blocks of a Type B device

A Type B device shall transmit mode-B0 discovery beacons only in the discovery channel. After the completion of a mode-B0 discovery beacon transmission, the Type B device shall transmit a mode-C0 Poll frame, followed by scanning for a mode-C0 response and a mode-B0 response. The transmission of the mode-B0 discovery beacon and mode-C0 poll frame shall be separated by a BIFS and completed in the TXOP obtained. The scanning shall start a SIFS after the end of the mode-C0 poll frame transmission. The Type B device shall first scan for responses to the transmitted mode-C0 poll frames for a duration of C-SCAN, after which the Type B device shall scan for responses to the previously transmitted mode-B0 discovery beacon for a duration of B-SCAN. If the Type B has multiple antenna that cover multiple sectors, after the completion of the scanning in the first sector, The Type B device shall switch to the next sector and repeat the same discovery-poll block for every sector the device can cover using a different antenna block, or beam. The timing of Type B discovery blocks is depicted in Figure 174.

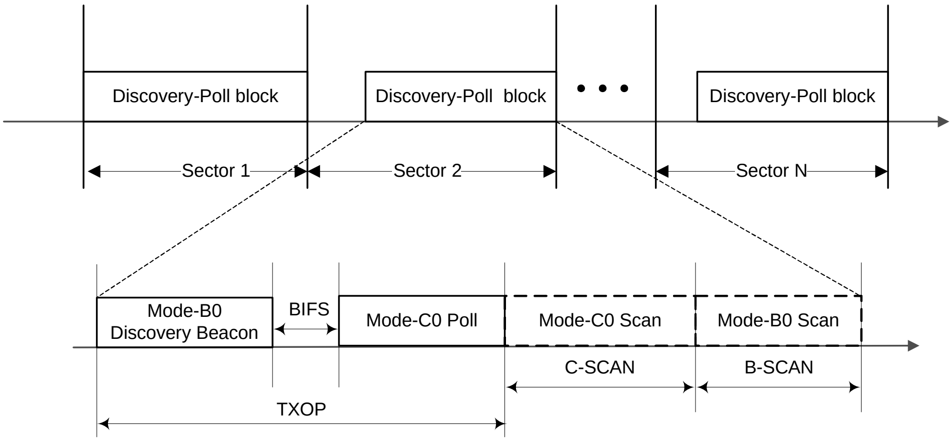
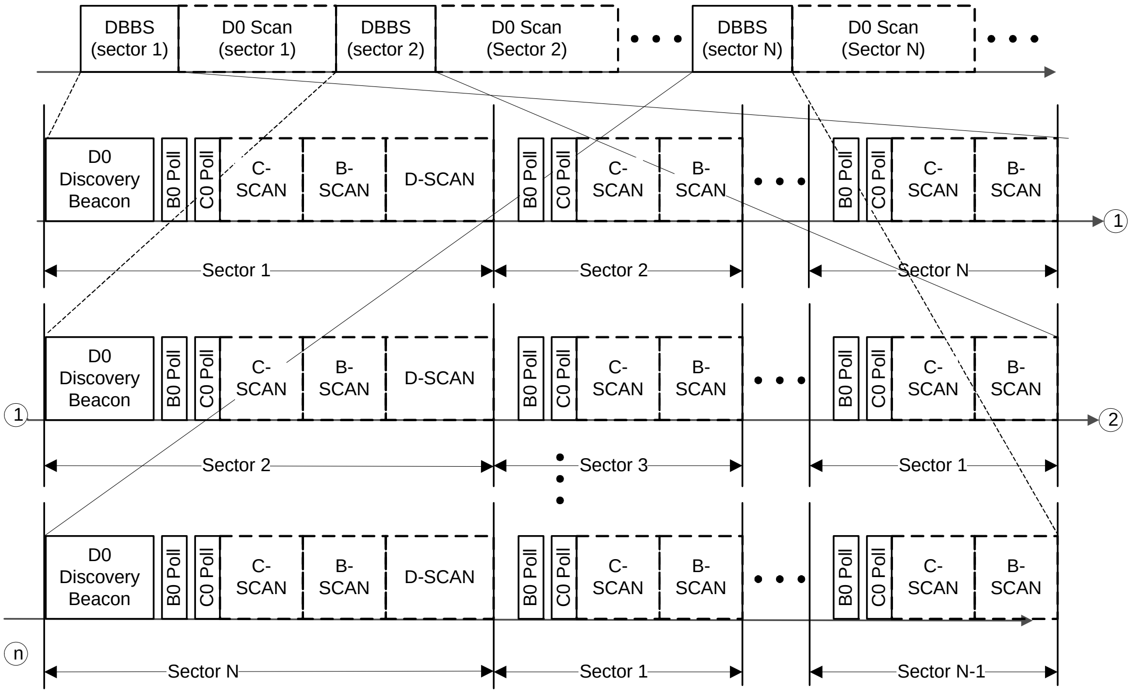


Figure 174 - Discovery beacon blocks of a Type B device

15.5.1.3 Randomization of discovery period

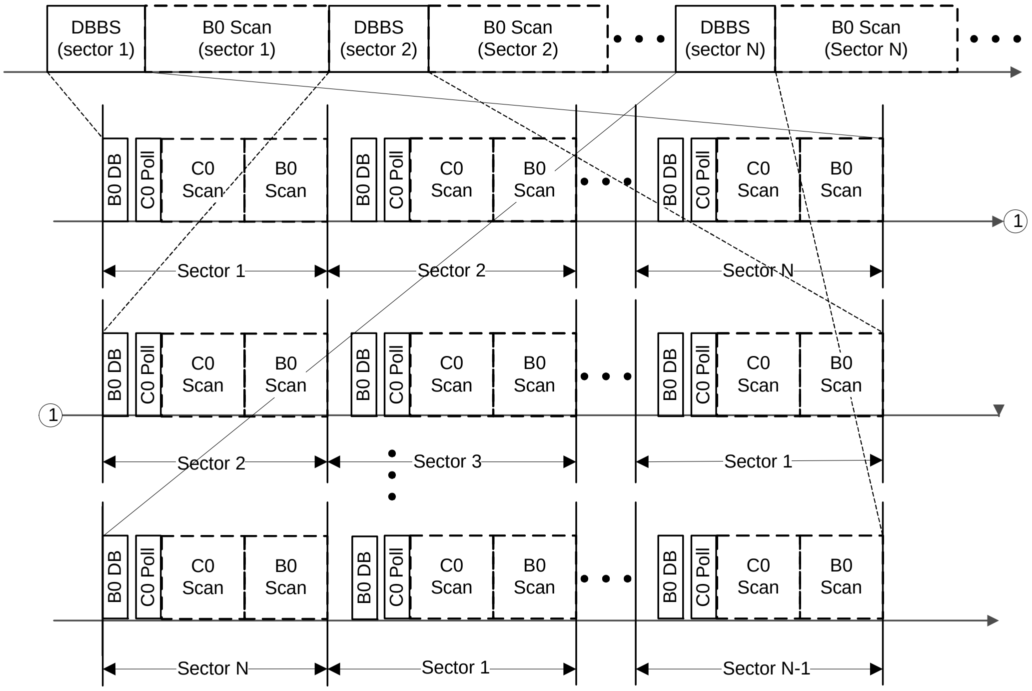
Each time after transmitting a set of discovery beacon blocks as described in 15.5.1.1 and 15.5.1.2, a device shall schedule another transmission of the same set of discovery beacon blocks at a time randomly drawn from a uniform distribution over the set {mDBPMin, mDBPMax} in number of superframes measuring from the start of the previous transmission of such a set of discovery beacon blocks. Before the scheduled transmission of the next set of discovery beacon blocks, a Type A device shall scan for mode-D0 discovery beacons or responses; and a Type B device shall scan for mode-B0 discovery beacon and Poll frames. The device shall repeat the above randomized discovery procedure until a response to the transmitted discovery beacons or Poll frames is received as described in 15.5.2, 15.5.3 and 15.5.4.

A device shall switch to the next sector to transmit the next set of discovery beacon blocks using a different antenna block or beam, if it covers multiple sectors using a number of antenna blocks, or beams. The order of antenna blocks, or beams used when switching shall be the same as the order used for transmitting its poll frame blocks (for a Type A device) or discovery-poll blocks (for a Type B device). The discovery period of a Type A device and a Type B device are specified in Figure 175 and Figure 176 respectively.



DBBS – Discovery Beacon BlockS

Figure 175 - Discovery period of a Type A device



DBBS – Discovery Beacon BlockS
DB – Discovery Beacon

Figure 176 - Discovery period of a Type B device

15.5.2 Transmission and reception of mode-A0 beacon frames

A Type A device, that has unreserved MAS, shall not transmit mode-A0 beacons in any channel unless it has (1) discovered another device with which the Type A device intends to exchange MPDUs, as described in 15.3 and (2) completed the antenna training if the newly discovered device is capable of antennna training.

Before a Type A device transmits any frames other than discovery beacons using the antenna beam determined in 15.3, it shall scan for mode-A0 beacons for at least one superframe, or at least two superframes if no mode-A0 beacon frame is received, using the same antenna beam. The device shall follow the following rules based on the scanning result:

A. If the device receives one or more mode-A0 beacon headers, but no beacon frames with a valid FCS during the scan, the device should scan for an additional superframe.

B. If the device has not selected a beacon slot to transmit mode-A0 beacons to exchange a MPDU with another device using any of its antenna beams, it shall follow the following rules:

- If the device receives no mode-A0 beacon frame headers during the scan, it shall create a new BP and send a mode-A0 beacon in the first beacon slot after the signalling slots;
- otherwise, if the device receives one or more mode-A0 beacons during the scan, it shall not create a new BP. Instead, prior to communicating with the newly discovered device, the device shall transmit a mode-A0 beacon in a beacon slot chosen from up to mBPExtension beacon slots located after the highest numbered unavailable beacon slot it observed in the last superframe and within mMaxBPLength after the BPST.

C. If the device already selected a beacon slot to transmit mode-A0 beacons to exchange a MPDU with another device using a different antenna beam, it shall follow the following rules:

- If the device receives one or more mode-A0 beacons that indicate a BPST that is not aligned with its own, it shall start the beacon relocation process as specified in 15.5.5.11;
- otherwise, if either of the following conditions is true, it shall transmit an additional mode-A0 beacon, using the same antenna beam as the one used in the scan, in a beacon slot chosen from up to mBPExtension beacon slots located after the highest numbered unavailable beacon slot it observed in the last superframe and within mMaxBPLength after the BPST.

  If the device receives no mode-A0 beacon frame headers during the scan - or - the mode-A0 beacon frame received indicates an aligned BPST to its own when transmitting any mode-A0 beacons; then, a Type A device shall follow the rules specified in 15.5.5.

With the exception of transmitting its own beacon as described in 15.5.2 and 15.5.5, a device shall not transmit frames in the current superframe during the BP length indicated in the most recent beacon received from any neighbour in the previous mMaxLostBeacons+1 superframes. A device shall not change beacon slots to a slot earlier than the highest-numbered unavailable beacon slot in the last superframe.

15.5.2.1 Discovery of additional devices

After starting the transmission of mode-A0 beacons in a channel, a Type A shall make a DRP reservation with Reservation Type set to Absence no later than mMaxDiscoveryLatency superframes. The length of this DRP reservation shall be greater than the minimal time needed by the Type A device to transmit a Type A discovery beacon block that consists of a mode-D0 beacon, mode-B0 and mode-C0 Poll frames. Within this reservation, the Type A device shall change to the discovery channel and transmit a discovery beacon block as described in 15.5.1.1. The discovery beacon sent in discovery channel after starting the transmission of mode-A0 beacons in a channel shall include a Channel Change IE to indicate which channel the device sends the mode-A0 beacons.

15.5.3 Transmission and reception of mode-B0 beacon frames

A Type B device shall not transmit mode-B0 beacons with Status set to Ready in any channel unless it has located another device with which the Type B device will exchange MPDUs. Before a Type B device transmits any frames other than mode-B0 beacons with Status set to Discovery, it shall scan for mode-B0 beacons, mode-C0 Poll and response frames for at least one superframe or at least two superframes if no mode-B0 beacon frame is received. If a Type C Poll or response frame is detected, the device shall transmit a Preempting frame as described in section 15.19.13. If the channel is detected as idle, it shall create a new BP and send a mode-B0 with Status set to Ready in the first beacon slot after the signalling slots and a mode A0 beacon with Status set to Dual in the second beacon slot after the signalling slots. If the device receives one or more mode-B0 beacon headers only, but no beacon frames with a valid FCS during the scan, the device should scan for an additional superframe. If the device receives only mode-B0 beacons during the scan, it shall not create a new BP. Instead, prior to communicating with another device, the device shall transmit a mode-B0 beacon with Status set to Ready and a mode-A0 beacon with Status set to Dual in two consecutive beacon slots chosen from up to mBPExtension beacon slots located after the highest numbered unavailable beacon slot it observed in the last superframe and within mMaxBPLength after the BPST. Otherwise, the Type B device shall not transmit a mode-B0 beacon with Status set to Ready in that channel.

When transmitting its dual-beacon, a Type B device shall follow the rules specified in 15.5.5.

With the exception of transmitting its own beacon as described in this sub clause and 15.5.5., a Type B device shall not transmit frames during the announced BP length of any of its neighbours.

A Type B device shall listen for duration of one superframe when it skips its beacon transmission as specified in 15.5.5.8. If transmission other than dual beaconing is detected using Physical sensing, the Type B device shall not move its dual beacon slots.

15.5.4 Transmission and reception of mode-C0 poll frames

A Type C device shall not transmit mode-C0 Poll frames in any channel unless it is operating in the master mode. Before a Type C device transmits mode-C0 Poll frames, it shall scan for mode-C0 Poll frames for at least one superframe, or at least two superframes if no mode-C0 Poll frame is received, using the scanning procedure described in 15.19.3. If the device receives one or more mode-C0 Poll frame headers, but no mode-C0 Poll frames with a valid FCS during the scan, the device should scan for an additional superframe. If the channel is detected as idle, it may start master-slave operation by sending mode-C0 Poll frames as described in 15.19.4. If the device receives only mode-C0 Poll frames during the scan, it shall not start a new master-slave operation in the current channel. Instead, the device may initiate a Master-Slave association according to the procedure described in clause 15.19.6.1. Otherwise, the Type C device shall not transmit mode-C0 Poll frames in the channel.

15.5.5 Superframe

Refer to 7.2.4.4 for the timing structure of the superframe. A Type A or Type B device shall use superframe timing structure for any frame exchange in a Data Channel. Superframe timing structure may be used in the discovery channel for MPDU exchange subject to preemption by device discovery or antenna training frame exchanges. A device shall not use superframe timing structure for MPDU exchange in Discovery Channel, if the initial scanning indicates that Discovery Channel is busy. A Type A or Type B device shall suspend all transmission using superframe timing structure in Discovery Channel for at leas mMaxNeighbourDetectionInterval, If the device receives a beacon with Status set to preemptive. The device shall access the channel after the suspension as it is powered up for the first time as described in 15.3.

15.5.5.1 Beacon period

Each superframe starts with a BP, which has a maximum length of mMaxBPLength beacon slots. The length of each beacon slot is mBeaconSlotLength. Beacon slots in the BP are numbered in sequence, starting at zero. The first mSignalSlotCount beacon slots of a BP are referred to as signalling slots and are used to extend the BP length of neighbours. Beaconing devices shall transmit beacons in the BP and listen for neighbour’s beacons in all beacon slots specified by its BP length in each superframe, except as described in 15.5.1. When transmitting in a beacon slot, a device shall start transmission of the frame on the medium at the beginning of that beacon slot. A device shall transmit beacons using the PHY modes based on the type of beacons to be sent as described in 15.5. The transmission time of beacon frames shall not exceed mMaxBeaconLength. This allows for a guard time of at least mGuardTime and pSIFS between the end of a beacon and the start of the next beacon slot. Figure 177 illustrates an example of a BP observed by a device in a given superframe.

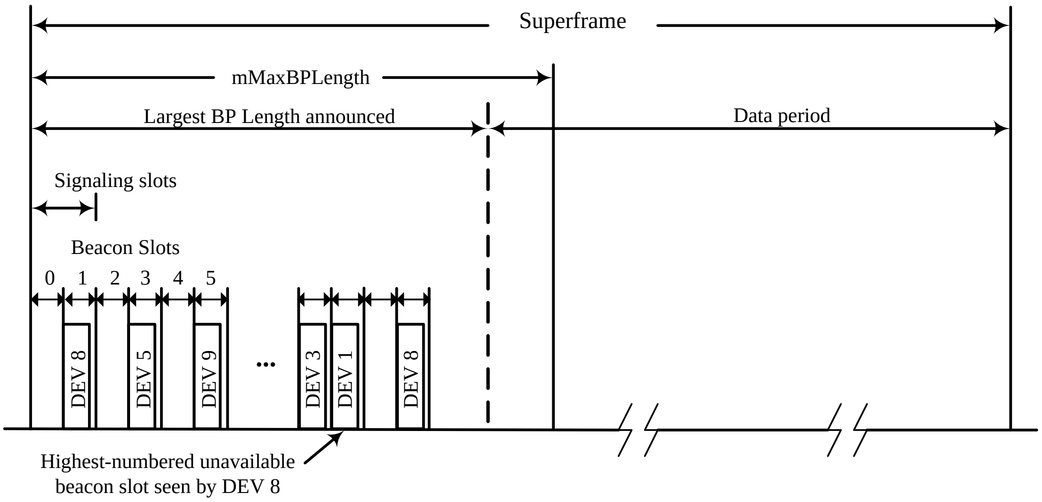


Figure 177 - Example BP structure

When a device selects an initial beacon slot after scanning for beacons as described in 15.5.2 and 15.5.3, the device shall transmit a beacon only if it selected a slot within mMaxBPLength after the BPST.

15.5.5.2 Beacon slot state

A device shall consider a beacon slot unavailable if in any of the latest mMaxLostBeacons+1 superframes:

- The beacon slot was considered to be occupied (according to Table 86); or

- The beacon slot was encoded as occupied (according to Table 86) in the BPOIE of any beacon received by the device.

A device shall consider a beacon slot available in all other cases.

15.5.5.3 BP length

A device shall consider a beacon slot to be monitored if in any of the latest mMaxLostBeacons+1 superframes:

- The device received a beacon frame in that beacon slot that is aligned to its BPST;

- The device received a beacon frame with an invalid FCS within 2×mGuardTime of that beacon slot boundary; or

- The beacon slot was encoded as occupied (according to Table 86) with a DevAddr not equal to BcstAddr in the BPOIE of any beacon received by the device.

A device shall announce its BP length in its beacon as a count of beacon slots starting from the BPST. The announced BP length shall include a) the device's own beacon slot in the current superframe, b) all monitored beacon slots in the BP of the prior superframe, and c) the beacon slot indicated in any beacon received in a signalling slot in the prior superframe.

The announced BP length shall not include more than mBPExtension beacon slots after the latest of a, b, and c, above. The announced BP length shall not exceed mMaxBPLength. Power-sensitive devices generally should not include any beacon slots after the last monitored beacon slot in their announced BP length.

The BP length reported by a device varies, as new devices become members of its extended beacon group, and as the device or other devices in its extended beacon group choose a new beacon slot for beacon slot collision resolution or BP contraction.

15.5.5.4 Neighbours

A device shall consider another device to be a neighbour if it has received a beacon from that device within the last mMaxLostBeacons+1 superframes, and the latest beacon from the device indicated a BPST aligned with its own. If a device has not received a beacon from another device for the last mMaxLostBeacons+1 superframes, it shall not consider the device a neighbour. A device shall not consider a received beacon with the Signalling Slot bit set to one as received from a neighbour.

15.5.5.5 Beacon slot collision

If a device detects a beacon collision as described in 15.5.5.9 it shall choose a different beacon slot for its subsequent beacon transmissions from up to mBPExtension beacon slots located after the highest-numbered unavailable beacon slot it observed in the last superframe and within mMaxBPLength after the BPST.

15.5.5.6 Use of signalling slots

If the beacon slot in which a device will transmit its beacon in the current superframe is located beyond the BP length indicated in any beacon the device received from a neighbour in the previous superframe, the device shall also transmit the same beacon, except with the Signalling Slot bit set to one, in a randomly selected signalling slot, except as follows:

- A device should follow recommendations in 15.5.5.11.3, if applicable.

- If a device transmits a beacon in a signalling slot for mMaxLostBeacons+1 consecutive superframes, it shall not transmit a beacon in a signalling slot in the next mMaxLostBeacons+1 superframes, and it should not transmit a signalling slot beacon for an additional aperiodic interval that does not exceed mMaxSignalingSlotBackoff superframes

Subject to the preceding exceptions, a device also may send a beacon in a signalling slot in response to abnormal conditions, such as failure to receive a beacon from a neighbour that previously did not include the device's beacon slot in its BP Length, or failure of a neighbour to report reception of the device's beacon in its BPOIE.

A device may consider a beacon received in a signalling slot as if it were not a received beacon, except to report reception as required in 15.1.10.4 and to process the Beacon Slot Number field as required in 15.5.5.3 and 15.5.5.9.

15.5.5.7 Required reception interval

An active mode device shall listen for neighbours' beacons in the first N beacon slots in each superframe, where N is the greater of its BP Length values for the current and previous superframes, as defined in 15.5.5.3. At a minimum, the device shall listen for intervals such that it would receive a frame with a reception time within mGuardTime of the start of any of the N beacon slots.

If a device received a beacon with invalid FCS, or detected a medium activity that did not result in reception of a frame with valid HCS in a signalling slot in the previous superframe, no BP Length adjustment is required, but it shall listen for beacons for an additional mBPExtension beacon slots after its BP length indicated in the current superframe, but not more than mMaxBPLength beacon slots.

15.5.5.8 Skipping beacon transmission

An active mode device shall transmit a beacon in each superframe, except as follows: In order to detect beacon slot collisions with neighbours, a device shall skip beacon transmission aperiodically, and listen for a potential neighbour in its beacon slot. A device shall skip beacon transmission, but not any associated signalling slot beacon, at least every mMaxNeighbourDetectionInterval. When a device skips beacon transmission, it shall act as if the skipped beacon were transmitted.

15.5.5.9 Beacon slot collision detection

A device shall consider itself involved in a beacon slot collision with another device in its extended beacon group if one of the following events occurs:

- Its beacon slot is reported as occupied in the BPOIE in any beacon it receives in the current superframe, but the corresponding DevAddr is neither BcstAddr nor its own DevAddr used in the previous superframe.

- After skipping beacon transmission in the previous superframe, its beacon slot is reported as occupied in the BPOIE in any beacon it receives in the current superframe, and the corresponding DevAddr is not BcstAddr.

- When skipping beacon transmission in the current superframe, it receives a MAC header of type beacon frame in its beacon slot.

- It receives a signalling slot beacon aligned with one of its own signalling slots, with the Beacon Slot Number field set to its own beacon slot.

Certain events indicate a potential beacon slot collision. A device should consider the possibility of a beacon slot collision and take appropriate action if one or more of the following anomalous events occurs, or occurs consistently over multiple superframes:

- The device's beacon slot was reported as occupied and the corresponding DevAddr was BcstAddr in the BPOIE of a beacon it received in the current superframe, and it sent a beacon in its beacon slot in the previous superframe.

- After skipping beacon transmission in the previous superframe, its beacon slot is reported as occupied in the BPOIE in any beacon it receives in the current superframe and the corresponding DevAddr is BcstAddr.

- When skipping beacon transmission in the current superframe, it receives a PHY indication of medium activity in its beacon slot that does not result in correct reception of a frame header.

In reaction to events that indicate a potential beacon slot collision, a device should:

- consider itself involved in a beacon slot collision and change slots as required in 15.5.5.5;

- skip beacon transmission; or

- send a beacon in a signalling slot, subject to requirements in 15.5.5.6.

At a minimum, a device shall execute at least one of these recommended reactions in the next superframe if in mMaxBeaconSlotCollisionDetectionLatency consecutive superframes one or more of the anomalous events described above occurs, and the device has not executed a recommended reaction in those mMaxBeaconSlotCollisionDetectionLatency superframes.

Other events can also indicate a potential beacon slot collision. For example, if a device's beacon slot is frequently reported as unoccupied in the BPOIE of a beacon it receives, it could indicate a collision, and the device may take action as described above.

15.5.5.10 BP contraction

A device shall consider its beacon to be movable if in the previous superframe it found at least one available beacon slot between the signalling slots and the beacon slot it indicates in its beacon in the current superframe. However, for purposes of BP contraction, a device may consider an unoccupied beacon slot to be occupied for up to mMaxMovableLatency superframes, if it detects conditions that indicate contraction into that beacon slot might lead to a beacon slot collision, such as a previous beacon slot collision or indication of poor link conditions in that beacon slot.

A device that includes a Hibernation Mode IE in its beacon shall consider its beacon to be non-movable during the announced hibernation period.

A device not involved in a beacon slot collision or a BP merge shall shift its beacon into the earliest available beacon slot following the signalling beacon slots in the BP of the next superframe, if in each of the latest mMaxLostBeacons+1 superframes:

- The device's beacon was movable; and

- The device did not receive a beacon from a neighbour that indicated a beacon slot after its own and had the Movable bit set to one; and

- The device did not receive a beacon from a neighbour that contained a BPOIE that encoded a beacon slot after its own as Movable (per Table 86).

However, if in the last mMaxLostBeacons+1 superframes the device received a beacon from a neighbour that indicated a BP Length that did not include the device's beacon slot, and that beacon had the Movable bit set to one, the device should not change to an earlier beacon slot in the next superframe.

Figure 178 shows some examples of BP contraction.

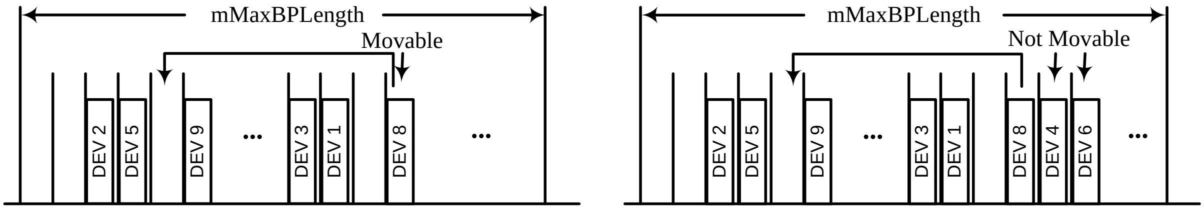


Figure 178 - Illustration for BP contraction by example device

15.5.5.11 Beacon relocation

Due to changes in the propagation environment, mobility, or other effects, devices using two or more unaligned BPSTs may come into range. This causes overlapping superframes. A received beacon that indicates a BPST that is not aligned with a device's own BPST is referred to as an alien beacon. The BP defined by the BPST and BP length in an alien beacon is referred to as an alien BP.

Synchronization problems could cause the beacon of a fast device to appear to be an alien beacon. A device shall consider a BPST to be aligned with its own if that BPST differs from its own by less than 2xmGuardTime. A device shall consider an alien BP to overlap its own if its BPST falls within the alien BP or if the alien BPST falls within its own BP. A device shall not consider a beacon that has the Signalling Slot bit set to one to be an alien beacon.

If a device does not receive an alien beacon for up to mMaxLostBeacons superframes after receiving one in a previous superframe, it shall use information contained in the most-recently received beacon as if the alien beacon were received at the same offset within the current superframe.

15.5.5.11.1 Overlapping BPs

If the BPST of a Type A device falls within an alien BP, the device shall relocate its beacon to the alien BP according to the following rules:

1. The device shall change its BPST to the BPST of the alien BP.

2.The device shall follow normal BP join rules as specified in 15.5.2 and 15.5.3 to relocate its beacon to the alien BP.

3.The device shall not send further beacons in its previous BP.

15.5.5.11.2 Non-overlapping BPs

If a Type A device detects an alien BP that does not overlap in time with its own BP, it shall merge BPs according to the following rules.

1.The device shall include in its beacon a DRP IE with Reservation Type set to Alien BP for the alien BP. Since the MAS boundaries may not be aligned, the device may need to include an additional MAS in the reservation to completely cover the alien BP. If the device received multiple beacons from the alien BP, it shall include all MASs used by the largest reported BP length in the reservation. If the MASs occupied by the alien BP change over time, the device shall update the DRP IE accordingly.

2. The device shall follow normal BP join rules as specified in 15.5.2 and 15.5.3 to relocate its beacon to the alien BP within mBPMergeWaitTime if the alien BPST falls within the first half of the superframe, or within 1.5×mBPMergeWaitTime if the alien BPST falls within the second half of the superframe.

A Type A device that transmits or receives a beacon in its own BP that contains a DRP IE with Reservation Type set to Alien BP shall listen for beacons during the MASs indicated in the reservation.

A Type B shall not allocate to an alien BP unless the received alien beacon is of mode-B0 with Status set to Master. When relocating its beacon to an alien, the Type B shall use the same procedure as defined for Type A device above.

A Type C device shall never relocate its beacon to an alien BP. The Type C device shall immediately cease its Master-Slave operation on that channel

15.5.5.11.3 Use of signalling slots after BP merge

After changing its BPST, regardless of whether due to overlapping or non-overlapping BPs, if a device is required to send a beacon in a signalling slot according to 15.5.5.6, it should wait for a random number of superframes before sending a beacon in a signalling slot. The device should choose the random number with equal probability in the range zero to the BP Length declared in its last beacon before relocating to the alien BP.

15.6 Distributed reservation protocol (DRP)

The DRP enables devices to reserve one or more MASs that the device can use to communicate with one or more neighbours. All devices that use the DRP for transmission or reception shall announce their reservations by including DRP IEs in their beacons (see Distributed reservation protocol (DRP) IE). A reservation is the set of MASs identified by DRP IEs with the same values in the Target/Owner DevAddr, Owner, Reservation Type, and Stream Index fields.

Reservation negotiation is always initiated by the device that will initiate frame transactions in the reservation, referred to as the reservation owner. The device that will receive information is referred to as the reservation target.

15.6.1 Reservation type

Each DRP IE, whether included in a beacon or separately transmitted during explicit DRP negotiation, specifies a reservation type. A device shall decode all DRP IEs in all beacons received from neighbours and shall not transmit frames except as permitted by the reservation type. For all reservation types, a device shall not initiate a frame transaction in a reservation block if that transaction would not complete pSIFS plus mGuardTime before the end of the reservation block.

Reservation types are defined and summarized in Table 104.

Table 104 - Reservation types

Reservation Type	Description	Reference
Alien BP	Prevents transmission during MASs occupied by an alien BP.	15.6.1.1
Hard	Provides exclusive access to the medium for the reservation owner and target.	15.6.1.2
Soft	Permits DCA for reservation target to send command, or control frames, but the reservation owner has preferential access.	15.6.1.3
Private	Provides exclusive access to the medium for the reservation owner and target. Channel access methods and frame exchange sequences are out of scope of this specification.	15.6.1.4
Absence	Indicates the medium time where the reservation owner returns to Discovery Channel to find additional device	15.5.2.1

15.6.1.1 Alien BP reservations

A device shall announce an alien BP reservation to protect alien BPs as described in 15.5.5.11.2. A device shall not transmit frames during an alien BP reservation except possibly to send a beacon in the alien BP.

15.6.1.2 Hard reservations

In a hard reservation, devices other than the reservation owner and target(s) shall not transmit frames. Devices other than the reservation owner shall not initiate frame transactions.

A device shall not transmit a data or aggregated data frame in a hard reservation unless the Delivery ID field is set to a Stream Index that is the same as the Stream Index for the reservation and the DestAddr of the frame is the same as the Target DevAddr for the reservation or the DestAddr of the frame matches the DevAddr of any target of an established multicast reservation. The reservation owner may transmit any command or control frame in a hard reservation.

15.6.1.3 Soft reservations

In a soft reservation, devices access the medium following DCA rules. The reservation owner may access the medium with the LIFS and without performing backoff. It may begin transmission at the beginning of each reservation block. It may initiate an additional frame transaction after any transaction it initiated but shall not initiate such a transaction later than SIFS after the end of the previous frame transaction. The reservation owner shall not transmit a data or aggregated data frame without backoff unless the Delivery ID field is set to a Stream Index that is the same as the Stream Index for the reservation and the DestAddr of the frame is the same as the Target DevAddr for the reservation or the DestAddr of the frame matches the DevAddr of any target of an established multicast reservation. The reservation owner may transmit any command or control frame without backoff. Neighbours of a reservation owner shall follow DCA rules to access the medium. Neighbours of a reservation target that are not neighbours of the reservation owner shall not access the medium.

15.6.1.4 Private reservations

The channel access method and frame exchange sequences used during a private reservation are out of the scope of this standard. Standard frame formats and frame types shall be used during a private reservation. In a private reservation, neighbours of the reservation owner and target(s) shall not transmit frames.

15.6.2 Reservation waveform

Each DRP IE, whether included in a beacon or separately transmitted during explicit DRP negotiation, specifies the waveform that shall be used for all the frame transactions in the reservation.

15.6.3 DRP availability IE

The DRP Availability IE identifies the MASs where a device is able to establish a new DRP reservation.

The combination of information from DRP Availability IEs and DRP IEs allows an owner to determine an appropriate time for a new DRP reservation. Device shall mark a MAS unavailable if (1) a neighbour includes it in a DRP IE with a target other than the device, whether the Reservation Status bit is zero or one, and (2) the Antenna Index in the received DRP IE is the same as the Antenna Index to be used by the device for a new DRP reservation.

In order to facilitate the DRP negotiation process, devices that are aware of existing neighbours' DRP reservations should mark the corresponding MASs as unavailable as described above.

A device shall mark a MAS unavailable if the device includes it in a DRP IE with the Reservation Status bit set to one. It shall mark a MAS unavailable if any BP occupies any portion of that MAS, based on information in any beacon received in the latest mMaxLostBeacons+1 superframes.

15.6.4 DRP reservation negotiation

There are two mechanisms used to negotiate a reservation: explicit and implicit. For explicit negotiation, the reservation owner and target use DRP Reservation Request and DRP Reservation Response command frames to negotiate the desired reservation. For implicit negotiation, the reservation owner and target use DRP IEs transmitted in their beacons. For either negotiation mechanism, the reservation owner completes the negotiation by including an appropriate DRP IE in its beacon.

A device shall not negotiate for MASs that are marked as unavailable, unless the MASs are referenced only in a DRP IE with Reason Code set to Denied.

A Type B device shall not negotiate for a new reservation or more MASs for an existing reservations if non-Type B transmission excluding Type C transmission is detected during the device's beacon period.

A device shall announce in the MAC Capabilities IE in its beacon whether it is capable of explicit DRP negotiation. A device shall not initiate an explicit DRP negotiation with devices that do not support it.

A device shall only initiate negotiation for a reservation as the reservation owner.

For reservations of type Alien BP, there is no negotiation with neighbours. A device shall include the appropriate DRP IE with Reservation Status set to ONE on detection of an alien BP, as specified in 15.6.1.1.

15.6.4.1 Negotiation

When negotiating a reservation, the reservation owner shall set the Target/Owner DevAddr field of the DRP IE to the DevAddr of the reservation target. It shall set the Reservation Status bit to ZERO and the Reason Code to Accepted in the DRP IE. For new streams, the Stream Index shall be set to a value that is currently not used with this Target DevAddr and has not been used as such for mMaxLostBeacons+1 superframes. To negotiate additional MASs for an existing stream, the Stream Index shall be set to the value used for the existing stream. The device shall set the Antenna Index as the Antenna Index in the beacon transmitted to the reservation target.

A reservation owner shall not transmit unicast frames within reserved MASs in a hard, soft or private reservation unless it and the recipient included DRP IEs with the Reservation Status bit set to ONE in their most-recently transmitted beacons.

When negotiating a reservation, a reservation target shall set the Target/Owner DevAddr field of the DRP IE to the DevAddr of the reservation owner. The device shall set the Antenna Index as the Antenna Index in the beacon transmitted to the reservation owner. If a unicast reservation is granted, it shall set the Reservation Status bit to one and the Reason Code to Accepted. If a multicast reservation is granted, it shall set the Reservation Status bit to the same value included in the DRP IE by the reservation owner, and shall set the Reason Code to Accepted. If the reservation is not granted, it shall set the Reservation Status bit to zero. If the reservation cannot be granted due to a conflict with its own or its neighbours' reservations, the reservation target shall set the Reason Code to Conflict. If the reservation is not granted, it shall set the Reason Code to Denied. If the reservation target cannot grant the reservation immediately, it may set the Reason Code to Pending, and deliver a final response later. For a unicast reservation, the reservation target shall set the DRP Allocation fields to match those in the request. For a multicast reservation, it shall set the DRP Allocation fields to match the request, or to include a subset of the MASs included in the request.

15.6.4.2 Explicit negotiation

To start explicit DRP negotiation, the reservation owner shall send a DRP Reservation Request command frame to the target device, as defined in 14.5.1.

On reception of a DRP Reservation Request command the reservation target shall send a DRP Reservation Response command, as defined in 14.5.2, to the reservation owner. The fields in the DRP IE shall be set according to 14.7.8. If the reservation cannot be granted due to a conflict with its own or its neighbours' reservations, the reservation target shall include a DRP Availability IE in the DRP Reservation Response command frame.

In a DRP Reservation Response command frame for a multicast reservation, the reservation target shall include a DRP Availability IE for a Reason Code other than Denied. Final multicast reservations are established implicitly, as described in 15.6.4.3.

15.6.4.3 Implicit negotiation

Implicit negotiation is carried out by transmitting DRP IE(s) in beacon frames. A device that supports the DRP shall parse all beacons received from neighbours for DRP IE(s) whose Target/Owner DevAddr field matches either the device's DevAddr or a multicast DevAddr for which the device has activated multicast reception. From this initial selection, the device shall process the DRP IE(s) that are new with respect to DRP IE(s) included in the most recently received beacon from the same device as a DRP reservation request or a DRP reservation response.

To start implicit negotiation, a reservation owner shall include a DRP IE that describes the proposed reservation in its beacon. The device should continue to include the DRP IE for at least mMaxLostBeacons+1 consecutive superframes or until a response is received.

On reception of a unicast DRP reservation request in a beacon, the reservation target shall include a DRP reservation response in its beacon no later than the next superframe, with fields set as described in 15.6.4.1. If the Reason Code indicates Conflict, the reservation target shall include a DRP Availability IE in its beacon.

As long as the reservation owner includes a unicast DRP reservation request in its beacon, the reservation target shall continue to include the DRP reservation response in its beacon. The reservation target shall not change the Reservation Status bit to ONE if there is a reservation conflict with its neighbours.

On reception of a multicast DRP reservation request, a reservation target shall include a reservation response DRP IE in its beacon no later than the next superframe if it is a member of the targeted multicast group. The fields in the DRP IE shall be set according to 14.7.8. If the Reservation Status bit in the response is ZERO, the reservation target shall include a DRP Availability IE in its beacon unless the Reason Code is set to Denied.

A device that elects to receive traffic in an already established multicast reservation does not negotiate the reservation. To join an established multicast reservation that does not conflict with other existing reservations, a device shall include corresponding DRP IE(s) in its beacon with Reservation Status bit set to ONE and Reason Code set to Accepted.

A device that cannot join an established multicast reservation because of an availability conflict may inform the source by including the corresponding DRP IE(s) in its beacon with Reservation Status bit set to ZERO, and the Reason Code set to Conflict. The device shall also include the DRP Availability IE in the beacon.

15.6.4.4 Negotiation conclusion

To conclude negotiation for a unicast reservation, the reservation owner shall set Reservation Status to ONE in the DRP IE in its beacon after receiving a beacon from the reservation target that contains a corresponding DRP IE with Reservation Status set to ONE. To conclude negotiation for a multicast reservation, the reservation owner may set Reservation Status to ONE in a DRP IE in its beacon in the next superframe after transmitting the same DRP IE with Reservation Status set to ZERO, regardless of responses from potential multicast recipients. If a reservation conflict exists, the reservation owner shall not set the Reservation Status bit to ONE except as specified in 15.6.6.

15.6.5 DRP reservation announcements

Once negotiation for a reservation successfully completes, the reservation owner and target shall include DRP IE(s) in their beacons that describe the reservation. Within each DRP IE, the Reason Code shall be set to Accepted and the Reservation Status bit shall be set to ONE. The devices shall include the DRP IEs in each beacon transmitted until the reservation is modified or terminated.

15.6.6 Resolution of DRP reservation conflicts

Devices engaged in independent DRP negotiation could attempt to reserve the same MAS, or due to mobility, devices could have reserved the same MAS. A device is considered to conflict with an existing DRP reservation if (1) a MAS included in the device reservation is included in the existing reservation, (2) the Antenna Index of the existing reservation is the same as the Antenna Index in the beacon that includes the DRP IE of the existing reservation, and (3) the Antenna Index of the existing reservation is the same as the Antenna Index of the device's own reservation. A device might detect a conflict during a DRP negotiation or after a reservation has been established. Reservations of type Alien BP never conflict with other reservations of type Alien BP.

A device shall apply the following rules to a conflict between a DRP IE included in its beacon and another DRP IE included by a neighbour:

1. If the device's reservation is of type Alien BP, the device shall maintain the reservation.

2. If the neighbour’s reservation is of type Alien BP, the device shall not transmit frames in conflicting MASs. If the device is the reservation target, it shall also set the Reason Code in its DRP IE to Conflict.
3. If the device is a Type B device, the device shall maintain the reservation
4. If the neighbour is a Type B device, the device shall not transmit frames in conflicting MASs. If the device is the reservation target, it shall also set the Reason Code in its DRP IE to Conflict.
5. If the device's DRP IE has the Reservation Status bit set to ZERO and the neighbour’s DRP IE has the Reservation Status bit set to ONE, the device shall not set the Reservation Status bit to ONE and shall not transmit frames in conflicting MASs. If the device is the reservation target, it shall also set the Reason Code in its DRP IE to Conflict.
6. If the device's DRP IE has the Reservation Status bit set to ONE and the neighbour’s DRP IE has the Reservation Status bit set to ZERO, the device may maintain the reservation.
7. If the device's DRP IE and neighbour’s DRP IE have the Reservation Status bit set to the same value and one of the following conditions is true, the device may maintain the reservation.
    a. The device's DRP IE and neighbour’s DRP IE have the Conflict Tie-breaker bit set to the same value and the device's occupied beacon slot number is lower than the beacon slot number of the neighbour; or
    b. The device's DRP IE and neighbour’s DRP IE have the Conflict Tie-breaker bit set to different values and the device's occupied beacon slot number is higher than the beacon slot number of the neighbour.
8. If the device's DRP IE and neighbour’s DRP IE have the Reservation Status bit set to ZERO and one of the following conditions is true, the device shall not set the Reservation Status bit to ONE. If the device is the reservation target, it shall set the Reason Code in its DRP IE to Conflict.
    a. The device's DRP IE and neighbour’s DRP IE have the Conflict Tie-breaker bit set to the same value and the device's occupied beacon slot number is higher than the beacon slot number of the neighbour; or
    b. The device's DRP IE and neighbour’s DRP IE have the Conflict Tie-breaker bit set to different values and the device's occupied beacon slot number is lower than the beacon slot number of the neighbour.
9. If the device's DRP IE and neighbour’s DRP IE have the Reservation Status bit set to ONE and one of the following conditions is true, the device shall not transmit frames in conflicting MASs. It shall remove the conflicting MASs from the reservation or set the Reservation Status to ZERO. If the device is the reservation target, it shall set the Reason Code in its DRP IE to Conflict.
    a. The device's DRP IE and neighbour’s DRP IE have the Conflict Tie-breaker bit set to the same value and the device's occupied beacon slot number is higher than the beacon slot number of the neighbour; or
    b. The device's DRP IE and neighbour’s DRP IE have the Conflict Tie-breaker bit set to different values and the device's occupied beacon slot number is lower than the beacon slot number of the neighbour.

When a reservation owner withdraws a reservation or part of a reservation due to a conflict, it shall invoke a backoff procedure prior to requesting additional MASs in any reservation. The device shall initialize the backoff window BackoffWin to mDRPBackoffWinMin. When the backoff algorithm is invoked, the device shall select a random number N uniformly from [0, BackoffWin-1]. The device shall not request additional MASs for N superframes. If a further negotiation fails due to a conflict, the device shall double BackoffWin, up to a maximum of mDRPBackoffWinMax. After a negotiation completes, the device shall generate a new backoff N. If a device does not request any MASs for 4×BackoffWin superframes, the device may terminate this backoff procedure and request MASs at any time unless another conflict occurs.

If a reservation target sets Reason Code to Conflict in any DRP IE in its beacon, it shall include a DRP Availability IE in the same beacon.

15.6.7 BPST realignment and existing DRP reservations

A device that realigns its BPST as described in 15.5.5.11 may assert new DRP reservations with Reservation Status bits set to ONE in the new beacon so long as they are equivalent to its old DRP reservations with the Reservation Status bit set to ONE in the prior BP. For this purpose, two DRP reservations are equivalent if their corresponding Target/Owner DevAddr, Stream Index, and Reservation Type fields are the same and the number of MASs claimed by the new reservation is less than or equal to the number claimed by the old reservation.

A device that realigns its BPST shall not assert DRP reservations with MASs that conflict with any BP it announced or detected. The device shall not assert DRP reservations with MASs that conflict with reservations with Reservation Status equal to ONE announced in the new BP unless no other MASs are available. Any conflict with existing reservations shall be resolved according to the procedures specified in 15.6.6.

15.6.8 Modification and termination of existing DRP reservations

A reservation owner may reserve additional MASs for a stream by negotiating an addition to the reservation using a DRP IE with the same Target/Owner DevAddr, Stream Index, and Reservation Type. Once negotiation has completed successfully, the reservation owner should combine the DRP IEs. When combining DRP IEs, the reservation owner shall set the Reason Code to Modified until a DRP IE is received from the reservation target that describes the combined reservation.

A reservation owner may remove MASs from an established reservation without changing the Reservation Status bit in the DRP IE. If a reservation owner removes some MASs from an established reservation, it shall set the Reason Code in its DRP IE to Modified until the reservation target has changed its DRP IE to match.

A reservation target may remove MASs from an established reservation without changing the Reservation Status bit in the DRP IE due to a conflict, as described in 15.6.6 or due to reception of a Relinquish Request IE. If the reservation target is unicast, the reservation owner shall remove the same MASs from the reservation or terminate the reservation in the current or following superframe.

To terminate a reservation, the reservation owner shall remove the DRP IE from its beacon.

If a reservation owner changes or removes a DRP IE, the reservation targets shall update or remove the corresponding DRP IE from their beacons in the current or following superframe.

To terminate a reservation, a reservation target shall set the Reservation Status bit to ZERO and the Reason Code to an appropriate value, as if responding to an initial reservation request. The reservation owner shall terminate the corresponding reservation or set the corresponding Reservation Status bit to ZERO in the current or following superframe.

If a reservation owner or target does not receive a beacon or any other frame from the other participant in the reservation for more than mMaxLostBeacons superframes, it shall consider the reservation terminated, and shall remove the corresponding DRP IE(s) from its beacon.

15.6.9 Retransmit procedures in DRP reservations

In a hard DRP reservation block, if the reservation owner transmits a frame with ACK Policy set to Imm-ACK or B-ACK, but does not receive the expected acknowledgement frame, it may retransmit the frame within the same reservation block if the reservation block has not been released.

In a soft DRP reservation block, the reservation owner may retransmit a frame with no backoff, as described in 15.6.1.3. Devices other than the reservation owner that retransmit frames in a soft DRP reservation block shall follow the DCA rules defined in 15.2

A device shall not retransmit a frame earlier than pSIFS after the end of an expected acknowledgement, whether or not it receives the expected frame. A device shall not retransmit a frame in the current reservation block if there is not enough time remaining in the reservation block for the entire frame transaction.

15.7 Coexistence and interoperability

This clause specifies mechanisms to prevent potential interference among devices of different types and to facilitate transmission between devices of different types.

15.7.1 Coexistence

Devices of different types are given priorities when accessing the channel or making DRP reservations. A Type A/B device may scan for Type C Poll frames with Status set to Polling any time during its superframe or before starting transmitting mode-A0 or mode-B0 beacons with Status set to Ready. If such mode-C0 Poll frames are received during the scanning, the device shall send a channel release request according to 15.19.13.2.

A Type B device shall send dual-beacon to announce its DRP reservation as specified in 15.7.1.1. A Type C device shall follow the rules specified in 15.7.1.2 to avoid interfering with the transmission of coexisting Type A and Type B devices.

15.7.1.1 Dual-beaconing

When it is required to transmit a dual-beacon in this specification, a Type B device shall select two consecutive beacon slots according to the rules specified in 15.5.5 and 15.7.2. The Type B device shall transmit a mode-B0 beacon with Status set to Ready in the first beacon slot and a mode-A0 beacon of the same payload with Status set to Dual in the second slot. The timing of the dual beaconing is specified in Figure 179.

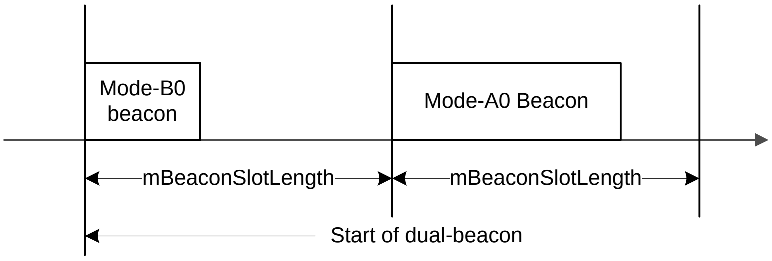


Figure 179 - Timing of a dual beacon

15.7.1.2 Channel sensing of a Type C device

A Type C device shall access discovery channel as described in 15.3.3. A Type C device shall not transmit any frames in any channel unless it has determined the channel is free of non-Type C transmission.

In addition, a Type C device shall aperiodically skip all transmission (including Poll frames) for duration of one superframe, and scan for non-Type C transmission. The Type C device shall perform such skipping-and-scanning at least every mMaxNeighbourDetectionInterval. Upon the detection of non-Type C transmission, the Type C device shall suspend its transmission for at least mMaxNeighbourDetectionInterval. To determine the channel or a reservation block is free of non-Type C transmission, a Type C device shall perform channel scanning according to 15.19.3. The Type C device shall access the channel after the suspension as it is powered up for the first time as described in 15.3.3.

15.7.2 Interoperability

Interoperability between different types of devices is accomplished using Master-Slave operation. The MSPr shall select a channel to exchange MSDUs using the explicit channel selection process before the master device starts the transmission of mode-A0 or mode B0 beacons with the Status set to Ready in a channel. If the master device has already started or joined a beacon group by transmitting mode-A0 or more-B0 beacons in a selected channel, the slave device shall switch to the selected channel as indicated in the Channel Change IE in the discovery beacon sent by the master device. A Type A device shall discover additional slave devices as specified in 15.7.2.1, if there is sufficient medium time available for frame exchange of Poll frame blocks (as specified in 15.5.1.1).

15.7.2.1 Discovery of slave devices in a data channel

After starting the transmission of mode-A0 beacons with Status set to Ready in a channel, a Type A device shall make a DRP reservation of type Private to transmit Poll frame blocks if there are MASs available, The Type A device shall scan for responses to the transmitted Poll frame blocks as specified in 15.5.1.1. If a mode-D0 beacon with Response is received correctly in a channel, the Type A device shall start a Type A-B MSPr following the procedure specified in 15.7.2.2.

A Type A device may adjust the reservation used for transmission of DBBS to release the reserved MASs for data transmission. The Type A device may terminate an existing reservation used for transmission of DBBS, if all MASs are needed for data transmission.

15.7.2.2 Type A-B MSPr

After starting the transmission of mode-A0 beacons in a channel, a Type A master device shall make a DRP reservation of type Private to transmit B-Poll frames in every superframe. The Type A device shall follow the same the rules specified in 15.5.5 to select two consecutive beacon slots for its Type B slave device to send its dual-beacon and indicate the timing of the selected beacon slots in the B-Poll frame it sends. The Type A master shall indicate the selected two slots as Occupied in the BPOIE in the mode-A0 beacon it sends before sending the B-Poll frame in the current superframe.

After a Type B slave device switches to a selected channel to exchange MPDUs with its master, the Type B device shall not transmit any frames until it receives a B-Poll frame from the Type A master device. The Type B device shall send a B-Response frame a SIFS after the reception of a B-Poll frame. In addition, the Type B device shall transmit a dual-beacon at the time indicated in the received B-Poll frame.

15.7.2.3 Type A/B-C MSPr

After starting the transmission of mode-A0 or mode-B0 beacons with Status set to Ready in a channel, the Type A or B device shall make a DRP reservation of type Private to transmit Type C Poll frames in the channel.

A Type C slave device shall not transmit any frames after it switches to a selected channel until it receives a Type C Poll frame from its master device. The Type C device shall follow the rules specified in 15.19 to send a Type C response and to exchange MPDUs with its master device.

15.8 Synchronization of devices

Each beaconing device shall maintain a beacon period start time (BPST). The device shall derive all times for communication with its neighbours based on the current BPST. The device shall adjust its BPST in order to maintain superframe synchronization with its neighbours, through the reception of neighbours' beacons. Since a Type B device cannot decode beacons transmitted by Type A devices, the Type B device cannot determine timing information needed for synchronization with Type A devices. Therefore, a Type A device shall always synchronize with its slowest Type B neighbour. The Type A device determines the difference between the actual reception time and the expected reception time of the mode-A0 beacon transmitted by a Type B device. The beacon's actual reception time is an estimate of the time that the start of the beacon preamble arrived at the receiving device's antenna. The expected reception time is determined from the Beacon Slot Number field of the received beacon and the receiving device's BPST. If the difference is positive, then the neighbour is slower. Otherwise, the neighbour is faster. A Type A device that synchronizes with a Type B device is a referred to as a forced synchronization device. A forced synchronization device shall set the Status of each transmitted beacon as ForcedSynchronization.

If a Type A device is not a forced synchronization device and receives mode-A0 beacons only from Type A neighbours, the device shall synchronize with a neighbour that is a forced synchronization. If none of the Type A device's neighbours is forced synchronization device, the Type A device shall synchronize with its slowest neighbour. A Type B device shall synchronize with its slowest Type B neighbour.

To maintain superframe synchronization with a slower neighbour, the device shall delay its BPST by the difference. To maintain superframe synchronization with a faster neighbour, the device shall advance its BPST by the difference. Any adjustment of the BPST shall be limited to a maximum of mMaxSynchronizationAdjustment per superframe. The adjustment to BPST may occur at any time following the detection of a slower device, but shall be done before the end of the superframe.

A device shall not use a beacon with the Signalling Slot bit set for synchronization. If a device does not receive a beacon from a neighbour, the device may use historical measurements to estimate the impact on superframe synchronization and increment its BPST accordingly. This estimate may be applied for up to mMaxLostBeacons consecutive superframes. Beacon transmit time and measured beacon receive time shall be accurate to at least mClockResolution.

A Type B or Type C device that is operating in the slave mode shall derive all times for communication with its master from the Poll frames transmitted by the master device.

15.8.1 Clock accuracy

MAC sublayers shall maintain a clock at least as accurate as mClockAccuracy. All time measurements, such as MAS boundary and frame reception time measurements, shall be measured with a minimum resolution of mClockResolution.

15.8.2 Synchronization for devices in hibernation mode

Devices in hibernation mode may become unsynchronized beyond the mGuardTime value during hibernation. A device in hibernation mode shall wake up at least one superframe before it will send a beacon and shall synchronize to the slowest clock in the beacon group during this superframe.

15.8.3 Guard times

Due to inaccuracy in the superframe synchronization and drift between synchronization events, the MAS start times of different devices are not synchronized perfectly. To ensure a full SIFS interval between transmissions in adjacent MASs, the devices shall maintain a SIFS interval and guard interval at the end of a reservation block. Guard times apply to all boundaries of DRP reservation blocks and BPs.

Figure 180 is an illustration of how a device uses the guard interval to maintain a SIFS interval between transmissions in adjacent reservation blocks. The length of the guard interval, mGuardTime, depends on the maximum difference between devices' MAS boundary times. The difference arises from synchronization error and drift. The guard time is determined as follows:

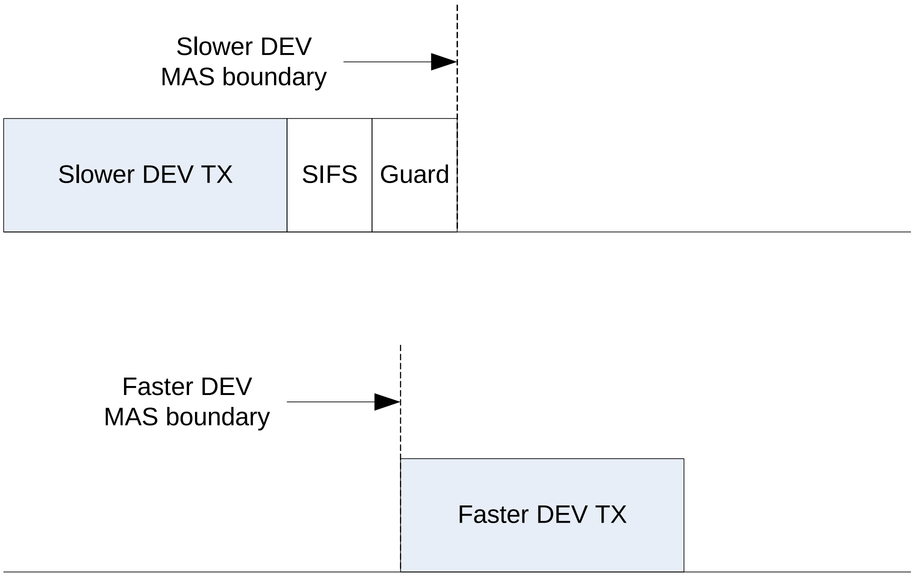


Figure 180 - Guard time

$$\text{mGuardTime} = \text{MaxSynchronizationError} + \text{MaxDrift} \quad (99)$$

where MaxSynchronizationError is the worst case error in superframe synchronization and MaxDrift is the worst case drift. Synchronization is achieved during the BP as described in 16.6. For purposes of determining guard time, MaxSynchronizationError is calculated as twice mClockResolution. Drift is a function of the clock accuracy and the time elapsed (SynchronizationInterval) since a synchronization event. The maximum drift, MaxDrift, is calculated using the worst case value for clock accuracy, mClockAccuracy, and the longest SynchronizationInterval:

$$\text{MaxDrift} = 2 \times \text{mClockAccuracy (ppm)} \times 1\text{E-}6 \times \text{SynchronizationInterval} \quad (100)$$

where SynchronizationInterval = (mMaxLostBeacons+1) × mSuperframeLength. Propagation delay will also affect timing uncertainty, but in a short-range network propagation delays are small. At 10 m range, the propagation delay is around 33 ns. This is much smaller than mClockResolution and it is ignored in calculating the length of the guard interval.

A device transmitting in a reservation block may start transmission of the preamble for the first frame at the point where it calculates the start of the reservation block to be based on its local clock. For frames that use No-ACK or B-ACK acknowledgement policy, the transmitting device shall ensure that there is enough time remaining in the reservation block to transmit the frame and allow for a SIFS plus mGuardTime before the end of the reservation block as calculated by that device.

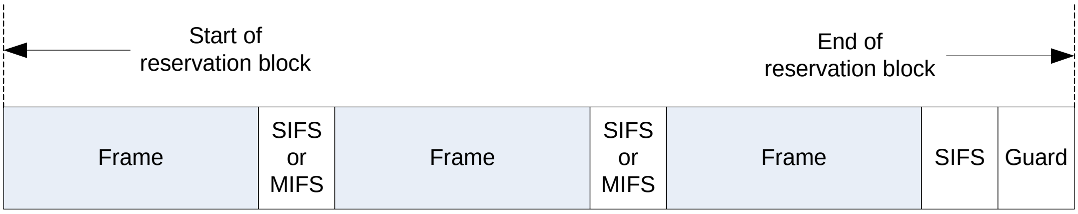


Figure 181 - SIFS and guard time in a DRP reservation block - No-ACK

If Imm-ACK is used, or a B-ACK is requested by the last frame, the transmitting device shall also ensure there is enough time for a SIFS interval, the ACK, another SIFS interval, and the guard time, as shown in Figure 182. A device shall be able to receive a frame that is transmitted within the bounds of allowable transmission, accounting for the worst case drift. A device shall begin listening mGuardTime prior to the start of a DRP reservation block, the start of a BP, or the start of a MAS in which the device announced it would be available.

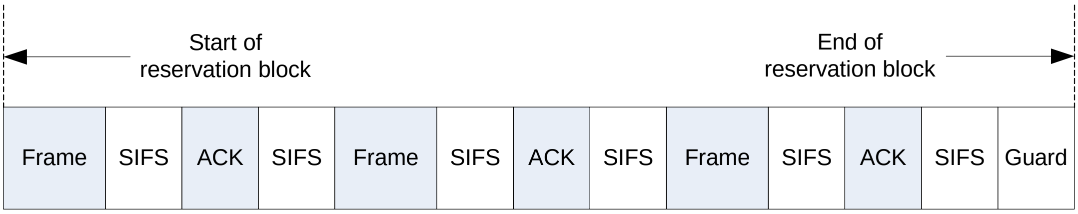


Figure 182 - SIFS and guard time in a DRP reservation block - Imm-ACK

15.9 Fragmentation and reassembly

A source device may fragment each MSDU/MCDU.

A device shall not fragment any MSDU/MCDU to more than mMaxFragmentCount fragments. Fragments may be of varying sizes. Once the MSDU/MCDU is fragmented and a transmission attempted, the device shall not refragment the frame. The device shall not create frame fragments smaller than mMinFragmentSize.

The device shall set the Fragment Number field in the first fragment to zero. It shall set each subsequent fragment to the Fragment Number field in the previous fragment plus one. The device shall not increment the Fragment Number field when a fragment is retransmitted.

A device shall assign the same Sequence Number to all fragments of an MSDU/MCDU.

The device shall completely reassemble an MSDU/MCDU in the correct order before delivery to the MAC client. The device shall discard any MSDU/MCDU with missing fragments. If the No-ACK policy is used, the recipient device shall discard an MSDU/MCDU immediately if a fragment is missing. Otherwise, a recipient device shall discard the fragments of an MSDU if the MSDU is not completely received within an implementation-dependent timeout.

If B-ACK is used, unacknowledged fragments from multiple MSDUs belonging to the same stream may be retransmitted in the same sequence. In this case it is the responsibility of the recipient device to deliver the MSDUs in the correct order to the MAC client.

If a source device discards a fragment of an MSDU/MCDU, the device shall discard all fragments of the MSDU/MCDU.

15.10 Aggregation

A transmitter may aggregate multiple MSDUs with identical Delivery ID into a single data frame. A device shall aggregate no more than mAggregationLimit MSDUs into an aggregated data frame. Two types of aggregation mechanisms are defined in this clause (1) MAC-layer only aggregation and (2) Integrated PHY/MAC aggregation. A device that uses these two aggregation mechanisms shall use the frame format as defined in 14.2.2.

A source device initiates the use of an aggregation mechanism with a recipient device for frames either from the same stream or of the same user priority. If the recipient device accepts use of the aggregation mechanism, it indicates the maximum number and size of the frames it can buffer. The source device includes a number of MSDUs in the aggregated frame, limited by the announced buffer size and maximum number of frames.

On receipt of such a frame, the recipient device returns a B-ACK frame giving feedback on the MSDUs received and indicating the buffer space available for the next aggregated frame.

A source device may invoke multiple instances of the aggregation mechanism with the same recipient device, each for a different stream or user priority. A source device may also invoke the aggregation mechanism with multiple recipient devices.

A source device may transmit an aggregated frame to any potential recipient device advertising aggregation capability in its MAC Capabilities IE. A source device shall initiate use of the aggregation mechanism by transmitting a frame with Frame Subtype set to Aggregation Request. After transmitting the frame, the source device shall follow the rules of operation as described below.

When receiving a frame with Frame Subtype set to Aggregation Request from a source device for a stream or user priority, the recipient device shall respond as follows:

- To acknowledge receipt of the frame but reject the request for starting transmission of a new aggregation frame, the recipient device shall respond with a B-ACK frame with no frame payload.
- To accept the request for starting transmission of a new aggregation frame, the recipient device shall respond with a B-ACK frame with a frame payload indicating the allowed maximum size (in frames and octets) for the next aggregation frame. The recipient shall acknowledge the received frame by indicating its reception in the acknowledgement window.

After transmitting a frame with Frame Subtype set to Aggregation Request, the source device expects to receive a B-ACK frame in response and takes one of the following actions:

- If the source device does not receive a B-ACK frame, it shall assume that the recipient device did not receive the request frame. To continue operation, the source device shall retransmit the request frame using medium access rules as described in 15.6.
- If the source device receives a B-ACK frame with no frame payload, it shall treat the transmitted frame as received and consider this use of the aggregation mechanism to be rejected.
- If the source device receives a B-ACK frame with a frame payload and with either Frame Count or Buffer Size set to zero, it shall process the acknowledgement as described below. To continue transmission of aggregated frames, the source device shall retransmit the requesting frame independently of whether the frame was indicated as received or not.
- If the source device receives a B-ACK frame with a frame payload containing non-zero values for both Frame Count and Buffer Size, then it shall process the acknowledgement as described below.

The source device processes the B-ACK frame acknowledgement as follows:

- MSDUs being held for retransmission with a sequence number earlier than the one indicated by the Sequence Control field were not received correctly from the reception of the last aggregated sequence, but shall not be retransmitted.

- MSDUs being held for retransmission with sequence and fragment number within the acknowledgement window (specified by the Sequence Control field and the Frame Bitmap field) with corresponding bit set to one were received and shall not be retransmitted.
- Other MSDUs being held for retransmission should be retransmitted in the next aggregated frame, ordered by increasing sequence and fragment numbers.

After receiving a B-ACK frame with non-zero values for Frame Count and Buffer Size, the source device may transmit a new aggregated frame. The total number of MSDUs included in the new aggregated frame must not exceed the Frame Count value specified in the B-ACK frame and the sum of the lengths of the frame payloads shall not exceed the Buffer Size value specified in the B-ACK frame. Within an aggregated frame, the MSDUs shall be ordered by increasing sequence and fragment numbers. Due to retransmissions, this ordering might not hold from one aggregated frame to the next and MSDUs transmitted in an aggregated frame might not have consecutive sequence and fragment numbers.

When the recipient device receives a frame with Frame Type set to Aggregation, it shall respond using SIFS with a B-ACK frame. To continue operation, the B-ACK frame shall contain a frame payload. If the recipient device receives a frame with a valid HCS but an invalid FCS and with ACK Policy set to B-ACK Request, the device shall also respond with a B-ACK frame with a frame payload. Within the B-ACK frame payload, the recipient device shall set the Frame Count and Buffer Size fields to limit the size of the next sequence of frames. It shall also set the Sequence Control and Frame Bitmap fields to indicate to the source device which frames should be retransmitted.

A recipient device may implement a timeout that indicates when to stop waiting for missing frames, allowing some MSDUs to be released to the MAC client and the buffer resources to be freed.

15.11 Channel bonding

A device may use channel bonding as described in 10.2.2.6 to exchange MPDU or MSDUs with devices that also support channel bonding. The device shall access the bounded channel as accessing an unbounded channel defined in this clause, except for transmission of beacon frames.

When using channel bonding, a device shall transmit beacons in a designated beaconing channel. For each bonded channel, the designated beaconing channel shall be selected as given in Table 105 only as defined in 15.5. The device shall include a CBOIE in its beacons that describes the proposed bonded channels as defined in 14.7.5.

Table 105 - Designated beaconing channels for bonded channels

Bonded Channel BAND_ID	Designated beaconing channel BAND_ID
5	2
6	3
7	3
8	3
9	3
10	3

Before accessing any of the channels, 1, 2 or 4, a device shall scan for beacons in all designated beaconing channels for which their corresponding bonded channels overlap that channel along the direction determined by the antenna training in 15.18. If a CBOIE with Status set to One is included in the received beacons during the scan, the device shall consider the channels indicated in the Bonded Channel Bitmap field of the CBOIE busy and shall not access those channels, except the designated beaconing channel.

A device shall not initiate or accept channel bonding unless the channels to be bonded, excluding the designated beaconing channel, are free of any transmission along the direction determined by the antenna training. A Type A or Type B device may consider a channel free of transmission if only Type C devices are using the channel.

A device shall not initiate or accept a new channel bonding unless the new bonded channel is identical to the existing bonded channel initiated or accepted by the device. Two bonded channel are considered identical if they are composed of the same channels.

To initiate channel bonding, a source device shall include a CBOIE that describes the proposed bonded channel with Status set to Zero in its beacon. The device should continue to include the CBOIE for at least mMaxLostBeacons+1 consecutive superframes or until a response is received.

A recipient device shall include a CBOIE in its beacon no later than the next superframe after receiving a CBOIE with Target DevAddr field matches the device's DevAddr. If the channels indicated by the Bonded Channel Bitmap field of the received CBOIE are available for channel bonding as describe above, the recipient device may set the Status to One in the CBOIE. Otherwise, the device shall set the Reason Code to an appropriate value as described in Table 96. A source device shall not change the Status in the CBOIE from Zero to One until the recipient device has set the Status to One in the CBOIE in its beacons. The recipient device shall include the CBOIE in its beacons until the source device removes CBOIE from its beacons regardless of the value of Bonding status field in its CBOIE.

15.12 Acknowledgement policies

This Clause defines three acknowledgement policies: no acknowledgement (No-ACK), immediate acknowledgement (Imm-ACK) and block acknowledgement (B-ACK).

A device shall acknowledge all received unicast frames with the ACK Policy field set to either Imm-ACK or B-ACK Request and DestAddr set to the DevAddr of this device. The device shall acknowledge the reception without regard to security validation. A device that receives a frame with a higher Protocol Version than it supports shall discard the frame without acknowledgement.

15.12.1 No-ACK

A frame with ACK policy set to No-ACK, as defined in 14.2.3.3, shall not be acknowledged by the recipient. The transmitting device MAC sublayer assumes the frame has been successfully transmitted and proceeds to the next frame upon completion of current frame. All broadcast and multicast frames shall have ACK Policy set to No-ACK.

15.12.2 Immediate ACK

On reception of a frame with ACK Policy set to Imm-ACK, a device shall respond with an Imm-ACK frame, as defined in 14.4.1**,** transmitted pSIFS after the end of the received frame.

15.12.3 Block ACK

The B-ACK mechanism allows a source device to transmit multiple frames and to receive a single acknowledgement frame from the recipient indicating which frames were received and which need to be retransmitted.

A source device initiates the use of the B-ACK mechanism with a recipient device for frames either from the same stream or of the same user priority. If the recipient device accepts use of the B-ACK mechanism, it indicates the maximum number and size of the frames it can buffer. The source device transmits a sequence of frames to the recipient, each from the same stream or of the same user priority, limited by the announced buffer size and maximum number of frames. The initial frames in the sequence are all transmitted with ACK Policy set to B-ACK. The final frame in the sequence is transmitted with ACK Policy set to B-ACK Request. On receipt of such a frame, the recipient device returns a B-ACK frame giving feedback on the frames received and indicating the buffer space available for the next B-ACK sequence.

A source device may invoke multiple instances of the B-ACK mechanism with the same recipient device, each for a different stream or user priority. A source device may also invoke the B-ACK mechanism with multiple recipient devices.

15.12.3.1 Initiation

A source device may activate the B-ACK mechanism independently for any stream or user priority traffic to any potential recipient device advertising B-ACK capability in its MAC Capabilities IE. A source device shall initiate use of the B-ACK mechanism by transmitting a frame with ACK Policy set to B-ACK Request to the recipient device. A source device shall use a dedicated Sequence Number counter for each stream or user priority traffic using the B-ACK mechanism with a recipient. After transmitting the frame, the source device shall follow the rules of operation as described in 15.12.3.2.

When receiving a frame with ACK Policy set to B-ACK Request from a source device for a stream or user priority traffic not currently using the B-ACK mechanism, the recipient device shall respond as follows:

- To acknowledge receipt of the frame but reject the request for starting a new instance of B-ACK mechanism, the recipient device shall respond with a B-ACK frame with no frame payload.
- To accept the request for starting a new instance of B-ACK mechanism, the recipient device shall respond with a B-ACK frame with a frame payload indicating the allowed maximum size (in frames and octets) for the next B-ACK sequence. The recipient shall acknowledge the received frame by indicating its reception in the acknowledgement window.

A recipient device may also accept a request to use the B-ACK mechanism even if the request frame has an invalid FCS. To accomplish this, the recipient device shall respond with a B-ACK frame with a frame payload that indicates the allowed maximum size for the next B-ACK sequence, but without acknowledgement of the frame with the invalid FCS.

A recipient device, even though it advertises B-ACK capability in its MAC Capabilities IE, may reject a request to use the B-ACK mechanism for any reason, including a temporary unavailability of resources or a lengthy setup process requiring a delayed start time. Thus, after being rejected, a source device may keep trying to initiate use of the B-ACK mechanism by sending the next frame with ACK Policy set to B-ACK Request.

15.12.3.2 Operation

After transmitting a frame with ACK Policy set to B-ACK Request, the source device expects to receive a B-ACK frame in response and takes one of the following actions:

- If the source device does not receive a B-ACK frame, it shall assume that the recipient device did not receive the request frame. To continue B-ACK operation, the source device shall retransmit the request frame with the same ACK Policy using applicable medium access rules as described in 15.6 and 15.6.

- If the source device receives a B-ACK frame with no frame payload, it shall treat the transmitted frame as received and consider this use of the B-ACK mechanism to be terminated.
- If the source device receives a B-ACK frame with a frame payload and with either Frame Count or Buffer Size set to zero, it shall process the acknowledgement as described below. To continue the B-ACK operation, the source device shall retransmit the requesting frame with the same ACK Policy, independently of whether the frame was indicated as received or not. If the requesting frame was indicated as received, the source device alternatively may transmit a zero-length payload frame with the same Sequence Control and Delivery ID to the recipient device.
- If the source device receives a B-ACK frame with a frame payload containing non-zero values for both Frame Count and Buffer Size, then it shall process the acknowledgement as described below. To continue the B-ACK operation, the source device shall send frames with ACK Policy set to B-ACK or B-ACK Request as described below.

The source device processes the B-ACK frame acknowledgement as follows:

- Frames being held for retransmission with a sequence number earlier than the one indicated by the Sequence Control field were not received in the last B-ACK sequence, but shall not be retransmitted.
- Frames being held for retransmission with sequence and fragment number within the acknowledgement window (specified by the Sequence Control field and the Frame Bitmap field) with corresponding bit set to ONE were received and shall not be retransmitted.
- Other frames being held for retransmission should be retransmitted in the next sequence, ordered by increasing sequence and fragment numbers.

After receiving a B-ACK frame with non-zero values for Frame Count and Buffer Size, the source device may transmit a sequence of frames. Each sequence of frames shall consist of zero or more frames with ACK Policy set to B-ACK followed by a single frame with ACK Policy set to B-ACK Request. The total number of frames must not exceed the Frame Count value specified in the B-ACK frame and the sum of the lengths of the frame payloads shall not exceed the Buffer Size value specified in the B-ACK frame. The sequence of frames may be transmitted in multiple DRP reservation blocks and may be interleaved with frames to other recipients or of other streams or user priorities, subject to all the medium access rules. Within a sequence, the frames shall be ordered by increasing sequence and fragment numbers. Due to retransmissions, this ordering might not hold from one sequence to the next and frames transmitted within a sequence might not have consecutive sequence and fragment numbers.

When the recipient device receives a frame with ACK Policy set to B-ACK Request, it shall respond using SIFS with a B-ACK frame. To continue operation, the B-ACK frame shall contain a frame payload. If the recipient device receives a frame with a valid HCS but an invalid FCS and with ACK Policy set to B-ACK Request, the device shall also respond with a B-ACK frame with a frame payload. Within the B-ACK frame payload, the recipient device shall set the Frame Count and Buffer Size fields to limit the size of the next sequence of frames. It shall also set the Sequence Control and Frame Bitmap fields to indicate to the source device which frames should be retransmitted.

A recipient device may implement a timeout that indicates when to stop waiting for missing frames, allowing some MSDUs to be released to the MAC client and B-ACK buffer resources to be freed. A recipient device may also implement a timeout to expire an instance of the B-ACK mechanism that appears to be inactive.

15.12.3.3 Termination

To terminate use of the B-ACK mechanism, the source device shall transmit a frame from the appropriate stream or of the appropriate user priority to the recipient device with ACK Policy set to anything other than B-ACK or B-ACK Request.

The recipient device may terminate use of the B-ACK mechanism by responding to a frame with ACK Policy set to B-ACK Request with a B-ACK frame with no frame payload.

15.13 Probe

The Probe IE and Application-specific Probe IE may be used in beacons and probe commands to request one or more IEs from the target device identified in the probe IE. Target devices are not required to respond with all requested IEs. If a target device supports the Probe command frame for one or more IEs, it shall set the Probe bit in its MAC Capabilities IE to ONE, or otherwise it shall set the bit to ZERO.

A device shall include a MAC Capabilities IE or a PHY Capabilities IE in its beacon if it is the target of a Probe IE received in a beacon that includes the MAC Capabilities IE Element ID or the PHY Capabilities IE Element ID, respectively.

On reception of either probe IE in a beacon, a target device shall include a response in its beacon for the next mMaxLostBeacons superframes.

On reception of either probe IE in a Probe command frame, a target device should respond with a Probe command frame addressed to the sender within one superframe or include a response in its beacon for the next mMaxLostBeacons superframes.

In the Probe command frame or beacon, the target device shall include:

- A Probe IE, with Target DevAddr set to the DevAddr of the requester, that includes no Requested Element IEs to reject the probe; or
- One or more requested IEs.

15.14 Multi-rate support

In device discovery (15.3) or antenna training, device shall transmit beacons or control frames using one of discovery modes as specified in 10.2.5.

In frame exchange other than device discovery or antenna training, a device shall transmit beacons using one of common PHY modes according to its device type, hence at the rate of the corresponding mode. More specifically, in a BP, a Type A device shall transmit beacon using mode-A0; while a Type B device shall transmit its beacons using both mode-A0 and mode-B0 (dual beaconing) and a Type C device shall transmit Type C Poll Frames using mode-C0.

Devices shall transmit non-beacon frames only at data rates supported by the intended recipient, based on information from the recipient's PHY Capabilities IE.

A recipient device may suggest the optimal data rate to be used by a source device, for example, to increase throughput and/or to reduce the frame error rate using explicit or implicit transmit rate control (TRC) mechanisms. For explicit TRC, the recipient sends the TPRC command frame to the source device. In addition, a source device may send a Link Feedback Request command frame to request a recipient device provide feedback on the quality of the link. The recipient sends a Link Feedback Response command frame to the source device SIFS after the reception of a Link Feedback Request command frame from the source device. For implicit TRC, the recipient includes a Link Feedback IE in its beacon. The data rate in the Link Feedback IE or the received link feedback command frame should be interpreted as the maximum data rate that the source device should use for this particular link, for an acceptable frame error rate. The source device should either follow the recommendation, or determine a data rate based on the received feedback on the quality of the link, which should not exceed the data rate recommended in the received Link Feedback command frame. The method to determine the optimal data rate in the recipient is beyond the scope of this Standard.

15.15 Transmit power control (TPC)

A device shall not transmit frames at a higher transmit power level than that used for its most-recently transmitted beacon or poll frame.

A recipient device may recommend a transmit power level change to be used by a source device using explicit or implicit TPC mechanisms. For explicit TPC, the recipient sends a TPRC command frame to the source device to recommend transmit power change. In addition, a source device may send a Link Feedback Request command frame to request a recipient device provide feedback on the quality of the link. The recipient sends a Link Feedback Response command frame to the source device SIFS after the reception of a Link Feedback Request command frame from the source device. For implicit TPC, the recipient includes a Link Feedback IE in its beacon to recommend change in transmit power. A device that receives a Link Feedback IE or a Link Feedback command frame should either follow the recommendation of change, or should determine its transmit power based on the feedback on the quality of the link in the received Link Feedback Response. The method to determine transmit power is out of the scope of this Standard, but the recipient device might use the signal to noise ratio, received signal strength, frame error ratio or other parameters to determine the transmit power change.

15.16 Power management mechanisms

This Clause specifies the power management mechanism of a device during its various operational modes. Part of this clause should be considered as informative.

15.16.1 Power management modes

A device may be in one of two power management modes during a superframe.

15.16.1.1 Active Mode

In the active mode the device sends and receives beacon frames in the current superframe. A device in active mode may switch between two power states during a superframe:

(a) Awake: device is able to transmit or receive frames.

(b) Sleep: device does not transmit or receive frames. Most of its Tx and Rx units are turned-off to save power except the timing and control units that should recover the system in relatively short time. The peer device(s) are not informed of the device transition to Sleep state.

15.16.1.2 Hibernation Mode

In the hibernation mode the device does not send or receive beacons or other frames in the current superframe; however, a device shall announce in previous superframe(s) that it plans to enter the hibernation mode.

15.16.2 Power state transitions at active mode

15.16.2.1 Power state transition from awake to sleep state

During the discovery process (device discovery, Antenna training or association), a device shall not switch to Sleep state.

Type A/B device may switch to Sleep state in the following cases:

1. After the end of BP till beginning of its first DRP reservation block in the current superframe.
2. Between two DRP reservation blocks in the current superframe
3. The device completes its DRP session in the current superframes.

Type C device may switch to Sleep state in the following cases:

1. Type C device may transient to Sleep state at the end of CEP of the current MSP until the next MSP, if the device has no CTR reservation in the DEP of current MSP.

2. Type C device may transient to Sleep state until the next MSP, if the device completes its transmission and/or reception at its last CTR in the DEP of the current MSP.

15.16.2.2 Power state transition from sleep to awake state

A device shall transient from Sleep to Awake state according to the following rules:

(1) Type A/B device shall be in the Awake state mGuardTime prior to its BPST in every superframe to participate in the transmission and reception of beacons.

(2) Type C slave device shall be in the Awake state mMasterSlaveGuardTime prior the transmission of Type C poll frame by the master device.

(3) If Type A/B device has any pending frame to be transmitted in the DRP reservations in the current superframe, it shall be in Awake state mGuardTime prior the start of each relevant DRP reservation block to start its transmission.

(4) If Type C device has any pending frame to be transmitted in DEP in the current Master-Slave Period, it shall be in Awake state mMasterSlaveGuardTime prior the DEP to start its transmission.

(5) If Type A/B device expects to receive transmissions from other devices in a DRP reservation block, as indicated in the beacons of those devices, it shall be in Awake state mGuardTime prior to the start of the reservation block for the reception of the planned transmission.

(6) If Type C device expects to receive transmissions from other device in a DEP, it shall be in Awake state mMasterSlaveGuardTime prior to the start of the relevant channel time at the DEP of the current superframe.

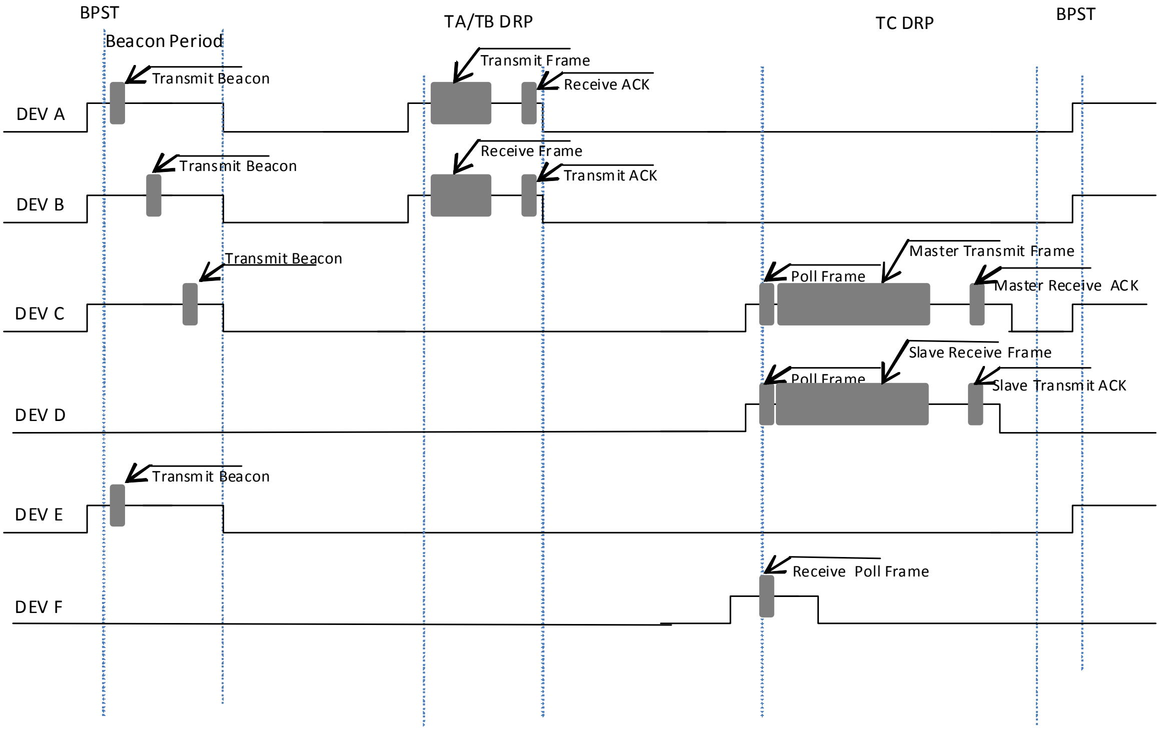


Figure 183 - Power state transition for Type A/B and Type C devices in active mode

Figure 183 illustrates the power state transition for devices in active mode.

(a) DEV A is a Type A/B device that has pending frames to be transmitted in a DRP reservation block in the current superframe.

(b) DEV B is a Type A/B device that is expecting to receive a planned transmission from DEV A in a DRP reservation block in the current superframe.

(c) DEV C is a Type A/B device that uses as Type C master device and has data traffic pending to be transmitted to the DEV D (Type C slave device) via Type C DRP in the current superframe.

(d) DEV D is a Type C slave device that is expecting to receive a planned transmission from DEV C via the Type C DRP in the current superframe.

(e) DEV E is a Type A/B device that does not have any traffic pending in its transmission queues, and is not expecting any planned transmission from other devices.

(f) DEV F is a Type C slave device that does not have any traffic pending in its transmission queues, and is not expecting any planned transmission from other devices.

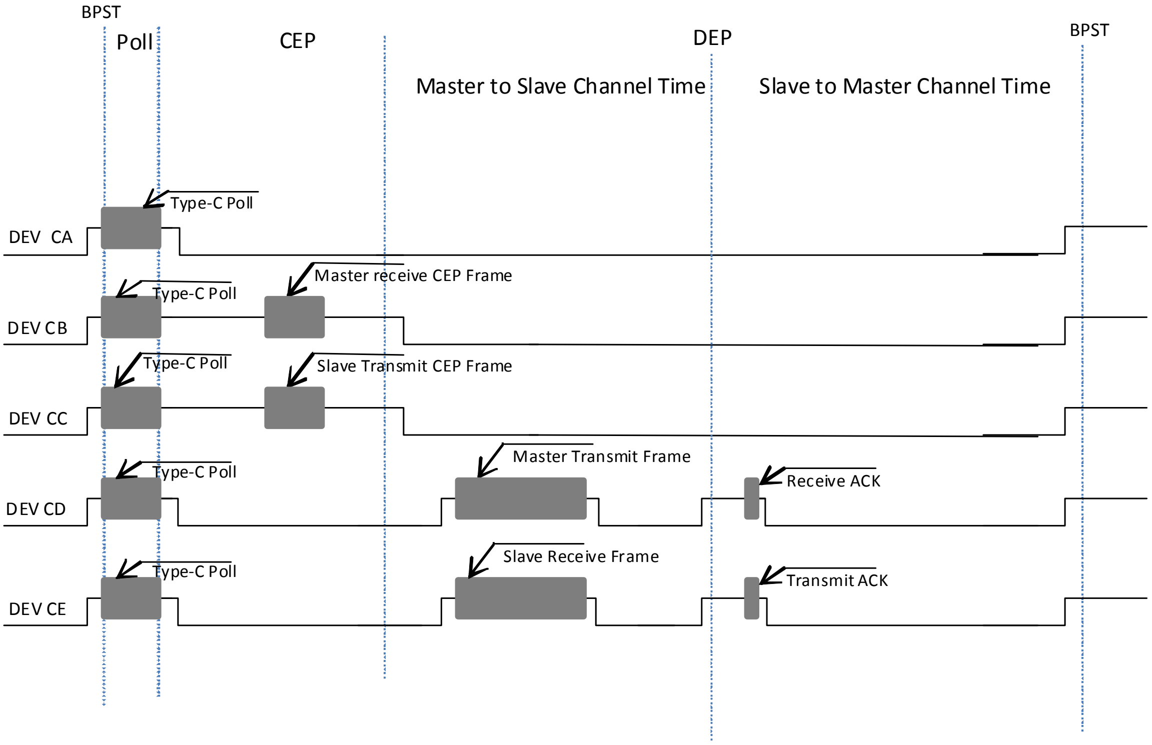


Figure 184 - Power state transition for Type C to Type C devices link in active mode

Figure 184 illustrates the power state transition for devices in active mode.

(a) DEV CA is a Type C master device that does not have any traffic pending in its transmission queues, and is not expecting any planned transmission from the TC slave device.

(b) DEV CB is a Type C master device that is expecting to receive a planned transmission from DEV CC via CEP in the current superframe.

(c) DEV CC is a Type C slave device that has data pending to be transmitted via CEP in the current superframe.

(d) DEV CD is a Type C master device that has data traffic pending to be transmitted via DEP master to slave channel time in the current superframe.

(e) DEV CE is a Type C slave device that is expecting to receive a planned transmission from DEV CD via DEP master to slave channel time in the current superframe.

15.16.3 Hibernation mode operation for Type A/B devices

Type A or Type B device using hibernation mode shall transient to and from hibernation mode according to the following rules:

(1) The device shall signal its intent to go into hibernation mode by including a Hibernation Mode IE in its beacon, as defined in 14.7.11. The Hibernation Duration field in the Hibernation Mode IE shall contain a non-zero value that specifies the duration of the hibernation period. If the device is associated to a Type C slave device, it shall terminate the Master-Slave operation with the Type C device by disassociating with the Type C slave device using the procedure as described in 15.19.6.3.

(2) The device may signal its intent to go into hibernation mode in several superframes. The value of the Hibernation Countdown field in the Hibernation Mode IE shall be set to

indicate the number of remaining superframes before the device enters hibernation mode. In each successive superframe, the device shall reduce the value of the Hibernation Countdown field by one. If this field is set to zero, the device enters hibernation mode at the start of the next superframe.

(3) During hibernation mode, the device shall not send a beacon or other frame. The device should terminate all established DRP reservations before entering hibernation.

(4) A device may leave hibernation mode prior to the end of its announced hibernation period by sending its beacon.

(5) A device in hibernation mode shall scan for beacons during the BP for one or more superframes immediately prior to the end of its hibernation period, in order to re-establish synchronization.

(6) If a device in hibernation mode finds that its former beacon slot is still available in the extended beacon group, the device may transmit a beacon in that beacon slot. Otherwise, the device shall transmit a beacon as if it was doing so for the first time.

Active mode devices in the presence of hibernation mode devices shall operate as follows:

(a) If an active mode device receives a neighbour's beacon that includes a Hibernation Mode IE, the device shall consider all DRP reservations with that neighbour to be terminated at the start of its hibernation period. An active mode device shall not commence any communication with a hibernation mode device until that device leaves hibernation mode. After receiving a beacon that includes a Hibernation Mode IE with Hibernation Countdown less than or equal to mMaxLostBeacons, an active mode device that misses the remaining expected beacons shall consider the device to be in hibernation mode as indicated in the Hibernation Mode IE.

(b) If an active mode device does not receive an expected beacon from a hibernation mode device, it shall treat the beacon slot of that device as occupied and non-movable, but shall not indicate the beacon slot as occupied by the hibernation mode device in its BPOIE, until the beacon is received, for up to mMaxHibernationProtection.

During a neighbour's hibernation period an active mode device shall continue to mark the hibernation mode device's beacon slot as occupied and non-movable in its BPOIE. If the active mode device receives another neighbour's beacon in the hibernation mode device's beacon slot, the device shall still advertise the hibernation mode device's DevAddr in its BPOIE.

(a) If an active mode device has unicast traffic for a hibernation mode device, it should buffer its traffic until the hibernation mode device enters active mode.

(b) If an active mode device has multicast or broadcast traffic it should not delay transmission of the traffic, even if it is aware that some intended recipients are in hibernation mode. It may buffer its multicast traffic for a hibernation mode device until the intended recipient enters active mode, and then deliver the buffered multicast data.

15.16.4 Hibernation mode operation for Type C device

A Type C device shall not enter Hibernation mode, nor declare itself as hibernation anchor.

15.16.5 Hibernation anchor operation in Type A/B devices

Active mode devices that are capable of acting as a hibernation anchor should indicate hibernation anchor capability in its MAC Capabilities IE. A device that indicates such capability should include a Hibernation Anchor IE in its beacon to convey information about neighbours in hibernation mode. A device may terminate its role as a hibernation anchor at any time, but at that time it should remove indication of the capability from its MAC Capabilities IE.

Devices, such as those that were recently off or in hibernation mode, may not have information about the hibernation state of their neighbours. These devices may use the information provided by Hibernation Anchor IEs for scheduling communication with neighbours in hibernation mode.

Upon reception of a beacon containing a Hibernation Mode IE in which the Hibernation Countdown is set to zero, a hibernation anchor should include a Hibernation Anchor IE. It shall set the Wakeup Countdown field in the Hibernation Anchor IE based on the Hibernation Duration field in the received Hibernation Mode IE. It shall decrement the Wakeup Countdown field in each successive superframe until the field reaches zero. After it transmits a beacon with a Hibernation Anchor IE that contains a Hibernation Mode Device Information field with Wakeup Countdown set to zero, it shall remove the corresponding Hibernation Mode Device Information field from the Hibernation Anchor IE. It shall not include a Hibernation Anchor IE if there are no Hibernation Mode Device Information fields in the IE.

If the hibernation anchor receives a beacon from a hibernation mode device prior to the end of the announced hibernation duration, the hibernation anchor shall remove the corresponding Hibernation Mode Device Information field from the Hibernation Anchor IE in the next beacon.

After receiving a neighbour’s beacon that includes a Hibernation Mode IE with Hibernation Countdown less than or equal to mMaxLostBeacons, a hibernation anchor device that misses the remaining beacons from the neighbour shall consider the device to be in hibernation mode as indicated in the Hibernation Mode IE and should include that device in the Hibernation Anchor IE.

15.17 ASIE operation

Zero or more ASIEs may be included in each beacon. ASIEs may appear within the IE area in a beacon as defined in 15.1.10. Unrecognized ASIEs shall be ignored. The format of the ASIE payload is defined by the owner of the value in the ASIE Specifier ID field and is outside the scope of this document.

15.18 The antenna training and tracking protocol (ATTP)

In order to maximise the link performance between a pair of devices for high data rate communication, fine antenna training is needed. Coarse antenna training (sector selection) takes place during the device discovery phase with low data rate transmissions, but fine antenna training is required for high data rate communication. Fine antenna training allows a pair of devices to steer their transmit and receive beam patterns in such a way that the link performance between them, and hence the achievable link data rate, are optimised.

Since there is a single data path, there are two possibilities for creating a high gain antenna with good coverage. Both possibilities involve multiple antennas of which the signals are selected or combined.

1. The device uses a switch to select one of the antennas. The combination of antennas and switch is called a **switched beam antenna** (SBA), since each antenna can be seen as covering a sector.
2. The individual antenna signals pass through adjustable amplifiers and phase shifters and are added (in receiver) or split (in transmitter) on their way to/from the AD/DA converter. This is called a **phased antenna array** (PAA).

The two possibilities can be combined: each antenna in a sector antenna array may itself be a phased array.

Sector selection in a SBA, which shall have at most 4 sectors, is part of the discovery mode protocol 15.3.

The Antenna Training and Tracking Protocol (ATTP) enables a pair of devices to train their phased antenna arrays as needed before and during communication, in order to achieve high data rate communication. The ATTP protocol has been defined to be flexible and allow freedom in implementation of specific antenna training algorithms. Some of the features of the ATTP protocol include:

- Supports iterative training for phased antenna arrays with up to 36 elements;
- Supports training amongst devices with heterogeneous antenna capabilities (in terms of antenna types, number of antenna elements, training sequence length, and so on);
- Allows training of both transmit and receive antenna weight settings;
- Allows antenna tracking to be done during data communication.

Antenna training takes place in the discovery channel after the devices have found each other, and thus selected the sector they use, through the discovery protocol 15.3. After training, the devices go to a data channel. In a data channel devices can update their antenna weight settings. This is called antenna tracking 15.18.8.

After training, the transmit antenna weight settings that a device uses depend on the intended destination and the receive antenna weight settings depend on the source. A device may maintain a list of its antenna weight settings for each partner with which the device has trained, so that it can re-use these weights when appropriate, rather than re-train its antennas from scratch.

A Type A device shall support the training of its own antenna and of the antenna of another Type A device.

A Type B device may support the training of the receive and transmit antenna weight settings of a Type A device.

Type C devices do not support antenna training.

Antenna training is done with the help of training frames that contain an antenna training sequence (ATS) 10.1.4. Upon reception of one or more training frames, a device determines 1) the receive antenna weight settings that it will use subsequently, and, if requested, 2) the transmit antenna weight settings of the other device.

For receiver training, the transmitter shall include a number of training symbols in the ATS of a single training frame. The number of training symbols shall be signalled via the NUM_RXTS field of the ATIF. These training symbols shall all be transmitted using the same transmit antenna weights.

There are two modes for transmitter training:

1. **closed loop mode:** a device determines its transmit antenna weight settings based on feedback from the other device;
2. **open loop mode:** a device uses the same antenna weight settings for transmission as for reception.

In closed loop mode, a PAA device that needs to train its own transmit array shall send a training frame in which it includes a number of training symbols at the end of the ATS that are transmitted using different transmit antenna weight settings. The training matrix 15.18.4 describes the number of these symbols and the transmit antenna weight settings that shall be used for each of these symbols.

15.18.1 ATTP functional description

During the process of device discovery, devices shall include the ACIE, which contains information about the RX and TX antenna types and the number of elements of the transmitting device, in their discovery beacons. When a device wants to initiate antenna training with another device, the RTT/CTT handshake shall be employed with the

TRAINING_CONTINUATION bit set to 0. RTT/CTT frames carry the ATIE used to initiate and negotiate antenna training. It is also used to configure the antenna training process that follows the RTT/CTT handshake.

The antenna training process is specified in Figure 185. As can be seen, the antenna training process consists of one or more consecutive iterations, as configured by the ATIE. Each iteration includes a segment for the initiator→respondent (I→R) link and one for the R→I link, since both initiator and respondent may need to train their antennas. The TRAINING (TRN) frame is described in 10.1 and 14.4.6.3. It contains an Antenna Training Sequence (ATS) which consists of the training symbols for training of receive and transmit antenna weight settings. Finally, feedback frames feeds back to the receiver on the status of the antenna procedure, which is done through the AFIE. Each frame transmission is separated by SIFS or MIFS, as indicated in Figure 185. As a TRN frame may contain training sequences for both transmit and receive antenna weight training, so a TRN frame transmitted by I can train the receiver antenna weight settings of R and (together with the following feedback) the transmitter antenna weight settings of I. Similarly, a TRN frame transmitted by R can train the receiver antenna weight settings of I and (together with the following feedback) the transmitter antenna weight settings of R.

The full antenna training procedure does not need to be completed after a single RTT/CTT exchange. In case the antenna training procedure is to be done in steps, it shall be done so in units of iterations as defined in the ATTP protocol. When device I desires to resume the antenna training procedure with device R, it shall do so by retransmission of another RTT/CTT exchange with the TRAINING CONTINUATION bit set to 1. In an antenna training procedure between two devices, the initiator device that transmits the first RTT shall also be the one that transmits any other RTTs for that same antenna training process.

Depending upon the settings of the TRAINING_DESIRED and FEEDBACK_NEEDED fields in their ATIEs, the initiator and respondent device send TRN packets containing training symbols for receiver and/or transmitter antenna weight training in the ATS. The feedback for transmitter antenna weight training (i.e., the AFIE) shall be included in a TRN frame. Table 106 shows how the TRAINING_DESIRED and FEEDBACK_NEEDED fields impact the training packets and the transmission of the AFIE.

The summarized operation of the ATTP protocol is as follows:

Step 1: When initiator device I desires to train its own antenna and/or the antenna of respondent device R, I shall transmit a RTT frame to R;

Step 2: If R is "available", it shall respond with a CTT frame addressed to initiator device S within SIFS time;

Step 3: I shall send a TRAINING frame with training sequences according to negotiated parameters;

Step 4: R shall send a TRN packet containing the AFIE within SIFS at the end of the transmission.

Steps 3 and 4 shall be repeated over one or more iterations, with at most 4 iterations for each direction.

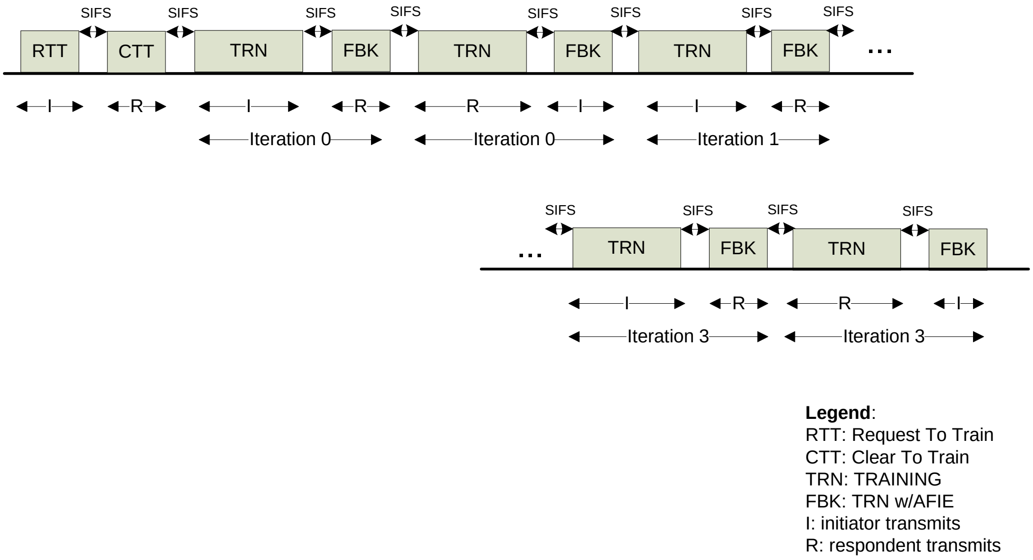


Figure 185 - The generalized ATTP frame exchange

Table 106 - Configurability of the ATTP frame exchange

		R parameters		
		TRAINING_DESIRED=1 *FEEDBACK_NEEDED=1*	*TRAINING_DESIRED=1* *FEEDBACK_NEEDED=0*	*TRAINING_DESIRED=0* *FEEDBACK_NEEDED=0/1*
I parameters	*TRAINING_DESIRED=1* *FEEDBACK_NEEDED=1*	I sends: ATS (receiver, transmitter), ACK R sends: ATS (receiver, transmitter), FEEDBACK	I sends: ATS(receiver, transmitter), R sends: ATS(receiver), FEEDBACK	I sends: ATS(transmitter) R sends: ATS(receiver), FEEDBACK
	TRAINING_DESIRED=1 *FEEDBACK_NEEDED=0*	I sends: ATS (receiver), FEEDBACK R sends: ATS(transmitter, receiver)	I sends: ATS(receiver) R sends: ATS(receiver)	R sends: ATS(receiver)
	TRAINING_DESIRED=0 *FEEDBACK_NEEDED=0/1*	I sends: ATS (receiver), FEEDBACK R sends: ATS (transmitter), FEEDBACK	I sends: ATS(receiver)	no ATS and FEEDBACK frames are sent

15.18.2 Antenna training sequence transmission and frame type

The actual antenna training sequences are transmitted as part of the TRAINING (TRN) frames depicted in Figure 185. In other words, the TRAINING frame is a regular PPDU frame in which the ATS field is used for the transmission of training sequences.

Obviously, the transmission of training sequences from a source device to a destination device is impacted by:

- The negotiated parameters in the ATIE between source and destination nodes;
- The antenna types at source and destination.

Clause 10.1 describes the PPDU frame format, which includes the ATS field. The ATS field shall be used for the transmission of training symbols employed during the antenna training process. The ATTP protocol has been defined to support dynamic antenna training and to be configurable in order to allow flexibility in implementation.

TRN frames (see 14.4.6.3) shall be used in any frame exchange for the purpose of antenna training

The ATTP parameters are given in the antenna training indicator field (ATIF) in the variable length PHY header 10.1.2.1 and in the antenna training frame IE (ATFIE) 14.7.3.3 which shall be included in training frames. Existence of the ATIF is indicated through the ATIF_EXISTENCE field in the PHY header.

15.18.3 Antenna training feedback information

The feedback information is carried in the AFIE 14.7.3.4. The AFIE shall be transmitted in a TRN frame.

For transmitter antenna weight training of a phased array in closed loop mode, the transmitter shall include a number of training symbols in a single frame. This number shall be equal to the number of columns of the training matrix 15.18.4. Each of these symbols is sent with different antenna weights, according to the training matrix. The receiver shall send a TRN frame containing the AFIE within SIFS after receiving the frame. There are several kinds of possible feedback, see 15.18.5. The type of feedback that shall be given is configured via the ATIE in the RTT and CTT control frames.

15.18.4 Training matrix in closed loop mode

A Type A transmitter with N antenna elements shall train its transmit antenna weights using a training matrix T of size N by K as defined below. The antenna weight of the n^{th} antenna element in the transmission of the k^{th} training symbol for transmit antenna training ($n = 1,\dots,N$; $k = 1,\dots,K$) shall be equal to the matrix element T_{nk}.

The complex Hadamard matrices $H(K)$, listed below, satisfy $H(K)H(K)' = KI_K = H(K)'H(K)$ and have the property that all matrix elements take values in $\{1, j, -1, -j\}$. This eases the implementation of the corresponding antenna weights. These matrices exist when K is even.

For a transmitter with N antennas, K shall be chosen follows: if $N \leq 16$, K is chosen as the lowest even number that is not smaller than N, i.e., $K = N$ if N is even and $K = N+1$ if N is odd. If $N>16$, K is chosen as the lowest multiple of 4 that is not smaller than N. Finally, the training matrix T shall be chosen as the first N rows of $H(K)$.

The following lists the complex Hadamard matrix used to construct the training matrix.

$$H(2) = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad (101)$$

$$H(6) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & j & -j & -j & j \\ 1 & j & -1 & j & -j & -j \\ 1 & -j & j & -1 & j & -j \\ 1 & -j & -j & j & -1 & j \\ 1 & j & -j & -j & j & -1 \end{pmatrix} \tag{102}$$

$$H(10) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -j & -j & -j & -j & j & j & j & j \\ 1 & -j & -1 & j & j & -j & -j & -j & j & j \\ 1 & -j & j & -1 & -j & j & -j & j & -j & j \\ 1 & -j & j & -j & -1 & j & j & -j & j & -j \\ 1 & -j & -j & j & j & -1 & j & j & -j & -j \\ 1 & j & -j & -j & j & j & -1 & -j & -j & j \\ 1 & j & -j & j & -j & j & -j & -1 & j & -j \\ 1 & j & j & -j & j & -j & -j & j & -1 & -j \\ 1 & j & j & j & -j & -j & j & -j & -j & -1 \end{pmatrix} \tag{103}$$

$$H(14) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & j & -j & j & j & -j & -j & -j & -j & j & j & -j & j \\ 1 & j & -1 & j & -j & j & j & -j & -j & -j & -j & j & j & -j \\ 1 & -j & j & -1 & j & -j & j & j & -j & -j & -j & -j & j & j \\ 1 & j & -j & j & -1 & j & -j & j & j & -j & -j & -j & -j & j \\ 1 & j & j & -j & j & -1 & j & -j & j & j & -j & -j & -j & -j \\ 1 & -j & j & j & -j & j & -1 & j & -j & j & j & -j & -j & -j \\ 1 & -j & -j & j & j & -j & j & -1 & j & -j & j & j & -j & -j \\ 1 & -j & -j & -j & j & j & -j & j & -1 & j & -j & j & j & -j \\ 1 & -j & -j & -j & -j & j & j & -j & j & -1 & j & -j & j & j \\ 1 & j & -j & -j & -j & -j & j & j & -j & j & -1 & j & -j & j \\ 1 & j & j & -j & -j & -j & -j & j & j & -j & j & -1 & j & -j \\ 1 & -j & j & j & -j & -j & -j & -j & j & j & -j & j & -1 & j \\ 1 & j & -j & j & j & -j & -j & -j & -j & j & j & -j & j & -1 \end{pmatrix} \tag{104}$$

The complex Hadamard matrix *H(K)* for K=4, 8, 12, 16, 20, 24, 28 and 32 is formed as the Kronecker product $H(K/2) \otimes H(2)$. Finally *H(36)* equals the Kronecker product

$H(6)\otimes H(6)$. Here the Kronecker product of a matrix *A* of size *p x q* and a matrix *B* of size *r x s* is the matrix $C = A\otimes B$ of size *pr x qs* with elements $C_{(k-1)r+m,(l-1)s+n} = A_{k,l}B_{m,n}$, where *k=1,...,p*, *l=1,...,q*, *m=1,...,r*, and *n=1,...,s*.

15.18.5 Feedback method in closed loop mode

After receiving the training frame containing the K training symbols, the receiver shall give feedback about the transmit antenna weight settings the transmitter should use in subsequent transmissions. As shown in Table 84, there are four feedback choices which can be grouped into two classes: index feedback and transmit weight feedback.

In index feedback, the feed back is the column of the training matrix which gave the best reception quality. This index i_{max} is encoded using 8 bits as specified in Table 108 and transmitted in the `Selected Index' field of the AFIE 14.7.3.4.

In transmit weight feedback, the feed back is the *N* transmit weights ν_i, *i* =1,...*N*. There are two ways in which the weights can be fed back. In the first method a codebook is used and in the second method each transmit antenna weight is quantised and fed back. Furthermore, for the codebook method can be further subdivided into a Fourier codebook or a Walsh codebook.

15.18.5.1 Codebook based feedback

There are two types of codebook, the Fourier codebook and the Walsh codebook. Their constructions are described below. Every Type A device shall support feedback based on the Fourier codebook, it may also support the Walsh codebook.

The transmit beamforming vector ν is estimated at the receiver side. In order to feed the knowledge of ν back to the transmitter, a vector quantization approach is used. Let ***C*** be the Fourier or Walsh codebook matrix of size *NxM* as defined below and let c_i denote the i^{th} column of ***C***. The receiver determines the index i_{max} as $i_{\max} = \arg\max_{1\le i\le M}\left|c_i^{\,H}v\right|$, i.e., i_{max} is the index of the codeword that is most correlated to the estimated beamforming vector ν. Then the binary representation of the index i_{max} shall be fed back to the transmitter in the `Selected Index' field of the AFIE, using 8 bits as specified in Table 107.

Table 107- Representation of the beamforming index

i_{max}	$b_7,...,b_0$	i_{max}	$b_7,...,b_0$	i_{max}	$b_7,...,b_0$	i_{max}	$b_7,...,b_0$
1	00000000	33	00100000	65	01000000	97	01100000
2	00000001	34	00100001	66	01000001	98	01100001
3	00000010	35	00100010	67	01000010	99	01100010
4	00000011	36	00100011	68	01000011	100	01100011
5	00000100	37	00100100	69	01000100	101	01100100
6	00000101	38	00100101	70	01000101	102	01100101
7	00000110	39	00100110	71	01000110	103	01100110
8	00000111	40	00100111	72	01000111	104	01100111
9	00001000	41	00101000	73	01001000	105	01101000
10	00001001	42	00101001	74	01001001	106	01101001

Table 107- Representation of the beamforming index (concluded)

i_{max}	$b_7,...,b_0$	i_{max}	$b_7,...,b_0$	i_{max}	$b_7,...,b_0$	i_{max}	$b_7,...,b_0$
11	00001010	43	00101010	75	01001010	107	01101010
12	00001011	44	00101011	76	01001011	108	01101011
13	00001100	45	00101100	77	01001100	109	01101100
14	00001101	46	00101101	78	01001101	110	01101101
15	00001110	47	00101110	79	01001110	111	01101110
16	00001111	48	00101111	80	01001111	112	01101111
17	00010000	49	00110000	81	01100000	113	01110000
18	00010001	50	00110001	82	01010001	114	01110001
19	00010010	51	00110010	83	01010010	115	01110010
20	00010011	52	00110011	84	01010011	116	01110011
21	00010100	53	00110100	85	01010100	117	01110100
22	00010101	54	00110101	86	01010101	118	01110101
23	00010110	55	00110110	87	01010110	119	01110110
23	00001110	47	00101110	79	01001110	111	01101110
24	00010111	56	00110111	88	01010111	120	01110111
25	00011000	57	00111000	89	01011000	121	01111000
26	00011001	58	00111001	90	01011001	122	01111001
27	00011010	59	00111010	91	01011010	123	01111010
28	00011011	60	00111011	92	01011011	124	01111011
29	00011100	61	00111100	93	01011100	125	01111100
30	00011101	62	00111101	94	01011101	126	01111101
31	00011110	63	00111110	95	01011110	127	01111110
32	00011111	64	00111111	96	01011111	128	11111111

Upon receiving the index i_{max}, the transmitter shall use the corresponding codeword $c_{i_{\max}}$ to approximate the actual beamforming vector ν.

15.18.5.2 Construction of Fourier codebook

The Fourier codebook is constructed based on a Fourier transformation matrix and a difference set. Let N be the number of transmit antennas. Compute $M = 2^{2+\lfloor \log_2(N) \rfloor} - 1$ and $K = 2^{1+\lfloor \log_2(N) \rfloor} - 1$. Define the Fourier matrix $\boldsymbol{F}_M$ of size MxM as $[\mathbf{F}_M]_{n,m} = \frac{1}{\sqrt{M}} \exp[-j2\pi(n-1)(m-1)/M]$. The Fourier codebook matrix $\boldsymbol{C}_F$ shall be chosen as the normalized NxM submatrix of $\boldsymbol{F}_M$ such that the chosen column indices are

{1,2,...,*M*} and the chosen row indices are {*r1,r2,...,r*$_N$}, where $R_K = \{r_1, r_2, ..., r_K\}$ is a modulo *M* difference set as given in Table 108, i.e.

$$[\mathbf{C}_F]_{n,m} = \frac{1}{\sqrt{N}} \exp[-j2\pi(r_n - 1)(m-1)/M] \text{ .}$$

Table 108 - Parameters of Fourier codebook matrix

N	*M*	*K*	$r_1,...,r_K$
2,3	7	3	2, 3, 5
4,...,7	15	7	1, 6, 8, 11, 12, 14, 15
8,...,15	31	15	2, 3, 4, 5, 7, 9, 13, 16, 17, 18, 24, 25, 28, 30, 31
16,...31	63	31	1, 8, 10, 12, 15, 16, 19, 23, 26, 28, 29, 31, 32, 36, 37, 38, 40, 45, 46, 48, 50, 51, 52, 55, 56, 57, 59, 60, 61, 62, 63
32,...,36	127	63	2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 16, 17, 18, 19, 21, 23, 24, 25, 30, 31, 33, 34, 35, 37, 40, 41, 45, 47, 49, 50, 56, 58, 59, 60, 61, 65, 66, 67, 69, 70, 72, 73, 76, 79, 81, 84, 89, 92, 93, 94, 97, 99, 100, 102, 106, 110, 111, 114, 115, 117, 119, 121

15.18.5.3 Construction of Hadamard codebook

The Hadamard codebook is constructed based on the Hadamard matrix $H(2^k)$ of size $2^k x 2^k$, as given in 15.18.4, with the addition that $H(64) = H(32) \otimes H(2)$. Let N be the number of transmit antennas. Compute $M = 2^{\lceil \log_2 N \rceil}$. The Hadamard codebook $\mathbf{C}_H$ shall be chosen as the *NxM* submatrix of the Hadamard matrix *H(M)* such that the chosen column indices are {1,2,3,...,M} and the chosen row indices are {1,2,3,...,N}. The *(n,m)* element of C_π is specified as $[\mathbf{C}_H]_{n,m} = \sqrt{\frac{1}{N}} \cdot [H(M)]_{n,m}$.

15.18.5.4 Quantised weights

The phases d_i and amplitudes a_i of the *N* components of the estimated transmit beamforming vector v are uniquely determined by $v_i = a_i \exp(j\pi d_i / 180)$, where $a_i \geq 0$ and $-11.25^\circ \leq d_i < 348.75^\circ$. The receiver calculates $n_i = \left[\frac{d_i}{22.5^\circ} \right]$, where [.] denotes rounding to the nearest integer. Each n_i is an integer in {0,1,,…,15}.

The feedback shall consist of *N* octets, the first four bits, $b_0,...,b_3$ of the ith octet shall comprise the 4-bit representation of n_i, as specified in Table 109, the last four bits $b_4,...,b_7$ are reserved for future use and shall be set to zero. The N octets shall be transmitted in the first 8N bits of the `Quantised transmitter weights' field, the remaining bits of this field shall be set to zero.

Table 109 - Quantised weights feedback

n_i	$b_3,...,b_0$	n_i	$b_3,...,b_0$
0	0000	8	1000
1	0001	9	1001
2	0010	10	1010
3	0011	11	1011
4	0100	12	1100
5	0101	13	1101
6	0110	14	1110
7	0111	15	1111

The following table summarizes the possible modes of feedback and indicated whether they are mandatory or optional for the transmitter, i.e., whether the transmitter shall be able to interpret that kind of feedback in closed loop mode, and the receiver, i.e., whether the receiver shall be able to give that kind of feedback when requested to do so. The source and destination indicate their `feedback capabilities' in the appropriate field of the ACIE (14.7.3.1), the transmitter decides upon a method that is supported by both. In feedback methods 2, the training matrix shall be a submatrix of the complex Hadamard matrix, as described above.

Table 110 - Summary of feedback modes

Kind of feedback	Mandatory/optional for transmitter (interpreting feedback)	Mandatory/optional for receiver (giving feedback)
index of best received training symbol	mandatory for Type A	optional for Type B
transmission weights with Fourier codebook	mandatory for Type A	mandatory for Type A
transmission weights with Walsh codebook	optional for Type A	optional for Type A
transmission weights with quantised coefficients	optional for Type A	optional for Type A

15.18.6 Implicit feedback in open loop mode

In open loop mode, devices exchange training frames for training the receive antenna weight settings and each device uses the trained antenna weight settings for transmission as well. Therefore there is no separate training of transmit antenna weight settings.

15.18.7 Iterative antenna training

Let S and D denote two stations with a PAA. The optimal antenna weight settings for station S depend on the antenna weight settings of station D, in general. Similarly, the optimal antenna weight settings for station D depend on those for station S. Therefore the training-feedback cycle is repeated a few (at most four) times, in the second to fourth iteration the transmitter and receiver use the optimal antenna weight settings from the previous iteration. The ITERATION field in the ATIF of the training frames counts the iterations. A device shall set ITERATION to 0 after the RTT and CTT exchange with TRAINING CONTINUATION bit equal to ZERO, and increment it by 1 after it has successfully received the explicit 15.18.7.1 or implicit 15.18.7.2 feedback. The iterative antenna training process ends when both S and D have completed four iterations, or when both S and D have included an AFIE with status field equal to 0 to indicate that they are done training.

The spreading factor of the training sequence is adaptive, i.e., its value can be reduced when the link quality improves. In the first iteration it shall be fixed to the maximum value, 128, while the spreading factors of the training sequence in the following iterations are computed by the receiver and fed back to the transmitter via the Requested_DISC_REP parameter in the ATFIE included in training frames in the reverse direction. Finally, the spreading factor of the training sequence in the final iteration shall be equal to 4. The adaptive spreading applies to both closed loop mode 15.18.7.1 and open loop mode 15.18.7.2.

15.18.7.1 Closed loop iterative antenna training

In this clause it is described in more detail how the ATTP protocol shall be carried out for two devices, S and D, that both indicate in their ATIE that they want training and need feedback. Here stations S and D are both Type A stations with phased antenna arrays of N_S and N_D antennas, respectively. The link from station S to station D is trained, which means that station S's transmit antenna weight settings are trained, as well as station D's receive antenna weight settings, so station S sends training frames to station D and station D sends feedback frames to station S. Also the reverse link from station D to station S, i.e., station S's receive antenna weight settings and station D's transmit antenna weight settings, must be trained, which means that station D must also send training frames to station S, and station S must also send feedback frames to station D. The training frame and feedback frame can be sent simultaneously by including the AFIE in the training frames. The following describes the training protocol that shall be carried out by devices S and D in this case.

During discovery and the RTT/CTT exchange, stations S and D shall inform each other of their antenna training parameters through the ACIE and ATIE information elements. After the RTT and CTT exchange the iterative training process starts.

In the first iteration (ITERATION = 0) station S shall send a training frame, i.e., a frame with Antenna Training Indicator Field (see 10.1.2.1.3) and Antenna Training Sequence (see 10.1.4), with NUM_RXTS=N_D training symbols for receiver antenna weight training and with spreading factor DISC_REP=128. It determines the number of columns K_S of its training matrix according to 15.18.4 and shall also include NUM_TXTS=K_S training symbols in the ATS, and it shall transmit these symbols according to the training matrix of size $N_S \, x \, K_S$ as defined in 15.18.4.

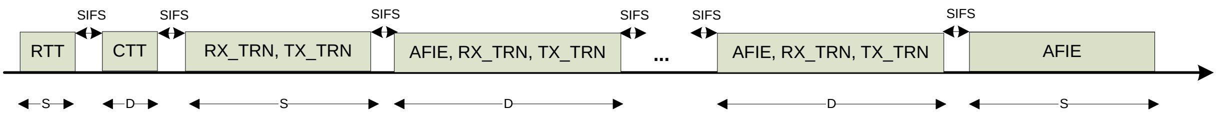
Station S shall tell station D what spreading factor station D should use in its first iteration through the ATFIE field REQUESTED_DISC_REP. In this case it makes sense to set REQUESTED_DISC_REP=128, as neither station S's receive antenna weight settings nor station D's transmit antenna weight settings have been trained yet. Station S shall inform D how many training symbols it wants for training its receive antenna weight settings through the ATFIE field REQ_NUM_RXTS. Here it makes sense to set REQ_NUM_RXTS=N_S so that it can search the full N_S - dimensional space for its optimum receive antenna weight settings when it receives the next frame from D.

Station D calculates its optimal receive antenna weight settings for station S's current transmit antenna weight settings, since it has received N_D training sequence for that purpose, and it also calculates station S's transmit antenna weight settings that are optimal for the receive antenna weight settings that station D is currently using. Within

SIFS after receiving the training frame from station S, station D shall send back a training frame, with ITERATION=0 which contains NUM_RTXS=N_S training sequences for receiver training of station S, as well as K_D training sequences using the training matrix of size N_D x K_D as defined in [clause Training Matrix in Closed Loop Mode]. Onto this training frame, station D shall also feed back the requested transmit antenna weight settings of station S (by including the AFIE) as well as the requested spreading factor for the next iteration, e.g., REQ_DISC_REP=64 and the requested number of training symbols REQ_NUM_RXTS in station S's next frame (in the ATFIE). Since station D has already some receive gain, it may choose to use a value lower than N_D. Station D shall send this frame using the spreading factor, requested by station S in the previous transmission, i.e. DISC_REP=128. Station S trains its receive antenna weight settings and calculates the transmit antenna weight settings of station D that are optimal for these receive antenna weight settings of station S.

In the second iteration (ITERATION = 1) station S shall send a training frame to station D, using the spreading factor DISC_REP=64 as requested by station D in the previous step. WIth the exception of the last symbols for transmit antenna weight training, this frame shall be transmitted using the transmit antenna weight settings that were fed back by station D in the previous iteration.This frame shall contain the requested number of training symbols for receiver training of station D, as well as KS training symbols, which shall be transmitted using transmit settings according to the training matrix. These are used for training station S's transmit antenna weight settings that are optimal for the new receive antenna weight settings of station D. Furthermore, station S includes the AFIE containing feedback to station D about station D's new transmit settings. Station D shall reply with a similar frame (with ITERATION=1), and since it now has some transmit gain and station S also has some receive gain, the spreading factor can be reduced even further.

The 3rd iteration and the 4th iteration proceed similarly as the 2nd iteration. The spreading factors and the number of training symbols for receiver antenna weight training can be reduced with each iteration as the link quality increases. There are at most four iterations. After the last iteration, station S feeds back the final transmit antenna weight settings of station D. This scenario is specified in Figure 186.

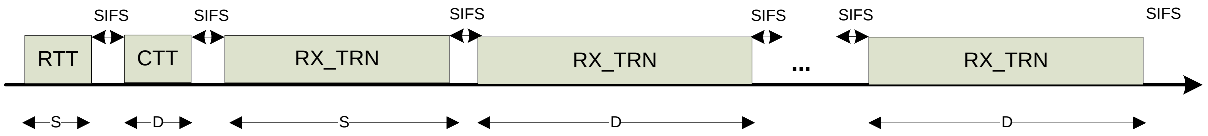


Legend:
RTT: Request To Train
CTT: Clear To Train
AFIE: antenna feedback IE
RX_TRN: Training symbols for receiver antenna weight setting
TX_TRN: Training symbols for transmitter antenna weight setting
S: device S transmits
D: device D transmits

Figure 186 - Example of closed loop training

15.18.7.2 Open loop iterative antenna training

In the open loop mode Figure 187 the TRN frame does not contain an AFIE. Instead the station S trains its receive antenna weight settings and uses the same settings in transmit mode. Again, just like in the closed loop training protocol, the spreading factor can be decreased in the course of the iterative algorithm, depending on the perceived signal strength.



Legend:
RTT: Request To Train
CTT: Clear To Train
RX_TRN: Training symbols for receiver antenna weight setting
S: device S transmits
D: device D transmits

Figure 187 - Example of open loop training

15.18.8 Antenna tracking

Antenna tracking takes place during communication between source and destination, and does not employ the RTT/CTT handshake. Whenever antenna tracking is required in either I→R or R→I direction, the ATS segment shall be included in the PPDU frame to allow for antenna tracking.

In performing antenna tracking, two options are possible:

1. The devices can use the same configuration as in the last RTT/CTT negotiation procedure for antenna tracking;
2. The devices may renegotiate the antenna training parameters for use during antenna tracking.

For case (1) above, the device shall indicate the presence of the ATS field and initiate the antenna tracking process by way of transmitting training sequences. For case (2), the process is as follows:

Step 1: When tracking is needed, the source device shall include the ATIE in the ATS field of the transmitted PPDU frame. The ATIE shall be confirmed back from the destination node on the following transmitted frame.

Step 2: Training sequences shall be transmitted within the ATS field in the following PPDU frame(s) after the ATIE negotiation between transmitter and receiver is completed.

Tracking shall use a single iteration of the training algorithm. When explicit feedback is used, and the device giving feedback is Type B, then method 1 (index), feedback shall be used; if the device giving feedback is Type A, method 2 (antenna weight) feedback shall be used. The training matrix shall be of size N x 3 and shall be chosen as $T_{nk} = W_n^{current} \exp[-j2\pi(r_n - 1)(c_k - 1)/M]$, where M and the numbers r_n are as in the Fourier codebook, clause 15.18.5, and the numbers c_k are given by $c_1 = M, c_2 = 1, c_3 = 2$.

Here $W_n^{current}$ stands for the transmit antenna weights that the transmitter is currently using. The three columns of the training matrix correspond to an angular sweep around the current antenna weights.

15.19 Master-slave operation

The communication between Type C devices uses a Master-Slave operation consisting of a Type C device operating as a Master device role, and another Type C device operating as a Slave device role. A Slave device shall associate with a Master device before using master-slave operation for data communication. Together, the Master device and the associated Slave device form a Master-Slave Pair (MSPr). A pair of Type C devices shall not form a MSPr in a channel unless both devices have scanned the channel and determined it to be unused by other devices (i.e. Type A or Type B devices).

In addition to Type C device, a Type A or Type B device may also use the Master-Slave operation to communicate with a Type C device. In this case, a DRP shall be reserved to facilitate such communication, as described in 15.19.2.1.

In a Master-Slave operation, a master device shall send Type C Poll frames to broadcast its existence, as well as information such as the timing structure of the MSP (described in 15.19.1) and channel time assignments, if any. An associated slave device communicates with the master device according to that received timing information. An unassociated slave device shall not send any frames except for association request commands.

Detailed master-slave operation procedures are described in this clause.

15.19.1 Master-slave timing structure

The period of channel time used for communication using the Master-Slave operation by a MSPr is referred to as Master-Slave Period (MSP). The length (in time) of a MSP is called the MSP duration. The length (in time) from the start of a MSP to the start of the next MSP is called the MSP interval. In most Type C to Type C device communication, the MSP interval will be the same as MSP duration, however, for Type A/B to Type C device communication (as described in 15.19.2.1) the MSP interval may be longer than the MSP duration. The maximum length of MSP Interval, mMaxMSPInterval, shall be equal to one Type A/B superframe (described in 7.2.4.4), which is the SuperframeLength.

A MSP shall consist of at least a Poll frame slot, a variable-length Command-Exchange Period (CEP) and a variable-length Data-Exchange Period (DEP), as shown in Figure 188.

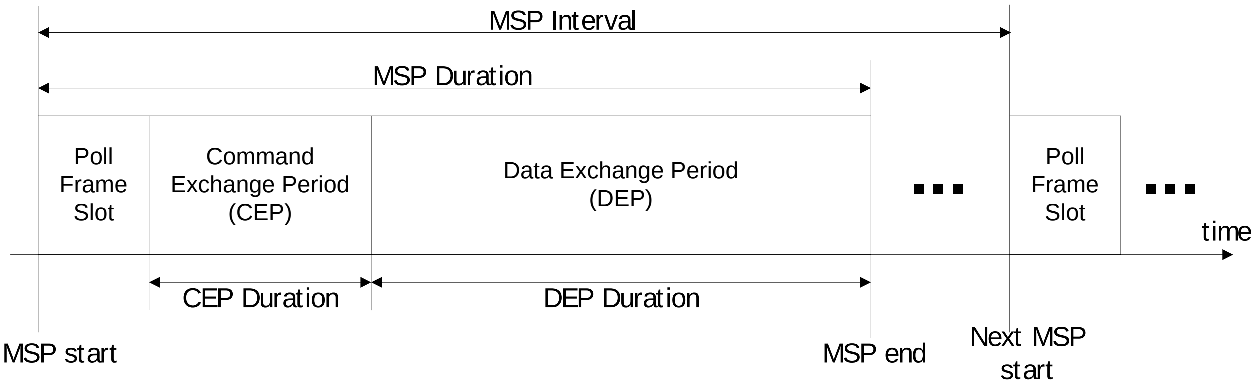


Figure 188 - Master-slave timing structure

If a Master device is already associated with a Slave device, the MSP shall consists of an additional Poll Frame Response Slot after the Poll Frame Slot as shown in Figure 189.

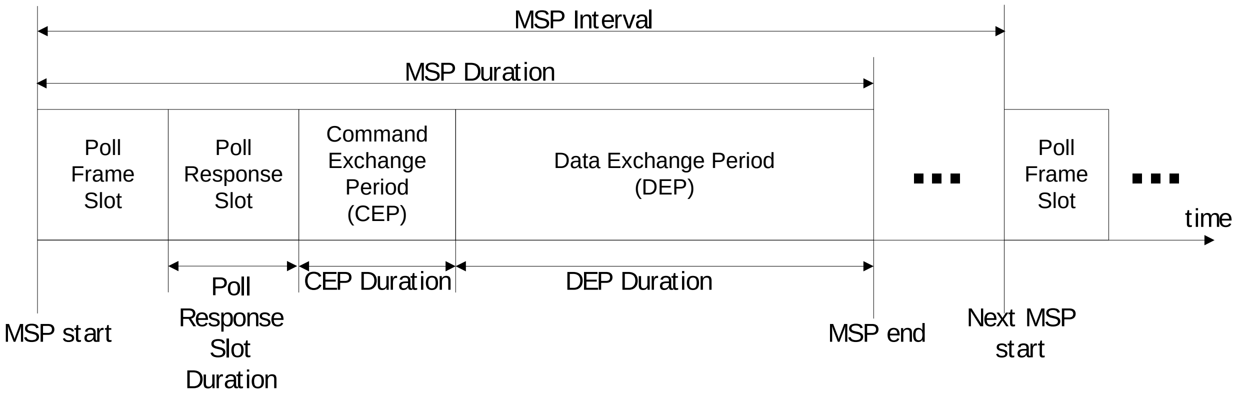


Figure 189 - Master-slave timing structure after association

The Poll Frame Slot at the start of a MSP is used by a Master device to transmit Type C Poll frame.

The Poll Frame Response Slot is used by a Slave device to acknowledge the reception of the Poll frame sent by its associated Master device. The length of the Poll Response Slot shall be:

$$\text{Poll Response Slot Duration} = \text{mTCBeaconSlotLength} + \text{SIFS\_C} \qquad (105)$$

The CEP is the period of time where Master device and Slave device may send command and data frames using Contention-based assess mechanism described in 15.19.7.2. If the Poll frame is sent by an unassociated Master device, the CEP begins immediately after the end of the Poll Frame Slot. If the Poll frame is sent by an associated Master device, the CEP begins after the end of the Poll Frame Response Slot. The length of a CEP (i.e. CEP duration) is decided by the Master device and shall be announced in the Type C Poll frame.

The DEP is the period of time where Master device and Slave device may reserve time slots called Channel Time Reservations (CTRs) to send data without contention. The DEP may contain zero, one or more CTRs as shown in Figure 190. A Master device shall include CTR information (e.g. start and end time) in its Type C Poll frame transmission.

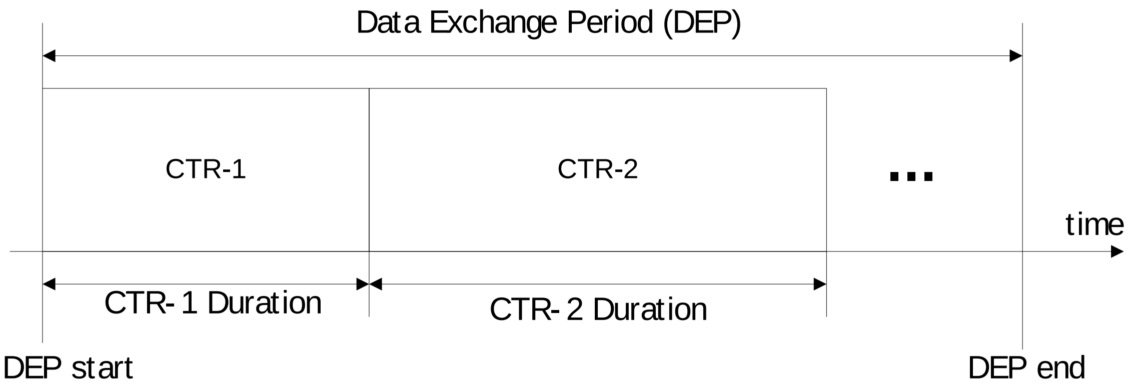


Figure 190 - Data exchange period (DEP)

An associated Slave device shall not transmit any frame in the entire duration of a MSP if it is not able to receive its Master device’s Type C Poll frame at the start of the MSP.

It shall be noted that for the sake of simple illustration, the Master-Slave timing structure with or without the Poll Frame Response Slot (i.e. Figure 189 and Figure 188 respectively) may interchangingly be used in the rest of the specifications. However, it is important to note that in all circumstances, the Poll Frame Response Slot shall be present (i.e. as in Figure 189) if the Master device is associated, and it shall not be present (i.e. as in Figure 188) if the Master device is not associated.

15.19.2 Master-slave operation between Type A, Type B and Type C devices

This clause describes the Master-Slave communication for (i) Type C device with a Type A or Type B device, and (ii) Type C device with another Type C device.

15.19.2.1 Type A/B to Type C master-slave operation

A Type A or Type B device may become a master device and start sending Type C Poll frames to manage communication with a Type C slave device.

In order to do this, the Type A/B device shall first follow the discovery procedure as specified in 15.3. To respond to a Type A/B discovery poll frame in the discovery channel, a Type C device shall start association procedure with the Type A/B device, as specified in 15.19.6.1. After discovering a Type C device, the Type A/B device shall perform channel selection procedure specified in 15.4.

To communicate with the Type C device, the Type A or a Type B device shall reserve a DRP according to clause Table 15.6. After the DRP is successfully reserved, the Type A or Type B device may begin sending Type C Poll frame with Status bit set to Polling at the beginning of that DRP. In this case, this reserved DRP is equivalent to a MSP.

A Type C device which received a Type C Poll frame with Status bit set to Polling from a master device may associate with that master device and start communication according to the procedures described in the clause 15.19.6.1 and after.

The channel timing structure of this Type A/B to Type C Master-Slave operation is shown in Figure 191.

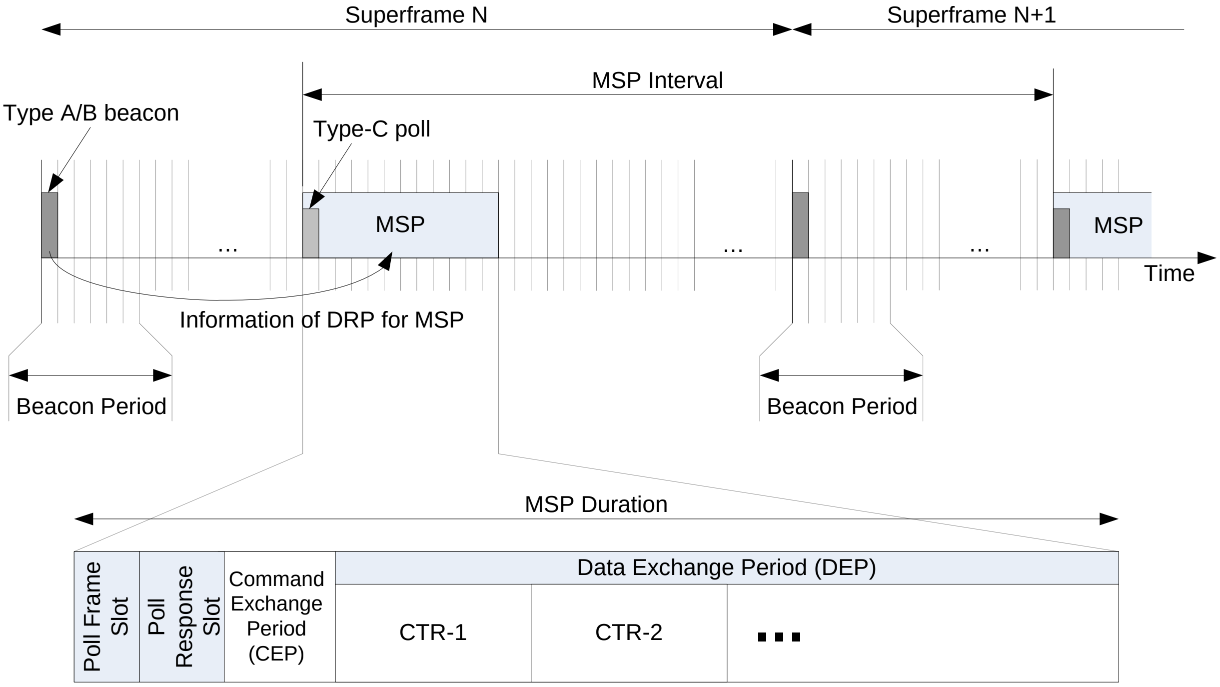


Figure 191 - Channel timing structure of the Type A/B to Type C master-slave operation

15.19.2.2 Type C to Type C master-slave operation

A Type C device may become a master device and start sending Type C Poll frames to manage communication with a Type C device if a channel is determined as empty according to the procedure described in the clause 15.19.3.

The channel timing structure of this Type C to Type C Master-Slave operation is shown in Figure 192.

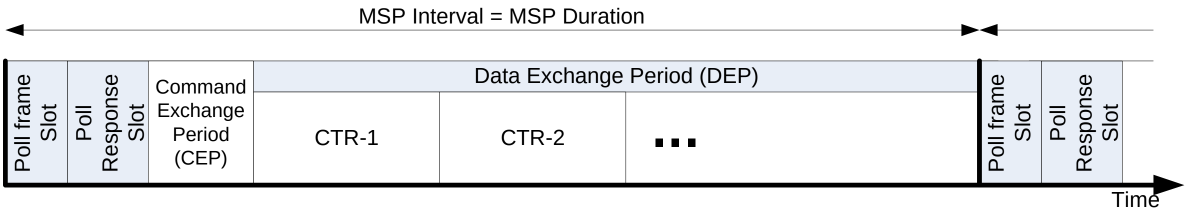


Figure 192 - Channel timing structure of the Type C to Type C master-slave operation

15.19.3 Channel scanning

Channel scanning is used to assess whether a channel is empty, and if not, whether there exists a Master device. A Master device prior to starting a Master-Slave operation shall perform channel scanning to access if the channel is empty. If the channel is not empty (i.e. used by a Type A or Type B device), it shall refrain from using that channel for Master-Slave operation.

In performing channel scanning, a device shall set itself to receive mode for a specified period of time to detect for any transmission. A Master device is considered detected when a Type C Poll frame sent by that Master device is received. In addition to searching for Type C poll frames, the searching device may optionally collect statistics of the channel and save them in a ChannelRatingList (as described in 14.8.13). Figure 193 below shows the MSC of the channel scanning procedure.

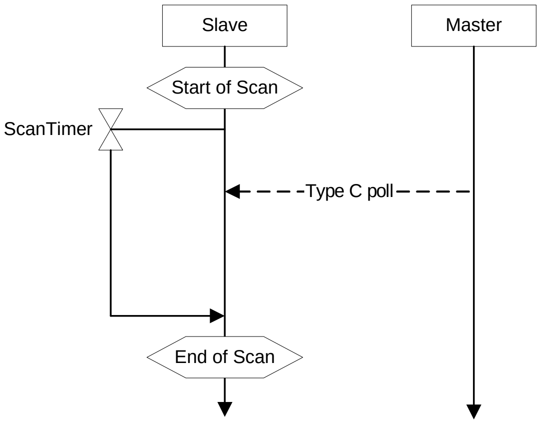


Figure 193 - MSC for channel scanning

15.19.4 Starting a MSPr

Prior to starting a MSPr, a device shall has recently completed a channel scan for Master device detection described in 15.19.3, and will have selected a channel in which to start its Master-Slave operation. A device shall not start a MSPr in a channel if it detects the channel is busy. The device should select the channel with least amount of interference based on last scanning result. To start a MSPr, a device shall initialize itself to become a Master role and begin transmitting Type C Poll frame with Status bit set to Polling, once every Master-Slave Period (MSP). The Type C Poll frame shall be sent using mode-C0. After starting to transmit Type C Poll frames with Status bit set to Polling, the Master device shall wait for a Slave device to initiate association with it using the association procedure described in 15.19.6.1.

15.19.5 Terminating a MSPr

To terminate a MSPr, a Master device may use the Disassociation procedure described in 15.19.6.3 with a Reason code field of MSPr Termination.

In some situations, a Master device may not have sufficient time to perform the above-mentioned disassociation procedure. If so, a Master device may terminate a MSPr by stopping its Type C Poll frame transmissions. Slave device that are associated with the Master device shall be able to determine the loss of MSPr after mMaxLostPolls number of MSPs where no Type C Poll frames from the Master device are received.

15.19.6 Association and disassociation

An association between a Master device and a Slave device shall be established before the Master device and the Slave device can perform Master-Slave operations such as data communication. Upon the completion of the necessary Master-Slave operations, the Master-Slave association may be disassociated to reduce power consumption and potential interference to surrounding devices.

15.19.6.1 Association

Prior to association, an unassociated Slave device should have previously received a Master device's Type C Poll frame transmission and hence determined the Master device's MSP timing information such as the MSP duration, CEP duration etc. Before association, any frame sent by an unassociated Slave device shall be transmitted in the CEP of a MSP whose Type C Poll frame was successfully received. To initiate the association procedure, the unassociated Slave shall send an Association Request command, as described in 14.8.3, during the CEP. The Slave device shall also start an AssocTimeout timer. The duration of AssocTimeout shall be determined by upper layer and will be communicated to the Master device via the Association Timeout field in the Association Request command, as described in 14.8.3. The Association request command shall only be sent to an unassociated Master device. The association state of a Master device shall be indicated in the Associated field in the Poll frame transmitted by the Master device, as described in 15.19.8.2.

Upon the successful reception of the Association Request command, the Master device shall acknowledge by sending an Imm-ACK back to the requesting Slave device and shall determine (the determination procedure and/or criteria are beyond the scope of this standard) whether the requesting Slave device shall be permitted to be associated. The Master device shall then send the Association Response command with the appropriate reason code field as described in 14.8.4, back to the requesting Slave device during the CEP to inform the Slave device of the outcome of the association request.

Upon the successful reception of the Association Response command, the requesting Slave device shall acknowledge the successful reception of the Association Response command by sending an Imm-ACK back to the Master device and stop its AssocTimeout Timer. If the reason code in the received Association Response command indicates a successful association, the requesting slave device shall switch its association state from Unassociated to Associated.

Upon the reception of the Imm-ACK for its previously sent Association Response command, the Master device shall consider the Association procedure to be completed. If the reason code field in the previously sent Association Response command indicates a successful association, the Master device shall regard itself as associated with the requesting Slave device.

During the association procedure, if the requesting Slave device received an AssocTimeout timer expiry notification, it shall abort the association procedure and remain in the Unassociated state. Figure 194 shows the MSC for a successful association procedure.

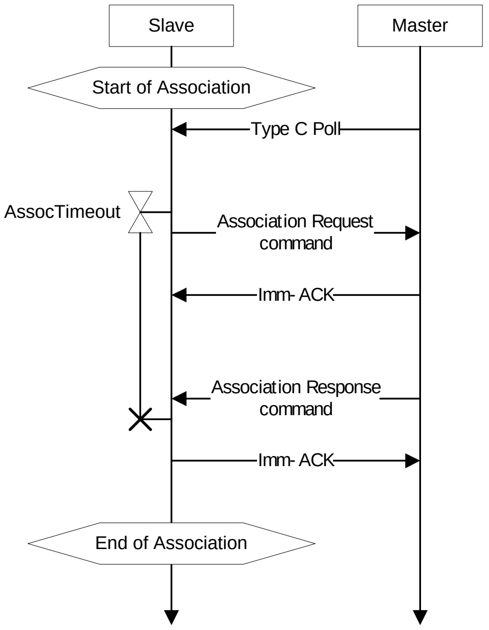


Figure 194 - MSC for association procedure.

After being associated, a Slave device shall always use the MSP timing information in the Type C Poll frame sent by the Master device to determine the start of the next MSP. In addition, it shall acknowledge all received Type C Poll frames by sending a Type C Poll Response frame in the Poll Frame Response Slot as described in 15.19.8.3.

After being associated, a Master device shall indicate its association state using the Associated field in the poll frame as described in 15.19.8.2.

After Master device have completed association with a Slave device, the MSP timing structure with Poll Response slot shall be used, as described in 15.19.1.

15.19.6.2 MSPr services

A Master device may optionally support MSPr Services information notification to a newly associated Slave device. In order to request Master device to send MSPr Services information, a Slave device shall set the MSPr Services inquiry bit in the Association Request command during the Association procedure (as described in 15.19.6.1) with the Master device.

After an Association procedure is completed between a Master device and a Slave device, if the MSPr Services inquire bit was set in the Association Request command, the Master device, if supported, shall send an MSPr Services command (described in 14.8.5) during the CEP to the Slave device. Upon reception of the MSPr Services command, the Slave device shall acknowledge the successful reception by returning an Imm-ACK to the Master device. In addition, the Slave device may forward the MSPr Services information to its upper layer. Figure 195 shows the MSC of a successful MSPr Services information notification.

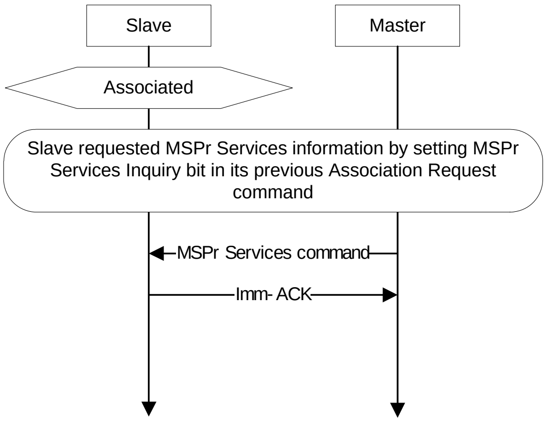


Figure 195 - MSC for MSPr service information notification procedure.

15.19.6.3 Disassociation

To terminate Master-Slave operation, the Disassociation procedure is used to disconnect the Master device and the associated Slave device from the MSPr. The disassociation procedure can be initiated by either Master device or Slave device by sending a Disassociation command and setting the appropriate reason code field.

In most of the case, a disassociation results from a normal completion of services in the upper layer. In this case, the reason code field shall be set to Normal Completion.

Another reason for Master device initiated disassociation may be due to MSPr termination, as described in 15.19.5. In this case, the Reason code field used for disassociation shall be set to MSPr Termination.

For the case of a Type C device, if it receives a Channel Release request, as described in 15.19.13 from a Type A or Type B device, it shall prepare to stop MSPr operations in the current channel. In such situation, the Type C device may, depending on available time in the current CEP, initiate disassociation to stop the Master-Slave operation as well as to inform the associated peer device of the requested Channel Release request. In this case, the Reason code field of Channel Release shall be used.

To initiate disassociation, the initiator device shall send a Disassociation Request command (described in 14.8.6) during the CEP to the associated target device. The Disassociation request command shall be sent with No-ACK specified as the ACK policy in the MAC header as described in 14.2.3.3. Upon the reception of a Disassociation Request command, the target device shall regard itself as disassociated with the requesting initiator device. Upon sending the Disassociation request command, the initiator device shall be considered as disassociated with the Target device. Figure 196 shows the MSC for a disassociation procedure.

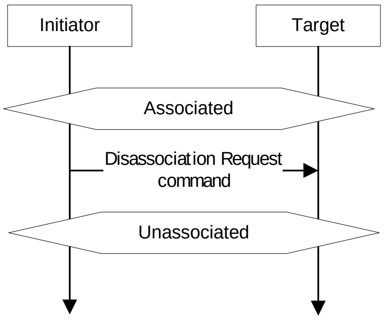


Figure 196 - MSC for disassociation procedure.

Upon disassociation, a Slave device shall not send Type C Poll Response for any Type C Poll frame it receives.

Upon disassociation, a Master device shall indicate itself as disassociated using the Associated field in the Poll frame as described in 14.8.1.

15.19.7 Channel access

The channel time is divided into MSP with each MSP beginning with a Type C Poll frame transmitted by a Master device. The MSP consists of at least 3 parts: A Poll frame slot for Type C Poll frame transmission by the Master device, a Command Exchange Period (CEP) and a Data Exchange Free Period (DEP). A MSP may additionally consist of a Poll Response Slot if the Master device is associated, as described in 15.19.1. A Master or Slave device may transmit command or data frame during the CEP period asynchronously according to the clause 15.19.7.2. In order to transmit data frames without contention during the DEP, a Slave device shall request for channel time reservations (CTRs) from the Master device beforehand as described in clause 15.19.7.3.

15.19.7.1 Interframe space (IFS)

The Master-Slave operation uses the same Interframe space as described in clause 15.1.6.

15.19.7.2 Channel access in CEP

Basic Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) contention based method is used for channel access during the CEP. The basic mechanism in the CSMA/CA used here involves a transmitting device to first sense the medium to be idle for a random length of time before transmission. This process of waiting for idle medium before transmission is termed as backoff. Device shall use the Clear Channel Assessment (CCA) capability of the PHY to assess whether the channel is busy or idle.

With exception of Imm-ACK frame, all frames sent during the CEP shall use the below described backoff algorithm.

The Backoff algorithm uses parameters in Table 111.

Table 111 - Backoff algorithm parameters

Parameter	Description	Value/range
retry_count	Keep track of the number of times the current frame is being retransmitted	0 - mMaxRetryCount
Max_backoff	An array of maximum backoff value corresponding to number of retry (i.e. retry_count)	[3, 5, 7]

Table 111 - Backoff algorithm parameters (concluded)

Parameter	Description	Value/range
pBackoffSlot	The length of time for each backoff slot. It is a PHY dependent parameter (clause 11.2) that is based on the amount of time it requires for PHY to sense the channel.	pSlotTime
backoff_counter	A backoff counter of integer value selected randomly over a uniform distribution from a given range. The range of values selectable depends on the current retry_count value.	If retry_count < 3: Selection range = [0, Max_backoff(retry_count)] Otherwise: Selection range = [0, Max_backoff(2)]

The following are the steps of the backoff algorithm:

1. retry_count is set to 0.
2. backoff_counter is selected randomly from [0, Max_backoff(retry_count)]. If retry_count is greater than 2, the ranged of [0, Max_backoff(2)] shall be used instead.
3. Transmitting device performs CCA for a duration of pBackoffSlot to assess whether the channel is busy.
4. If the channel is busy, transmitting device shall suspend the backoff_counter and assess the channel to be idle for SIFS before resuming the backoff_counter (go back to step 3).
5. If the channel is idle, transmitting device shall decrement the backoff_counter if it is greater than 0. If however, the backoff_counter has already reached 0, the transmitting device shall transmit the current frame (if there is time left in the CEP to do so), increment retry_count and go to step 6. If there is not time left in the CEP to transmit the current frame, the transmitting device shall not transmit the current frame and wait for the start of next CEP and go to step 3.

NOTE
In determining whether there is sufficient time for transmission of the current frame, applicable time required for ACK reception and IFS shall be factored in as well.

6. If No ACK was expected, the transmitted frame shall be considered as received successfully. The transmitting device shall then exit the backoff algorithm.
7. If an Imm-ACK was expected and was received within SIFS, the transmitted frame shall be considered as received successfully. The transmitting device shall then exit the backoff algorithm.
8. If however, an Imm-ACK was expected but was not received within SIFS, the transmission shall be considered to have failed. If retry_count has not reached pMaxRetryCount, the transmitting device shall perform retransmission (go back to step 2). pMaxRetryCount is the maximum number of times MAC will attempt to transmit a frame. The value for pMaxRetryCount shall be implementation dependent and shall be communicated by the upper layer to the MAC. If retry_count has reached pMaxRetryCount, the current frame shall be considered as failing all transmission attempts. The transmitting device shall exit the backoff algorithm and report to upper layer of the transmission failure of the current frame.

The backoff counter shall be suspended outside the CEP or that there is insufficient time remaining in the CEP for the transmitting device to send the frame (including time for receiving device to return an Imm-ACK should the transmitting device expects it). At any time during the backoff algorithm, if the transmission timeout specified for the frame expires, the backoff algorithm shall be exited and transmitting device shall abort the attempted transmission.

15.19.7.3 Channel time management in DEP

The Data Exchange Period (DEP) shall be used for sending frame without contention. Prior to sending frames in the DEP, the channel time used for sending frames (including the required time for IFS and for Imm-ACK transmission, if any) must be reserved. Channel time in DEP may be reserved by either Master device or Slave device. Once a channel time reservation is completed successfully, information of the channel time reservation such as start and end time, shall be included in every Type C Poll frame. The reserved channel time shall remain reserved until the reservation is terminated or that the Slave device is disassociated.

15.19.7.3.1 Channel time reservation creation

Either the Master device or Slave device can initiate a channel time reservation. Both the Master device and the Slave device use the same procedure to create a channel time reservation (CTR).

The initiator device, requests channel time reservation by sending a Channel Time Request command described in 14.8.7 to the target device.

The initiator device shall include a RequestTimeout value in the Channel Time Request command, as specified in 14.8.7, to indicate the maximum amount of time that the target device is given to respond to the request. The Channel time reservation procedure shall be considered to have failed if there is no response is received after RequestTimeout length of time.

Upon receiving a Channel Time Request command, the target device shall acknowledge the request by sending an Imm-ACK back to the initiator device. The target device shall respond using a Channel Time Response command (described in 14.8.8) with a result code containing the outcome of the reservation.

Upon the reception of the Channel Time Response command, the initiator device shall acknowledge by sending back an Imm-ACK regardless of the outcome of the reservation.

If the outcome of the reservation is successful, the master device shall include the channel time reservation information in its Type C Poll frame starting from the next MSP.

After a successful CTR reservation, the initiator device may begin to transmit frames in the reserved CTR of the following MSPs that has the CTR information contained in the corresponding Type C Poll frame of that MSP. Figure 196 and Figure 197 show the MSC of the Master device initiated and Slave device initiated CTR reservation request procedure respectively.

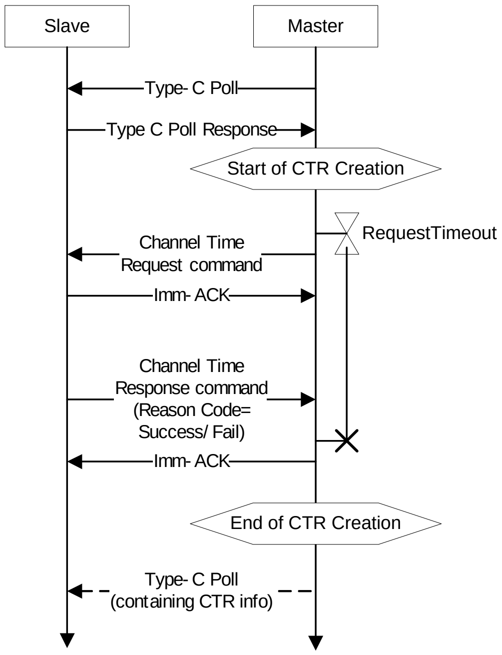


Figure 197 - MSC of Master device initiated channel time reservation request procedure

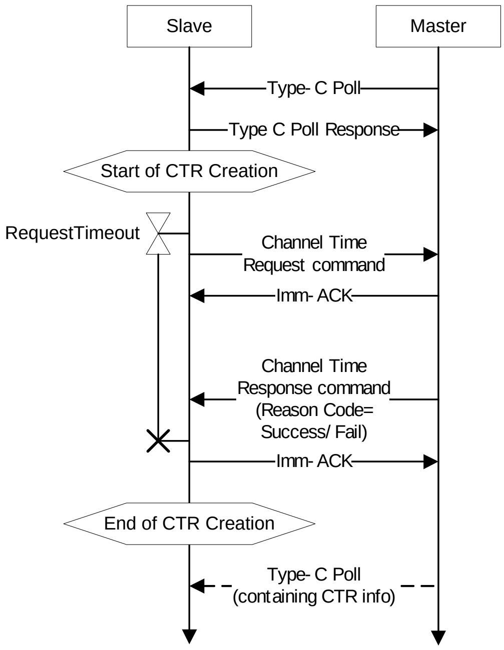


Figure 198 - MSC of slave device initiated channel time reservation request procedure.

15.19.7.3.2 Channel time reservation termination

A CTR may be terminated by either the Master device or Slave device.

The master device shall terminate a CTR by removing its corresponding CTR IE from the Type C Poll frame starting from the next MSP. Upon the reception of a Type C Poll frame without its CTR IE, a Slave device shall be notified of the CTR termination. Figure 199 shows the MSC of the Master device initiated channel time reservation termination procedure.

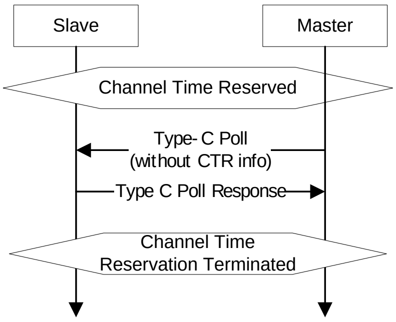


Figure 199 - MSC for Master device initiated time reservation termination

To terminate channel time reservation, a Slave device shall send a Channel Time Termination command (described in 14.8.9) to its Master device. Upon the reception of a Channel Time Termination command, a Master device shall acknowledge with an Imm-ACK and proceed to de-allocate the necessary resource starting from the next MSP. Once the Slave device receives the Imm-ACK acknowledgement for its Channel Time Termination command, the termination procedure shall completed. After the channel time reservation is terminated, the Master device shall remove the corresponding CTR IE from its Type C Poll frame starting from the next MSP. Figure 200 shows the MSC of the Slave device initiated channel time reservation termination procedure.

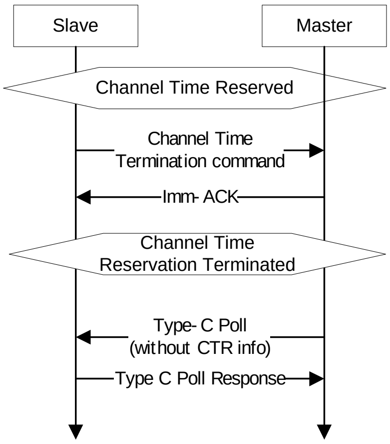


Figure 200 - MSC for slave device initiated channel time reservation termination.

15.19.7.3.3 Channel time management

After a CTR is created, a Master device or a Slave device may modify the reservation to lengthen or shorten the reservation.

To modify channel time reservation, the initiator device shall send a Channel Time Modify Request command described in 14.8.10 to the target device. The initiator device shall include a RequestTimeout value in the Channel Time Modify Request command, as specified in 14.8.10, to indicate the maximum amount of time the target device will be given to respond to the request. The Channel time Modify reservation procedure shall be considered to have failed if there is no response is received after RequestTimeout length of time.

Upon receiving a Channel Time Modify Request command, the target device shall acknowledge using Imm-ACK, and responds accordingly using a Channel Time Modify Response command (described in 14.8.11) with a result code containing the outcome of the modification.

Upon the reception of the Channel Time Modify Response command, the initiator device shall acknowledge by sending back an Imm-ACK regardless of the outcome of the modification.

If the outcome of the modification is successful, the master device shall include the corresponding modified CTR IE in its Type C Poll frame starting from the next MSP. Figure 201 shows the MSC of a channel time reservation modification procedure.

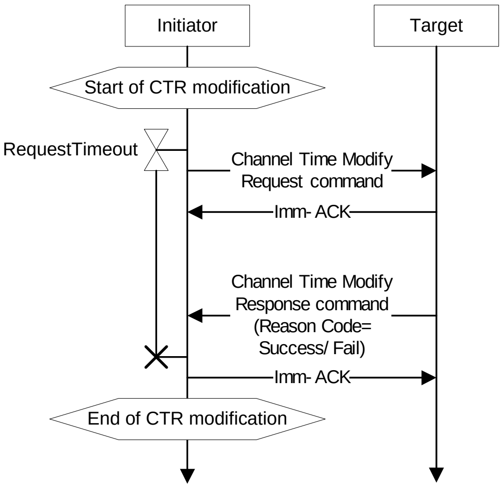


Figure 201 - MSC for channel time reservation modification procedure.

15.19.7.3.4 DEP duration modification

In the described procedures of Channel time reservation creation (in 15.19.7.3.1), and Channel time management (in 15.19.7.3.3), both Master device and Slave device shall check their own resources prior to creating or modifying a CTR. In some cases, a Master device may try to extend the DEP duration if there is insufficient time remaining in the DEP to fulfil the Master devices's or a Slave device's CTR request.

If a Master device receives a Channel Time Request command frame or a Channel Time Modification Request command frame from Slave device and finds that there is insufficient time remaining in the DEP, a Master device may respond to the Slave device with a Channel Time Response command frame or a Channel Time Modification Response command frame with Reason Code field is set to Wait until DEP modification.

After sending the Channel Time Response command frame or Channel Time Modification Response command frame with Reason Code of Wait until DEP modification, the Master device shall extend the DEP based upon the device type. If the Master device is Type A or Type B device, the Master device shall modify its own DRP according to the clause 15.6.8. If Master device is Type C device, Master device shall modify its MSP interval according to the clause 15.19.12.2.

After receiving a Channel Time Response command frame or a Channel Time Modification Response command frame with Reason Code of Wait until DEP modification, a Slave device shall wait for several MSPs and may send a Channel Time Request command frame or a Channel Time Modification Request command frame to Master device again.

15.19.8 Synchronization

To maintain time synchronization, a Master device shall transmit Type C Poll frame at the start of every MSP. The Type C Poll frame shall contain timing information for Slave device to synchronize its MSP timing with the Master device. At the start of a MSP, the Master device shall start transmitting Type C Poll frame and reset its MSP start time based on the start of transmission of its Type C Poll frame. On the reception of a Type C Poll frame, a Slave device shall reset its MSP start time based on the beginning of the Type C Poll frame preamble. In addition, the Type C Poll frame shall contain timing information such as Next MSP start time and CEP duration. If a Slave device does not receive Type C Poll frame in a particular MSP, it shall reset the MSP time to 0 at the point where it expects to hear the Type C Poll frame.

A period of time equals to mSIFS shall be allocated at the end of a Poll frame slot.

To allow for device transmission and reception turnaround, if a Poll Response slot is present, a mSIFS period shall be allocated at the end of the Poll Response frame transmission. Similarly, a mSIFS period shall be allocated at the end of the CEP, as well as at the end of every CTR.

Additionally to safe guard against clock drift, Master device shall allocate a period of time equals to mMasterSlaveGuardTime between the last CTR and the end of the DEP. mMasterSlaveGuardTime shall be calculated as:

$$\text{mMasterSlaveGuardTime} = \text{mMaxMSPInterval} * \text{mClockAccuracy(ppm)} * 10\text{-}6 * 2 = 655.36\text{ns} \qquad (106)$$

Figure 202 below shows the detailed MSP timing relative to the start of Type C Poll frame when there is a Poll Response Slot.

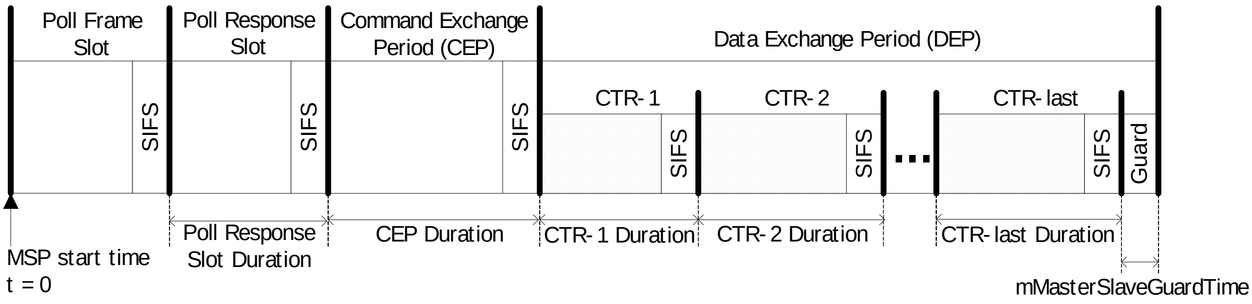


Figure 202 - Detailed MSP (Poll Response Slot) timing relative to start of Type C poll frame

Figure 203 below shows the detailed MSP timing relative to the start of Type C Poll frame when there is a no Poll Response Slot.

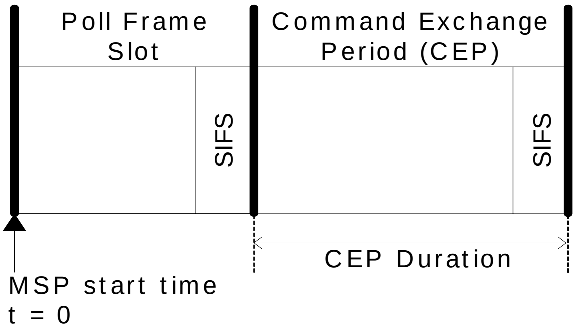


Figure 203 - Detailed MSP (no poll response slot) timing relative to start of Type C poll frame

15.19.8.1 Timer accuracy

A compliant implementation shall maintain the accuracy of the timer at no less than mClockAccuracy.

15.19.8.2 Type C poll generation

Master device shall schedule and transmit Type C Poll frame at the start of every MSP such that the time between successive Type C Poll transmission equals to the announced MSP Interval with an error of no more than mClockAccuracy times the MSP interval.

A Type C Poll frame shall include a MSP timing IE containing the timing information of the MSP.

The Type C Poll frame shall indicate in every Poll frame, the current Association state of the Master device using the field Associated as described in 14.8.1.

The Type C Poll frame shall indicate in every Poll frame, the status of the Master device in the Status bit specified in 14.8.1. If a Type A/B device is sending out the Poll frame for the purpose of discovering Type C devices, it shall set the Status bit to Discovery. If a Type A/B/C device is sending out the Poll frame for the purpose of setting up an MSPr for communication with a Type C Slave device, it shall set the Status bit to Polling.

15.19.8.3 Type C poll reception

An associated Slave device shall use the received time of Master device's Type C-Poll frame to determine the current MSP start time. Using the MSP Interval information in the MSP timing IE included in the Type C Poll frame, Slave device shall be able to determine the expected Type C Poll frame transmission time for the next MSP and hence shall schedule for Type C Poll frame reception for the next MSP. To avoid missing the reception of the Poll frame due to faster Master device's clock, and/or due to slower Slave device's clock, Slave device shall schedule to listen for the expected Poll frame mMasterSlaveGuardTime, as described in 15.19.8, before the expected Poll frame transmission time.

The Type C Poll frame may additionally contain timing information for CTRs, if any, within the DEP and these information shall be used by Slave device to determine the start and end time of any CTRs reserved.

As long as a slave device is associated with a Master device, it shall always send a Type C Poll Response frame back to the Master device in the Type C Poll Response Slot after receiving a Type C Poll frame from the Master device. The Type C Poll Response frame shall include the MSP Timing information and shall always be sent using mode-C0. Figure 204 shows the MSC for a Slave device sending a Type C Poll Frame Response after receiving a Type C Poll frame from its associated Master device.

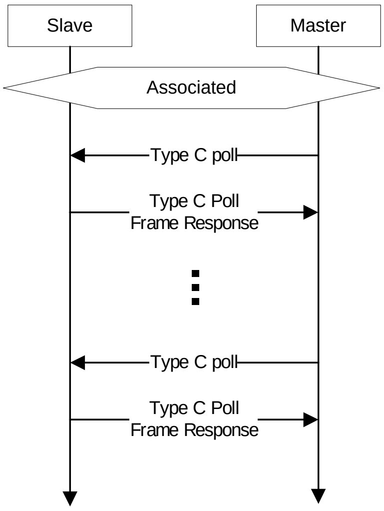


Figure 204 - MSC for Type C poll frame response

15.19.8.4 Link loss detection

If no Type C Poll Response frame is received from an associated Slave device for consecutive mMaxLostPolls Type C Poll frames, the Master device shall regard the Slave device as disassociated due to link loss by updating the Associated field (as described in 14.8.1) in its Poll frame transmission.

On the other hand, if a Slave device does not receive any Type C Poll frames from its associated Master device for consecutive mMaxLostPolls MSPs, the Slave device shall regard itself to be disassociated from the Master device due to link loss.

15.19.9 Fragmentation and de-fragmentation

Fragmentation and de-fragmentation shall not be performed within the MAC. Implementer shall only send MSDU with maximum size of pMaxFramePayloadSize through the MAC SAP. Fragmentation and de-fragmentation, if any, shall be implemented above the MAC for Master-Slave operations.

15.19.10 Acknowledgment and retransmissions

Master-Slave operations shall support a subset of the ACK scheme (described in 15.12) consisting of No ACK (described in 15.12.1) and Immediate ACK (described in 15.12.2).

The Imm-ACK frame shall be returned by the destination device within a time equals to mSIFS. If no Imm-ACK frame is received after mSIFS, the sender of the frame shall retransmit the frame. A frame shall only be allowed to be retransmitted up to mMaxRetryCount times. To improve the system performance over the multipath channels, retransmitted frames shall be transmitted using a bit-reversal scheme in the PHY as described in 10.4.6.1.3. In the bit-reversal scheme for a retransmission frame, the manner that the bits in the frame are to be reversed and scrambled, is based on the retransmission times (i.e. the number of times the frame has been retransmitted including the current attempt) of the frame. Thus, information on the retransmission times of the frame shall be made available to the PHY which will perform the bit-reversal scheme as described in 10.4.6.1.3. Various methods may be used to provide PHY with the necessary retransmission times of a retransmission frame. For example, a compliant device may be implemented such that the PHY extracts the retry bit information from the MAC header (described in 14.2.3.6). Alternatively, a compliant device may be implemented such that the MAC uses a different MAC-PHY interface primitive to pass a retransmission frame to

PHY for transmission. As the internal communication between the PHY and MAC is implementation-specific, the method to be used shall be beyond the scope of this standard.

15.19.11 Link check

During Master-Slave operation, a Master device shall actively perform link check to determine the quality of the channel or whether a Slave device is still in the MSP after some inactivity. In additional, it may request a Slave device to perform a link check on its behalf. The following sub clause describes these operations.

15.19.11.1 Aperiodic channel scanning

During Master-Slave operation, in order to assess the channel quality and to detect any potential interference, a Master device shall schedule itself and its associated Slave device to perform aperiodic channel scan as specified in 15.7.1.2.

To announce its intention for both the Master device itself and the Slave device to perform aperiodic channel scan, the Master device shall include in its Type C Poll frame, a Scan Countdown IE, as specified in 14.7.23, indicating the number of MSPs to the scheduled aperiodic channel scan. The ScanCountdown field in the Scan Countdown IE shall be decremented by 1 after each MSP. And when a Scan countdown field of a Scan Countdown IE in a Type C Poll frame reaches zero, both the Master device and Slave device shall perform a channel scanning at the start of the next MSP. During the channel scan, both Master device and Slave device shall not transmit any frames and shall listen to the medium for the entire duration.

15.19.11.2 Remote scan

In addition to the mandatory aperiodic channel scan, that is to be performed by both Master and Slave device, as described in 15.19.11.1, a Master device may at any time request its Slave device to perform a channel scan and report the scan results back to the Master device.

The Master device initiates the remote scan procedure by sending a Remote Scan Request command (described in 14.8.12) to the associated Slave device and starting the RequestTimeout timer. The duration of RequestTimeout shall be determined by upper layer and will be communicated to the Slave device via the Request Timeout field in the Remote Scan Request command, as described in 14.8.12. The Slave device shall acknowledge the successful reception of the Remote Scan Request command by sending a Master-Slave Imm-ACK back to the Master device. The Slave device shall then proceed to perform channel scan in the same way described in 15.19.3. Once the channel scan is completed, the Slave device shall send a Remote Scan Response command (described in 14.8.13) containing the channel scan results to the Master device. The Master device shall acknowledge the successful reception of Remote Scan Response command using Master-Slave Imm-ACK, stop the RequestTimeout timer, and may forward the remote scan response command to the upper layer. The Master device may make use of the received remote scan results to make decision to stop the Master-Slave operation, or to shift the Master-Slave operation to another better quality channel. Figure 205 shows the MSC of the described remote scan procedure.

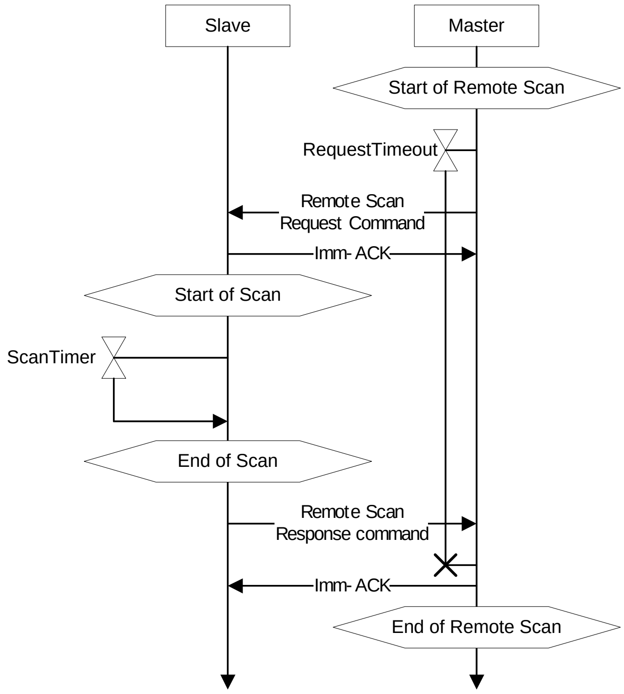


Figure 205 - MSC for remote scan procedure

15.19.12 Changing MSP parameters

A Master device may change the parameters of the MSP at any time. To change the MSP parameter, the appropriate parameter change IE (described in the following sub-clauses) shall be broadcasted by the master device. In order to ensure that an associated Slave device is informed of the MSP parameter change and have sufficient time to adjust to the change, the Master device shall broadcast the appropriate parameter change IEs for at least mMaxLostPolls number of Type C Poll frames before adopting the new MSP parameters.

15.19.12.1 Changing MSP Interval

The Master device may change the MSP Interval. To initiate the MSP Interval change, the Master device shall include the new MSP Interval in a MSP Interval Change IE, as described in 14.7.16, and broadcast it for at least nMaxLostPolls number of MSPs. After broadcasting the MSP Interval Change IE containing the new MSP Interval for at least nMaxLostPolls number of MSPs, the Master device shall adopt the new MSP Interval and transmit its Type C Poll frame for the next MSP based on the new MSP Interval as shown in Figure 206.

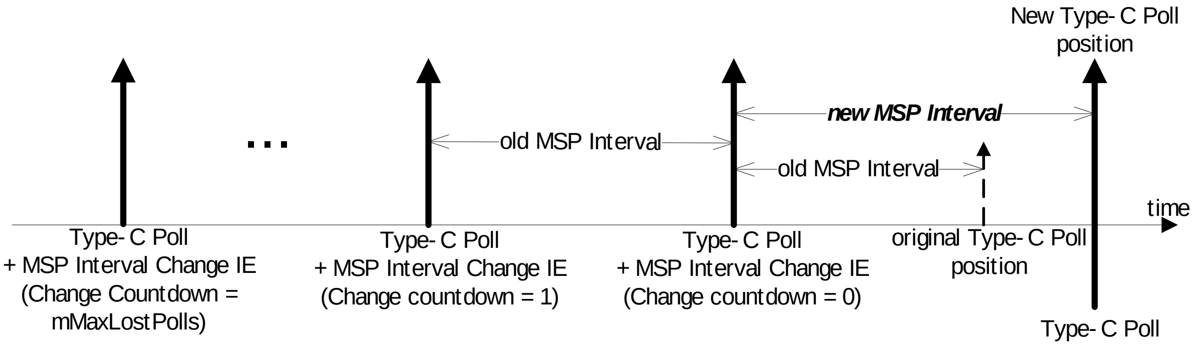


Figure 206 - Changing MSP interval

15.19.12.2 Changing operating channel

The Master device may change the current operating channel to another channel. To initiate the changing of operating channel, the Master device shall include the new operating channel number in the Channel Change IE, as described in 14.7.6, and broadcast it for at least mMaxLostPolls number of MSPs. After broadcasting the Channel Change IE containing the new MSP Interval for at least mMaxLostPolls number of MSPs, the Master device shall switch to the indicated new channel for its Master-Slave operation and transmit its Type C Poll frame for the next MSP in the new channel as shown in Figure 207.

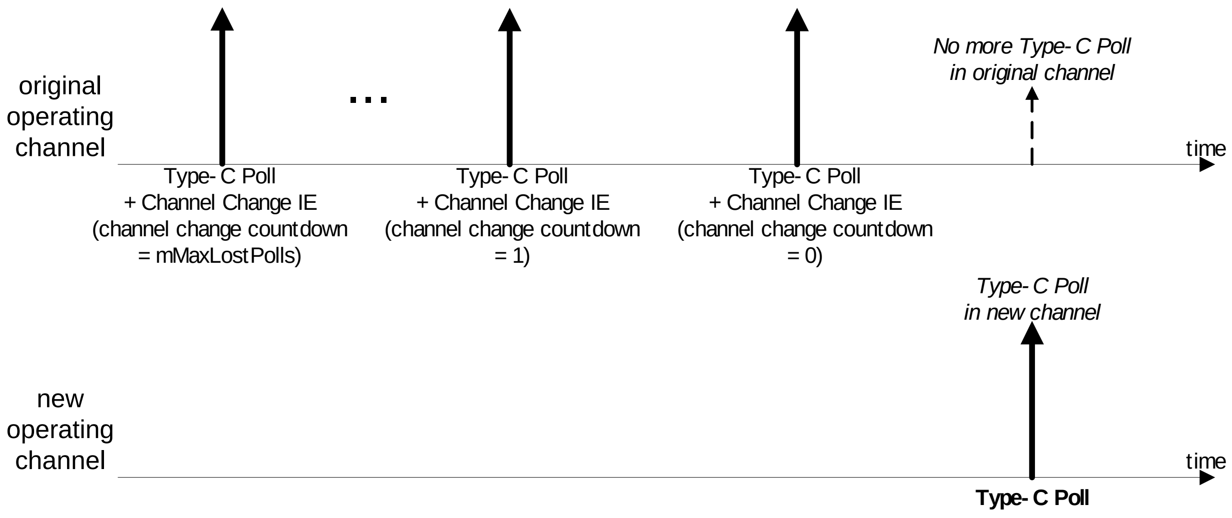


Figure 207 - Changing operating channel

After switching to the new channel, the Slave device shall scan for the Master device using the channel scanning procedure, described in 15.19.3, in the new channel.

15.19.13 Channel release

It may be possible that when a pair of Type C device forms a MSPr in a channel, there are no Type A or Type B device present but, after some time, a Type A or Type B device starts up and discovers a MSPr. In such situations, the Type A or Type B device may use Channel Release mechanism described in this clause to preempt the Type C devices on the current channel. Preempting the Type C devices on the current channel means to instruct the Type C devices to stop using the current channel for any transmission.

15.19.13.1 MSP timing information announcement

In addition to Master device's announcement of the MSP timing via the Type C Poll frame, a Slave device shall also include the MSP timing via the Type C Poll Response frame. In this way, Type A and Type B devices shall be able to determine the start of the CEP where it can initiate a Channel Release Request, as described in 15.19.13.2, in regards to any discovered Type C device.

15.19.13.2 Channel release request

For an initiator device to request Channel Release to a target device in a MSPr, a Channel Release Request command shall be sent by the initiator device during the CEP. In order to allow both Master device and Slave device to be able to know of the ongoing Channel Release request, the Channel Release Request command shall be sent as a broadcast frame, by indicating broadcast in the DestAddr field as described in 15.1.1.

As Channel Release request command is of higher priority compared to other MSPr commands, channel access in the CEP shall use the backoff algorithm described in 15.19.7.2 with some modifications. For a Channel Release request command, the initiator shall always select the value for backoff_counter randomly over a uniform distribution in the range of [0, 2] regardless of number of retry, to ensure that the initiator device is able to content successfully for most of the time.

Upon reception of a Channel Release Request command, a device shall not transmit any frame in that channel except for disassociation request with Reason code field set to Channel Release, or ACK frame to such a disassociation request frame.

Figure 208 illustrates a case when a Type A/B device sends Channel Release request to a Master device.

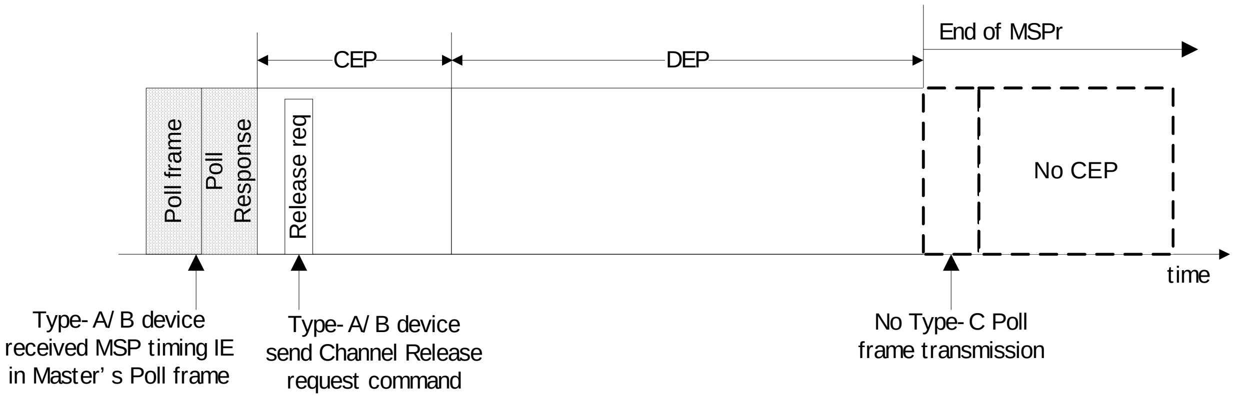


Figure 208 – Channel preemption of a master device

If there is sufficient time left in the CEP of the current MSP, the device receiving the Channel Release request shall initiate disassociation from the MSPr with Reason code field set to Channel Release, as described in 15.19.6.3, to properly terminate the MSPr. Figure 209 illustrates a case when a Master device disassociates the MSPr after receiving a Channel Release request from a Type A/B device.

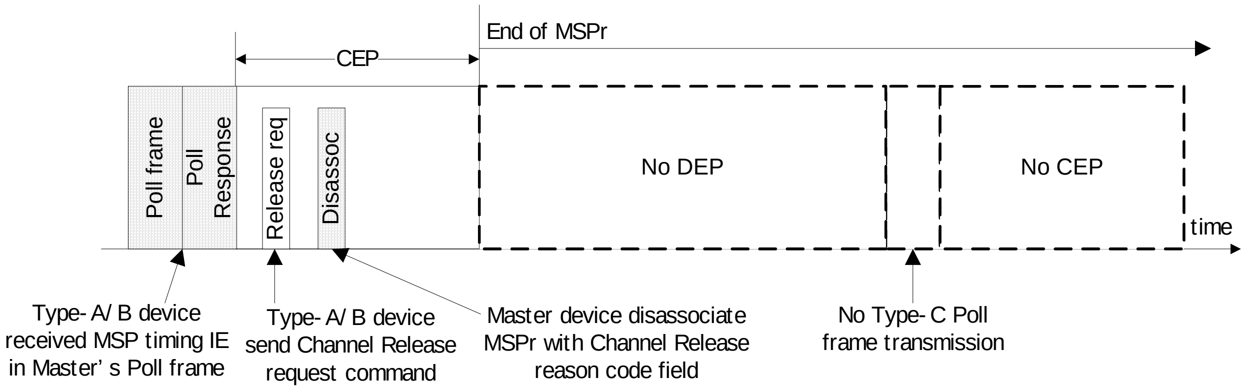


Figure 209 - Channel preemption of a master device with proper MSPr termination

If the target device for the disassociation is a Master device, the Master device, after being disassociated, shall stop transmitting Type C Poll frame as well as all Master-Slave operations in the current channel for at least mMaxNeighborDetectionInterval, After such suspension the Type C device may access the channel as specified in 15.3.3. Figure 209 illustrates a case when a Slave device disassociates the MSPr after receiving a Channel Release request from a Type A/B device.

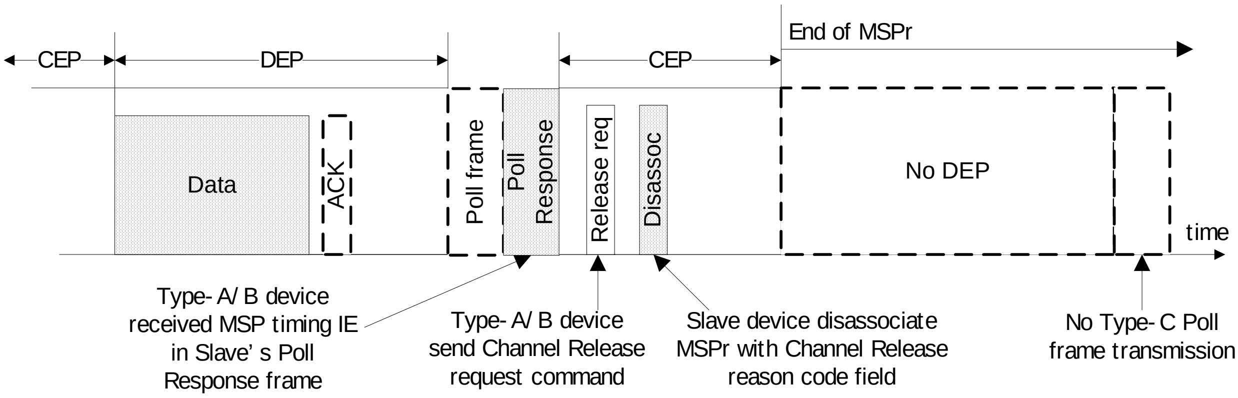


Figure 210 - Channel release of a slave device with proper MSPr termination

15.20 Transmit switched diversity (TSD) operation

The transmit switched diversity (TSD) is used to achieve diversity gain from shadowing or blockage. To perform the TSD operation, the source must have multiple antennas either sharing one common RF chain or using independent RF chains, respectively, as shown in Figure 211.

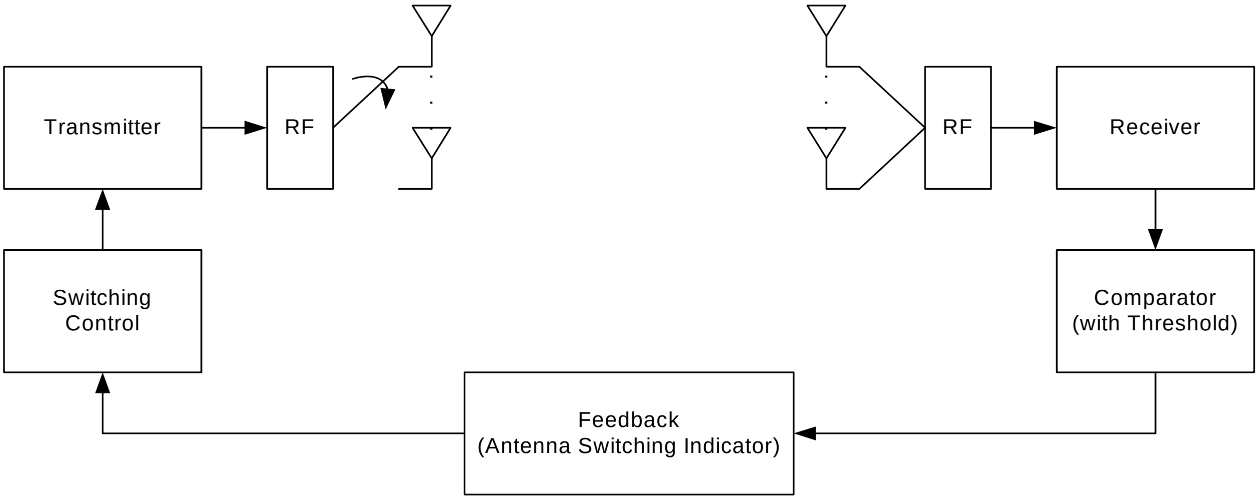


Figure 211 - Structure of transmit switched diversity system

15.20.1 TSD initiating procedure

The source shall send a TSD SET Request command to the destination using an arbitrarily selected antenna among multiple TX antennas, with the TSD Status set to BEGIN, as defined in 14.5.10. The destination, upon receiving the TSD SET Request command, shall send a TSD SET Response command, as defined in 14.5.11 to the source, proving the capability of the TSD operation of the destination.

15.20.2 Antenna switching

There are two mechanisms for switching TX antenna: reactive and proactive switching. In case of reactive switching, the destination sends the TSD Switch command frame with the TSD Switch field set to “Switch” to the source and then the source unconditionally switches to any other available TX antenna. In case of proactive switching, the source individually collects the channel status from every logical channel (link) which corresponds to each of the TX antennas, and switches the TX antenna to an optimal TX antenna which provides the maximum data rate among all TX antennas. The reactive switching is a simple way of the TX antenna switching which can overcome the link interruption, while the proactive switching can be used along with adaptive modulation and coding schemes to provide the maximum data rate, but it requires relatively complicated operating procedure.

15.20.2.1 Reactive switching

Before commencing data transmission, the source may arbitrarily select one antenna from multiple transmit antennas. At each frame transmission, the destination shall compare a certain metric based on the received SNR with a predetermined threshold after receiving data transmitted by the source. If the received SNR is smaller than the threshold, the destination shall feedback a TSD Switch command frame with the TSD Switch field set to “Switch” to the source. Note that the threshold may be selected by the receiver to satisfy the required criterion for a given data rate. Determining threshold may be decided by the implementers to get the required performance.

15.20.2.2 Proactive switching

In order to support proactive switching, the source should have the status information of each channel which corresponds to each of the TX antennas. To collect channel status for every TX antenna, the source shall be periodically operated in channel scan mode to acquire the status information of the channel which corresponds to each of the inactive antennas. When the data transmission is idle, the source and the destination may be operated in channel scan mode. During channel scan mode, the source shall stop using the current active TX antenna and switch to any TX antenna to acquire the channel state information by sending TSD SET Request command frame with the TSD Status field set to "SCAN" and by receiving TSD SET Response command frame with channel state information. Based upon the channel information acquired from all TX antennas, the source shall choose the optimal TX antenna which provides the best performance. The proactive switching may occur at every event of data rate change due to bad channel condition. However, if the current data rate is below the predetermined threshold, the proactive switching shall be periodically done to prevent staying with the current TX antenna despite bad channel conditions.

15.21 MAC sublayer parameters

Table 112 contains the values for the MAC sublayer parameters.

Table 112 - MAC sublayer parameters

Parameter	Value
mAccessDelay	1251 μs
mAggregationLimit	63
mBeaconSlotLength	21.3 μs
mBPExtension	6 beacon slots
mBPMergeWaitTime	128 superframes
mClockAccuracy	20 ppm
mClockResolution	1 μs
mDRPBackoffWinMax	16 superframes
mDRPBackoffWinMin	2 superframes
mGuardTime	4.7 μs
mInitialMoveCountdown	3 × mMaxLostBeacons
mMasLength	64 μs
mMaxBeaconLength	mBeaconSlotLength - pSIFS - mGuardTime
mMaxBeaconSlotCollisionDetectionLatency	16
mMaxBPLength	48 beacon slots
mMaxDiscoveryLatency	128
mMaxFragmentCount	8
mMaxFramePayloadSize	pMaxFramePayloadSize
mMaxHibernationProtection	128 superframes
mMaxLostBeacons	3
mMaxMovableLatency	32

Table 112 - MAC sublayer parameters (concluded)

Parameter	Value
mMaxNeighbourDetectionInterval	128 superframes
mMaxSignalingSlotBackoff	128
mMaxSynchronizationAdjustment	4µs
mMinFragmentSize	1
mSignalSlotCount	2 beacon slots
mSuperframeLength	256 x mMASLength
mTCBeaconSlotLength	8 µS
mCWMin	15
mCWMax	1023
mDBPMin	3 superframes
mDBPMax	7 superframes
C-SCAN	256 µS
B-SCAN	256 µS
D-SCAN	mSuperframeLength
BIFS	pMIFS
LIFS	pSIFS + mMaxBeaconLength
MIFS	pMIFS
SIFS	pSIFS
MIFS_C	pMIFS_C
SIFS_C	pSIFS_C
mMaxLostPolls	mMaxLostBeacons

Table 113 contains the values of the PHY dependent parameters used by the MAC sublayer for the PHY.

Table 113 - PHY-dependent MAC sublayer parameters for the PHY

Parameter	Value
pCCADetectTime	Defined in Tables 49 and 50
pClockAccuracy	20 ppm
pMaxFramePayloadSize	65535 octets
pSlotTime	8 µs

16 Security

This Clause specifies the security mechanisms needed to provide the security service introduced in 7.2.4.9. 16.1 reviews these security mechanisms. 16.2 defines security modes that govern the security operation of devices. 16.3 specifies the 4-way handshake procedure for two devices to establish pair-wise temporal keys (PTKs) and a secure relationship. This Clause also describes how a device may solicit or distribute group temporal keys (GTKs) within a secure relationship. 16.4 describes the procedures for frame reception and replay prevention. 16.5 provides the parameters needed in applying the AES-128 CCM cryptography to compute the message integrity code (MIC) and encrypt the secure payload for secure frames.

16.1 Security mechanisms

The security mechanisms specified in this Standard control the security operation of devices by setting appropriate security modes. They allow devices to authenticate each other, to derive PTKs, and to establish secure relationships. They also enable devices to solicit or distribute GTKs within established secure relationships. In addition, the security mechanisms provide replay attack prevention measures through the use of secure frame counters (SFCs) and replay counters. The security mechanisms specify the parameters needed in applying the AES-128 CCM to protect the privacy and integrity of unicast and broadcast/multicast traffic using PTKs and GTKs, respectively. Privacy is protected by encrypting the secure payload, while integrity is protected by including a MIC.

Two devices use a shared master key to establish a secure relationship. The establishment and management of master keys are additional security facilities that need to be provided outside the MAC sublayer.

16.1.1 Security operation

Security modes are defined to control the level of security required of a device in its communications with other devices. Three security modes are provided. Mode 0 allows a device to communicate without security protection. Mode 1 allows a device to use both secure and non-secure frames for data exchanges. Mode 2 restricts a device to use security facilities in transmitting and receiving certain frames.

A device announces its selected security mode in the Beacon Parameters field in its beacons.

16.1.2 4-way handshake

The 4-way handshake mechanism enables two devices to use a shared master key to authenticate the identity of each other and to establish a new PTK for protecting certain frames exchanged between the two devices. By way of a successful 4-way handshake, the two devices establish a secure relationship with each other.

A device initiates a 4-way handshake with another device only if it has determined that it shares a master key with that device. The master key is not exposed in the 4-way handshake; it is specified by a master key identifier (MKID).

16.1.3 Key transport

Two devices establish a new PTK via a 4-way handshake. The PTK is derived from a shared master key and two new random numbers generated by the two devices. A PTK is never transmitted directly in any frame, encrypted or not.

Two devices, after establishing a secure relationship via a successful 4-way handshake, distribute their respective GTKs for protecting their broadcast traffic to each other, if applicable. Additionally, a device may distribute GTKs for protecting certain multicast traffic addressed to those devices with which the device has a valid secure relationship. A device may also request, or solicit, GTKs used to protect multicast traffic from the multicast source devices.

A GTK is solicited or distributed by use of the GTK commands and is sent in encrypted form.

16.1.4 Freshness protection

Freshness protection insures that no parties can successfully replay previously captured messages as an attack. This Standard defines secure frame counters and replay counters on a per-temporal key basis to provide freshness protection.

16.1.5 Data encryption

Data encryption uses a symmetric cipher to protect data from access by parties not possessing the encryption key. This key is a PTK for unicast traffic transmitted between two devices and a GTK for broadcast/multicast traffic transmitted from a sender to a group of recipients.

AES-128 counter mode is used for data encryption in this Standard.

16.1.6 Frame integrity protection

Frames are protected from modification by other parties by message authentication using a MIC. The MIC also provides assurance that the sender of the frame possesses the correct temporal key. This key is shared among a group of devices or only between two devices. The MIC is a cryptographic checksum of the message to be protected.

AES-128 cipher block chaining **-** message authentication code (CBC-MAC) is used for MIC calculation in this Standard.

16.2 Security modes

The security mode indicates whether a device is permitted or required to establish a secure relationship with another device for data communications.

Two devices establish a secure relationship by a 4-way handshake based on a shared master key as described in 16.3.

Once two devices establish a secure relationship, they shall use secure frames for frame transfers between them as specified in Table 114 and Table 115**.** Either device shall discard a received frame from the other device if the frame is required to be a secure frame but was transmitted as a non-secure frame.

Data and aggregated data frames shall be transmitted using the temporal key specified by the TKID associated with the corresponding MSDU. Command and control frames, when transmitted as secure frames in a secure relationship, shall employ a temporal key currently possessed in that secure relationship.

In Table 114, "N" indicates a non-secure frame, and "S" indicates a secure frame.

Table 114 - Frame protection in a secure relationship

Frame type or subtype	Frame protection	Meaning
Beacon frame	N	Beacon frames shall be sent as non-secure frames.
Imm-ACK control frame	N	Imm-ACK frames shall be sent as non-secure frames.
B-ACK control frame	N	B-ACK frames shall be sent as non-secure frames.
RTT control frame	N	RTT frames shall be sent as non-secure frames.
CTT control frame	N	CTT frames shall be sent as non-secure frames.
Application-specific control frame	N, S	Application-specific control frames may be sent as secure or non-secure frames.
DRP Reservation Request command frame	N, S	DRP Reservation Request frames may be sent as secure or non-secure frames.
DRP Reservation Response command frame	N,S	DRP Reservation Response frames may be sent as secure or non-secure frames.
Probe command frame	N, S	Probe frames may be sent as secure or non-secure frames.
PTK command frame	N, S	PTK frames may be sent as secure or non-secure frames.
GTK command frame	S	GTK frames shall be sent as secure frames.
Application-specific command frame	N, S	Application-specific command frames may be sent as secure or non-secure frames.
Data frame	S	Data frames shall be sent as secure frames.
Aggregated data frame	S	Aggregated data frames shall be sent as secure frames.
TRN control frame	N, S	The TRN control frame may be sent as secure or non-secure.

Table 115 specifies the values of the Encryption Offset (EO) field in secure frames

Table 115 - EO values in secure frames

Frame type or subtype	EO value
Application-specific control frame	Application defined
DRP Reservation Request command frame	Length of Secure Payload
DRP Reservation Response command frame	Length of Secure Payload
PTK command frame	0
GTK command frame	0
Probe command frame	Variable
Application-specific command frame	Application defined
Data frame	Variable
Aggregated data frame	Length of (Aggregation Header + Aggregation Header Pad octets)

16.2.1 Security mode 0

A device operating in security mode 0 shall use non-secure frames to communicate with other devices. Such a device shall not establish a secure relationship with any other device.

If a device operating in this mode receives a secure frame, the MAC sublayer shall discard the frame.

16.2.2 Security mode 1

A device operating in security mode 1 shall use non-secure frames to communicate with devices operating in security mode 0. The device shall also use non-secure frames to communicate with devices operating in security mode 1 with which it does not have secure relationships. The device shall use secure frames according to Table 114 and Table 115 to communicate with another device operating in security mode 1 with which it has a secure relationship. It shall not establish secure relationships with other devices unless those devices are also operating in security mode 1.

A device operating in security mode 1 may or may not respond to command frames received from other devices with which it does not have a secure relationship.

If a device operating in security mode 1 receives a secure frame from a device with which it does not have a secure relationship, the MAC sublayer shall discard the frame.

If a device operating in mode 1 receives a non-secure frame from a device with which it has a secure relationship, but the frame is required to be a secure frame per Table 114, the MAC sublayer shall discard the frame.

A device that chooses to enable security mode 1 must understand and accept the responsibility that comes with receiving non-secure frames. The device shall instruct the higher layers to handle the received non-secure frames in a safe and secure manner.

A compliant MAC sublayer shall never use security mode 1 by default. Security mode 1 shall be entered from either mode 0 or mode 2. Requiring that a device explicitly select this mode serves as an indication that the device is aware of the security responsibilities it accepts when enabling security mode 1.

16.2.3 Security mode 2

A device operating in security mode 2 shall not establish a secure relationship with devices operating in either security mode 0 or security mode 1. The device shall use secure frames based on Table 114 and Table 115 to communicate with another device operating in security mode 2 and having a secure relationship with it. A device operating in security mode 2 shall establish a secure relationship with another device operating in the same security mode by a 4-way handshake prior to data exchanges.

If a device operating in mode 2 receives a secure frame from a device with which it does not have a secure relationship, the MAC sublayer shall discard the frame.

If a device operating in mode 2 receives a non-secure frame that is required to have frame protection per Table 114, regardless of whether the device has a secure relationship with the device transmitting the frame, the MAC sublayer shall discard the frame.

16.3 Temporal keys

Two devices establish a secure relationship based on a shared master key by employing a 4-way handshake to derive a PTK as described in this Clause**.** They may establish a PTK for each master key they share. Two devices have a secure relationship as long as they possess a currently installed PTK. A device's DevAddr is part of the information used in deriving a PTK. Once a PTK is established, it shall not be changed due to a change in the device's DevAddr.

A device solicits a GTK from, or distributes a GTK to, another device sharing a PTK as also described in this Clause.

Master keys are identified by MKIDs. A device is not required to include an MKID IE in its beacon, nor is it required to advertise every MKID it possesses in the MKID IE included in its beacon. They may advertise some or all of the MKIDs they possess in an MKID IE in their beacons. A device may probe another device for the MKIDs possessed by that device by addressing an appropriate Probe IE in a beacon or Probe command to that device. A device shall list all the MKIDs it possesses in the MKID IE in response to a probe request for its MKIDs.

16.3.1 Mutual authentication and PTK derivation

This Standard uses a 4-way handshake to provide mutual authentication and PTK generation for two devices sharing a master key. To perform a 4-way handshake, the two devices assume the roles of "initiator" and "responder", respectively. A 4-way handshake consists of four messages, called message 1, message 2, message 3, and message 4, that are sent back and forth between the two devices. The device sending message 1 becomes the initiator. The other device becomes the responder.

16.3.1.1 4-way handshake message 1

The initiator shall begin a 4-way handshake by composing and sending message 1 in a PTK command to the responder. In this command, the initiator shall specify the MKID for use in the 4-way handshake, propose a TKID for the PTK to be derived, and include a unique 128-bit cryptographic random number, I-Nonce. The proposed TKID shall be different from any TKID currently installed in the initiator's local MAC sublayer or being used in an in-progress 4-way handshake involving this initiator device. The I-Nonce shall be generated anew each time the initiator starts a new 4-way handshake.

On reception of message 1, the responder shall verify that the requested TKID is unique (i.e., not currently installed for an active temporal key or requested by an in-process 4-way handshake exchange). The responder shall perform the following steps:

1. Generate a new 128-bit cryptographic random number, R-Nonce.

2. Derive the PTK and KCK as specified in 16.3.4.

3. Construct and send message 2 in a PTK command.

16.3.1.2 4-way handshake message 2

The responder shall send message 2 to the initiator as specified in 16.3.1.1. In this command, the responder shall include an appropriate Status Code, the newly generated R-Nonce, and the PTK MIC value computed for the message using the newly derived KCK according to 16.3.5. If the proposed TKID in message 1 is not unique, the responder shall so indicate in the Status Code.

On reception of message 2, the initiator shall perform the following steps:

1. Derive the PTK and KCK as specified in 16.3.4.
2. Recalculate the PTK MIC for the received message using the KCK according to 16.3.5. If the recalculated PTK MIC does not match the PTK MIC field from this message, discard and disregard message 2 and abort the 4-way handshake. Otherwise, consider this message a proof that the responder holds the correct master key, and proceed to the next step.
3. Check the Status Code returned in the received message. If the Status Code indicates an abortion of the 4-way handshake by the responder, stop the 4-way handshake as well. If the Status Code indicates a conflict of the proposed TKID at the responder, restart the 4-way handshake with a different TKID. If the Status Code indicates a normal status, proceed to the next step.
4. Construct and send message 3 in a PTK command.

16.3.1.3 4-way handshake message 3

The initiator shall send message 3 to the responder as specified in 16.3.1.2. In this command, the initiator shall include the same I-Nonce as contained in message 1 and a PTK MIC computed for this message using the newly derived KCK according to 16.3.5.

On reception of message 3, the responder shall perform the following steps:

1. Verify the PTK MIC for this message using the KCK according to 16.3.5. If the calculated PTK MIC does not match the PTK MIC field from this message, discard and disregard message 3 and abort the 4-way handshake. Otherwise, consider this message a proof that the initiator holds the correct master key, and proceed to the next two steps.
2. Construct and send message 4 in a PTK command.
3. Install the PTK using the appropriate MLME primitives.

16.3.1.4 4-way handshake message 4

The responder shall send message 4 to the initiator as specified in 16.3.1.3. In this command, the responder shall include the same R-Nonce as contained in message 2 and a PTK MIC computed for this message using the KCK according to 16.3.5.

On reception of message 4, the initiator shall perform the following step:

1. Verify the PTK MIC for this message using the KCK according to 16.3.5. If the calculated PTK MIC does not match the PTK MIC field from this message, discard and disregard message 4 and abort the 4-way handshake; otherwise, install the PTK.

16.3.2 GTK exchange

Upon successful completion of a 4-way handshake and installation of the resulting PTK, the initiator and responder each shall use GTK command frames (with Message Number set to 1) to distribute their respective GTKs for broadcast traffic to each other. Each may also use a GTK command to distribute a GTK for protecting certain multicast traffic to an intended recipient with which it holds a valid PTK.

On reception of a valid GTK command frame marked as Message Number 1, a device shall verify that the GTKID is a unique TKID. The device shall then respond with a GTK command frame with Message Number set to 2 and Status Code set to the appropriate value.

A recipient may request a GTK for certain multicast traffic in the form of a GTK command (with Message Number set to 0) from the source device if it holds a valid PTK with the source.

On reception of a valid GTK command marked as Message Number 0, the multicast source device shall respond with a GTK command marked as Message Number 1, which may or may not contain the requested GTK. The requesting device, upon receiving this GTK command and verifying the uniqueness of the proposed TKID, shall further return a GTK command with Message Number set to 2 and Status Code set to the appropriate value.

A source device distributing a GTK shall check the Status Code indicated in the returned GTK command (Message Number set to 2). If the Status Code indicates a conflict of the proposed TKID at the recipient device, the source device shall propose a new TKID and re-distribute the GTK to the recipient. After receiving a returned GTK command from the recipient with the Status Code indicating a normal status, the source device shall use the new TKID to re-distribute the GTK to each of the devices to which it has previously distributed the GTK and with which it maintains a secure relationship.

A GTK shall be a 128-bit cryptographic-grade random number. A fresh GTK shall be generated when the distributing device establishes a new group relationship. 16.3.6 provides an example means of generating a fresh GTK.

16.3.3 Pseudo-random function (PRF) definition

A PRF is used in several places in the security specification. Depending on the use, the PRF may need to output values of 64 bits, 128 bits, and 256 bits. This Clause defines three PRF variants:

- PRF-64, which outputs 64 bits,
- PRF-128, which outputs 128 bits, and
- PRF-256, which outputs 256 bits.

In the following, K denotes a 128-bit symmetric key, N denotes a 13-octet nonce value, A denotes a unique 14-octet ASCII text label for each different use of the PRF, B denotes the input data stream, Blen specifies the length of this data stream, and || denotes concatenation. Blocks are each 16 octets long, and are defined as inputs to the AES-128 CCM for the MIC generation as specified in 16.5.

```
CCM-MAC-FUNCTION(K, N, A, B, Blen)
begin
    Form authentication block B_0 from flags = 0x59, N, and l(m) = 0
    Form authentication block B_1 from l(a) = 14 + Blen and A
    Form additional authentication blocks from B
        (with last block zero padded as needed)
    Form encryption block A_0 from flags = 0x01, N, and Counter_0 = 0
    R ← MIC (K, B_0, B_1, ..., A_0)
```

```
return $R$

PRF($K, N, A, B, Blen, Len$)
  for $i \leftarrow 1$ to ( $Len + 63)/64$ do
      $R \leftarrow R \,||\,$ CCM-MAC-FUNCTION($K, N, A, B, Blen$)
      $N \leftarrow N + 1$
return L($R, 0, Len$) = $Len$ most-significant bits of $R$
```

PRF-64($K, N, A, B, Blen$) = PRF($K, N, A, B, Blen, 64$)

PRF-128($K, N, A, B, Blen$) = PRF($K, N, A, B, Blen, 128$)

PRF-256($K, N, A, B, Blen$) = PRF($K, N, A, B, Blen, 256$)

16.3.4 PTK and KCK derivation

PRF-256 shall be employed to generate the PTK and KCK associated with a 4-way handshake as used in 16.3.1 based on the following parameters as defined in Table 116.

K - The PMK

N - B12-11= InitiatorDevAddr, B10-9= ResponderDevAddr, B8-6 = PTKID, B5-0 = zero

A - "Pair-wise keys"

B - I-Nonce || R-Nonce

Blen - 32

Table 116 - PTK and KCK generation parameters

Name	Size (octets)	Description
InitiatorDevAddr	2	DevAddr of device with role of initiator
ResponderDevAddr	2	DevAddr of device with role of responder
I-Nonce	16	Random number selected by initiator (in message 1)
R- Nonce	16	Random number selected by responder (in message 2)
PTKID	3	Negotiated TKID value for the PTK to be derived (in message 1)
PMK	16	A pre-shared pair-wise master key identified by the MKID (in message 1)

The PRF-256 is called with these parameters to compute a 256-bit key stream:

KeyStream $\leftarrow$ PRF-256($K, N, A, B, Blen$)

This key stream is then split to form the desired PTK and KCK. The least-significant 16 octets of KeyStream become the KCK while the most-significant 16 octets become the PTK, as specified in Table 117.

Table 117 - KCK and PTK source

Key	Source
KCK	KeyStream octets 0 through 15
PTK	KeyStream octets 16 through 31

16.3.5 PTK MIC generation

The 4-way handshake uses an "out-of-band MIC" calculation for the PTK MIC field in handshake messages 2-4. PRF-64 shall be used to provide the PTK MIC calculation. The PRF-64 parameters shall be defined as follows based on Table 116:

K - The KCK

N - B12-11 = InitiatorDevAddr, B10-9 = ResponderDevAddr, B8-6 = PTKID, B5-0 = zero

A - "out-of-bandMIC"

B - Fields from Message Number to I-Nonce/R-Nonce contained in the PTK command

Blen - Length in octets of *B* = 48

PTK MIC ← PRF-64(*K, N, A, B, Blen*)

16.3.6 Random number generation

To implement the cryptographic mechanisms outlined in this Standard, devices need to generate cryptographic grade random numbers. ISO/IEC 8802-11 Amendment 6 gives a detailed explanation of cryptographic grade random numbers and provides guidance for collecting suitable randomness. It recommends collecting random samples from multiple sources followed by conditioning with PRF. This method can provide a means for an implementation to create an unpredictable seed for a pseudo-random generation function. The example below shows how to distil such a seed using random samples and PRF-128.

```
LoopCounter = 0
Nonce = 0
while LoopCounter < 32 begin
    result = PRF-128(0, Nonce, "InitRandomSeed", DevAddr || Time || result || LoopCounter, dataLen)
    Nonce ← Nonce + 1
    result ← result || <randomness samples>
end
GlobalSeed = PRF-128(0, Nonce, "InitRandomSeed", DevAddr || Time || result || LoopCounter, dataLen)
```

Once the seed has been distilled, it can be used as a key for further random number generation. The 4-way handshake requires each party to supply a 128-bit random number. This number can be generated using the seed and PRF-128.

```
GenerateRandomNonce
begin
    N = DevAddr || DevAddr || zero
```

```
        Collect randomness samples
        result = PRF-128(Global Seed, N, "Random Numbers", <randomness samples>, length of samples)
    return result
```

16.4 Frame reception steps and replay prevention measures

A recipient device shall carry out the reception steps and replay prevention measures as specified in this Clause.

16.4.1 Frame reception

The MAC sublayer shall perform the following validation steps in sequence when receiving frames:

1. Validate the FCS. If this validation fails, discard the frame. Otherwise, acknowledge the received frame using the appropriate acknowledgment rules, and proceed to the next step.
2. Validate the Secure bit setting in the MAC Header and take the appropriate actions according to its security mode as specified in 16.2. If the frame is not discarded and the Secure bit is set to ONE, proceed to the next step.
3. Validate the TKID. If the TKID does not identify a currently installed PTK or GTK, discard the frame; otherwise, proceed to the next step.
4. Validate the MIC using the identified PTK or GTK as specified in 16.5. If this validation fails, discard the frame; otherwise, proceed to the next step.
5. Detect frame replay as specified in 16.4.2. If replay is detected, discard the frame; otherwise, update the replay counter that was set up for the PTK or GTK used for this frame as also specified in 16.4.2, and proceed to the next step.
6. Process the frame as specified in Clause 15, including duplicate frame filtering. If the frame was already received, discard it. Otherwise, proceed to the next step.
7. Decrypt the frame. This step may be taken in parallel with the MIC validation step.

16.4.2 Replay prevention

Each transmitting MAC sublayer shall set up a 48-bit SFC and initialize it to zero when a temporal key, PTK or GTK, is installed to it. The MAC sublayer shall increment the SFC by one before transmitting a secure frame - whether a new frame or a retry - that uses the temporal key, and shall set the SFN in that secure frame to the value of the SFC after the increment.

Each recipient MAC sublayer shall set up a 48-bit replay counter when a temporal key, PTK or GTK, is installed to it. The MAC sublayer shall initialize the replay counter to zero for an installed PTK, and to the GTK SFC for an installed GTK which was contained in the GTK command distributing the GTK.

Upon receipt of a secure frame with valid FCS and MIC, the recipient shall perform replay attack detection and protection as follows:

The recipient shall compare the SFN extracted from the received frame with the reading of the replay counter for the temporal key used by the frame. If the extracted SFN is smaller than or equal to the replay counter reading, the recipient MAC sublayer shall discard the frame; otherwise, the recipient shall set the corresponding replay counter to the received SFN.

The recipient shall insure that the frame passes FCS validation, replay prevention, and MIC verification before using the SFN to update its replay counter.

16.4.3 Implications on GTKs

Because a recipient maintains only one replay counter per installed temporal key, that recipient can receive traffic from only one source using a given temporal key. A scheme that allows multiple source devices to use the same GTK will result in frames sent from some of those sources being seen as replay attacks. To avoid this problem, each source device in a group is required to distribute a unique GTK to the recipients in the group.

16.5 AES-128 CCM inputs

AES-128 CCM provides confidentiality, authentication, and integrity for secure frames defined in this Standard. This Clause specifies the various fields required for AES-128 CCM operation.

16.5.1 Overview

AES, the Advanced Encryption Standard, is specified in FIPS PUB 197. AES-128 defines a symmetric block cipher that processes 128-bit data blocks using 128-bit cipher keys. CCM, counter with CBC-MAC, is specified in RFC 3610. CCM employs counter mode for encryption and cipher block chaining for authentication. AES-128 CCM combines AES-128 with CCM to encrypt and authenticate messages.

Encryption is done on part or all of the Secure Payload, while authentication is provided by a message integrity code (MIC) that is included in each secure frame. MIC also protects the integrity of the MAC Header and Frame Payload in a secure frame.

CCM has two input parameters - M (number of octets in authentication field) and L (number of octets in length field). For this Standard, M = 8 and L = 2.

CCM requires the use of a temporal key and a unique Nonce for each transmitted frame to be protected. The SFN is combined with frame addressing and temporal key identification information to provide a unique Nonce for every secure frame. Since every frame protection with a key requires a unique Nonce, temporal keys have a known lifetime. Each temporal key can be used to protect up to *n* frames, where *n* is the maximum value of the SFN. All security guarantees are void if a nonce value is used more than once with the same temporal key.

In the following figures in this Clause showing the format of Nonce and CCM blocks, the most-significant octet is represented to the left of the other octets.

16.5.2 Nonce

The CCM Nonce is a 13-octet field, consisting of the 2-octet SrcAddr, 2-octet DestAddr, 3-octet TKID, and 6-octet SFN for the current frame. The Nonce is used as a component of authentication block B_0, an input to CBC-MAC. It is also used as a component of input block A_i for CCM encryption. It provides the uniqueness that CCM requires for each instance of authentication/encryption. The CCM Nonce shall be formatted as shown in Figure 212. In this figure, each component of the Nonce is represented with the most-significant octet on the left and the least-significant octet on the right.

octets: 2	**2**	**3**	**6**
SrcAddr	DestAddr	TKID	SFN

Figure 212 - Nonce input to the CCM algorithm

16.5.3 CCM blocks

The CCM authentication blocks shall be formatted as shown in Figure 213 and further described below.

octets: 1	13	2	2	10	2	1	1	EO	0-15	P – EO	0-15
Flags (= 0x59)	Nonce	Encrypted data length l(m) = P – EO	Additional authenticated data length l(a) = 14 + EO	MAC Header	Encryption Offset (EO)	Security Reserved	0	Secure Payload portion not to be encrypted	Zero padding	Secure Payload portion to be encrypted	Zero padding
B_0			B_1					B_2, …, B_(M-1)		B_M, …, B_N	

Figure 213 - Input to CCM authentication blocks

16.5.3.1 Authentication block B_0

Authentication block B_0 is the first input block to the CBC-MAC algorithm. It shall be formatted as shown in Figure 214. The component l(m) is represented with the most-significant octet on the left and the least-significant octet on the right. The Nonce component is represented with the least-significant octet on the left and the most-significant octet on the right.

octets: 1	13	2
Flags = 0x59	Nonce	l(m)

Figure 214 - Format of authentication block B_0

16.5.3.2 Authentication block B_1

Authentication block B_1 is the second input block to the CBC-MAC algorithm. It shall be formatted as shown in Figure 215. In this block, the l(a) component is represented with the most-significant octet on the left and the least-significant octet on the right. The EO and MAC Header components are represented with the first octet transmitted into the wireless medium on the left and the last transmitted octet on the right.

octets: 2	10	2	1	1
l(a)	MAC Header	EO	Security Reserved	0

Figure 215 - Format of authentication block B_1

16.5.3.3 Authentication blocks B_2, …, B_N

Authentication blocks B_2, …, B_(M-1) and B_M, …, B_N, if any, are additional input blocks to the CBC-MAC algorithm. They shall be formatted as shown in Figure 216. They are formed by breaking the Secure Payload portion not to be encrypted into 16-octet blocks and the Secure Payload portion to be encrypted into 16-octet blocks. The last block constructed from the Secure Payload portion not to be encrypted is padded with zero values as needed to insure 16-octet block length. Likewise, the last block constructed from the Secure Payload portion to be encrypted is padded with zero values as needed to insure 16-octet block length. The padding octets are not transmitted onto the wireless medium.

octets: EO	0-15	P – EO	0-15
Secure Payload portion not to be encrypted	Zero padding	Secure Payload portion to be encrypted	Zero padding
B_2, …, B_(M-1)		B_M, …, B_N	

Figure 216 - Format of authentication blocks beginning from B_2

In each of the blocks B_2, …, B_(M-1) or B_M, …, B_N, the Secure Payload portion not to be, or to be, encrypted shall be represented with the earliest octet transmitted into the wireless medium on the left and the latest transmitted octet on the right. When needed, B_(M-1) and B_N are padded with zeros to the right.

16.5.3.4 Encryption blocks A_0, A_1, …, A_m

CCM uses encryption blocks A_0, A_1, …, A_m to generate key stream blocks that are used to encrypt the CBC-MAC and the Secure Payload portion to be encrypted. These blocks shall be formed as shown in Figure 217. In this figure, Counter *i* is a 2-octet monotonically incrementing counter that shall be initialized to 0 for each secure frame. It shall be incremented by one for each successive encryption block. The Counter *i* component of A_i shall be represented with the most-significant octet on the left and the least-significant octet on the right. The Nonce component shall be represented with the least-significant octet on the left and the most-significant octet on the right.

octets: 1	13	2
Flags = 0x01	Nonce	Counter *i*

Figure 217 - Format of A_i blocks

16.6 Token authentication

Tokens are used by devices to authenticate with each other indicating they belong to the same security domain. The user designates one device as the domain authorizer, and uses the domain authorizer to issue tokens to devices in which the user trusts.

16.6.1 Token issuance

A device transmits a token request to the domain authorizer when it needs to join the domain. The request contains the device's public key. The user verifies the request. The method of how to interact with the user is implementation related. A simple example is a flashing light flash button which the user can push to confirm the authorization. After the verification, the domain authorizer generates a token. The token is the signature of a hash value signed using the domain authorizer's private key. The hash value is the hash of the requesting device's public key. The domain authorizer then transmits the token, together with the domain authorizer's public key, back to the device. The device then verifies the token and stores it.

Two devices shall authenticate with each other that they belong to the same security domain before transmission starts. The tokens are exchanged and verified between two devices and the Revocation List is checked before key exchange. The Revocation List is described in 16.6.2.

16.6.2 Token revoke

The User uses the domain authorizer to revoke tokens by generating a Revocation List (RL). The Revocation List is the list of the hash values of each revoked device's public key and is signed by the domain authorizer's private key. After one device is added or removed from the list, a new RL is generated with a higher version number. After a new RL is generated, it is broadcast to all the devices in the domain. Devices receiving a higher version numbered RL will update its RL.

17 HDMI PAL

17.1 Introduction

This clause describes an HDMI pass-through protocol adaptation layer which preserves the HDMI content protection scheme. Figure 218 shows the wired HDMI interface. There are 3 channels of data, a clock channel, a display data channel and a CE control. The data channels are TMDS encoded where every 8 bits of data are encoded to 10 bits for purposes associated with wire transmission.

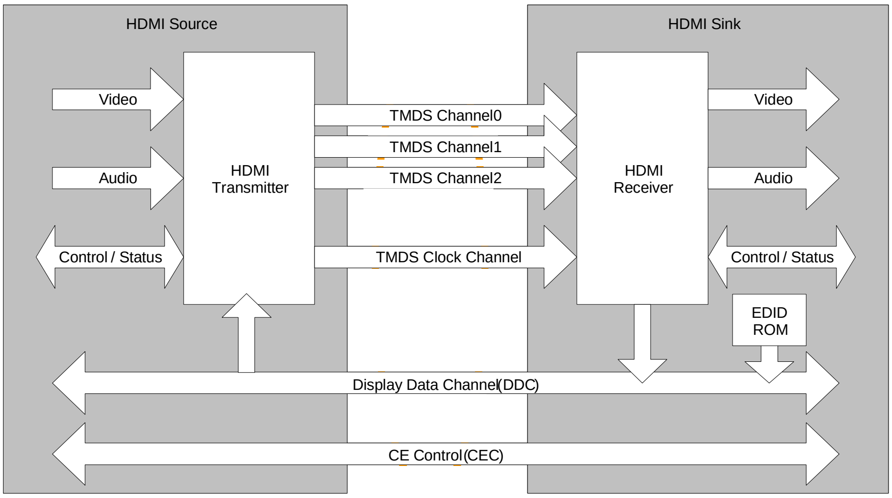


Figure 218 - Wired HDMI

The 60 GHz wireless solution is placed between the HDMI source and HDMI sink as shown in Figure 219 and Figure 220.

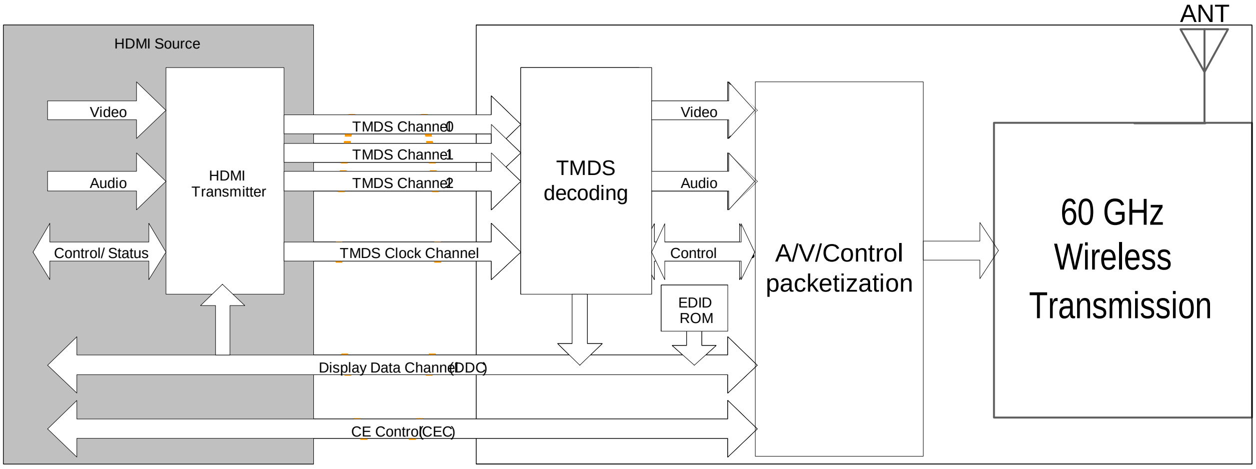


Figure 219 - Wireless HDMI transmitter

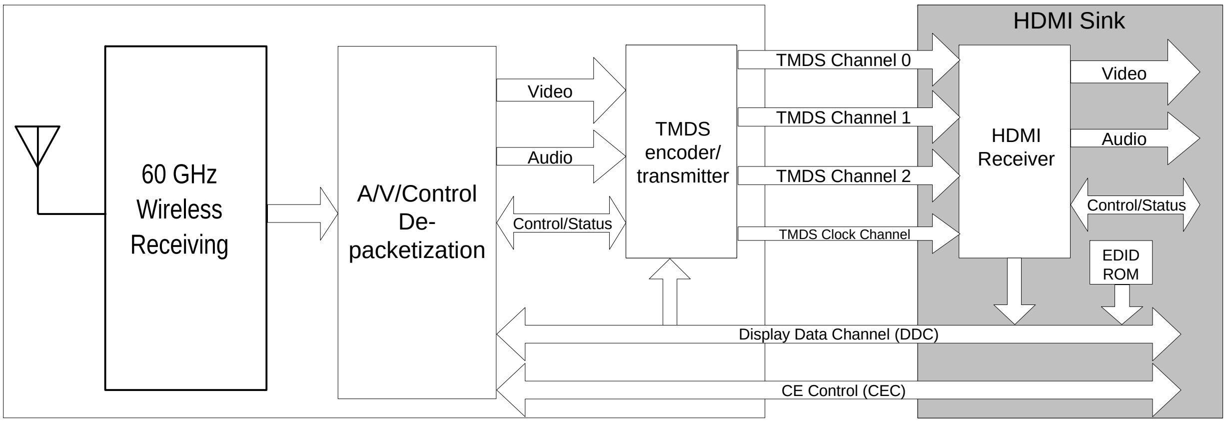


Figure 220 - Wireless HDMI receiver

The HD frame is made up of blanking periods, which may contain data islands, and active video periods as shown in Figure 221.

***720x480p video frame*:**

- 128 frames = ~2 seconds
- 64 frames = ~1 second
- 1 frame = ~15.5ms
- Vertical blanking = ~1.2 ms
- 1 TMDS clock = 1 pixel clock
- 1 TDMS period = 30 bits
- 4 different PAL payload type
  - Video Data Packet
  - Data Island Packet
  - Control Packet
  - DDC/CEC Packet
- Bits are numbered in little-endian format, i.e. the least-significant bit of a byte or word is referred to as bit 0.
- 1 frame = 450450 TMDS periods = 1.6MB

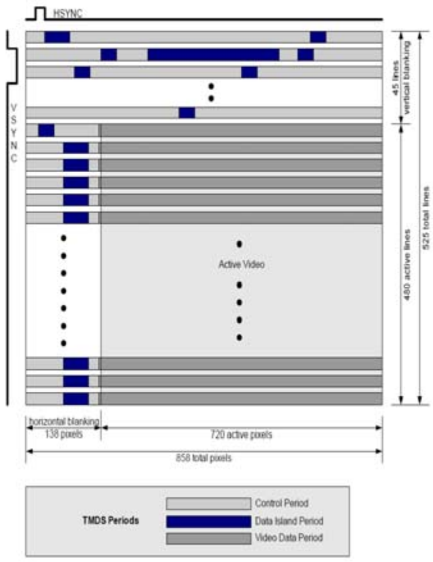


Figure 221 - Frame example

The nature of the frame will result in four kinds of packets: Video, Audio/Auxiliary, Control and DDC/CEC.

17.2 HDMI transmission

In the wireless HDMI transmitter, the TDMS coding is removed prior to transmission. The three data channels are then multiplexed together, along with the display data channel and the CE control. Both the Control Period and the Data Island Period contain the HSYNC and VSYNC information. If the data is video, then the pixel words are identified and the LSBs are flagged for unequal error protection (UEP). If the data is not video, then the data is flagged for equal error protection (EEP). The serial bit stream is then framed and presented to the MAC SAP.

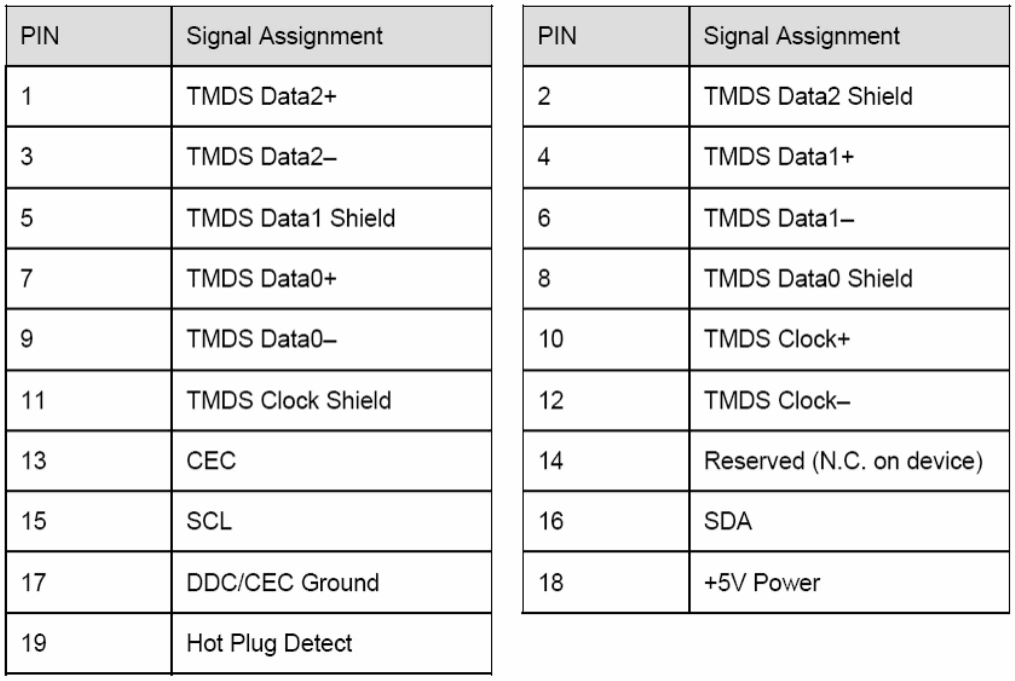
17.2.1 Identification of video vs. data

In each video frame, a Control Period is required between any two periods that are not Control Periods. It is used to identify the type of the following Period (Video Data Period or Data Island Period).

17.2.2 TMDS removal

There are four colour depths supported: 24, 30, 36 and 48 bits per pixel. The pixel encoding method is pre-negotiated as part of the E-EDID (Extended display identification data) structure.

The 10 bit code words are identified on the TMDS channels 0 through 2. The device extracts 10 bits of data per Clock period on each TMDS pin.

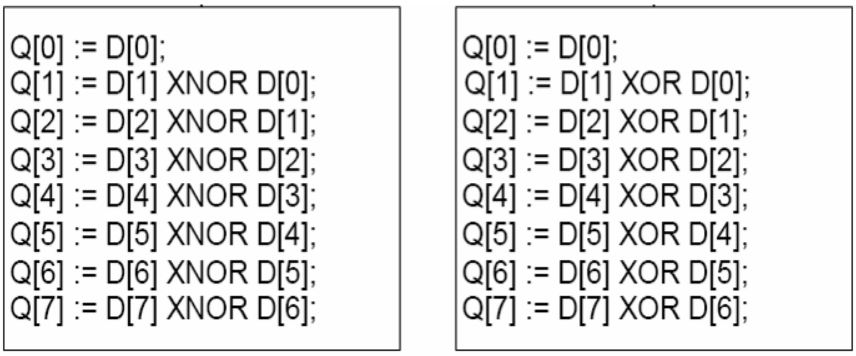


PIN	Signal Assignment
1	TMDS Data2+
3	TMDS Data2–
5	TMDS Data1 Shield
7	TMDS Data0+
9	TMDS Data0–
11	TMDS Clock Shield
13	CEC
15	SCL
17	DDC/CEC Ground
19	Hot Plug Detect

PIN	Signal Assignment
2	TMDS Data2 Shield
4	TMDS Data1+
6	TMDS Data1–
8	TMDS Data0 Shield
10	TMDS Clock+
12	TMDS Clock–
14	Reserved (N.C. on device)
16	SDA
18	+5V Power

Figure 222 - HDMI pin definitions

The 10th bit indicates if the first 8 bits need to be inverted. A value of 1 indicates the inversion has been made and a value of zero indicates that no inversion has been made. The 9th bit then indicates if the first 8 bits are pairwise XOR'ed or XNOR'ed. If the 9th bit has value of 1, then the first 8 bits are pairwise XOR'ed; otherwise, the first 8 bits are pairwise XNOR'ed.



```
Q[0] := D[0];
Q[1] := D[1] XNOR D[0];
Q[2] := D[2] XNOR D[1];
Q[3] := D[3] XNOR D[2];
Q[4] := D[4] XNOR D[3];
Q[5] := D[5] XNOR D[4];
Q[6] := D[6] XNOR D[5];
Q[7] := D[7] XNOR D[6];
```

```
Q[0] := D[0];
Q[1] := D[1] XOR D[0];
Q[2] := D[2] XOR D[1];
Q[3] := D[3] XOR D[2];
Q[4] := D[4] XOR D[3];
Q[5] := D[5] XOR D[4];
Q[6] := D[6] XOR D[5];
Q[7] := D[7] XOR D[6];
```

Figure 223 - TMDS coding XNOR/XOR scheme

17.2.3 Data type multiplexing

17.2.3.1 Video data

The video packet contains the PAL header and packet body. The leading guard band shall be stripped by the PAL and will not be part of payload. It is the receiver PAL's responsibility to insert the leading guard band into the HDMI Sink's TMDS data channel based on the PAL Header.

The layout of the packet body with "payload size = n bytes" is described below.

Name	***Offset***	***Size***	***Content***
PAL Header	0	32 bits	Header data
Pixel stream on TMDS 0	PAL Header+0	8 bits	Pixel data
Pixel stream on TMDS 1	PAL Header+1	8 bits	Pixel data
Pixel stream on TMDS 2	PAL Header+2	8 bits	Pixel data
.			
.			
Pixel stream on TMDS 0	PAL Header+n-3	8 bits	Pixel data
Pixel stream on TMDS 1	PAL Header+n-2	8 bits	Pixel data
Pixel stream on TMDS 2	PAL Header+n-1	8 bits	Pixel data

Figure 224 - Video packet body layout

17.2.3.2 Data island

The data island packet contains the PAL header and packet body. The leading guard band and trailing guard band shall be stripped by the PAL and will not be part of payload. It is the receiver PAL's responsibility to insert the leading guard band and trailing guard band into the HDMI Sink's TMDS data channel base on the PAL Header.

The minimum size of payload size = 32 TMDS periods (1 data packet * 32 Data TMDS period each). The maximum payload size = 576 TMDS periods (18 data packets * 32 Data TMDS period each).

1 Data packet = 32 TMDS periods = 30 bits * 32 = 960 bits = 120 bytes

The layout of the packet body with "payload size = n bytes" is described below.

Name	***Offset***	***Size***	***Content***
PAL Header	0	32 bits	Header data
Data packet clock 0 on TMDS 0	PAL Header+0	8 bits	MSB 8 bits of TMDS 0
Data packet clock 0 on TMDS 1	PAL Header+1	8 bits	LSB 2 bits of TMDS 0 + MSB 6 bits of TMDS 1
Data packet clock 0 on TMDS 2	PAL Header+2	8 bits	LSB 4 bits of TMDS 1 + MSB 4 bits of TMDS 2
Data packet clock 1 on TMDS 0	PAL Header+3	8 bits	LSB 6 bits of TMDS 2 + MSB 2 bits of TMDS 0
Data packet clock 1 on TMDS 1	PAL Header+4	8 bits	LSB 8 bits of TMDS 0
Data packet clock 1 on TMDS 2	PAL Header+5	8 bits	MSB 8 bits of TMDS 1
.			
.			
Data packet clock k on TMDS 2	PAL Header+n-1	8 bits	LSB 8 bits of TMDS 2

Figure 225 - Data packet body layout

17.2.3.3 Control

The control period contains a 4 bytes PAL header and packet body (control only). The preamble shall be stripped by the PAL and will not be part of payload. It is the receiver PAL's responsibility to insert the Preamble into the HDMI Sink's TMDS data channel base on the PAL Header.

The layout of the packet body with "payload size = n bytes" is described below.

Minimum Control Packet Payload Size = 4 TMDS periods = 15 bytes

Name	***Offset***	***Size***	***Content***
PAL Header	0	32 bits	Header data
Control packet clock 0 on TMDS 0	PAL Header+0	8 bits	MSB 8 bits of TMDS 0
Control packet clock 0 on TMDS 1	PAL Header+1	8 bits	LSB 2 bits of TMDS 0 + MSB 6 bits of TMDS 1
Control packet clock 0 on TMDS 2	PAL Header+2	8 bits	LSB 4 bits of TMDS 1 + MSB 4 bits of TMDS 2
Control packet clock 1 on TMDS 0	PAL Header+3	8 bits	LSB 6 bits of TMDS 2 + MSB 2 bits of TMDS 0
Control packet clock 1 on TMDS 1	PAL Header+4	8 bits	LSB 8 bits of TMDS 0
Control packet clock 1 on TMDS 2	PAL Header+5	8 bits	MSB 8 bits of TMDS 1
•			
•			
Control packet clock k on TMDS 2	PAL Header+n-1	8 bits	LSB 8 bits of TMDS 2

Figure 226 - Control packet body layout

17.2.3.4 DDC/CEC packet

The DDC/CEC packet contains the PAL header and packet body. The packet body includes the 6 bytes DDC/CEC message header, followed by the variable size message body. The format of the DDC and CEC message body can be find in the HDMI specification. The layout of the packet body is described below.

Name	***Offset***	***Size***	***Content***
PAL Header	0	32 bits	Header data
Message Header (Type)	PAL Header+0	8 bits	EDID Protocol: 1 CEC Protocol: 2
Message Header (Rd/Wrt)	PAL Header+1	8 bits	Read: 0 Write: 1
Message Header (Address)	PAL Header+2	8 bits	Address
Message Header (SegPtr)	PAL Header+3	8 bits	Segment Pointer
Message Header (MsgLen)	PAL Header+4	16 bits	Message Length: 1 - 65535
Message body byte 0	PAL Header+Message Header+0		
Message body byte 1	PAL Header+Message Header+1	8 bits	
•			
•			
Message body byte n	PAL Header+Message Header+n-1	8 bits	

Figure 227 - DDC/CEC packet body layout

17.3 HDMI reception

In the wireless HDMI receiver, the packet is received via the MAC SAP. The PAL header is read to indicate if the packet is video, audio, control or DDC/CEC. If the packet is video then the TDMS coding is applied prior to passing the video data onto the HDMI receiver. The three data channels are demultiplexed apart, along with the display data channel and the CE control.

17.3.1 TMDS encoding

For video packets the PAL needs to apply the TDMS coding prior to passing it to the HDMI receiver. The first stage produces a transition-minimized 9-bit code word from the input 8 bits. The second stage produces a 10-bit code word, the finished TMDS character, which will manage the overall DC balance of the transmitted stream of characters.

A two-stage process converts an input of 8 bits into a 10 bit code with particular desirable properties.

In the first stage each bit is either XOR or XNOR transformed against the previous bit, whilst the first bit is not transformed at all. The encoder chooses between XOR and XNOR by determining which will result in the fewest transitions; the ninth bit is added to show which was used, as specified in 17.2.2.

In the second stage, the first eight bits are optionally inverted to even out the balance of ones and zeros and therefore the sustained average DC level. The tenth bit is added to indicate whether this inversion took place.

Table 118 - TMDS encoding operators

Operator	Definition
D	The encoder input data set. D is 8-bit pixel data
cnt	This is a register used to keep track of the data stream disparity. A positive value represents the excess number of “1”s that have been transmitted. A negative value represents the excess number of “0”s that have been transmitted. The expression cnt{t-1} indicates the previous value of the disparity for the previous set of input data. The expression cnt(t) indicates the new disparity setting for the current set of input data.
q_m	Intermediate value.
q_out	These 10 bits are the encoded output value.
$N_1\{x\}$	This operator returns the number of “1”s in argument “x”.
$N_0\{x\}$	This operator returns the number of “0”s in argument “x”.

Upon entering a Video Data Period, the data stream disparity (cnt) shall be considered to be zero by the encoder.

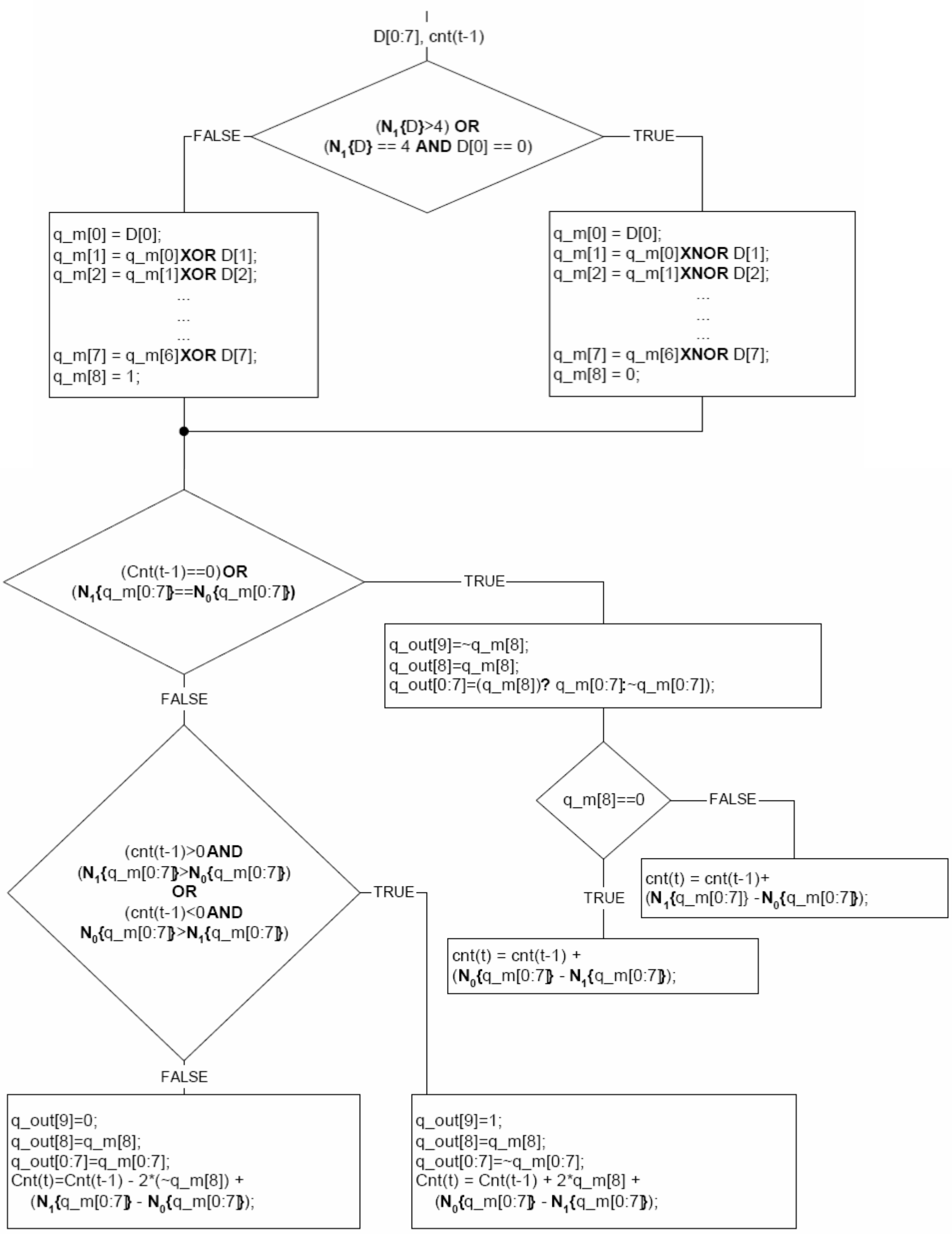


Figure 228 - TMDS decoding flowchart

17.3.2 Packet demultiplexing

The packet body is routed, as indicated, to the appropriate port of the HDMI receiver.

17.3.2.1 Video

Video packets are demultiplexed and distributed to TMDS data channels 0 through 2. The receiver PAL shall receive and examine the PAL header for packet type. If the type of packet is video, then a leading guard band shall be insert into TMDS data channel before the video stream.

After the leading guard band, the PAL shall encode the 8 bits video data into 10 bits format before mapping to the TMDS data channel. The 2-stage TMDS encoding process shall be performed before mapping into the TMDS data channel, starting with TMDS data channel 0 for the first video data byte in the video packet, as shown in the figure below.

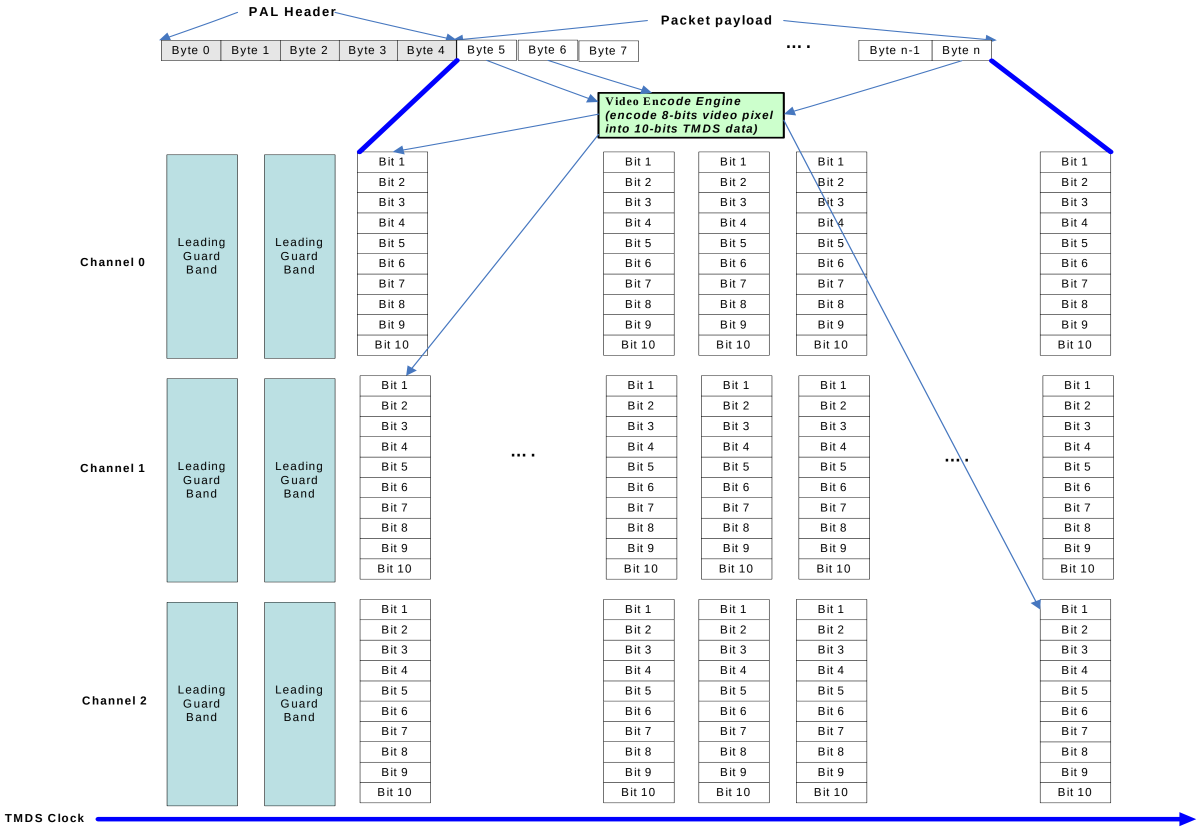


Figure 229 - Video packet distribution

17.3.2.2 Data (audio and auxiliary)

The receiver PAL shall receive and examine the PAL header for the packet type. If the type of packet is data, the leading guard band and trailing guard band shall be inserted into TMDS data channel before and after the data stream.

After the leading guard band the 10 bit data packet format needs to be unpacked into a TMDS data channel, starting with the TMDS data channel 0 for the first data byte in the data packet, as specified in the figure below.

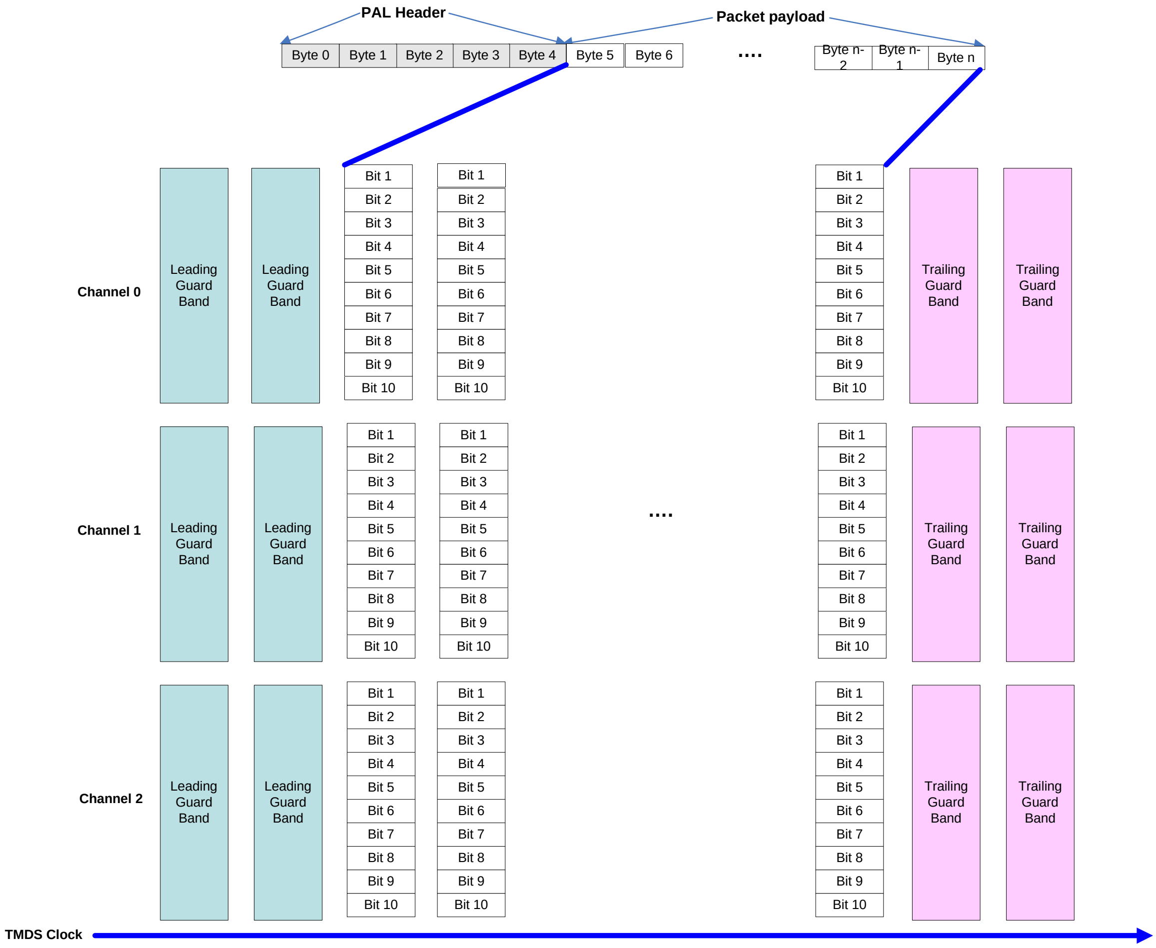


Figure 230 - Audio packet distribution

The following describes how the packet payload maps into 10 bit TMDS data channels.

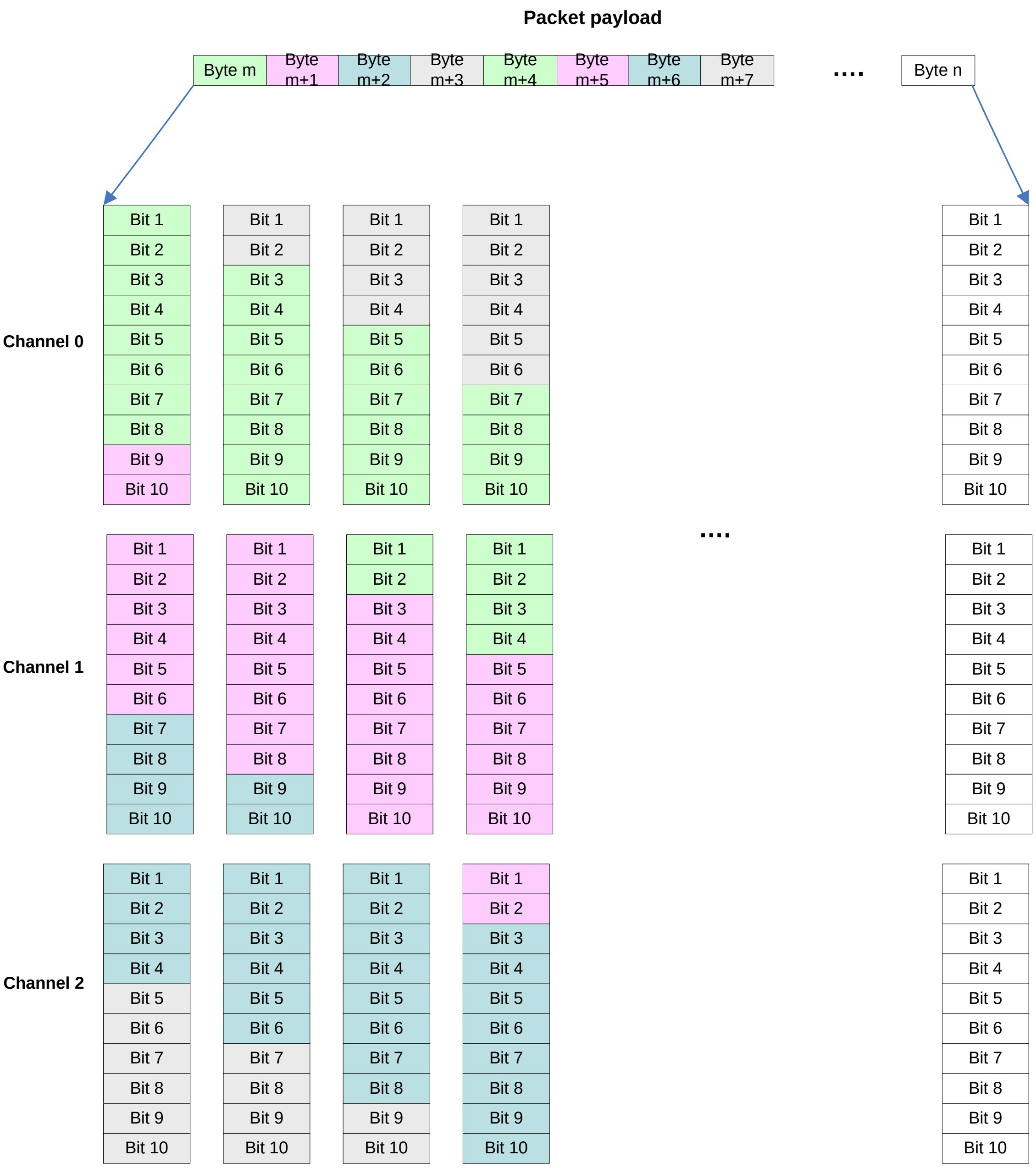


Figure 231 - Packet payload map

17.3.2.3 Control

The receiver PAL shall receive and examine the PAL header for the packet type. If the type of packet is control, a preamble shall be inserted into TMDS data channel after the control data, as specified in Figure 232.

The 10 bit format control data needs to be unpacked into TMDS data channels.

Figure 232 - Control packet map

17.3.2.4 DDC/CEC

The DDC/CEC packet will be prepended with the 4 byte PAL header, followed by the 6 byte Message Header to describe the type, access method, address and message length of the message.

The received DDC/CEC packet will be forwarded to either the Data Display Channel (DDC) or the CE Control channel base upon the information retrieved in the Message Header.

17.4 PAL header format

The PAL packet shall be formatted as specified in Figure 233.

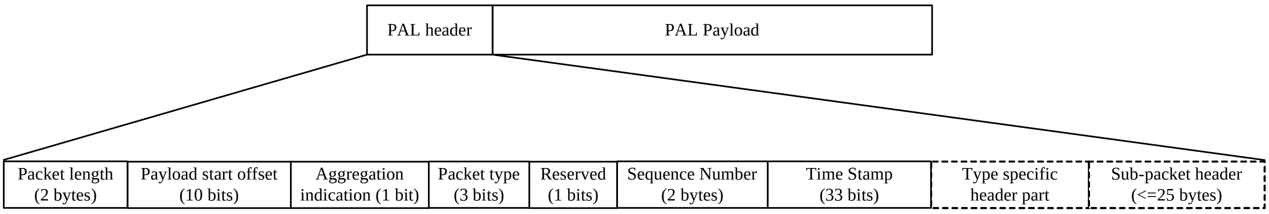


Figure 233 - PAL frame format

The packet length field indicates the length of the PAL packet in octets.

The packet start offset field indicates the starting position of the PAL packet payload in octets.

The packet type field in the PAL header shall be used only when there are no sub-packets in the PAL payload and shall be ignored when there are sub-packets in the PAL payload. The values for this field are given in Table 119.

Table 119 - Packet type

Packet Type	Description
0b00	Data
0b01	Video
0b10	Audio
0b11	CEC/DDC
0b100	Composite[1]
0b101-111	Reserved

[1] Composite refers to a packet containing more than one type of information (e.g. Uncompressed Video and Audio).

The aggregation indication bit is used to tell whether the PAL packet is an aggregated one or not. When the bit sets to "1", the Sub-packet information field shall exist within the PAL header which is formatted as specified in Figure 234.

The sequence number indicates the sequence number of the PAL packet.

The time stamp field is used for TMDS clock transfer and is obtained at PAL header generation time by loading the value from a 33 bit accumulator that is slaved to the TMDS clock frequency.

17.4.1 Sub-packet header

The construction of the sub-packets is shown in Figure 234. The number of sub-packets, *n*, shall be at most eight.

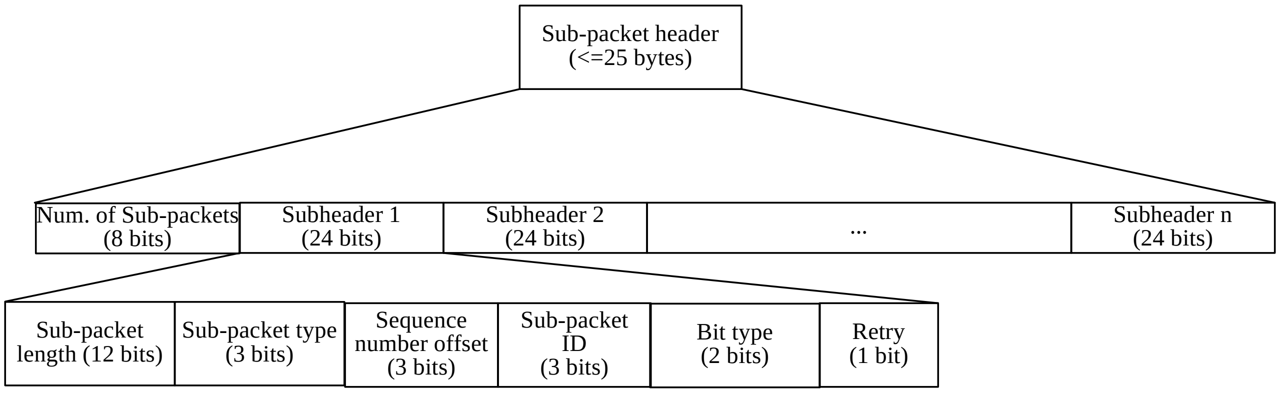


Figure 234 - The format of sub-packet information

The number of sub-packets indicates the total number of sub-packets within the PAL packet. In each subheader the:

- sub-packet length indicates the length of the sub-packet payload.
- sub-packet type indicates the type of the sub-packet as specified in Table 120.

Table 120 - Sub-packet type

Sub-packet Type	Description
0b000	Data
0b001	Video
0b010	Audio
0b011	CEC/DDC
0b100-111	Reserved

- sequence number offset indicates the three low order bits of the sequence number in the sequence number field of the PAL header.
- sub-packet ID indicates the identification number of the sub-packet within the PAL packet.
- bit type field indicates the information type of the bits at the sub-packet payload as specified in Table 121.

Table 121 - Bit type

Bit Type	Description
0b00	MSB/LSB combined
0b01	MSB only
0b10	LSB only
0b11	Reserved

- retry bit indicates whether the sub-packet is a re-transmitted sub-packet (the bit is set to "1" if it is a re-transmitted sub-packet).

17.5 PAL payload format

The PAL payload shall be formatted as specified in Figure 235 in case the aggregation indication field of the PAL header is set to 1 while there's no FCS field for non-aggregated PAL packet.

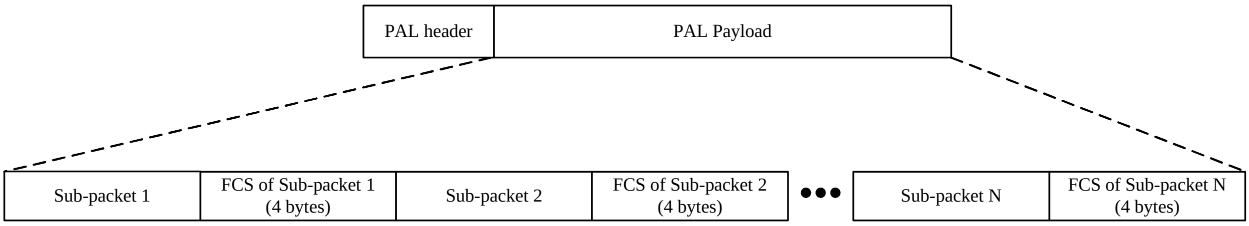


Figure 235 - PAL payload format

17.6 Block retransmission request

For PAL level block retransmission, the source PAL, which is supposed to transmit the aggregated PAL packet, shall request the source MAC to set its ACK policy as a No-ACK mode. The destination PAL, which is supposed to receive the aggregated PAL packet, shall request of the destination MAC that it transfer all the received frames, with the appropriate destination address, to the destination PAL even if they are in error.

The value of the number of sub-packets field shall be greater than zero except when the destination PAL requests a PAL level block retransmission of sub-packets. In this case the number of subpackets shall be set to zero (i.e. 0b00000000). The source PAL shall interpret this as meaning there are no following sub-headers and that the PAL payload body has a length of 32 octets and represents a sub-packet retransmission request bit map. In this retransmission request bit map, a value of "1" indicates that corresponding sub-packet needs to be retransmitted and a value of "0" indicates that corresponding sub-packet does not need to be retransmitted.

17.7 Type specific header fields

The type specific header field of the PAL header includes the header information specific to the PAL packet type. For uncompressed video, there are two UEP schemes: Sequential Packet Based UEP (S-UEP) and Parallelized Bit Based UEP (P-UEP) UEP. Type specific headers for S-UEP and P-UEP are specified in clauses 17.7.1 and 17.7.2.

17.7.1 S-UEP

The format of S-UEP PAL packet is specified in Figure 236.

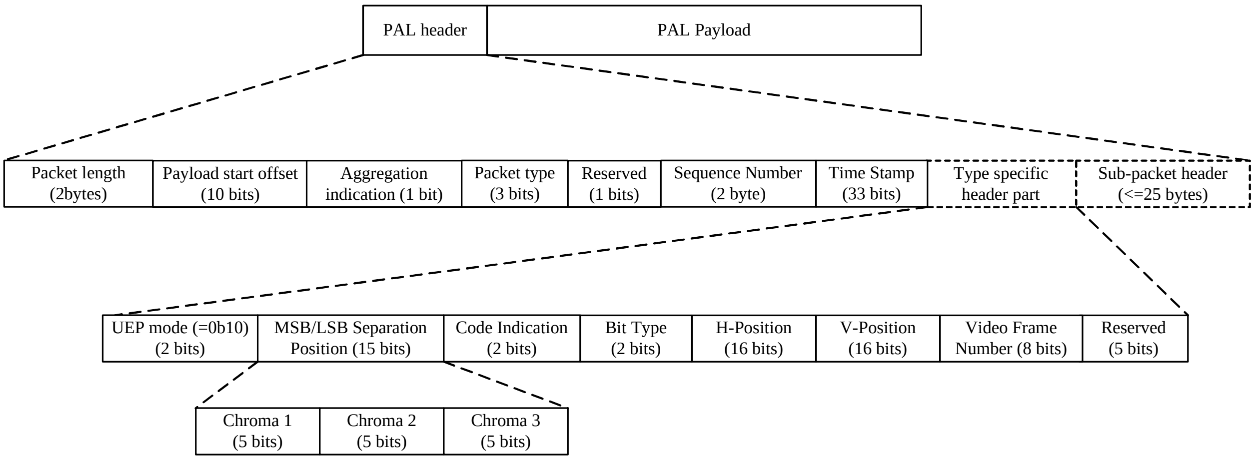


Figure 236 - S-UEP PAL packet format

The UEP mode definition is specified in Table 99 and set to 0b01 for S-UEP. The MSB/LSB separation field indicates the separating position of MSBs and LSBs for each chroma.

The chroma field within the MSB/LSB separation position field contains the separating position of MSBs and LSBs for each chroma as follows. Valid values $(0 \le n \le 31)$ are shown in Table 122, where X indicates colour depth per chroma.

Table 122 - Each chroma field encoding

5 bit value range	Interpretation
n=0	All bits are MSB
0<n<X	bits b_0 to b_{n-1} are LSB bits b_n to b_{X-1} are MSB
n=X	all bits are LSB
X+1≤n≤31	Reserved

The Code Indication field specifies which code is used for protecting the segregated MSB or LSB data. Code Indication field is shown in Table 123.

Table 123 - Code indication field

The value of Code Indication	Description
b00	None
b01	(7,4) Hamming Code
b10-b11	Reserved

The Bit Type field is specified in Table 121.

H-Position indicates the horizontal position of first video pixel in the payload of the PAL packet.

V-Position indicates the vertical position of first video pixel in the payload of the PAL packet.

The video frame number indicates the number of the frame to which the PAL packet payload belongs.

In order to support S-UEP functionality, the PAL shall include the procedures consisting of the following steps as a transmitter or a receiver.

The source PAL - which is in charge of the transmission - shall follow the next 4 steps.

In the first step, once the source PAL entity receives video streams from upper layer, it shall segregate MSB data from LSB data respectively for each video chroma such as RGB or YCbCr video data or Encrypted Video data by HDCP regardless of colour depth as shown in Figure 237. In addition, there is no limitation of partitioning of video data. In other words, the MSB/LSB separation position can be dynamically applied as 3M:5L/ 4M:4L/5M:3L for instance.

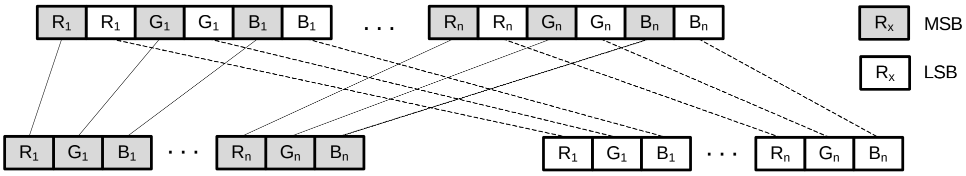


Figure 237 - The procedure framing MSB/LSB separately

In the second step, the source PAL entity shall protect the gathered MSB data by applying the code in accordance with the Code Indication field within PAL header. Figure 238 illustrates the (7, 4) hamming code is used for protecting MSB data for instance; however, if the Code Indication field is set to 0 then this step shall be skipped.

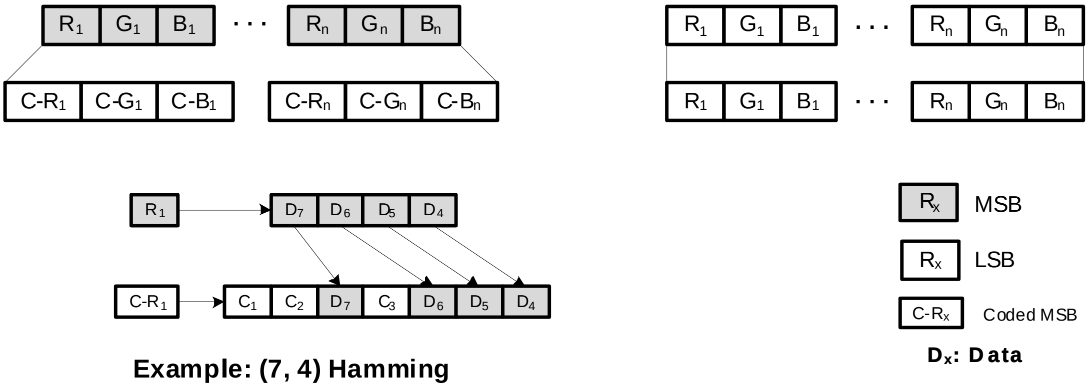


Figure 238 - The procedure adding some parity bits to the gathered MSB data

In the third step, the source PAL entity shall generate the MSB packet including only MSB data and the LSB packet consisting of only LSB data. Figure 239 specifies the MSB video packet consisting of only MSB data and the LSB packet consisting of only LSB data.

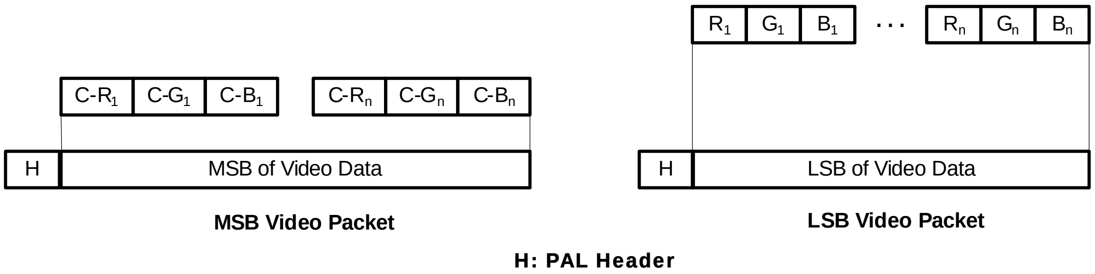


Figure 239 - The procedure generating the different MAC video frames

In the last step, the source PAL entity shall transfer a MSB or LSB packet or an aggregated packet to the lower layer through the SAP along with the primitive indicating whether it is a MSB packet or a LSB packet. The lower layer shall apply MCS for each packet or each subframe within an aggregated packet according to the corresponding Bit Type information, given by the PAL regarding each packet or each subframe. In this step, the different MCSs can be applied, which is determined by the lower layer. Then, the frame or the aggregated frame is sent to the peer device over the air.

The destination PAL - which is in charge of reception - shall follow the next 4 steps as well.

In the first step, the destination PAL receives a MSB or LSB packet or an aggregated packet from the lower layer via the SAP

In the second step, the destination PAL shall decode the packet using the corresponding coding mechanism according to the value of the Coding indication field within PAL header. If the Coding indication field is set to 0, then this step shall be skipped.

In the third step, the destination PAL shall put the MSB packet and the LSB packet together in order to build or recover the complete packet or video chroma. For this, the destination PAL shall put them together by regarding two packets with the different Bit Type and the same V-Position, H-Position and Video Frame Number as the packets consisting of the original video chroma. In addition, the destination PAL shall consider the value of MSB/LSB separation position field as well when it builds the original packet.

In the last step, the destination PAL shall transfer the PAL payload as a video stream to the upper layer.

17.7.2 P-UEP

The frame format to support P-UEP shall be formatted as specified in Figure 240.

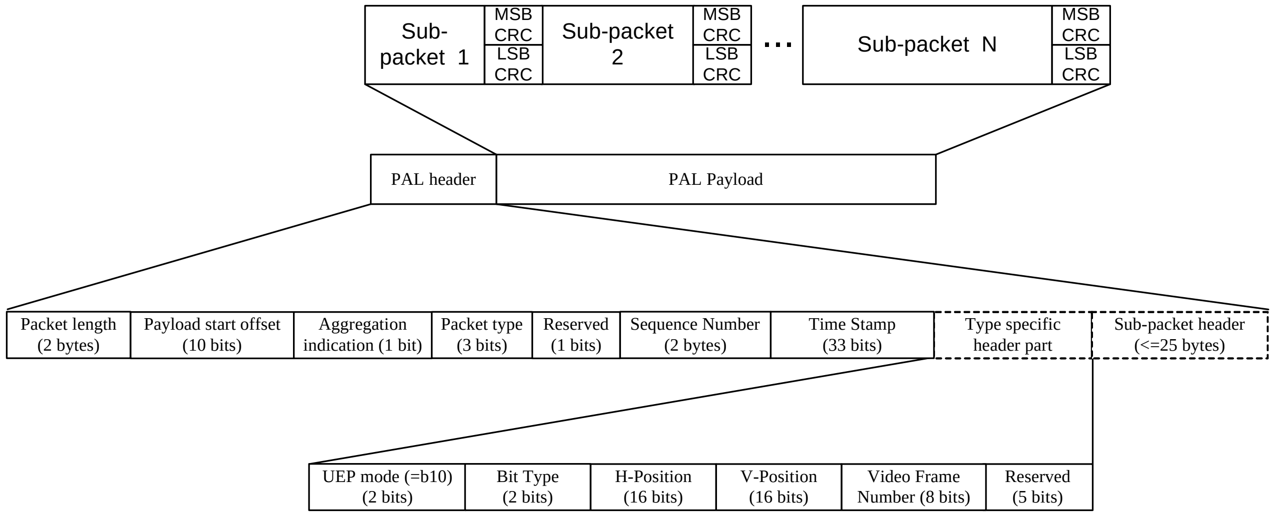


Figure 240 - P-UEP frame format

The UEP mode definition is shown in Table 121 and set to 0b10 for P-UEP.

Bit type field is specified in Table 121.

H-Position indicates the horizontal position of first video pixel in the payload of the PAL packet.

V-Position indicates the vertical position of first video pixel in the payload of the PAL packet.

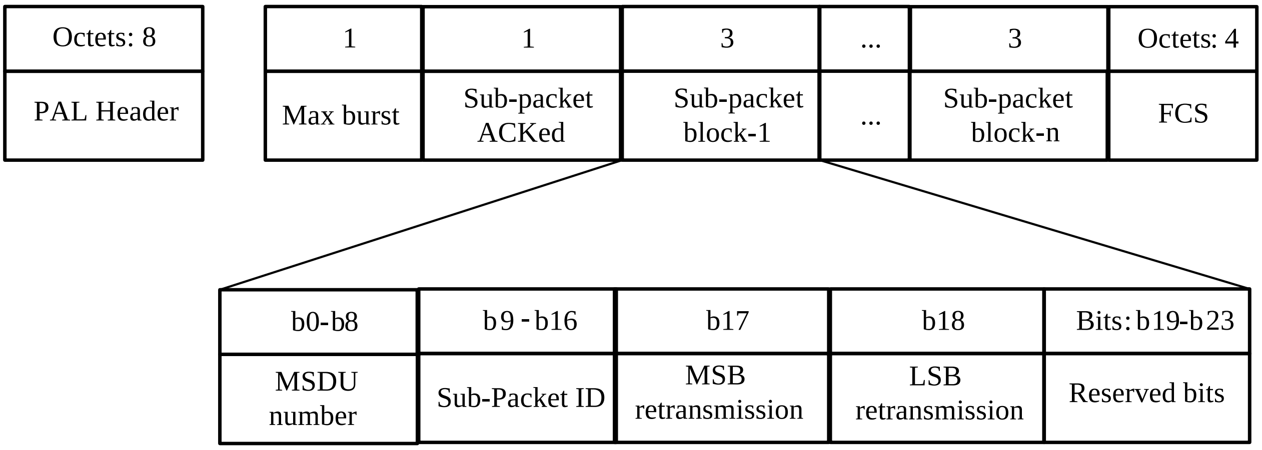
The video frame number indicates the number of the frame to which the PAL packet payload belongs.

Each sub-packet can carry control packet, data, audio, uncompressed video frame and etc. The header part uses the most robust modulation and coding scheme (MCS). Different sub-packets can use different MCSs. Video sub-packets use UEP MCS for MSBs and LSBs. Each sub-packet can also be divided into multiple smaller micro units with separate CRCs.

A selective ACK and re-transmission scheme shall be used for P-UEP video. The block ACK frame shall be formatted as specified in Figure 241.

For each sub-packet, there are separate CRCs for MSBs and LSBs of video information, i.e., the LSB CRC field contains the CRC calculated over the LSBs of video information and the MSB CRC field contains the CRC calculated over the MSBs of video information. When the transmitted sub-packet includes MSB CRC and LSB CRC, the receiver recognizes which part has an error so that it sends the information back to the transmitter using the Block-ACK, which is used at the transmitter for efficient retransmission.

If the transmitted sub-packet contains only MSB retransmission, the LSB retransmission indication in the Block-ACK shall be set to 0 and shall be ignored at the transmitter. Likewise, if the transmitted sub-packet contains only LSB retransmission, the MSB retransmission indication in the Block-ACK shall be set to 0 and shall be ignored at the transmitter.



Octets: 8	1	1	3	...	3	Octets: 4
PAL Header	Max burst	Sub-packet ACKed	Sub-packet block-1	...	Sub-packet block-n	FCS

b0-b8	b9 - b16	b17	b18	Bits: b19-b23
MSDU number	Sub-Packet ID	MSB retransmission	LSB retransmission	Reserved bits

Figure 241 - Block ACK frame format

To support P-UEP functionality, the PAL at the source side and destination side shall have the following procedure.

When the source PAL entity starts to receive a HDMI stream including uncompressed video and other possible information such as audio and control, it shall generate aggregated PAL packets which include video sub-packets and possibly other types of sub-packets as specified in Figure 240. Then the source PAL entity shall pass the PAL packets down to the lower layer through the SAP along with the primitive parameter information such as MCS modes for individual sub-packets. More robust modulation or coding scheme is applied to MSB bits than LSB bits in each video sub-packet.

When the destination PAL entity receives packets through SAP from the low layer, it shall de-packetize different sub-packets and re-construct the HDMI stream.

17.8 Video/audio format adaptation

Video/audio format adaptation shall be done to accommodate wireless link degradation. Usually it takes several seconds to switch the A/V formats at the video input and playback modules. To solve this long delay problem, this clause describes the fast format adaptation schemes at the wireless transmitter and receiver modules. These schemes are transparent to the application input and output formats.

17.8.1 Fast video format adaptation

Figure 242 illustrates the fast video format adaptation procedure.

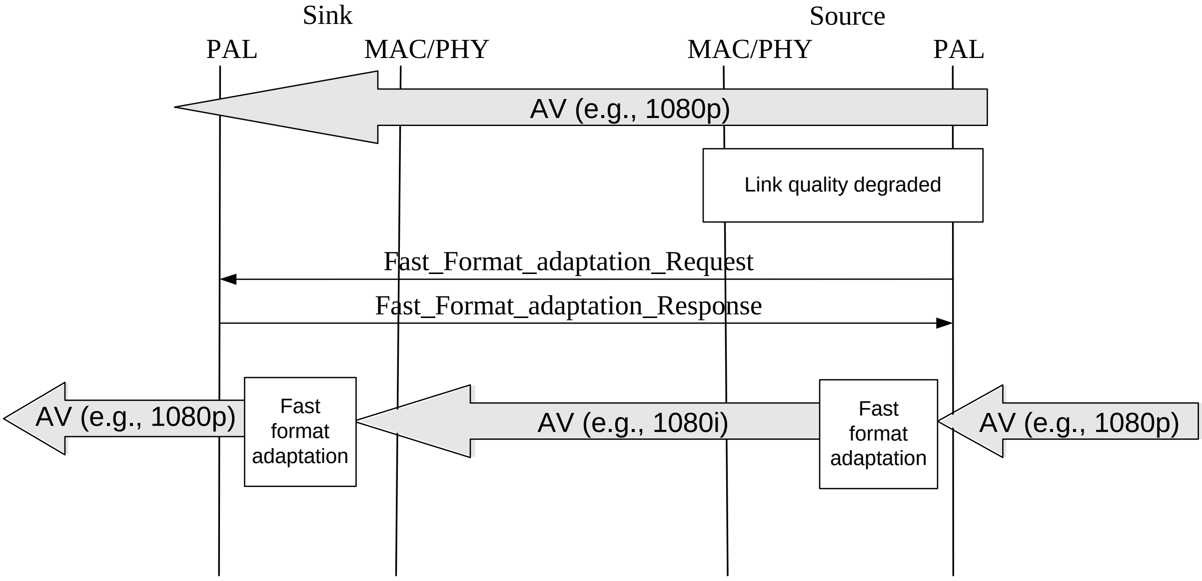


Figure 242 - Fast format adaptation procedure

17.8.1.1 Format adaptation before packetization

As shown in Figure 243, format adaptation can be done before pixel partitioning and packetization if the link quality degrades. Format adaptation will be used to fit into the reduced rate immediately after the video pixels enter the wireless module.

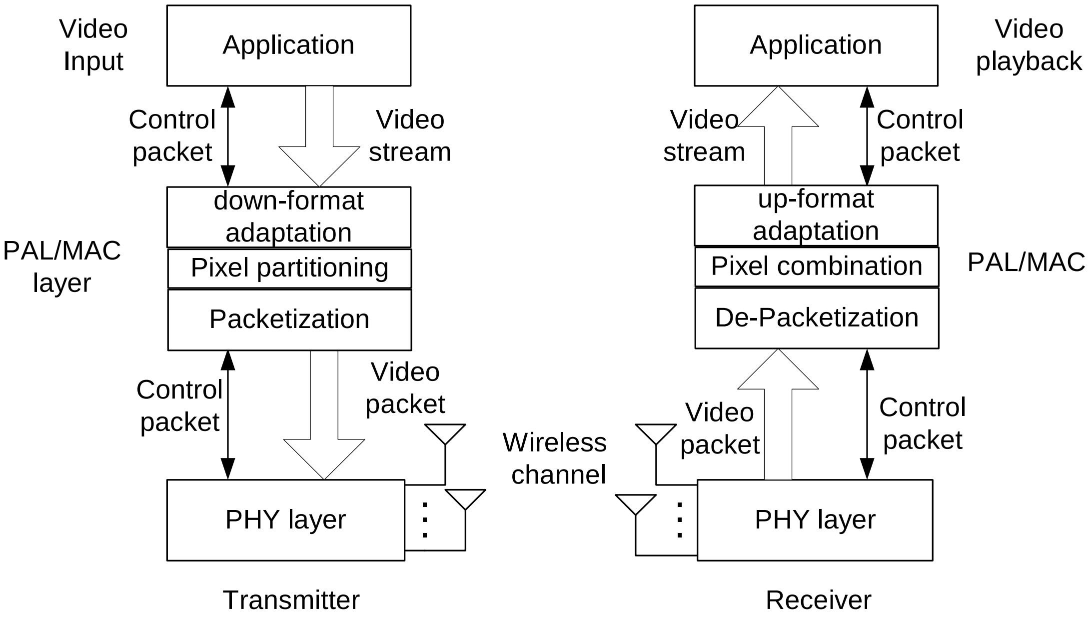


Figure 243 - Format adaptation before packetization

17.8.1.1.1 Pixel resolution reduction

The down-format adaptation function block at the sender side can discard some pixels to fit into the new video format. As a first example, in the case of conversion from 1080p (1920*1080 progressive) to 1080i ((1920*1080 interlaced), for a 1080p frame with even frame number, the format adaptation block will keep the even number line of pixels but discard the odd number line of pixels; however, for a 1080p frame with odd frame number, the format adaptation block will keep the odd number line of pixels but discard the even number line of pixels. The second example, in the case of conversion from 1080p (1920*1080 progressive) to 480p (640 *480 progressive), for each 1080p video frame: horizontally, the format adaptation block will keep one pixel among three neighbouring pixels in each line or calculate the average value of every three neighboring pixels; vertically, the format adaptation block will choose 4 lines from every 9 lines, for instance, the 2nd, 4th, 6th, and 8th line from each group of line 1 to 9. The up-format adaptation function block at the receiver side will recover the required original video format by copying pixel from neighbours horizontally or vertically and send the recovered video streams to display devices such HDTV.

17.8.1.1.2 Frame updating frequency reduction

The down-format adaptation function block can simply discard video frames periodically to adapt to the new frequency. The up-format adaptation function block at the receiver side will recover the required original video format by copying frames.

17.8.1.1.3 Colour depth reduction

The down-format adaptation function block will reduce the bit number for each pixel component. For example, in the case of conversion of colour depth from 10 bits to 8 bits, the down-format adaptation block just conduct 2 bits right shifting operation for each pixel component (R, R, B or Y,U, V). The up-format adaptation function block at the receiver side will recover the required original video format by doing 2 bits left shifting operation for each pixel component.

17.8.1.1.4 Combination of different rate reduction methods

Different combinations of the above three modes can be done together reduce data rate and also minimize video quality degradation. For example, the format adaptation function block may reduce the pixel resolution as well as reduce the frame update frequency rate to fit into the available data rate at the PHY layer.

17.8.1.2 Format adaptation after pixel partitioning and packetization

As shown in Figure 244, format adaptation is done after pixel partitioning and packetization, if the link quality degrades.

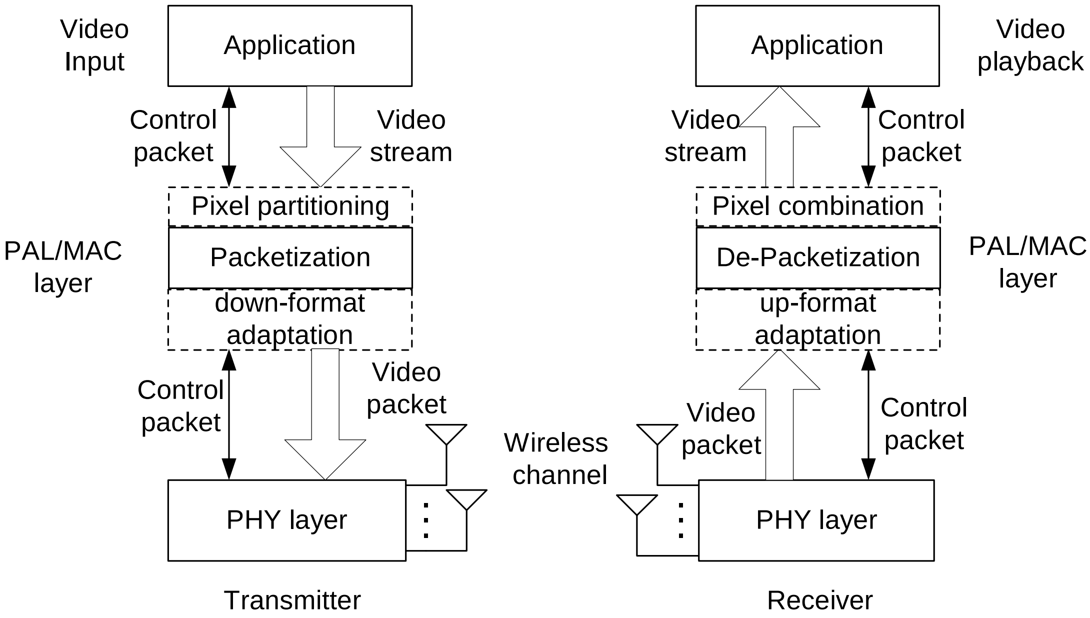


Figure 244 - Format adaptation after packetization

At the transmitter side, different pixel partitions will be put into different packets or sub-packets. If the link condition requires video data rate reduction, the transmitter can just simply discard packets or sub-packets of one or multiple pixel partitions. For example, as shown in Figure 245, 4 partitions are put into 4 separate sub-packets. When the PHY MCS reduced to half rate mode, the transmitter can simply discard two sub-packets (partitions) from the four to fulfil the rate reduction requirement. In the case that each sub-packet should keep the same duration regardless MCS, the two sub-packets after format adaptation operation may need to be further split into 4 sub-packets for transmission as specified in Figure 245.

At the receiver side, the discarded sub-packets can be re-constructed by copying from other sub-packets.

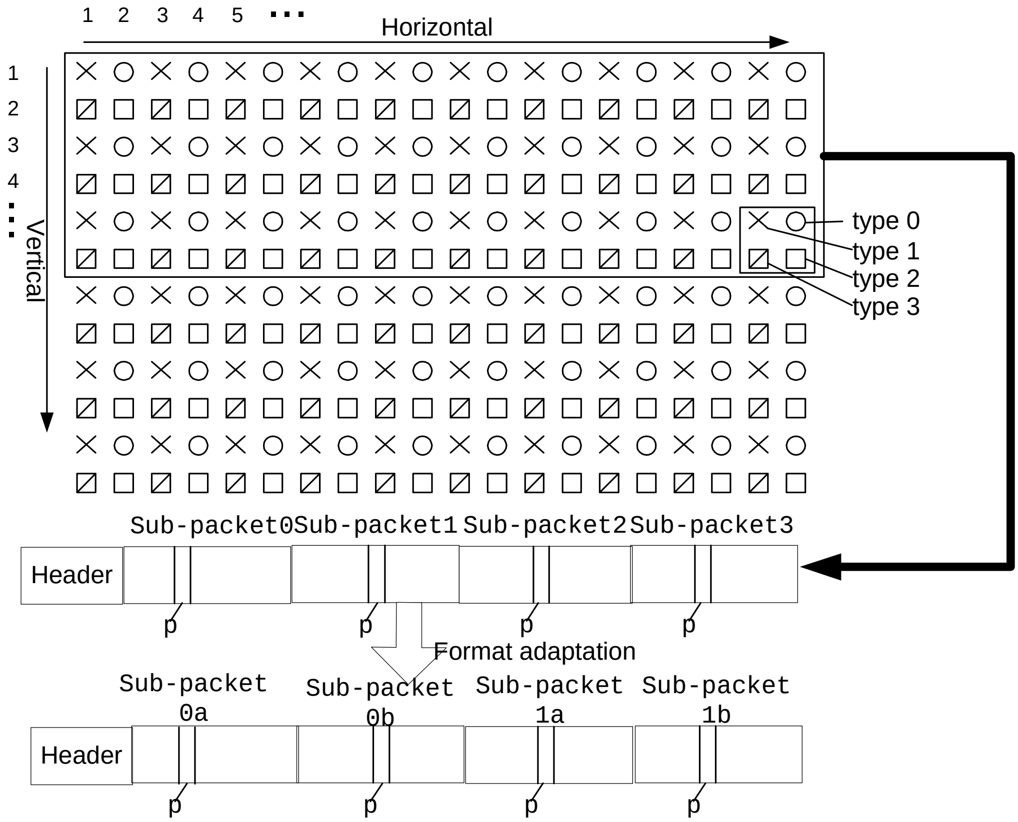


Figure 245 - An example of Pixel partitioning, packetization and format adaptation

17.8.2 Fast audio format adaptation

17.8.2.1 Audio channel number reduction

The down-format adaptation function block may discard audio samples from some channels to reduce data rate. The up-format adaptation function block at the receiver side will reconstruct all audio channels by copying audio samples of other channels.

17.8.2.2 Audio sampling rate/frequency reduction

The down-format adaptation function block may discard audio samples periodically or do down-sampling to adapt to the new sample rate. The up-format adaptation function block at the receiver side will recover the required original audio sampling rate by copying audio samples or by up-sampling or interpolation.

17.8.2.3 Audio channel sample depth (bits per sample) reduction

The down-format adaptation function block will reduce the number of bits for each audio channel sample. For example, in the case of conversion of sample depth from 10 bits to 8 bits, the down-format adaptation block just conducts 2 bits right shifting operation for each audio channel sample. The up-format adaptation function block at the receiver side will recover the required original audio format by doing 2 bits left shifting operation for each audio channel sample.

17.8.2.4 Combination of different rate reduction methods

Different combinations of the above three modes may be used to reduce data rate and also minimize audio quality degradation. For example, the format adaptation function block may reduce the channel number as well as reduce the audio sampling rate to fit into the available data rate at the PHY layer.

17.8.3 Control messages to support fast format adaptation

17.8.3.1 Fast format adaptation request message

The FAST_FORMAT_ADAPTATION_REQUEST message shall be formatted as specified in Figure 246. The sink uses this message to request the source to perform video and/or audio format adaptation.

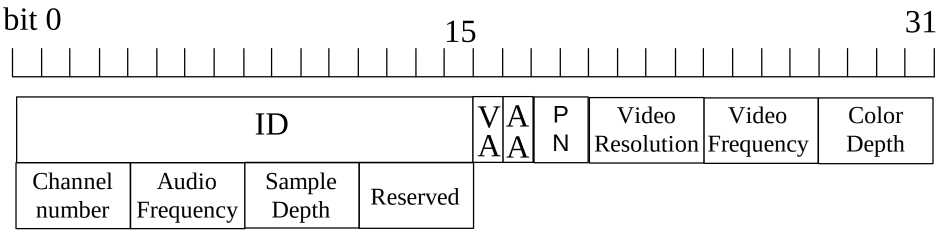


Figure 246 - Format of FAST_FORMAT_ADAPTATION_REQUEST message

The VA field indicates whether video format adaptation is requested or not.

The AA field indicates whether audio format adaptation is required or not.

The PN field indicates how many pixel partitioning are kept after video adaptation.

The Video resolution field indicates the sub-sampling rate for pixels.

The Video frequency field indicates the sub-sampling rate for video frame updating frequency.

The Colour depth field indicates how many bits are used for each pixel component.

The Audio channel number field indicates the number of audio channels of the new audio format.

The Audio frequency field indicates the sub-sampling rate for audio frequency.

The Sample depth field indicates how many bits are used for each audio sample at each audio channel.

17.8.3.2 Fast format adaptation response message

The FAST_FORMAT_ADAPTATION_RESPONSE message shall be formatted as specified in Figure 247. This message is sent by the source to report to the sink whether the source accepts format adaptation.

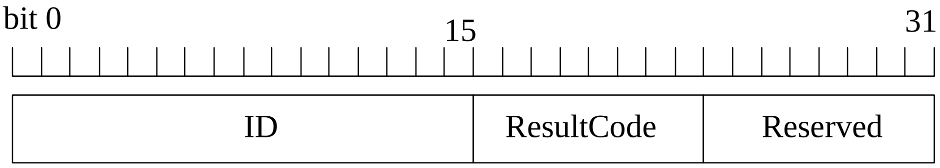


Figure 247 - Format of FAST_FORMAT_ADAPTATION_RESPONSE message

The ResultCode field is set to indicate the result of the request. Valid values are shown in Table 124.

Table 124 - Result Code field

Code	Meaning
0x00	Accepted
0x01	Rejected
0x02 - 0xFF	Reserved

17.9 Fast uplink channel allocation (FUCA)

Video streaming is an asymmetric point-to-point application in that a source (DVD player, set top box and so on) tends to transmit video streaming data to a sink (flat TV) in an unidirectional manner. In this case, it is difficult for a sink to “transmit” control signals or user data, such as HDMI Display Data Channel packets, due to the lack of available uplink time slots. Even though the vertical blanking interval can be used for uplink transmission, it's often not adequate for the amount of control and/or user data that needs to be sent. For instance, when it's necessary for a device to support the immediate acknowledgement for anti-blocking functionality, the delay associated with the periodic occurrence of the vertical blanking interval is too long. To mitigate this problem, this clause introduces the fast uplink channel allocation method (FUCA) which makes use of the horizontal blanking interval instead of the vertical blanking interval.

It turns out that the use of the horizontal blanking interval for data transmission can also be problematic due to its small time duration. In addition, at the source side, a device will require at least several µs for RF switching from TX to RX, or visa versa, for which the horizontal blanking interval is not long enough. For example, Figure 248 shows that the expected time for Tx-Rx switching is at least 4 µs.

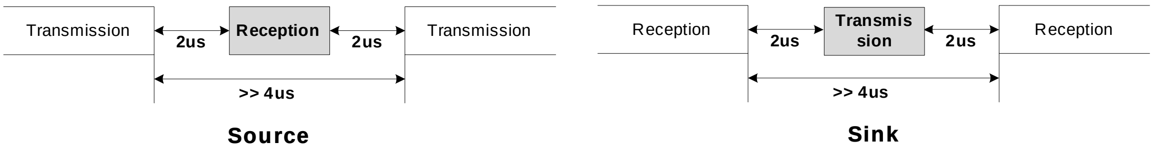


Figure 248 - Example of Tx-Rx switching time in terms of RF

Table 125 shows the real value of Hsync and Vsync for various video formats.

Table 125 - Hsync and Vsync values for various video formats

Format	V Freq	HRES	VRES	DE_CNT	DE_LIN	Rate 24bit	Rate 20bit	Hsync (us)	Vsync (ms)
640x480	60	800	525	640	480	1920	1600	6.34921	1.429
640x480	72	832	520	640	480	1920	1600	6.16371	1.6068
640x480	75	840	500	640	480	1920	1600	6.34921	0.533
800x600	60	1056	628	800	600	2400	2000	6.43376	0.743
800x600	72	1040	666	800	600	2400	2000	4.8125	1.376
800x600	75	1056	625	800	600	2400	2000	5.17172	0.533
1024x768	60	1344	806	1024	768	3072	2560	4.92339	0.786
1024x768	70	1328	806	1024	768	3072	2560	4.05735	0.674
1024x768	75	1312	800	1024	768	3072	2560	3.65854	0.533
1024x768	85	1376	808	1024	768	3072	2560	3.72472	0.582
1280x1024	60	1708	1066	1280	1024	3840	3200	3.91785	0.657
1280x1024	75	1708	1066	1280	1024	3840	3200	3.13428	0.525
1280x1024	85	1728	1072	1280	1024	3840	3200	2.84525	0.527
480i	59.94	1716	262/263	720	480	2160	1800	36.852	2.802/ 2.918
480p	60	858	525	720	480	2160	1800	5.10601	1.429
720p	60	1650	750	1280	720	3840	3200	4.98316	0.667
1080i	30	2200	562/ 563	1920	1080	5760	4800	3.77104	0.652/ 0.681
1080p	30	2200	1125	1920	1080	5760	4800	3.77104	0.1333
1080p	60	2200	1125	1920	1080	5760	4800	1.88552	0.667

17.9.1 FUCA operation

In general, the horizontal blanking interval and the active video line of video transmission systems are repeated alternatively as shown in Figure 249, where H stands for horizontal blanking interval and D stands for data. In this case, the length of the horizontal blanking period is too short to transmit any meaningful data in the direction of the sink to source if the Rx-Tx switching time is included.

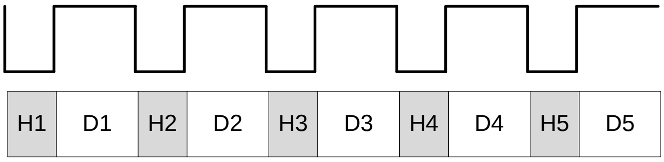


Figure 249 - Distribution of blanking interval in video transmission system

However, separating the active lines (AL) from the horizontal blanking intervals (HBI) and aggregating them respectively enables a sink to send a short amount of data back to the corresponding source. To this end, the FUCA map shall be generated by the source PAL according to information on aggregated data and delivered to the MAC followed by the transmission of the corresponding aggregated data. The FUCA map is specified in Figure 250.

Figure 250 shows the FUCA map format of 40 bits. The Pos (Position of combined HBI) field indicates where the combined HBI is located. The value ZERO means "Before the combined AL", the value ONE means "After the combined AL". Each HBI OA (Order and Attribute) Block field consists of Order of HBI field and Attribute of HBI field as specified in Figure 250. The Attribute of HBI field indicates the data type transmitted in the corresponding HBI. In other words, "0" means "No auxiliary data" and "1" means "Auxiliary data". The Order of HBI field indicates the exact location of corresponding HBI. The number of Aggregated Data specifies the total number of aggregated data which is limited to 8.

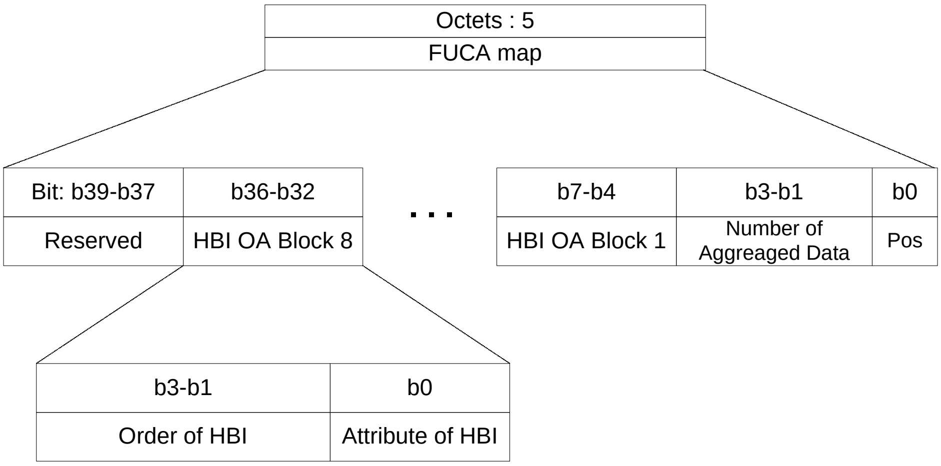


Figure 250 - FUCA map format

There are two ways to combine the HBI and AL as shown in Figures 251 and 252.

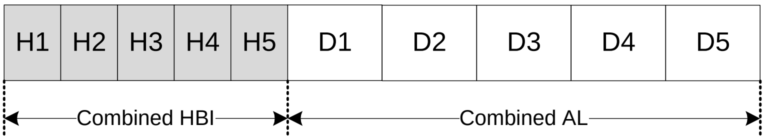


Figure 251 - Aggregation of HBI and AL before combined AL

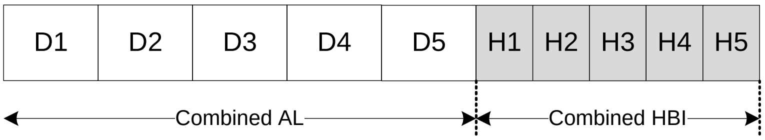
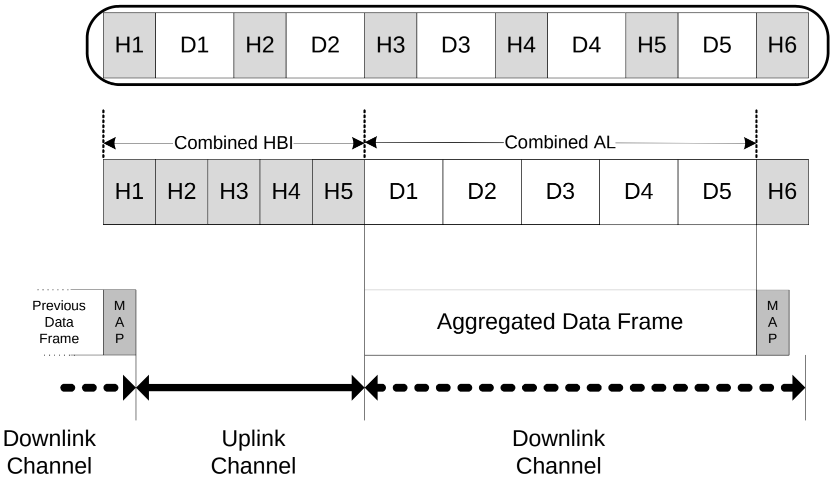


Figure 252 - Aggregation of HBI and AL after combined AL

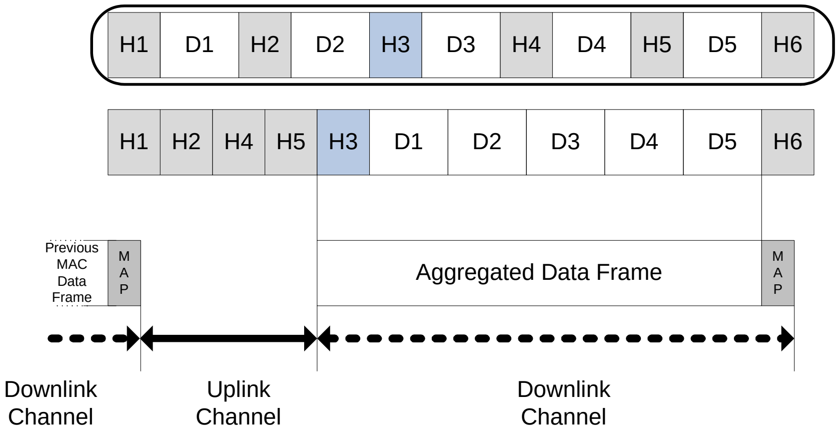
In the following paragraphs, a few cases using FUCA functionality are introduced. In the case of Figure 253, it is possible to use combined HBI for transmitting uplink control/data because there are 5 HBIs in series but no auxiliary data. In Figure 253, OA stands for "Order and Attribute", E stands for "Empty" horizontal blanking and E1-E2-E3-E4-E5 indicates the successive 5 empty HBIs.



Pos: 0; Number of Aggregated Data: 5; HBI OA: E1-E2-E3-E4-E5

Figure 253 - Example use of combined HBI

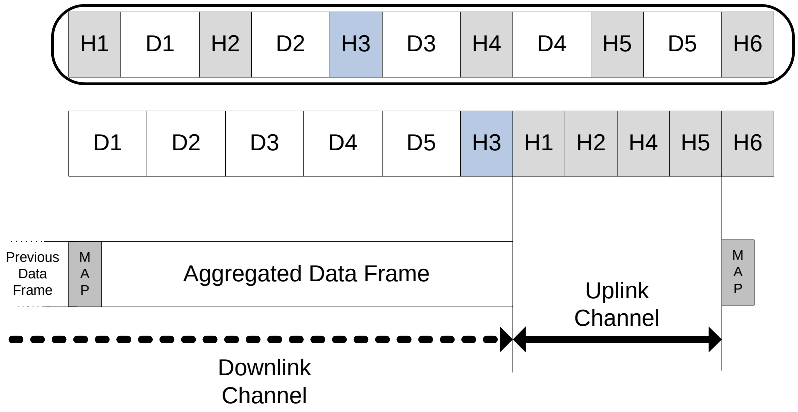
In the case of Figure 254, it's possible to use combined HBI for transmitting uplink control/ data because there is one auxiliary data in the third horizontal blanking interval (HBI OA: E1-E2-E3-E4-A3) where A stands for "Auxiliary data".



Pos: 0; Number of Aggregated Data: 5; HBI OA: E1-E2-E3-E4-A3

Figure 254 - Second example use of combined HBI

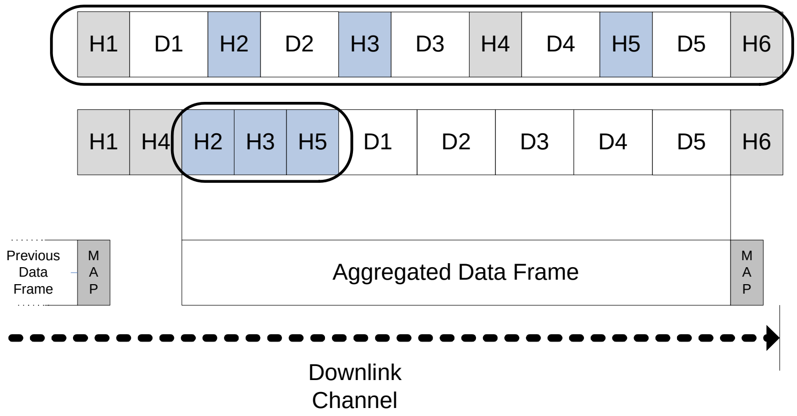
In the case of Figure 255, it is possible to use combined HBI for transmitting uplink control/ data similar to Figure 254. The only different is that the combined HBI follows the combined AL.



Pos: 1; Number of Aggregated Data: 5; HBI OA: A3-E1-E2-E4-E5

Figure 255 - Example use of combined HBI in case that HBI follows AL

In the case of Figure 256, it is not possible to use combined HBI for transmitting uplink control/data because three of the horizontal blanking intervals already contain auxiliary data.



Pos: 0; Number of Aggregated Data: 5; HBI OA: E1-E4-A2-A3-A5

Figure 256 - Example use of combined HBI

17.9.2 The procedure of data exchange using FUCA

Figure 257 shows the procedure of data exchange using FUCA. The source device collects m number of consecutive data frames and aggregates them as a single frame. Then, the source device creates the FUCA map based on the information of the horizontal blanking intervals. The source device shall send the FUCA command frame before transmitting the aggregated data frame. The FUCA command frame consists of the Next Frame Duration field, Sequence Number field and FUCA map field as described in Table 17.9.1. After sending the FUCA command frame, the source device shall send the data frame to the sink device.

The sink device shall compare the Sequence Number of the received FUCA command frame with that of the aggregated data frame received, after receiving the FUCA command frame, and decide whether the FUCA command frame is valid or not. If the Sequence Numbers between the two are the same, the FUCA map is considered as acceptable; otherwise, the sink device shall ignore the FUCA map.

In the latter case, the sink device doesn't make use of the uplink channel.

In the former case, the sink device uses the uplink channel as indicated in the FUCA map to transmit data from the sink device to the source device.

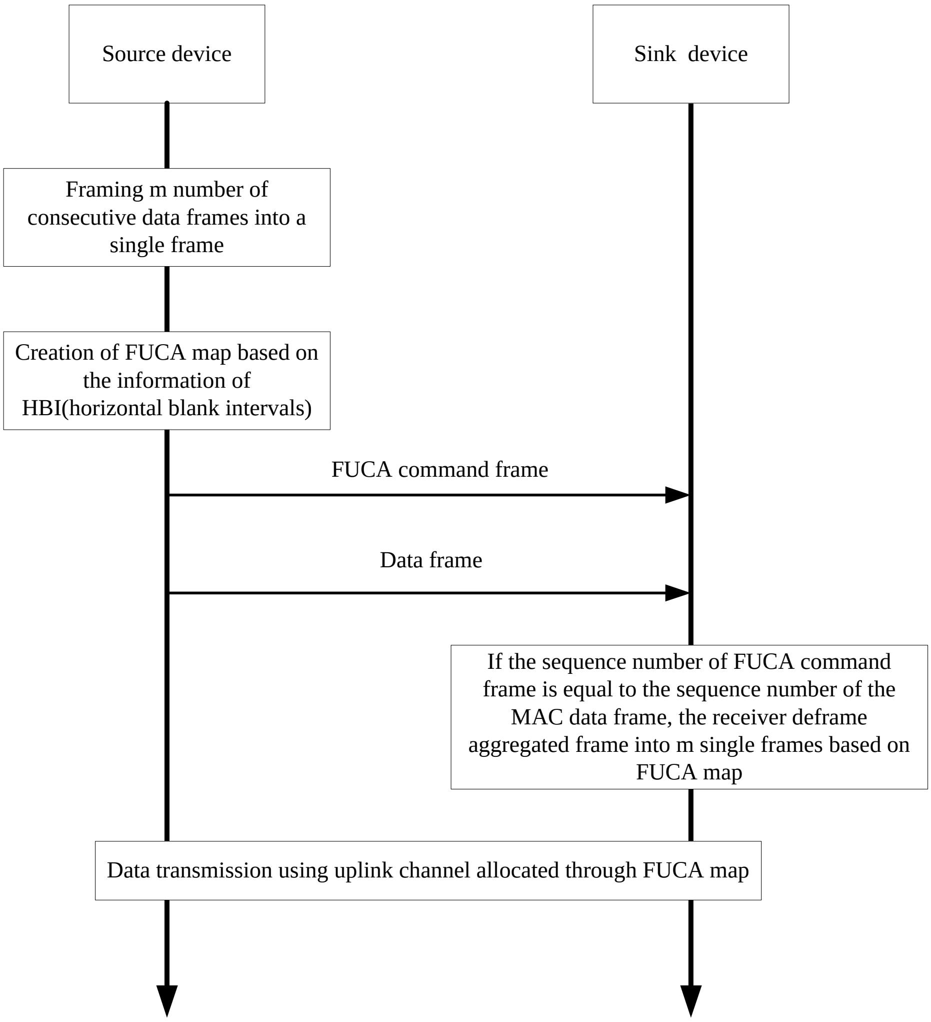


Figure 257 - Data exchange procedure for using FUCA

18 Out-of-band control channel

The out-of-band (OOB) control channel based on IEEE 802.11g operating on 2.4 GHz is optional for providing WPAN management and control with omni-directional transmission. A device indicates if it supports the OOB control channel by setting the OOB bit in the MAC Capabilities IE, as defined in clause 14.7.14.

Figure 258 illustrates the protocol architecture with OOB control channel. The high rate PHY/ MAC and low-rate OOB PHY/MAC with omni-directional transmission operate on the 60GHz and 2.4 GHz bands, respectively. This chapter describes a MAC convergence sub-layer which is used to coordinate the low-rate OOB and high-rate 60 GHz PHYs/MACs and to support functions such as scheduling and synchronization, device & service discovery, association control, and bandwidth reservation for the 60 GHz channels.

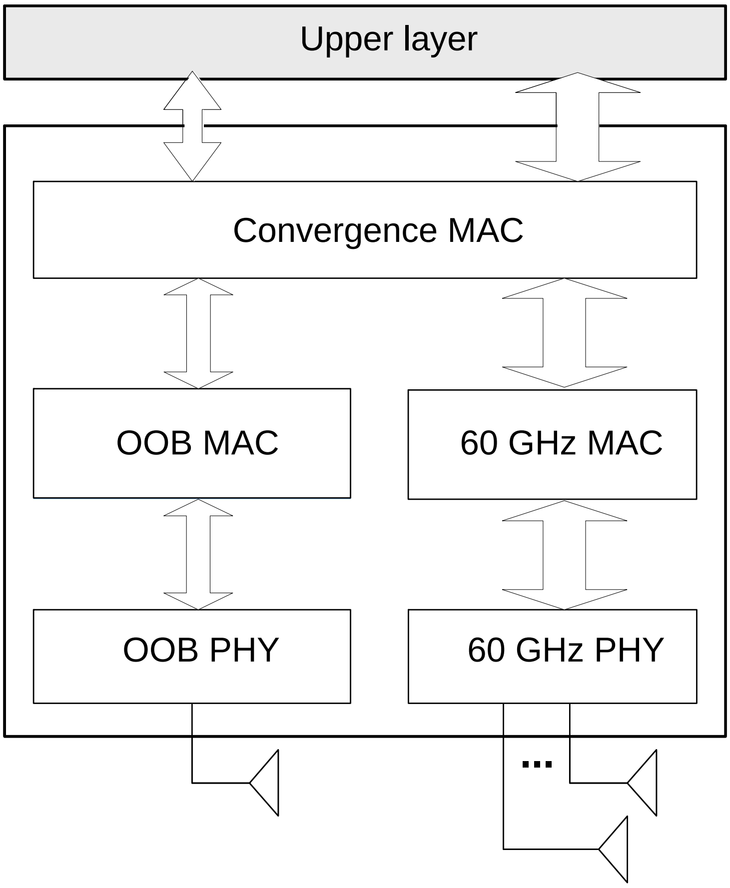


Figure 258 - Protocol architecture with OOB control channel

18.1 OOB operation

There are two OOB operation modes: ad hoc and infrastructure. The mode shall be indicated in the beacon control field of the OOB beacon.

Note: In this subclause, an initiator is the device that initiates the exchange of control messages with the other device. It can be either the source or destination of data transmission. A responder is the device that responds to the initiator. It can also be either the source or destination of data transmission depending on the role of the initiator in data transmission.

18.1.1 Ad hoc mode

In ad hoc mode, there is no coordinator and devices within the transmission range can communicate with each other, as shown in Figure 259. Each device periodically sends out OOB beacons in an OOB control channel.

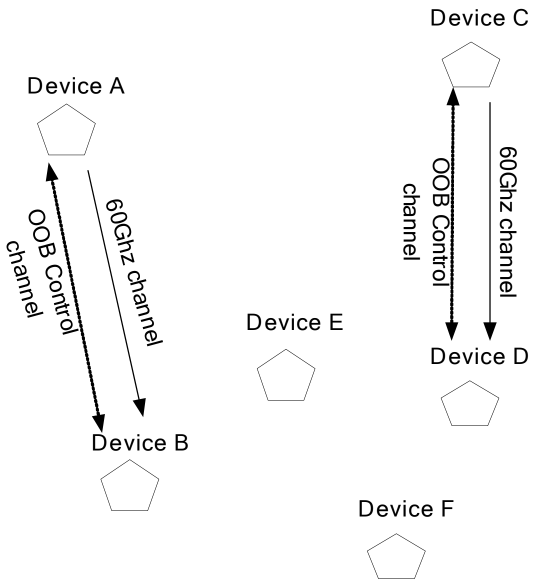


Figure 259 - Topology configuration for ad hoc transmissions

18.1.1.1 Device discovery with OOB control channel

Devices that support the OOB control channel shall follow the two-step device discovery procedure as follows:

A. Device discovery in the OOB control channel as specified in this subclause. The two devices then exchange their 60 GHz capabilities through OOB-60GHzCapabilityRequest and OOB-60GHzCapabilityResponse in the OOB control channel.

B. Device discovery at 60GHz as specified in clause 15.3.

During the step of device discovery in the OOB control channel, two devices that support OOB shall try to discover each other in the OOB control channels specified in a predefined order, starting with the default OOB control channel (i.e., Channel 6 of IEEE 802.11g at 2.4 GHz). Detailed procedure is specified as follows:

1) A device shall scan the OOB control channel for at least mMaxBeaconIntervalTime, measure the peak energy and analyze OOB beacons and other frames if any.

2) If the device receives an OOB beacon sent from the other device, it shall respond with an OOB beacon. If the other device receives the responding OOB beacon, then the two devices successfully discovered each other and the procedure ends.

3) If the device cannot receive an OOB beacon or other frames sent from the other device, then the following sub-procedure shall be applied.

    a) If the device is the initiator, it shall send out its own OOB beacon randomly at the free control channel time periods. If it receives a responding OOB beacon from the responder on or before mMaxDeviceDiscoveryTimeOut, then the two devices successfully discovered each other and the procedure ends; otherwise, it shall go to Step 4).

    b) If the device is the responder, it shall keep scanning the OOB control channel and waiting for the OOB beacons from the initiator. If the device receives an OOB beacon sent by the initiator on or before mMaxDeviceDiscoveryTimeOut, it shall respond with an OOB beacon; otherwise, it shall go to Step 4).

4) If not all OOB control channels scanned, select the next OOB control channel and go to Step 1); otherwise, the two devices fail to discover each other and the procedure ends.

18.1.1.2 Avoidance of interference caused by spatial re-use

After two devices successfully discover each other at both OOB and 60 GHz, the initiator shall choose a free 60GHz channel for the ad-hoc transmissions between the two devices. If no free 60GHz channel is available, the initiator shall choose a channel with enough free bandwidth for the ad-hoc transmission. If there is not enough bandwidth at any 60GHz channel, the initiator and the responder shall start the spatial-reuse transmission option of bandwidth reservation with procedure specified as follows.

Note: In the following procedure, a new transmission means the data transmission on the 60GHz which has not completed the bandwidth reservation process and has not started the formal data transmission yet. An existed transmission means the data transmission on the 60GHz channel is ongoing after finish the bandwidth reservation process.

1) The initiator chooses one 60GHz channel which has the largest free channel bandwidth among all available 60GHz channels as specified in clause 15.

2) The two devices shall perform antenna training as specified in 15.18.

3) The new transmission destination shall analyze OOB beacons received from other devices to see whether there is enough channel time available on the chosen 60GHz channel for the new transmission. If there is enough 60GHz channel time available, the two devices shall go to the DRP reservation procedure specified in clause 15.6; otherwise, go to step 4).

4) The new transmission destination shall analyze 60GHz beacons received from other devices to see whether there is enough 60GHz channel time available for the new transmission. If there is not enough 60GHz channel time available, the device shall report to the upper layer that not enough channel time available as specified in 15.6; otherwise, go to step 5).

5) From the available 60GHz channel time obtained at Step 4, the new transmission destination shall choose a 60GHz DLP which has the shortest time overlap with the existed transmissions detected at Step 3. Then the new transmission initiator and responder shall reserve the chosen 60GHz DLP through the procedure specified in 15.6.

6) The new initiator and the responder shall send out OOB-InterferenceDetectionRequest commands to request transmission destinations of the existing transmissions which have the DLP overlap with the new transmission to scan the 60GHz channel at the non-overlap portion of the new DLP. Within the non-overlap portion of the new DLP, testing data shall be transmitted between the new initiator and the responder with the decided directional transmission/receiving coefficients obtained at the antenna training stage. The selected transmission destinations of the existed transmissions shall report whether they can hear the testing data transmission through OOB-InterferenceDetectionRequest commands.

7) If no interference detected in Step 6, the new initiator and the responder successfully reserve the DLP and can go to transmission stage; otherwise, the new initiator and the responder will go back to step 5) to adjust its DLP and do interference detection again.

18.1.2 Infrastructure mode

Figure 260 illustrates the infrastructure mode where all devices have low-rate OOB control channel and 60GHz high rate channel with the coordinator.

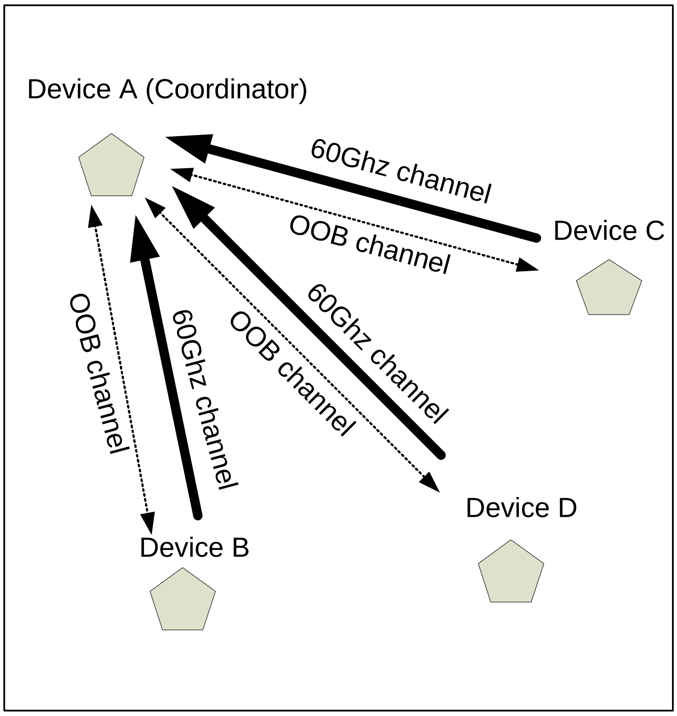


Figure 260 - Topology illustration for infrastructure mode

18.1.2.1 Device discovery with OOB control channel

Devices that support the OOB control channel shall follow the two-step device discovery procedure as in the ad-hoc mode, except that during the step of device discovery in the OOB channel, the coordinator periodically sends out OOB beacons in the OOB control channel. When a new device joins the OOB network, it shall scan each OOB control channel for at least mMaxBeaconIntervalTime. If an OOB beacon is detected, it may send the association command to the coordinator to join the OOB network. If no OOB beacon is detected and the device is coordinator capable, it may become the coordinator and send out its OOB beacons. After a device joins an OOB network or sets up its own OOB network, it may discover other devices by sending OOB-ProbeRequest and receiving OOB-ProbeResponse commands.

18.1.2.2 Avoidance of interference caused by spatial re-use

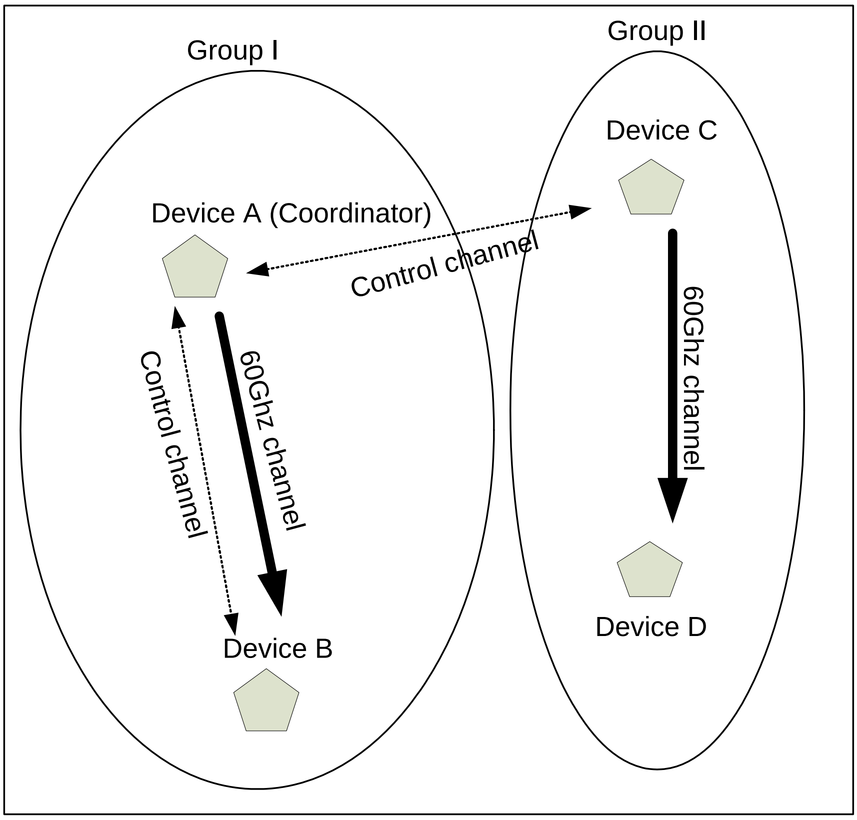


Figure 261 - Spatial re-use group topology with coordinator

The procedure of spatial-reuse transmission option of bandwidth reservation in the infrastructure mode is the same as in the ad-hoc mode, except that the coordinator schedules and controls the interference detection steps.

Spatial reuse for infrastructure mode with OOB is utilized by the coordinator to schedule multiple DLPs at the same 60GHz channel without interfering with each other. An example is given in Figure 261 where the streams between device A and B and between device C and D are on the same 60GHz channel without interfering with each other. So the coordinator device A can allocate overlapped DLPs to these two streams at the same 60 GHz channel.

18.1.3 Other OOB functions

18.1.3.1 Fast recovery with OOB from 60GHz channel blockage

When the 60GHz channel is suddenly blocked, a responder shall send an OOB-ChannelLossNotification message to the initiator in the OOB control channel. The two devices shall switch from the 60GHz data channel to the 60GHz discovery channel and perform antenna training again for 60GHz data channel recovery.

18.1.3.2 ACK transmission on the OOB channel

Devices may transmit OOB-BlockACK messages in the OOB control channel to provide acknowledgement for the data packets transmitted on the 60 GHz data channel.

18.2 OOB frame format

18.2.1 General frame format

The OOB commands shall be formatted as specified in Figure 262.

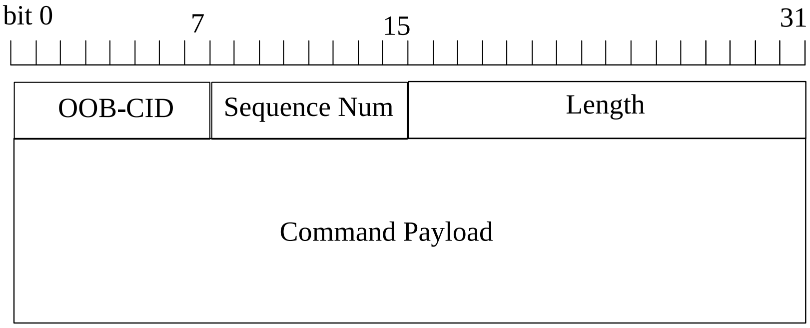


Figure 262 - General format of OOB commands

OOB-CID field contains the OOB command identification number of each OOB command. Sequence Num field is used for re-transmission.

Length field indicates the length of the command payload.

Table 126 - OOB-CID Commands

OOB-CID	Command Name
0x00	OOB-Beacon
0x01	OOB-60GHzCapabilityRequest
0x02	OOB-60GHzCapabilityResponse
0x03	OOB-InterferenceDetectionRequest
0x04	OOB-InterferenceDetectionResponse
0x05	OOB-ChannelLossNotification
0x06	OOB-BlockACK
0x07	OOB-ProbeRequest
0x08	OOB-ProbeResponse
0x09-0xFF	Reserved

18.2.2 OOB-beacon

The OOB-Beacon frame payload is specified in Figure 263.

Octets: 1	6	L1	...	LN
Beacon Control	Device Identifier	Information Element 1	...	Information Element N

Figure 263 - Payload format for OOB-beacon

The Beacon control field is specified in Figure 264.

Bits: 1	1	6
Ad-hoc mode	60GHz capable	Reserved

Figure 264 - Beacon control field format

The Ad-hoc mode bit shall be set to ONE if the OOB operates at ad hoc mode and set to "0" for the infrastructure mode.

The 60 GHz capable bit shall be set to ONE if the devices implements the 60 GHz PHY and MAC as specified herein.

The Device Identifier field is set to the EUI-48 (Annex C.1) of the device sending the beacon.

The information elements (IEs) that may be included in a beacon frame are listed in Table 83.

18.2.3 OOB-60GHz capability request

The OOB-60GHzCapabilityRequest is used to request in-band 60GHz capability information from a device. The payload format is specified in Figure 265.

Octets: 1	...	1
Information Element ID 1	...	Information Element ID N

Figure 265 - Payload format for OOB-60GHzCapabilityRequest

Information Element IDs can be the IDs of PHY capabilities IE, MAC capabilities IE or ATTP IEs specified in 14.7.

18.2.4 OOB-60GHz capability response

The OOB-60GHzCapabilityResponse is used to respond to OOB-60GHzCapabilityRequest command. The payload format is specified in Figure 266.

Octets: 1	...	1
Information Element 1	...	Information Element N

Figure 266 - Payload format for OOB-60GHzCapabilityResponse

Information Element can be the PHY capabilities IE, MAC capabilities IE or ATTP IEs specified in 14.7.

18.2.5 OOB-interference detection request

The OOB-InterferenceDetectionRequest is used to request a device to scan a 60GHz channel with the same antenna configuration for data communication in its reserved DLP. The payload format is specified in Figure 267.

Octets: N (0 to 32)
MAS bitmap

Figure 267 - Payload format for OOB-InterfenceDetectionRequest

The MAS bitmap is used to indicate during which MASs in the superframe that the device shall listen to the channel to detect interference. A bit in the MAS bitmap set to ONE means the device shall scan the channel at the corresponding MAS.

18.2.6 OOB-interference detection response

The OOB-InterfenceDetectionResponse is used to respond to OOB-InterfenceDetectionRequest command. The payload format is specified in Figure 268.

Octets: N (0 to 32)
MAS bitmap

Figure 268 - Payload format for OOB-InterferenceDetectionResponse

The MAS bitmap indicates during which MASs in the superframe the device has already detected 60GHz signals. A bit in the MAS bitmap set to ONE indicates the device has detected 60GHz signals at the corresponding MAS.

18.2.7 OOB-channel loss notification

The OOB-InterfenceDetectionRequest is used to report 60GHz channel loss. The payload format is specified in Figure 269.

Octets: 1
60GHz channel ID

Figure 269 - Payload format for OOB-ChannelLossNotification

60GHz channel ID is used to indicate at which 60GHz channel that the two devices lost their connections.

18.2.8 OOB-block ACK

The OOB-BlockACK acknowledges correct or incorrect receipt of the previous sequence of frames on the 60GHz channel and provides information for the transmission of the next sequence of frames as described in 15.12.3. The OOB-BlockACK frame payload is specified in Figure 270.

The fields in OOB-BlockACK frame payload are described in 14.4.2.

Octets: 2	1	2	0-n
Buffer Size	Frame Count	Sequence Control	Frame Bitmap

Figure 270 - Payload format for OOB-BlockACK

18.2.9 OOB-probe request

The OOB-ProbeRequest is used to search for another device on the OOB control channel. The payload format is specified in Figure 271.

Octets: 6
Targeting Device Identifier

Figure 271 - Payload format for OOB-ProbeRequest

The targeting Device Identifier field is set to the EUI-48 (see Annex C.1) of the device for which to be searched.

18.2.10 OOB-probe response

The OOB-ProbeResponse is used to respond to OOB-ProbeRequest on the OOB control channel. The payload format is specified in Figure 272.

Octets: 6
Targeting Device Identifier

Figure 272 - Payload format for OOB-ProbeResponse

The targeting Device Identifier field is set to the EUI-48 (see Annex C.1) of the device for which to be searched.

18.3 Convergence MAC sublayer parameters

Table 127 contains the values for the Convergence MAC sublayer parameters.

Table 127 - Convergence MAC sublayer parameters

Parameter	Value
mMaxBeaconIntervalTime	64 ms
mMaxDeviceDiscoveryTimeOut	256*mMaxBeaconIntervalTime

19 Relay operation

This clause specifies the relay operation required to provide an anti-blocking mechanism. When the direct path of two devices communicating each other is blocked or in bad channel condition, another path via a relay device is used to continue data transmission. A relay device is considered as a type A device with amplify-and-forward function. To support the functionality, it should include at least two RF modules capable of antenna training.

19.1 Relay path setup

Prior to data transmission via a relay device, the following setup procedures are needed. These are accomplished with device discovery and followed by antenna training in discovery channel.

19.1.1 Identification of relay capabilites

A device initiating relay operation needs to identify a relay device and to recognize whether its target device supports relay operation or not. The initiating device shall send beacons with the Relay Support Capability field in the MAC capability field, as described in 14.7.14, set to ONE. In case that any other device which responds to the initiating device is also relay supportable, it shall respond to this beacon with the Relay Support Capability field set to ONE. A relay device shall send a beacon with the Relay Capability field in the MAC capability field, as described in 14.7.14, set to ONE.

19.1.2 Association with a relay device

If a relay device exists, the device which initiated the association with the relay device shall send the relay device the Relay Reservation Request command frame with the Relay IE which contains the information of the source ID and the destination ID. Upon the reception of the Relay Reservation Request, the relay device shall send Relay Reservation Response command with Relay Reservation field as described in 14.7.21. After receiving the Relay Reservation Response, the source device shall consider this association procedure with the relay as completed. If the Relay Reservation field in the Relay Reservation Response indicates a successful reservation, the source shall regard the relay device as associated with itself.

19.1.3 Antenna training between the source and the destination

In case that a relay device is reserved, a source device shall set the Duration field of RTT control frame for antenna training as much as 3 times of that of normal antenna training sequence. Then, the antenna training between the source and the destination follows the antenna training procedure as described in 15.18. After completing the antenna training, they shall perform the explicit channel scan, as described in 15.4.1.1.

19.1.4 Antenna training between the source and the relay

A source device shall send a Relay Set Request command frame to the relay device. If the relay device sends a Relay Set Response command frame in response to it, the source device shall start antenna training by sending out a RTT control frame with the duration value of twice as long as the normal antenna training sequence period. After the antenna training procedure between the source device and the relay device is done, they shall perform the explicit channel scan where the source shall send the Channel Scanning Request using the same Channel Bitmap field as used between the source and the destination. After completing the explicit channel scan, the source device shall send the Relay Complete Request command frame to the destination device.

19.1.5 Antenna training between the relay and the destination

A destination device receiving the Relay Complete Request command frame shall start antenna training procedure between the destination and the relay device by sending RTT control frame with the duration value of normal antenna training sequence period. After antenna training procedure, the destination and the relay shall perform the explicit channel scan where the destination shall send the relay the Channel Scanning Request using the same Channel Bitmap field as used between the source and the destination. After completing the explicit channel scan, the destination device shall send the source Relay Complete Response command frame including a number of DRP available IEs resulted from the channel scan.

After the antenna training among the three devices, the source may decide which path should be used for data transmission in the data channel. For this end, the source shall decide the channel status of relay path by using the LQI between the source and the relay and the one between the relay and the destination which is obtained by R-D Link LQI field, as described in 14.7.22. If one of two LQIs is lower than predefined threshold, the source may consider the relay path not good enough and shall not send Channel Change Request to the relay device. Otherwise, the source shall decide the channel status based on two LQIs, which is implementation dependent. Then, after comparing the channel status of relay path to that of direct path, the source shall decide which channel status is better. Even if the channel status of direct path is not available due to antenna training failure of that path, the source may use the relay path in data channel instead.

After receiving Relay Complete Response command frame, the source device may search another relay device in discovery channel.

19.1.6 Transition to data channel

A source device receiving the Relay Complete Response command frame shall choose the best channel in order to move, along with the destination and relay device, and then shall send a Channel Change Request command frame to both relay device and destination device. If the source receives, from both devices, Channel Change Response command frames with Reason code set to Accepted, the three devices shall change to the agreed upon channel. Otherwise the source shall send revised Channel Change Request command frames. Figure 272 shows the antenna training and channel selection procedures for relay operation, as described from 19.1.3 to 19.1.6.

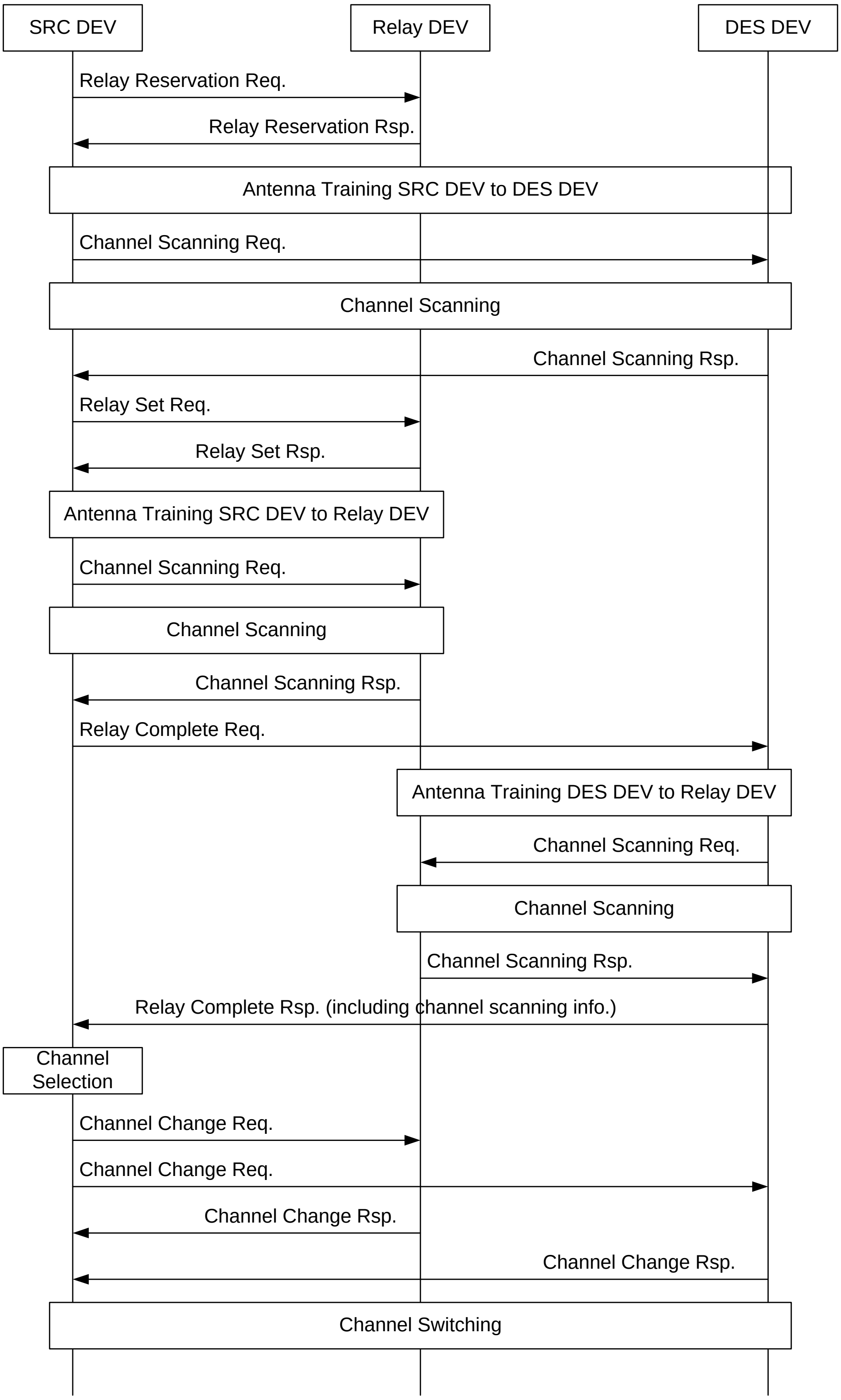


Figure 273 - Relay association procedure

19.2 Data transmission in relay operation

19.2.1 DRP reservation for relay operation in data channel

Both source device and destination device shall select two beacon slots to transmit its beacon to the corresponding device and relay device. The relay device shall also select two beacon slots to send its beacon to the source device and the destination device.

In the beaconing period, the source shall send its beacon including DRP IE to the destination and the relay in order to perform DRP reservation negotiation, as described in 15.6.4. This mechanism is identical to DRP reservation negotiation in terms of the source and the destination. In view of the relay device, an additional step it has to do is to listen to beacon frame from the source device and to check the Target/Owner DevAddr field of the DRP IE in the beacon frame. If the Owner DevAddr and the Target DevAddr fields are corresponding to those devices associated with the relay device, the relay shall perform the DRP reservation negotiation requested by the source. Once both negotiations for a reservation are successfully completed, the source and the destination shall announce the DRP reservation. After that, the relay may use the reserved MASs for relay operation.

19.2.2 Usage of relay device

A device capable of relaying shall be in one of the following two data transmission modes within a superframe:

- Normal mode: the device transmits and receives frames via just one path, such as direct path for example, until the path is no longer used due to blockage or channel degradation.

- Alternation mode: a source device transmits and receives frames with multiple paths using predetermined order of path usage in a time division manner. Only one path for data transmission by the source shall be permitted within the Path Change Interval, where the Path Change Interval indicates the time duration, in unit of MAS, which a source can change its path for transmission as defined in 14.7.22. The source device, which intends to use the Alternation mode, shall announce its use to the destination and the relay device by including Relay IE which contains Tx Mode and Path Order field, as defined in 14.7.22. The Relay IE may be sent in a beacon or a command frame. On receiving a beacon or a command frame including the Relay IE, the destination device shall send a response, whether in a beacon or a command frame, to the source device. And then, the source and the destination device shall send and receive frame in data channel via the chosen path according to the given path order. Also, the relay device shall relay frames when the relay path is chosen.

The Normal mode may be used in the case that the path change is not often required. The Alternation mode is appropriate for sending frames that are delay sensitive. Basically, both modes require the acknowledgement for frames sent. However, the Alternation mode may be also used for sending frames that do not require guaranteed delivery, such as No ACK policy.

19.2.3 Frame transfer in relay operation

A source device shall transmit MPDUs with a Path ID field using the reserved 2 bits in the frame control field of the MAC header, as described in 14.2.1. The Path ID shall be used to indicate which path is selected whenever a frame is transmitted by using relay operation, as defined in Table 128, where the Path ID field is only available in relay operation but ignored in non-relay operation and the number of paths shall have at least two paths for relay operation.

Table 128 - Path ID encoding

Value	Path ID
0	Direct Path

Table 128 - Path ID encoding (concluded)

Value	Path ID
1	Relay Path 1
2	Relay Path 2
3	Reserved

19.2.4 Frame reception in relay operation

On receiving a frame with Path ID and ACK Policy set to Imm-ACK or B-ACK Request from the source device, to acknowledge receipt of the frame, the destination device shall transmit Imm-ACK or B-ACK to the source device with Path ID of a received frame.

19.2.5 Data exchange using relay device

This clause describes the rules for data exchange to support relay operation in any reservations established by using DRP.

In the channel time reserved by DRP, the source device and the destination device can transmit and receive data through the direct path or the relay path via relay device.

At the beginning of DRP reservation, data exchange shall be performed over the direct path.

If the direct path is unavailable during the data exchange, the source may transmit data at the beginning of the next Path Change Interval via one of other paths.

The following describes the detailed rules according to the Transmission Mode.

19.2.5.1 Data exchange rules for Normal mode

After completing the relay association, the source device and the destination device shall exchange their first data through the direct path.

The data exchange within Path Change Interval, as defined in 14.7.22, shall be done only through one path. The length of Path Change Interval field shall be decided by the source. The source shall announce the value, as included in the beacon, to the relay and the destination.

The source device shall change the path and transmit data to other path for data exchange in the next Path Change Interval when it didn't receive ACK frame in current Path Change Interval. In this case, the currently remaining period of current Path Change Interval is no longer used for data exchange.

If the source device intends to change the path, the source shall start to transmit data to the other path after Data Sensing Time plus Path Switching Time at the beginning of the next Path Change Interval. Both values for Data Sensing Time and Path Switch Time are the same as pSIFS.

The destination device shall listen for Data Sensing Time whenever the Path Change Interval starts. If there's no data during Data Sensing Time from the start of the Path Change Interval, the destination shall change the path in order to receive data at once.

The source device may transmit data to the other path for data exchange when it decided to change the path on purpose according to channel status for instance.

An example of switching operation for Normal mode is specified in Figure 274. Figure 274 shows the destination listens for Data Sensing Time at the beginning of every Path Change Interval and does not switch to the relay path until no data is detected, where Path Change Interval is set to 3 MASs.

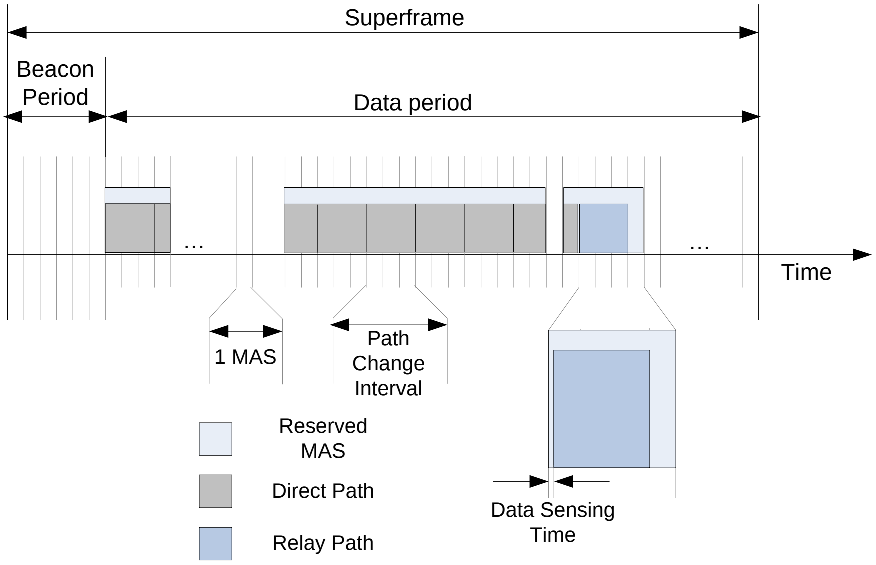


Figure 274 - Data transmission for Normal mode

19.2.5.2 Data exchange rules for Alternation mode

After completing the relay association, the source device and the destination device shall exchange their first data through the direct path.

The data exchange within Path Change Interval shall be done only through one path.

The source device and the destination device shall always change the path and transmit data to other path for data exchange in the next Path Change Interval, regardless of blockage.

19.2.6 Decision on path change

This subclause introduces three ways to decide whether the path in use is unavailable or not.

First, if the source device didn't receive an ACK frame in the Path Channel Interval, the path shall be considered as blocked.

Second, whether the path in use is available or not is determined according to channel status information which can be obtained by using Link Feedback mechanism as described in 14.5.4. If the chosen channel status information is lower than the predefined threshold, the source device may change the path to the other path.

Third, if ACK policy is set to B-ACK, the ratio between total number of subframes and the number of subframes requesting retransmission or frame error rate may be used as decision metrics.

Those methods may be used simultaneously for path change decision.

19.2.7 Path Change

If the path in use is no longer available based on ways described in 19.2.5, the path shall be changed to the other path.

After the path change is decided regardless of blockage or channel degradation, the source device shall send a frame using one of the valid paths to avoid the blocking.

The path selection rule is as follows. When the Transmission Mode field is equal to Normal, the source shall search for a path in an increasing order of Path ID in the available path's list. When Transmission Mode is equal to Alternation, the source first shall search for path according to the order of Path Order field. If all the paths in the Path Order are unavailable, the source shall search for path in an increasing order of available path's list.

If Transmission Mode is Normal mode, the source shall start to transmit data to other selected path after Data Sensing Time plus Path Switching Time at the beginning of the next Path Change Interval in order to avoid the case that the destination does not receive data due to the later path change compared to that of the source. If Transmission Mode is Alternation mode, the source shall start to transmit data to other selected path at the beginning of the next Path Change Interval without any delay.

19.2.7.1 Explicit path selection (command based)

When the source device intends to change the active path although ACK frame is received from the destination, the source device may send Relay Switch Request command frame, including Detour Start Duration field which indicates the start time of data transfer and Path ID of detour path, to the destination device via the active path.

After receiving the command, the destination shall respond to the source with the Relay Switch Response command frame.

Then, the source device shall transmit data through the other path as indicated in the time of Detour Start Duration field.

If the source does not receive Relay Switch Response command frame via the active path, it shall check whether other paths exist based on the list of available paths and repeat the exchange of the Relay Switch commands as described above.

If all the paths are considered as unavailable, the source and the destination shall return to the first blocked path and check then again. If the first blocked path is still unavailable, the source and the destination shall terminate the process of data transfer.

19.2.8 Relay link feedback via relay device

The source device may determine the best rate or best power level in current channel according to link feedback mechanism, as described in 14.5.4. However if relay path is used, other method is necessary because two link qualities are needed to decide the best rate. When the source device intends to acquire the current link qualities of relay path, it shall send the Link Feedback Request command to the relay device, as defined in 14.5.4.1. On receiving the command, the relay device shall send another Link Feedback Request command to the destination device and calculate the link quality information between the source device and the relay device. In response to the Link Feedback Request command, the destination device shall send the Link Feedback Response command to the relay device, including the link quality information between the relay device and the destination device. After receiving it, the relay device shall send the Link Feedback Response command to the source device, including the link quality status of worse link between two links: one is between the source device and the relay device; the other is between the relay device and the destination device. Upon receiving the response command, the source device may switch to the best rate adapted to the link quality status. The algorithm to select the best rate from the two link quality status is implementation dependent. Figure 275 shows the relay link feedback procedure via the relay device.

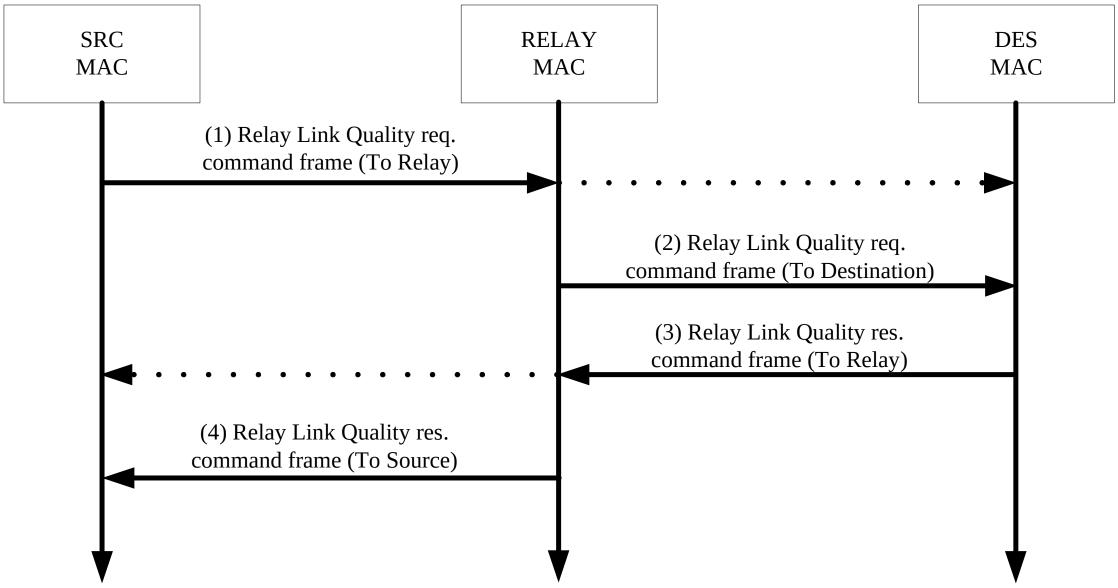


Figure 275 - Relay link feedback procedure

19.2.9 Scan of idle path

If the path in use is considered as an unavailable one, the path's ID shall be removed from a list of available paths and added to a list of unavailable paths in the MIB of the source and the destination device. On adding one path to the unavailable list, the corresponding timer for checking the path availability shall be invoked.

Using this timer, the source device shall check periodically whether the unavailable path is in resumption or not.

For the scan of the idle path, receiving the timer expiry notification, the source shall do path change mechanism, as described in 19.2.7. At that time, if the idle path is the relay path, the source shall acquire the channel status by using relay link feedback mechanism as described in 19.2.8. If the idle path is the direct path, the source shall acquire the channel status by exchanging the Link Feedback Request/Response command frames. If the channel status on the idle path is better than that of the active path, the source may switch into the idle path. Otherwise, the source may continue using the active path.

If the unavailable path is still unavailable after the path change, the source device shall reset the timer and continue to check the availability of the path.

If the previous unavailable path becomes available, the source device shall remove that path from the list of unavailable paths and add to the list of available paths again for future use.

19.2.10 Operation of relay device

A relay device may operate in two modes. One is a decoding mode and the other is an amplify-and-forward mode which means that the relay device amplifies the received data and forwards it to the destination bit by bit.

In discovery channel, a relay device shall be always in the decoding mode in order to listen to the other device's beacons or command frames and send its beacon.

During BP, the relay device shall always operate in the decoding mode as well in order to listen to any frame from two directions by setting both antennas to Rx mode

In any reservations established by using DRP, a relay device operates in an amplify and forward manner:

- If the relay operates as an amplify-and-forward, a source-sided antenna shall be initialized in Rx mode and the other one shall be done in Tx mode.

- During amplify-and-forward operation, a relay device, at the same time, shall decode the relayed frame in order to identify ACK policy from the frame header.

- After relaying a frame or frames transmitted from the source, a relay device shall switch its antenna mode based on the decoded information relating to ACK policy, which enables its antenna switching to be performed before the destination sends ACK frame.

Annex A
(normative)

MUX sublayer

The MUX sublayer is the MAC client as depicted in Figure 31 and routes data between the MAC sublayer and MUX clients.

A.1 MUX service

The MUX sublayer is expressed in terms of the MUX SAP, the MUX service, and the MUX client. Each MUX client is associated with a unique protocol. Service data units presented at the MUX SAP by the MUX client are therefore associated with that protocol.

The protocol is encoded in a MUX header as either:

- A protocol identifier and an OUI; or
- An IEEE Ethertype value [C.2].

The MUX service adds a MUX header to the MUX service data unit to construct a MUX protocol data unit. The MUX sublayer makes use of the service provided by the MAC sublayer for the transfer of its protocol data units.

On receipt of a MUX protocol data unit from the MAC sublayer, the MUX service removes the MUX header and delivers the transported service data unit to the appropriate MUX client based on the identified protocol.

A.2 MUX protocol data unit format

A MUX protocol data unit consists of a MUX Header and a MUX Payload and is defined in Figure A.1.

octets: 2 or 5	**N**
MUX Header	MUX Payload

Figure A.1 - MUX protocol data unit format

The MUX Payload field contains the MUX service data unit that is a payload data unit of the protocol identified in the MUX Header.

The first two octets of the MUX Header are encoded as unsigned binary values, and are delivered to the MAC sublayer in order from the octet containing the most-significant bits to the octet containing the least-significant bits. The octet order for this field is the reverse of that for most fields in this specification.

The MUX Payload is a sequence of octets labelled as MUX Payload[0] through MUX Payload[M-1]. Octets are passed to the MAC sublayer in ascending index-value order.

The MUX Header and MUX Payload together form the payload of the MAC sublayer, which appears to the MAC as a sequence of octets labelled as payload[0] through payload[P-1], as specified in 16.2.

There are three versions of the MUX Header, which are distinguished based on the value of the first two octets of the header.

A.2.1 MUX Header - OUI version

The first version has a length of five octets and is defined in Figure A.2.

octets: 2	**3**
Protocol ID (0x0000 - 0x00FF)	OUI

Figure A.2 - Format of first version of MUX Header

The Protocol ID field is restricted to values from 0 through 255 and is set to a value that identifies a protocol defined by the owner of the OUI specified in the OUI field. The OUI is a sequence of 3 octets, labelled as oui[0] through oui[2]. Octets of the OUI are passed to the MAC sublayer in ascending index-value order.

A.2.2 MUX Header - reserved version

The second version of the MUX Header has a length of two octets and is defined in Figure A.3.

octets: 2
Protocol ID (0x0100 - 0x05FF)

Figure A.3 - Format of second version of MUX Header

The protocol ID field has values between 256 and 1 535 as defined in Table A.1.

Table A.1 - Protocol ID in the MUX Header

Value	Description
0 - 255	Defined by the OUI owner
256	Protocol ID value for WiMedia Logical Link Control Protocol
257	Protocol ID value for Bluetooth
258 - 1 535	Reserved for future use

A.2.3 MUX Header - Ethernet type version

The third version of the MUX Header has a length of two octets and is defined in Figure A.4.

octets: 2
Ethernet Type (0x0600 - 0xFFFF)

Figure A.4 - Format of third version of MUX Header

The Ethernet Type field is restricted to values from 1 536 through 65 535 and is set to the value of an Ethernet type [C.2] identifying a protocol.

Annex B
(normative)

MAC policies

B.1 Reservation limits

A reservation consists of a row component and a column component.

Row component: A portion of a reservation that includes an equal number of MASs at the same offset(s) within every zone, optionally excluding zone zero, as indicated in the DRP IEs.

Column component: The portion of the reservation that is not a row component.

Rules stated in this Clause apply independently to a device whether it is a reservation owner or a reservation target. They do not apply to DRP IEs with Reservation Type set to Alien BP.

A device may consider contiguous reservation blocks from multiple column components in the same zone as if they were a single reservation block in a single column component.

A device shall not allocate more channel time than necessary for its optimal operation.

A device shall set the Unsafe bit of the DRP Control field of a DRP IE according to the following rule:

A) A device shall not identify more than Y consecutive MAS in the same zone within a column component in DRP IEs with the Unsafe bit set to ZERO, where Y is a function of the MAS number within the zone (counting from zero) of the earliest reserved MAS within the set of consecutive MASs, as shown in Table B.1.

Table B.1 - Reservation block size limits

First MAS number	Y
0	8
1	7
2	6
3	5
4	4
5	4

Table B.1 - Reservation block size limits (concluded)

First MAS number	Y
6	4
7	4
8	4
9	4
10	4
11	4
12	4
13	3
14	2
15	1

B) A device shall not set the Unsafe bit to ONE in DRP IEs except to comply with A).

A device shall not include MASs in zone zero in the column component of a reservation.

A device may at any time send a Relinquish Request IE in its beacon where the Target DevAddr identifies a device transmitting its beacon with one or more DRP IEs with the Unsafe bit of the DRP IE Control field set to one (unsafe DRP IEs). The device shall not set the Target DevAddr field to identify a device if that device does not include any unsafe DRP IEs in its beacon, unless forwarding a received Relinquish Request IE to its reservation owner, as specified in 15.1.10.21.

The Allocation fields of the Relinquish Request IE should identify MASs in one or more unsafe DRP IEs.

The Reason Code of the Relinquish Request Control field should be set to a valid Reason Code indicating the reason for requesting the identified MASs.

If a device receives a beacon that contains a Relinquish Request IE with Target DevAddr set to its own DevAddr that identifies MASs it includes in an unsafe DRP IE, it shall:

- Modify its DRP IEs to remove the identified MASs; or
- Modify its DRP IEs such that the Unsafe bit in any DRP IE that includes one or more identified MAS is set to zero per the previous rules in this Clause.

The device shall make this adjustment within mUnsafeReleaseLimit superframes after first receiving the Relinquish Request IE.

If a device requests a neighbour to release MASs in an unsafe DRP IE, the device shall not include a new unsafe DRP IE in its beacon or change a DRP IE to set the Unsafe bit to ONE until mOwnerUnsafeHoldoff superframes have passed.

If a device includes an unsafe DRP IE in its beacon and it receives a Relinquish Request IE that identifies MASs included in the DRP IE, it shall not include a new unsafe DRP IE in its beacon or change a DRP IE to set the Unsafe bit to ONE until the neighbour requesting release establishes a new DRP IE or mTargetUnsafeHoldoff superframes have passed.

B.2 Reservation locations

As referenced in herein, a reservation owner shall select MASs based on a minimum latency requirement of not less than 4 milliseconds, or on a medium utilization efficiency or power consumption requirement for a minimum reservation block length.

In the row component of a reservation, the reservation owner shall select reservation blocks such that the lowest MAS number selected within a zone is maximized, except that the reservation owner is not required to use more than one reservation block per zone.

In the column component of a reservation, the reservation owner shall select reservation blocks that meet its requirements such that each block is located within the first eight MASs of its zone, if possible. If not possible, the reservation owner shall select reservation blocks that meet its requirements and that minimize the highest MAS number selected in any zone.

If multiple potential zone locations meet the previous requirements, the reservation owner shall select reservation blocks in zones such that the latest used set in the following ordered list of sets is as early as possible:

[{8}, {4 or 12}, {2, 6, 10, or 14}, {1, 3, 5, 7, 9, 11, 13, or 15}].

If there are multiple possible zone locations that use the same latest set, the reservation owner should minimize the highest MAS number selected within the zones. The reservation owner shall place each reservation block at the earliest available location within its zone.

If the MASs available for reservation by the reservation owner change, it shall determine if its reservations would still meet the reservation location rules listed in B.2. If not, the reservation owner shall change its reservations within mCompactionLimit superframes such that they meet the reservation location rules.

A reservation owner may disregard the reservation location rules for MASs identified in DRP IEs with the Unsafe bit set to ONE according to the Y limit stated in B.1.

B.3 MAC policies parameters

Table B.2 contains the values for the MAC policies parameters.

Table B.2 - MAC policies parameters

Parameter	Value
mCompactionLimit	32
mOwnerUnsafeHoldoff	32
mTargetUnsafeHoldoff	2×mMaxLostBeacons
mUnsafeReleaseLimit	4

Annex C
(informative)

Bibliography

C.1 "Guidelines for use of a 48-bit Extended Unique Identifier (EUI-48™)",

http://standards.ieee.org/regauth/oui/tutorials/EUI48.html

C.2 IEEE STD 802.3, 1998 Local and Metropolitan Area Networks EtherType Field Tutorial, Rev 0.7, New York, Institute of Electrical and Electronics Engineers, Inc.

C.3 ISO/IEC 7498-1:1994, Information Technology-Open Systems Interconnection-Basic Reference Model: The Basic Model.

C.4 Assigned numbers for Specifier ID in Application-specific control frames, command frames, IEs and Probe IEs.

http://www.ecma-international.org/publications/standards/UWB_specifier_id.htm

C.5 IEEE Std 802®-2001, IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture.

Annex D
(informative)

Higher Layer Synchronization Support

This mechanism supports the process of synchronization among higher-layer protocol entities residing within different devices. The actual synchronization mechanism in the higher layer is out of the scope of this Standard. In principle, the MLME indicates the transmission/reception of frames with a specific multicast address in the DestAddr field of an MSDU of type data. The primitives covered in this Clause are listed in Table D.1.

Table D.1 - Higher layer synchronization support primitives

Service Primitive	Request	Indication	Confirm
MLME-HL-SYNC	1	2	3

Table D.2 lists the parameters that appear in the primitives of this Clause

Table D.2 - Parameters that appear in the primitives of this Clause

Name	Type	Valid range	Description
GroupEUI	EUI-48	Any valid multicast EUI-48	Specifies the multicast group to which the synchronization frames are addressed. A synchronization frame is a data type frame with higher layer synchronization information
ResultCode	Enumeration	SUCCESS, NOT_SUPPORTED	Indicates the result of the MLME-HL-SYNC.request
SrcEUI	EUI-48	Any valid multicast EUI-48	Specifies the EUI-48 of the device that transmitted the higher layer synchronization frame
SequenceNumber	Integer	As defined in the frame format	Specifies the Sequence Number of the higher layer synchronization frame received or transmitted

D.1 MLME-HL-SYNC.request

This primitive requests activation of the synchronization support mechanism.

The definition of this primitive is:

```
MLME-HL-SYNC.request(
    GroupEUI
    )
```

D.1.1 When generated

This primitive is generated by the DME when a higher layer protocol initiates a synchronization process.

D.1.2 Effect of receipt

This request activates the synchronization support mechanism at the device. The MLME subsequently issues an MLME-HL-SYNC.confirm that reflects the results of the higher layer synchronization support request. If the request has been successful, and the higher layer synchronization support mechanism has been activated, the MLME issues an MMLE-HL-SYNC.indication whenever a higher layer synchronization frame, which is a data type frame with the specified McstAddr in the DestAddr field, is received or transmitted.

D.2 MLME-HL-SYNC.indication

This primitive indicates the transmission or reception of a higher layer synchronization frame. The indication is delivered with respect to the start of the frame on the medium, whether transmitted or received by the MAC sublayer.

The definition of this primitive is:

```
MLME-HL-SYNC.indication(
    GroupEUI,
    SrcEUI,
    SequenceNumber
    )
```

D.2.1 When generated

This primitive is generated by the MLME when the successful reception or transmission of a higher layer synchronization frame is detected, as indicated by the PHY. The higher layer synchronization frame is identified by the McstAddr registered by an earlier MLME-HL-SYNC.request primitive, in the DestAddr field of a data type frame.

D.2.2 Effect of receipt

The DME is notified of the reception or transmission of a higher layer synchronization frame.

D.3 MLME-HL-SYNC.confirm

This primitive confirms the activation of the higher layer synchronization support mechanism.

The definition of this primitive is:

```
MLME-HL-SYNC.confirm(
    ResultCode
    )
```

D.3.1 When generated

This primitive is generated by the MLME as a result of an MLME-HL-SYNC.request to activate the higher layer synchronization support mechanism for a particular multicast address.

D.3.2 Effect of receipt

The DME is notified of the activation of the higher layer synchronization support mechanism. The result code of NOT_SUPPORTED is issued if the MLME does not support the higher layer synchronization support mechanism or if the address provided by the MLME-HL-SYNC.request is not a multicast address.