IEEE P802.15
Wireless Specialty Networks

IEEE P802.15.13

Further text for IEEE P802.15.13

Date: 2019-05-13

Author:

Name	Affiliation	Address	Phone	Email
Tuncer Baykas	Vestel			tbaykas@ieee.org

Abstract

This document contains proposed text for IEEE P802.15.13.

Hints & legend

Text marked in green refers to other documents
Text marked in cyan was taken from other proposals and at most slightly modified
Text marked in yellow indicates missing content or requires further technical discussion
[Text marked in red and in brackets is an editor's note]
Text marked in grey is added in this version

Open issues

- OWPAN ID / Coordinator address
- Security
- Format of the Variable Element Container element
- Include priority & stream reservation
- Queue state feedback

1 Overview

1.1 Scope

1.2 Purpose

2 Normative References

3 Definitions, acronyms, and abbreviations

3.1 Definitions

This clause lists terms that are used throughout the standard. The respective term is printed in **bold letters.** The definition is given after a colon. If a term has synonyms, i.e. other terms that are describing the same entity in this standard, these synonyms are listed in [brackets]. The definition may be substituted by a synonym specification, in which case one can refer to the definition of the synonym.

Contention Access Period (CAP):

CAP Slot: the compound of multiple superframe slots in the CAP.

Guaranteed Time Slot (GTS):

MAC Frame: Frame that is handled on the MAC sublayer.
[MAC Protocol Data Unit]

MAC Protocol Data Unit (MPDU):
[MAC frame]

Modulation and Coding Scheme (MCS): Rate adaptation parameters on the physical layer. This includes, for example, the type of modulation or details of the error-coding scheme.

Superframe Slot: the elements of each superframe in the beacon-enabled channel access mode. Superframe slots define a basic duration in the superframe. Other durations, i.e. of the beacon slots or CAP slot, are multiples of this superframe slot duration.

3.2 Acronyms and abbreviations

FCS Frame Check Sequence
DME Device Management Entity
TAIFS Turn Around Inter-Frame Space
LLC Link Layer Control

4 General description

4.1 Introduction

4.2 Components of an IEEE 802.15.13 OWPAN

An OWPAN constitutes of standard-compliant devices. Devices carry a MAC-48 address for identification and flat addressing in the network. Devices constitute of a standards-compliant MAC implementation and make use of a compliant PHY defined in the standard. Not all devices are required to implement the functionality to maintain an OWPAN. Devices, which support that functionality, are also referred to as coordinator-capable devices or coordinators if they actively maintain an OWPAN.

In each OWPAN, a single coordinator-capable device assumes the role of the coordinator. The coordinator is responsible for starting, maintaining and finally stopping the OWPAN. The coordinator is furthermore involved in all data transmission in the OWPAN. Hence, the single logical network topology of an OWPAN is the star topology, as detailed in 4.3.

Non-coordinator devices, subsequently also simply referred to as devices, implement less functionality than coordinators. Devices associate with an OWPAN in order to gain to gain layer 2 connectivity with the network. Coordinators and devices may have multiple optical frontends for transmission and reception to support MIMO communication.

4.3 Network service

An OWPAN denotes the network between devices. The MCPS-SAP provides as a service the transmission of MSDUs between devices, based on MAC-48 addresses. Moreover, coordinators may act as access points [bridges], connecting devices associated with the maintained OWPAN with peers in an external network.

4.3.1 Topology

All IEEE 802.15.13 OWPANs have a star topology. Thus, a single coordinator is involved in all data transmission between two devices or between external peers and the devices associated with the OWPAN as illustrated in Figure 4-1. Moreover, data transmissions between two devices of the same OWPAN must also be relayed by the coordinator.

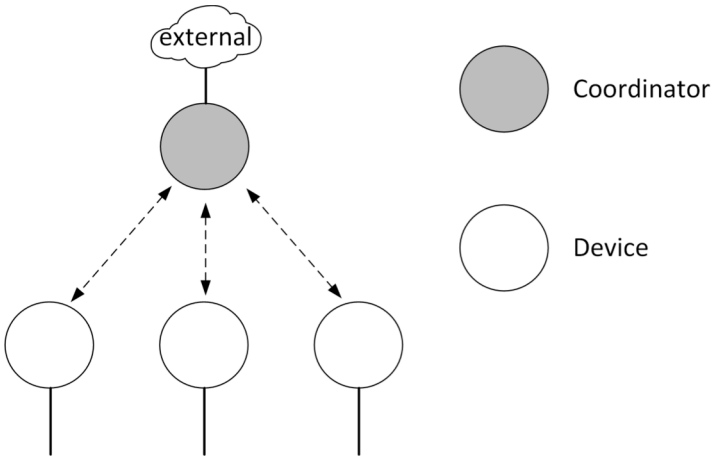

Figure 4-1: Basic star topology

Depending on the application, the star topology may have different characteristics. The subsequent clauses list different special cases of star topologies to be realized.

4.3.1.1 Distributed MIMO star topology

To improve transmission properties and to increase mobility support, the star topology may be realized to support MIMO principles. In that case, the coordinator may have multiple optical frontends (OFEs) for transmission and reception associated with its PHY. With each OFE, the coordinator is able to transmit the same or different signals. However, individual OFEs are not addressable by the device, making them transparent to it apart from the different pilot signals a device observes.

The realization of the distributed MIMO star topology is out of scope of this standard. For example, the OFEs may be distributed in space and connected to the single central coordinator instance via some fronthaul technology, e.g. according to IEEE 802.1CM-2018. To regard for such possibilities, the standard defines means that are helpful for realization. These are, for example, the possibility to transmit orthogonal pilot symbols from each OFE at the physical layer. Detailed implementation rules are, however, left out. The MAC furthermore supports these possible realizations through a channel access mechanism that is able to cope with large fronthaul delays, virtually increasing the propagation delay to the hundreds of microseconds.

The distributed MIMO star topology is illustrated in Figure 4-2. The OFEs of a coordinator may be placed in various ways. One possible placement is the distribution of OFEs over the targeted coverage area of the OWPAN.

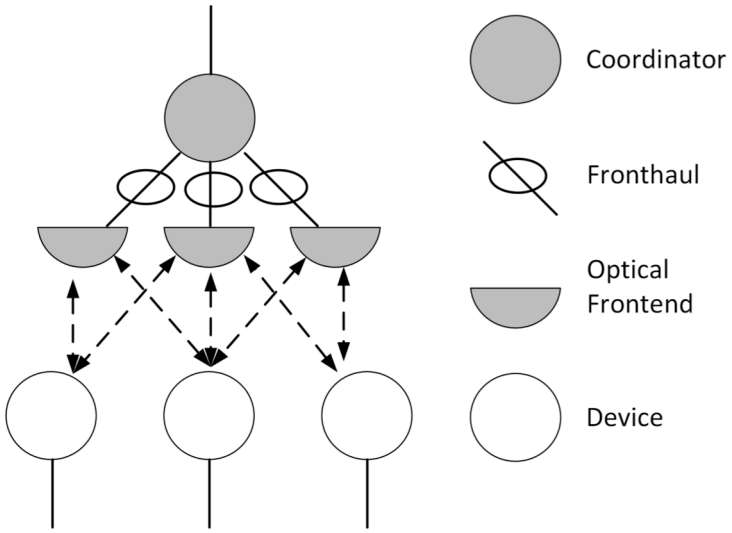

Figure 4-2: Distributed MIMO star topology

4.3.1.2 Unidirectional star topology (broadcast)

In the unidirectional star topology, the OWPAN comprises a coordinator only. The coordinator of the unidirectional star topology does not accept associations by devices. It may transmit frames having the broadcast address as destination address.

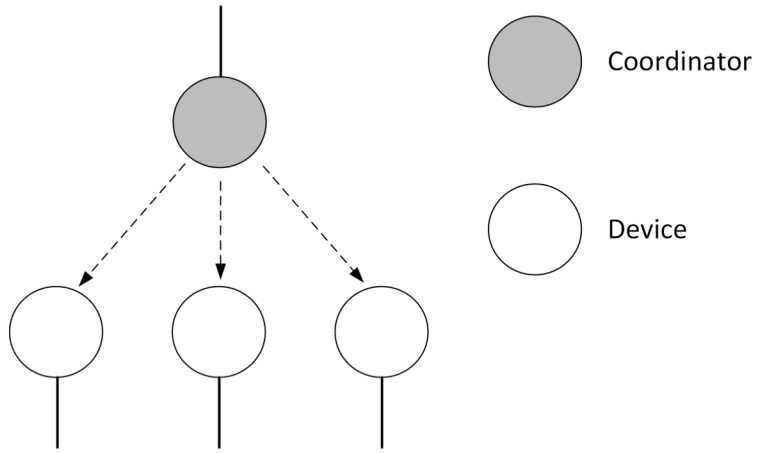

Figure 4-3: Unidirectional star topology

4.3.1.3 Coordinated star topology

Multiple coordinators deployed in the same area may be coordinated by a master coordinator. The corresponding topology is called coordinated star topology.

The functionality of the master coordinator is out of scope of the standard. It is currently anticipated, that coordinated star topology deployments will include devices from a single vendor, hence not requiring standardization of the interfaces between coordinators and the master coordinator.

The network interconnecting the master coordinator with the individual coordinators may be used for steering information but also for the hauling of data frames. It is also referred to as backhaul. The master coordinator shall provide a SAP to the higher layers in order to abstract the coordinated OWC network as a bridge according to [ieee 802 Bridge definition]. The coordinated topology is depicted in Figure 4-4.

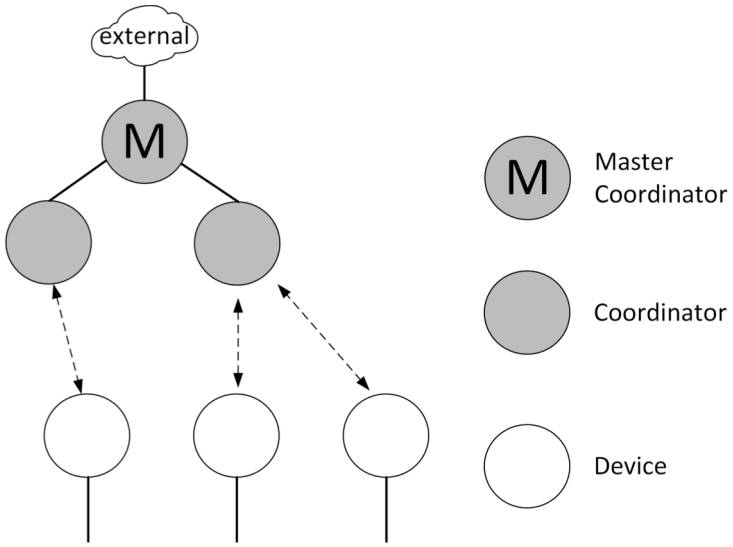

Figure 4-4: Coordinated star topology

As light is highly local, a single area is typically not equipped with multiple uncoordinated infrastructures from different providers. Thus, it is assumed that neighboring IEEE 802.15.13 OWPANs will be deployed in a

coordinated manner. If multiple OWPAN infrastructures overlap in their coverage area, they should always be coordinated by a master coordinator, managing resource allocations between the corresponding coordinators.

4.3.1.4 Radio frequency hybrid topology

The hybrid topology involves an optional RF-based connection at each device. The realization of the hybrid topology is out of scope of the standard. It is expected that the management of the alternate OWC- and RF-based connections is performed above the 802.15.13 MAC, for example according to 802.1AX.

4.3.1.5 Peer-to-peer topology

In the peer-to-peer topology, two devices seek to perform point-to-point communication with each other. In that case, one of the devices assumes the coordinator role, providing an ad-hoc OWPAN to the other device. Hence, the peer-to-peer topology is a special case of the star topology, involving a coordinator and a single non-coordinator device associated with the provided OWPAN.

4.3.1.6 Relay functionality

With the relay functionality, an intermediate relay is used to assist a transmission via a direct optical wireless link. With the relay functionality, each relay supports different duplexing and relay modes. For full duplex (FD), the relay receives and transmits data simultaneously, while in half duplex (HD), the relay receives the data in one time slot and retransmits it in another transmission slot. The relay supports two modes; amplify and-forward (AF), and decode-and-forward (DF).
a) In AF mode, the RD receives the data from the coordinator, which are then retransmitted after amplification.
b) In DF mode, the received data is decoded by the relay and then retransmitted to the destination device.
In case the device is disconnected from the coordinator, a relay search request is conducted, including the relay capabilities. The coordinator broadcasts a relay search request frame. Each relay replies back on the control channel with its own capabilities including duplexing and relaying modes. The coordinator selects the relay that provides the best connectivity. The coordinator sets up a relay link between itself and the device through the selected relay. A connection remains active until the direct link between the coordinator and the device is reinitiated and the coordinator requires a termination of the link between the coordinator and the relay.

4.3.2 Network Integration

An OWPAN provides three logically distinct transmission services:

1) Transmission from a coordinator to a device or from a device to a coordinator
2) Transmission from device to another device in the OWPAN or to a peer in an external network
3) Transmission from another device in the OWPAN or a peer in an external network to a device in the OWPAN

A fourth case, bridging, is currently not supported by the standard. In bridging, a peer in an external network behind the coordinator would be able to communicate with a peer in an external network behind an associated device.

In case 1), the transmitted frame can be a control or management frame, transmitted from the coordinator to the device or vice versa. Also, the frame may be a data frame from an higher layer application or protocol running on the device or coordinator with the destination MAC-48 address set to either the coordinator's or the device's address.

In case 2), a higher layer application or protocol running on the device transmits a frame with the destination set to a MAC-48 address other than the coordinator's unicast address. The destination address may belong to either another device of the OWPAN or a peer in another network, which the coordinator is connected with.

In case 3), …

The coordinator is expected to maintain a database of its associated devices and their corresponding MAC-48 addresses. If it receives an MSDU from a device,

Three address format is used to realize LAN integration… Create Text!

4.4 Coexistence

The high directivity of light imposes difficulties on coexistence schemes that are based on energy detection. This is in contrast to RF-based communication technologies with omnidirectional propagation characteristics. Through these omnidirectional characteristics, heterogeneous RF-technologies that feature not mutually decodable signals, can rely on refraining from transmissions after the channel is detected busy by exceeding a given signal energy threshold (CCA through energy detection).

Through the directivity, however, a device A is not able to infer that a second device B's transmission is currently received at device A's prospective receiver.

This standard restricts uncoordinated transmissions, i.e. random channel access, to the minimum required purposes such as association or reconnection. However, in case of alien technologies entering the coverage area of an IEEE 802.15.13 OWPAN, the behavior is unspecified. Currently, there is no coexistence coordination among different OWC standards.

4.5 Architecture

Similar to other IEEE 802 standards, the architecture of this standard is defined by a number of layers in order to group related functionality and simplify the standard. Each layer is responsible for a subset of the functionality included in the standard and offers services to the next higher layer.

Each layer includes interfaces that serve the exchange with other layers. More specifically, a lower layer provides its service to the next higher layer. The term used for the corresponding interfaces in this standard is service access point (SAP).

This standard specifies only exposed interfaces, which are likely to connect entities provided by different vendors. Currently, this are the MCPS-SAP and the MLME-SAP, as depicted in Figure 4-5. Other interfaces are assumed to be vendor-internal or do not require detailed specification.

Different functionalities of a layer are accessible through so-called primitives that make up a given SAP. The concept of primitives is further described in clause 4.5.2.

4.5.1 Layers

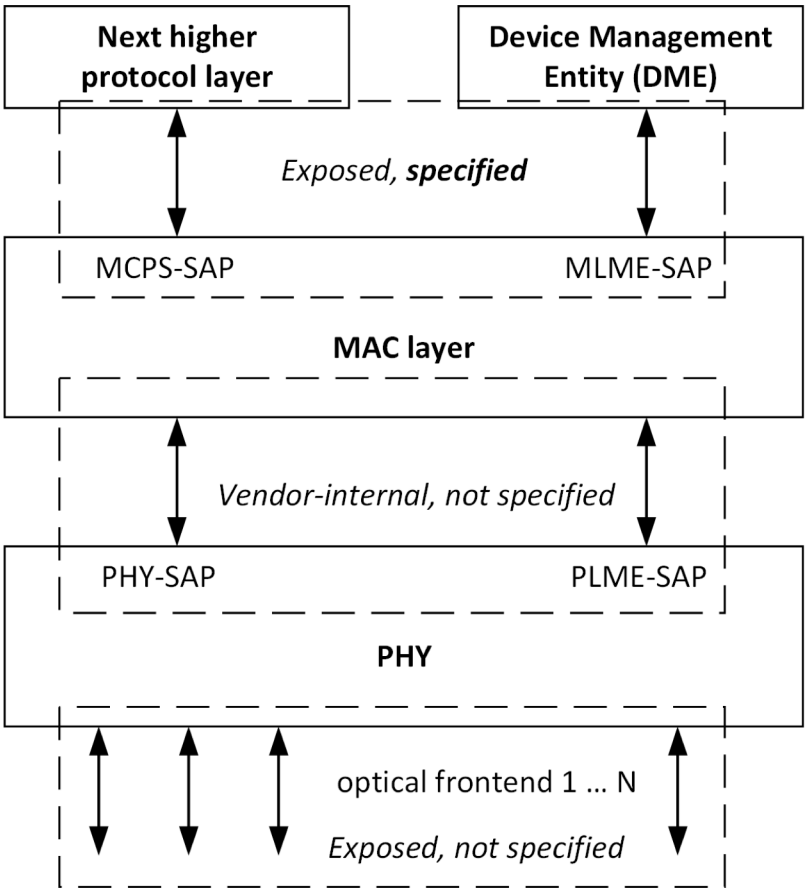

Figure 4-5: Architecture and interfaces of the IEEE 802.15.13 standard

Data to be transmitted via an OWPAN passes multiple layers. The collection of control- management and / or data bits to be passed between layers is also referred to as protocol data units (PDUs). Depending on the layers involved in the exchange of PDUs and the direction of exchange, PDUs have specific names. A data PDU passed to the MAC sublayer by a higher layer protocol is called MSDU. It enters the MAC sublayer through the MCPS-SAP for transmission over the OWPAN and leaves the MAC sublayer through the MCPS-SAP after successful transmission over the OWPAN.

After processing through the MAC sublayer (during transmission) or before processing through the MAC sublayer, the data unit exchanged with the PHY is called either MPDU (from the MAC perspective) or (PHY service data unit) PSDU (from the PHY perspective). The PSDU enters and leaves the PHY through the PHY-SAP.

In the transmit direction, the PHY processes PSDUs, yielding PPDUs, which represent the physical signal to be transmitted over the optical medium. After transmission over one or multiple OFEs, the PPDU is received by a PHY layer and processed into a PSDU, which subsequently traverses the layers up until the MCPS-SAP.

4.5.2 Concept of primitives

The services of a layer are the capabilities it offers to the user in the next higher layer or sublayer by building its functions on the services of the next lower layer. This concept is illustrated in Figure 4-6, showing the relationship of the service user and the service provider (next lower layer).

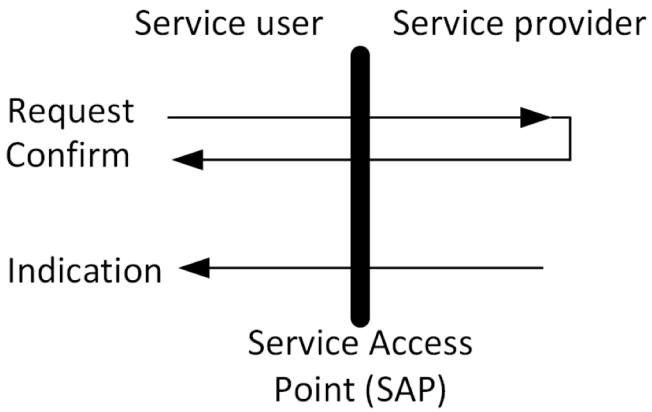

Figure 4-6: Request, confirm and indication primitives of a SAP

The services are specified by describing the information flow between the user and the layer. This information flow is modeled by discrete, instantaneous events, which characterize the provision of a service. Each event consists of passing a service primitive from one layer to the other through a layer SAP associated with a user. Service primitives convey the required information by providing a particular service. These service primitives are an abstraction because they specify only the provided service rather than the means by which it is provided. This definition is independent of any other interface implementation.

4.6 Functional Overview

This clause provides an overview on the functionality supported by the standard.

4.6.1 MAC sublayer

The MAC layer of this standard allows two modes of channel access operation….

4.6.2 PHY layers

This standard supports three distinct PHY layers.

4.6.2.1 PM-PHY introduction

4.6.2.2 LB-PHY introduction

4.6.2.3 HB-PHY introduction

4.6.3 Improving probability of successful delivery

Optical wireless transmission is prone to obstructions of the line of sight. To alleviate such blocking events and other failures of frame transmission, the IEEE 802.15.13 OWPAN employs mechanisms to improve the probability of successful data transmission. These mechanisms include deterministic channel access, automatic retransmission, and support of physical layer spatial diversity.

4.6.4 Addressing

In an OWPAN, flat addressing of devices is facilitated. Each device in an OWPAN has a unique MAC-48 address, consisting of 48 bits. The choice of an MAC-48 address can be used to integrate the OWPAN with other LANs, which are relying on the MAC-48 address format [Reference to Std. 802.1D & Std. 802.1Q].

To increase signaling efficiency, a device gets a shorter address assigned during the association process. The short address consists of 16 bits and may be used to identify the device in various control and management procedures or to facilitate addressing in the MAC frame.

Certain addresses are reserved for special purposes. For MAC-48 addresses, the reserved addresses shall be the same as in the [IEEE 802 MAC address details standard].

The short address 0x0000 shall not be used. The short address 0xFFFF shall be used as the broadcast address. A frame addressed to the broadcast address shall be received by all devices.

4.6.5 Duplex mode

All medium access is controlled by the coordinator of an OWPAN. The coordinator may allow for implicit full duplex transmission and reception of a device, if the device supports the *capFullDuplex* capability.

4.7 Conventions in this standard

This clause lists different conventions for the format, terminology and units within this standard.

4.7.1 Format conventions

Constants and attributes that are specified and maintained by the MAC sublayer are written in italics and without spaces. Constants have a general prefix of “a”, e.g., *aMinFragmentSize*. Variable attributes have a general prefix of “mac”, e.g., *macOwpanId*.

Names of frames, elements or fields are also written in italics. They are capitalized and might contain whitespaces, e.g., *Association Request* element.

4.7.2 Normative terminology

Requirements on conformant implementations of this standard are expressed using the following terminology:

a. ***Shall*** is used for mandatory requirements
b. ***May*** is used to describe optional functionality that the implementation is permitted to support
c. ***Should*** is used for recommended implementation and configuration choices

4.7.3 Power levels

Optical wireless communication utilizes intensity modulated light and direct detection at the receiver. Hence, electrical signal levels, as measured at the receiving DSP, do not relate to the received optical power level in the same way for all devices. Rather, the relationship between received optical power and received electrical power depends on implementation details, such as LED and photodetector characteristics, of a given device.

To compare signal levels, one must thus refer to the optical power and not the electrical power. When signal levels are specified in this standard, these are optical powers. This is the case for emitted signal levels and received signal levels as well.

5 MAC functional description

This clause specifies functions and procedures of the MAC sublayer. Procedures may be initiated by the MAC or the consequence of a MLME-SAP primitive invocation. The MAC makes use of the physical layer service and is responsible for the following tasks:

- Performing channel access and transmission in correspondence with the OWPAN's configuration
- Starting and maintaining an OWPAN
- Associating / disassociating with / from an OWPAN
- Fragmenting and aggregating MSDUs
- Providing a reliable link between two peer MAC entities
- Adapting to alternating channel conditions

The MAC frame formats supporting the function of the MAC are specified in **clause 6**. Services, MAC PIB attributes and device capabilities are specified in **clause 7**. The support for security is specified in **clause 8**.

5.1 MAC transmissions overview

This clause provides an overview over transmissions from the MAC layer perspective. It covers procedures for frame transmission, as initiated through the MCPS-DATA.request primitive until the start of PSDU processing through the PHY. Similarly, procedures for frame reception, starting after the successful reception of a PSDU through the PHY to the triggering of a MCPS-DATA.indication primitive are described.

An OWPAN can operate in beacon-enabled or non-beaconed enabled mode. Depending on which mode is used by the coordinator, channel access is performed in different ways. However, the remaining transmit and receive process are the same, regardless of the applied channel access mechanism.

5.1.1 Addressing

Within the OWC MAC, each device shall be addressable through a 6-octet MAC address compatible with IEEE 802 LANs as specified in [add reference to MAC48-addresses].

In addition, a 2 octet short address is issued to a device as part of the association process. The allocation of short addresses to associated devices is at the discretion of the coordinator implementation. The address 0x0000 shall not be used. Furthermore, the address 0xFFFF shall be regarded as the broadcast address and hence received by all devices. It shall not be allocated to an associating device.

5.1.2 The transmit process

The transmit process starts when the MAC receives an MSDU through the MCPS-SAP or when the MLME requests transmission of a management frame. A device shall keep MSDUs received from the higher layer in logical queues. A distinct queue should be used for every possible value of *Priority* as specified in 7.1.1.

A device prepares MPDUs for transmission in accordance with the maximum duration, as required by the applied channel access mode. Furthermore, a device shall ensure that the MPDU size does not exceed the maximum supported PSDU size of the used PHY.

The channel access mechanism used in the associated OWPAN (see clauses 5.2. and 5.3) regulates when to hand the resulting MPDU to the PHY for transmission over the medium.

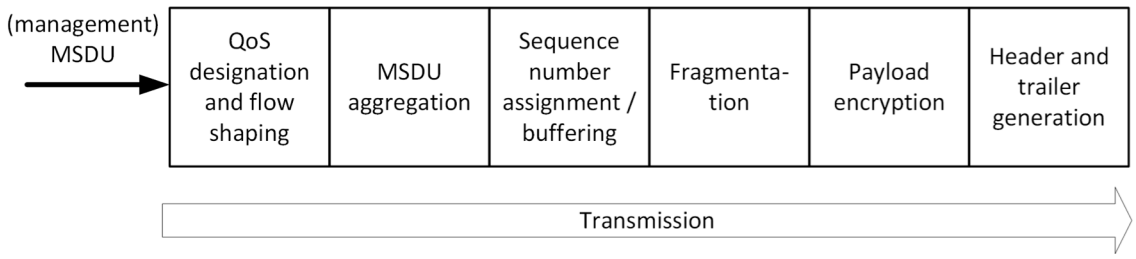

Figure 5-1: The MAC transmit process

5.1.3 The receive process

After a PSDU was successfully received by the PHY, it enters the MAC through the conceived PHY-SAP. The MAC shall first check the integrity of the frame based on the FCS field included in the frame. If the frame contains uncorrectable errors, the MAC shall discard the frame. If the frame was received successfully, the MAC may attempt parsing the frame and processing it further.

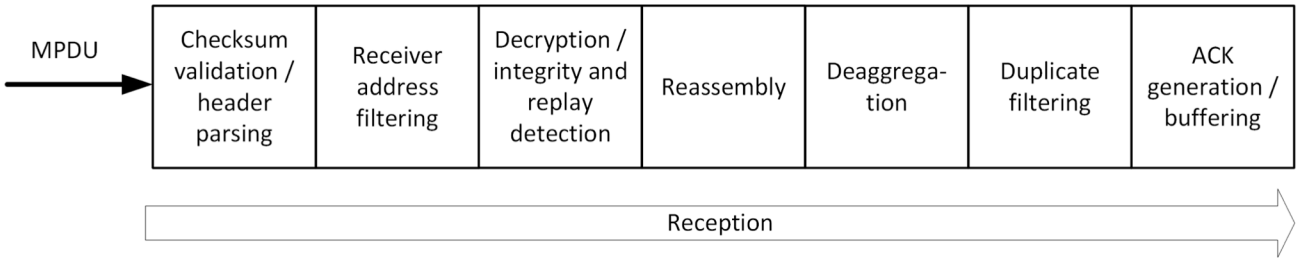

Figure 5-2: The MAC receive process

The MAC shall discard frames that have an unsupported *Frame Version* in their *Frame Control* element (clause 6.2.1.1).

The MAC shall filter frames based on the included receiver address. It shall discard all frames that are neither unicasts to the address associated with the device nor broadcasts address. The MAC shall also discard data and management frames that do not belong to the OWPAN it is associated with.

For remaining received frames that indicate the usage of security in their header, the MAC shall perform decryption, authenticity checking and replay detection and prevention as detailed in the respective security clause. For that purpose, the MAC makes use of the security information included in the *Auxiliary Security Header* of the frame as specified in the respective clause for the security type.

If the frame indicates to contain a fragment, the MAC shall buffer the frame and perform reassembly according to 5.5. Subsequently, it shall perform disaggregation according to 5.6.2 if the frame contains aggregated MSDUs.

The MAC shall discard duplicate received MSDUs. Finally, the MAC shall generate an acknowledgment for each successfully received MSDU according to 5.7.1 or 5.7.2.

5.2 Beacon-enabled channel access

If an OWPAN runs in beacon-enabled channel access mode, channel time is divided into subsequent superframes. Each superframe is composed of three major parts: a beacon transmission, an optional contention access period (CAP) and the contention free period (CFP).

Transmission of the beacon by the OWPAN coordinator is described in 5.2.2.

In the CAP, devices may access the channel randomly by means of slotted ALOHA. Random channel access in the CAP is only allowed for specific procedures and frame types as specified in 5.2.3.

All other frame transmissions happen in the CFP (see 5.2.4). The CFP consists of reserved resources, called GTSs, which are assigned to each device for a given superframe. The coordinator schedules and announces GTS allocations as described in 5.2.5.

5.2.1 Superframe structure

A superframe consists in total of *macNumSuperframeSlots* superframe slots. *macNumSuperframeSlots* is a variable determined by the OWPAN coordinator and announced to the devices in the beacon frame. The maximum number of superframe slots within a superframe is 65535 (see 6.6.1.6). Each superframe slot has a duration of *aSuperframeSlotDuration*. The number of superframe slots and their respective duration determine the total duration of each superframe.

The standard makes use of integer numbers of superframe slots to specify durations within the superframe. That can be durations of the CAP, CAP slots, GTS and other sub-parts of the superframe.

Each OWPAN coordinator defines the superframe structure for its coordinated OWPAN. Consecutive superframes of an OWPAN do not necessarily have to be adjacent but may have channel time between them that is not used by the OWPAN.

In the coordinated topology, the master coordinator determines when the superframe of each OWPAN starts and how long it is. The details of the coordinated topology are outside the scope of this standard.

Of the *macNumSuperframeSlots* superframe slots in a superframe, three consecutive slot groups are used for the beacon transmission, the CAP and CFP respectively as shown in Figure 5-3. The number of superframe slots reserved for the beacon transmission depends on the length of the beacon frame. The length of the CAP is determined by the OWPAN coordinator and may change from superframe to superframe. The remaining slots in the superframe are used for the CFP and can be used for frame transmissions between the devices and the coordinator.

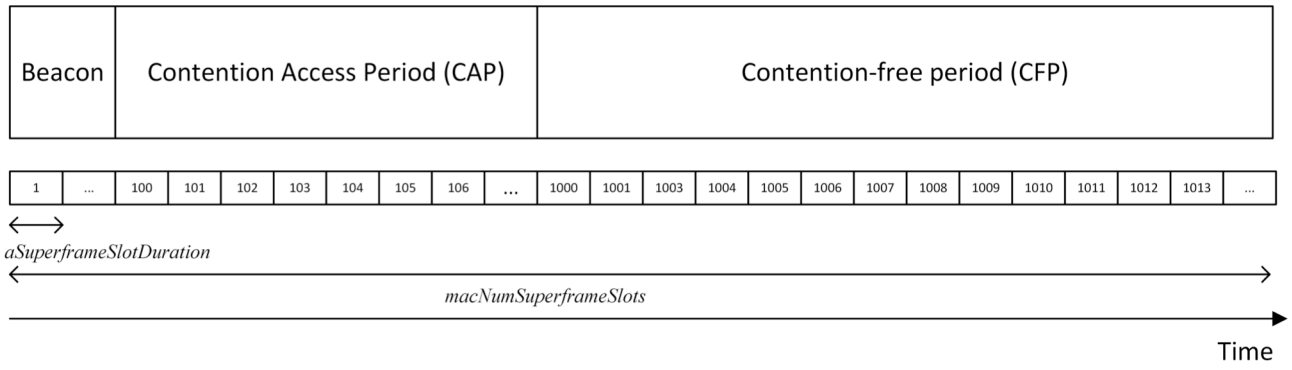

Figure 5-3: Superframe structure with exemplary number of superframe slots

5.2.2 Beacon transmission

In the beacon-enabled channel access mode, the coordinator shall transmit a beacon at the beginning of the superframe. The beacon frame is a control frame containing either solely the *Superframe Descriptor* element or the *Superframe Descriptor* element and additional elements via the *Variable Element Container* element. Beacons should be transmitted with a constant periodicity when possible. The modification of the superframe timing may, for example, be a case in which the beacon periodicity changes.

The coordinator shall maintain the *macBeaconNumber* PIB attribute and increment it by one for every started superframe and corresponding beacon transmission. The *macBeaconNumber* may wrap to 1 after reaching the highest possible value. The highest possible value is determined by the coordinator.

The coordinator shall embed the current *macBeaconNumber* into the *Superframe Descriptor* element of each beacon. Upon reception of a beacon frame, each associated device shall set their value of the *macBeaconNumber* attribute to the value in the received beacon frame.

Upon reception of a beacon frame, devices shall synchronize their clocks to the received beacon frame as described in 5.2.6. Moreover, devices which are either associated with the corresponding OWPAN or attempt to associate with the given OWPAN shall set its *macNumSuperframeSlots, macCapSlotLength* attributes of the MAC to the values contained in the received *Superframe Descriptor* element.

When multiple OFEs are used by the coordinator, the beacon frame shall be transmitted over all OFEs simultaneously. If the coordinator supports the *capMultiOfeEstimation* capability, it shall embed orthogonal pilot symbols in the beacon frame, as detailed in clause 5.8.4.

Devices shall expect the next beacon reception directly after the superframe. If no beacon frame is detected, devices shall keep listening for the next beacon frame in order to synchronize before attempting further transmissions.

5.2.3 Medium access in the CAP

The CAP shall only be used for frame transmissions in the

a) Association procedure (see 5.2.3.1)
b) Resource request procedure (see 5.2.3.2)

The CAP shall start with the superframe slot following the beacon and end before the beginning of the CFP on a superframe slot boundary. The length of the CAP is advertised in the beacon frame / *Superframe Descriptor* element (see 6.6.1.6). Both CAP and CFP periods may shrink or grow dynamically on a superframe-by-superframe basis in order to allow more random access transmissions in the CAP or more scheduled ones in the CFP.

The slotted Aloha scheme is used for contention-based access in the CAP. The superframe slots in the CAP are grouped in so-called CAP slots, which comprise *macCapSlotLength* superframe slots [TBD] each. The number of superframe slots per CAP slot determines the slot size for the slotted Aloha scheme and hence the effectiveness of collision prevention. *macCapSlotLength* is advertised in the beacon frame (clause 6.6.1.6).

A device willing to transmit shall choose a number of CAP slots **RS ("Random Slots")** uniform randomly from [1, **CW**], where **CW ("Contention Window")** is equal to *aInitialCapCw* for the first attempted transmission. Random number generators of all devices shall be statistically uncorrelated. Subsequently, the device shall wait for **RS** CAP slots before attempting transmission. The waiting process may extend over multiple superframes, until a total of **RS** CAP slots have passed. The transmission shall then be performed at the starting boundary of the next CAP slot.

Transmissions in the CAP may not be acknowledged like other frames, as defined in 5.7. If a device implicitly detects that a CAP transmission was not successful, e.g. by the fact that the expected response is never received, the device shall increment the variable **RC (“Retry Count”)** by one. **RC** shall initially be 0. How to detect unsuccessful CAP transmission depends on the specific procedure. Details are given in the respective clauses 5.2.3.1 and 5.2.3.2. The CAP transmission shall ultimately be given up, once **RC** exceeds an implementation-specific value.

For every failed transmission, the device shall double **CW** before attempting retransmission of the frame in the CAP. However, **CW** should not exceed *aMaximumCapCw*. For the retransmission, the device shall then wait again for a random number of CAP slots **RS**, drawn from [1, **CW**] and pursue retransmission at the start of the following CAP slot.

Following a CAP transmission, a device shall be continuously listening in the CFP in order to receive a potential response to the frame transmitted in the CAP.

The process of CAP transmission is visualized for the association and resource request procedure in Figure 5-4 and Figure 5-5 respectively.

5.2.3.1 Association procedure in the CAP

As a device does not have GTS assigned prior to association, the association request frame must be transmitted in the CAP. Hence, the requesting device begins the CAP transmission procedure after preparing a frame containing the *Association Request* element as described in 5.4.5.

If the device supports the *capMultiOfeEstimation* capability, it shall include a *Multi-OFE Feedback* element, containing the CSI obtained from the latest beacon frame reception in the same frame. If the beacon does not contain additional multi-OFE channel estimation pilots, the device shall not include the *Multi-OFE Feedback* element.

A flow chart of the association request procedure is given in Figure 5-4.

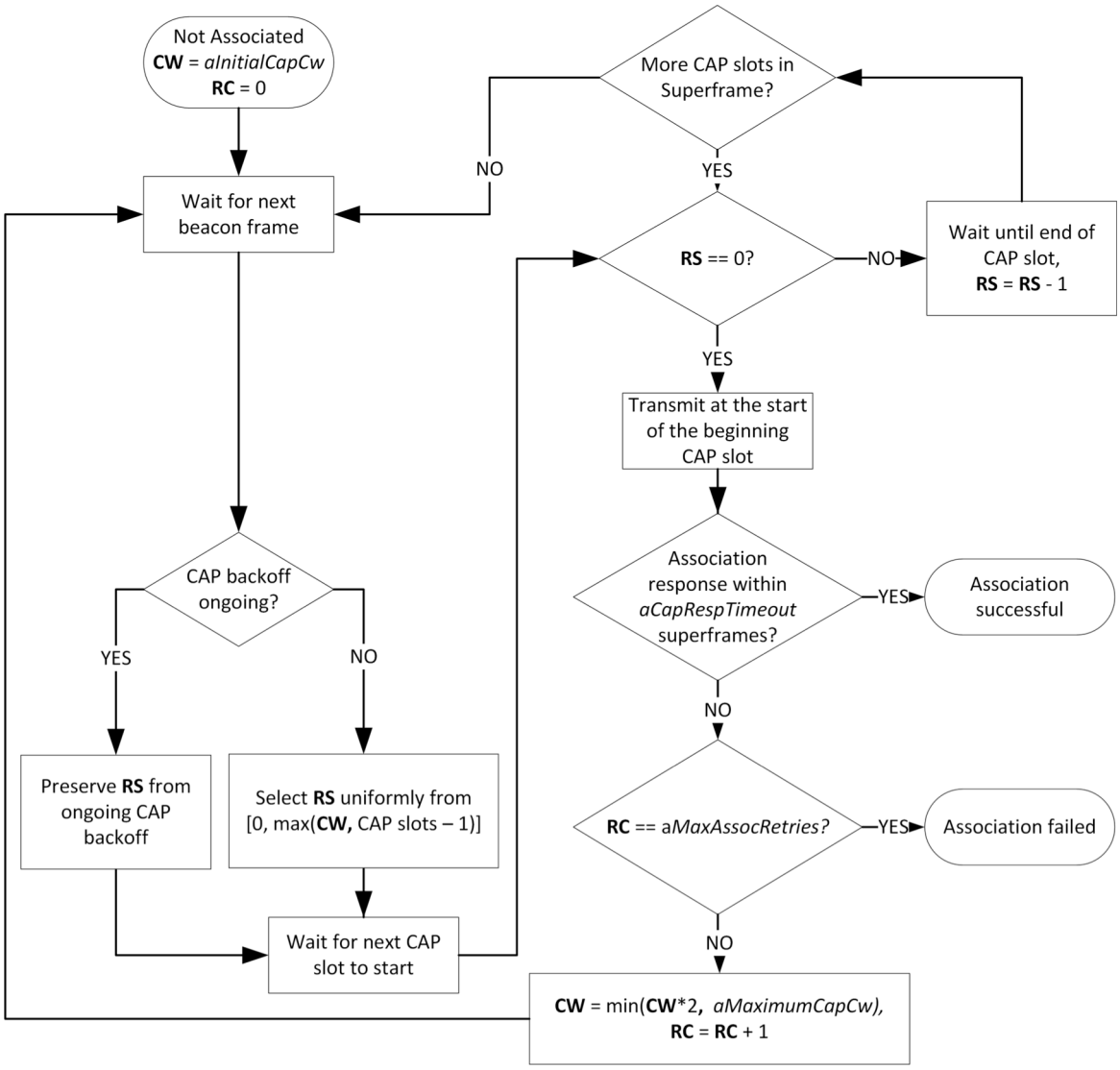

Figure 5-4: Flow chart of an association request in the CAP

After transmitting the *Association Request* element, the device shall stay in receive mode, expecting an *Association Response* element from the coordinator.

After transmitting the *Association Request* element in the CAP, a device may reattempt association through sending the *Association Request* element as described in 5.2.3 after it did not receive an *Association Response* element for at least *aAssociationTimeout*. A device shall not attempt association more than *aMaxAssocRetries* automatically.

5.2.3.2 Resource request procedure in the CAP

When a device does not have any or only insufficient GTS time allocated for its transmissions, it may perform the resource request procedure. For example, this may be the case after the connectivity from coordinator was interrupted and the coordinator stopped allocating GTSs for the device.

In that case, the device may transmit a resource request control frame containing the *XXXXXX* element in the CAP to signal the requirement for GTS time to the coordinator.

If the *capMultiOfeEstimation* capability was negotiated during association, the control frame shall include the *Multi-OFE Feedback* element, containing the multi-OFE CSI obtained from the latest beacon frame reception.

The procedure for a GTS request in the CAP is similar to the association procedure. The corresponding flow chart is depicted in Figure 5-5.

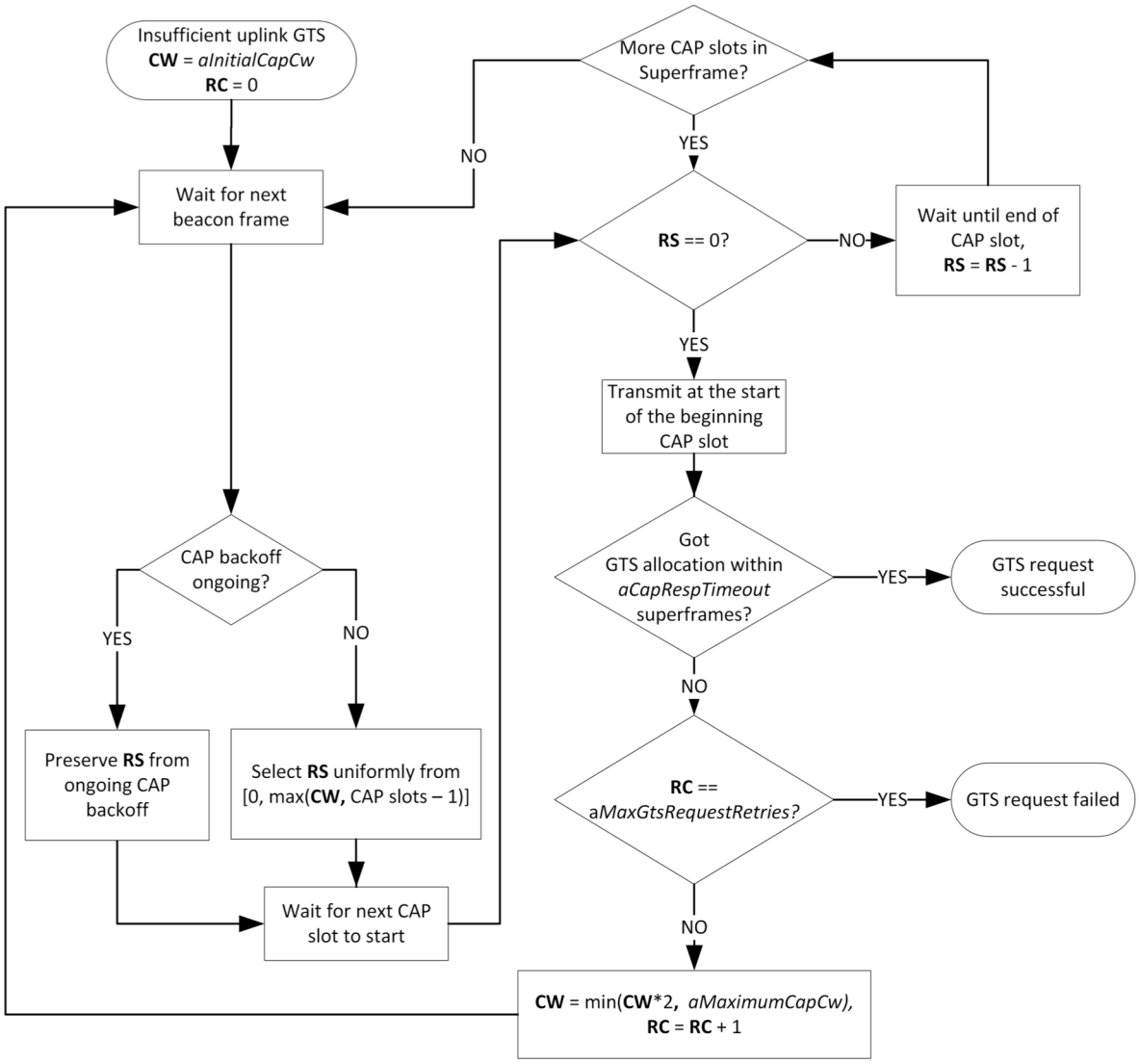

Figure 5-5: Flow chart of a GTS request in the CAP

After transmitting the resource request control frame, the device shall stay in receive mode, shall expec

If the device does not receive any or enough GTS allocation resources, …

5.2.4 Medium access in the CFP

Channel access in the CFP is based on a dynamic TDMA principle. Superframe slots can be reserved on a per-device basis in order to allow contention-free medium access. A group of adjacent superframe slots that is reserved for a specific device is called guaranteed time slot (GTS). The first superframe slot and a duration, given

in an integer number of superframe slots, define the position of a GTS in the superframe as described in clause 6.6.1.8.1. GTS shall only reside within the CFP.

A device shall keep a list of all its upcoming GTS after it received the corresponding *GTS Descriptor* element. A device shall only transmit in GTS that were assigned to it.

Devices should ensure that transmitted signals cannot interfere with transmissions in other GTS at any other device. This includes, for example, regarding for the total transmit delay introduced by the used PHY and the assumed propagation delay and range. A device shall ensure that its transmissions adhere to the rules for inter-frame spaces, as described in 5.2.7.

A device with a GTS may or may not make use of all the allocated time duration within the GTS. The selection of an MPDU for transmission is determined locally by the device depending on the number of pending frames in its queue and the value of their user priority fields and potentially other criteria.

The coordinator may perform transmissions to a device at any point in the CFP. Hence, all devices must listen for receptions during the whole CFP. Vice versa, the coordinator must be listening to the channel for receptions during each GTS.

5.2.5 GTS allocation and signalling

Only the OWPAN coordinator is entitled to allocate GTSs. Any allocated GTSs shall be located within the CFP.

If the coordinator has control over multiple spatially distributed OFEs, it may allocate the same superframe slots in different GTS for multiple spatially distant devices in order to facilitate spatial reuse of resources throughout the OWPAN's coverage area. However, the coordinator must ensure that transmissions from and to devices that share the same superframe slots do not interfere or the interference does not lead to packet losses.

Devices aid the coordinator in the GTS allocation process through providing information about their queue states and making flow reservations. For that purpose, devices may transmit *QueueState* elements to the coordinator.

Devices aid the coordinator at allocating GTSs in an interference-free manner through providing information about the signal strengths by which they receive the nearest OFE. For this purpose, devices shall transmit *Multi-OFE Feedback* elements to the coordinator if the *capMultiOfeEstimation* capability was negotiated during association.

The coordinator may move GTSs within the superframe on a superframe-by-superframe basis. This allows the coordinator the flexibility to rearrange GTS assignments, optimize the utilization of resources and prevent collisions of GTSs if visibility and signal strength varies among OFEs and devices due to mobility.

GTS allocations shall be advertised from the coordinator to the corresponding devices via control frames including *GTS Descriptor* elements. These control frames shall be unicasts and only be received by the devices for which the GTS allocations are designated. Devices shall infer that a GTS allocation concents itself based on that receiver address.

Each *GTS Descriptor* element shall include the superframe number of the superframe it gets valid in. A GTS allocation is only valid for one superframe.

5.2.6 Synchronization

All devices, whether they are associated with a beacon-enabled OWPAN or attempting association, shall be synchronized to the coordinator's clock before they start transmission or reception. The beacon sent at the

beginning of every superframe enables synchronization of the devices in the beacon-enabled OWPAN through time of arrival synchronization.

Each device in the OWPAN, including the coordinator, shall begin counting the first superframe slot at the beginning of the PHY preamble of the beacon, as shown in Figure 5-6. All superframe slots and hence timings within the superframe are thus relative to the start of the beacon preamble.

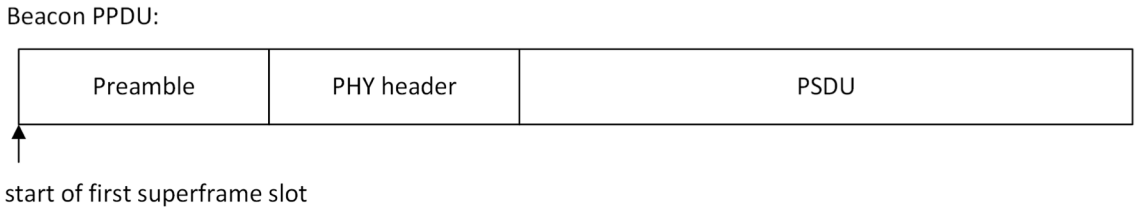

Figure 5-6: Timing relative to the beacon frame reception at every device

A compliant device implementation shall maintain the accuracy of the local time to be at least as accurate as *aClockAccuracy*.

5.2.7 Interframe spaces

[Maybe one can include a safety barrier for the maximum assumed propagation delay?]

The only IFS defined by this standard is the Turn Around Interframe Space (TAIFS) *aGtsTaifs*. The TAIFS is required to ensure sufficient turnaround time between transmissions. The turnaround time is defined as the maximum time a transceiver requires to switch from transmitting to being ready to receive or from receiving to starting a subsequent transmission. A transmitter has to ensure that its transmissions end at least a TAIFS before the end of the GTS in order to enable all receiving devices to fully utilize their GTSs from the beginning. The TAIFS shall be at least the maximum expected turn-around time as defined for each PHY.

If a device is able to ensure that all other devices are able to transmit and receive orderly in their GTSs, for example because they implement the *capFullDuplex* capability, it may disregard the requirement to finish transmissions at least a TAIFS before the end of its GTS.

Spaces between successive transmissions of a single transmitter are not strictly required. Receivers are expected to be able to process incoming frames fast enough to handle contiguous transmissions.

5.2.8 Guard time

In a TDMA system, guard times are required to keep transmissions in adjacent GTS from colliding when local clocks of devices are imperfectly synchronized, e.g. through drift caused by frequency inaccuracies of device-local clocks. A GTS is defined by the start time and the duration, as specified in the GTS element (see clause 6.6.1.8). Guard time is the time between the end of one GTS and the start of the next GTS.

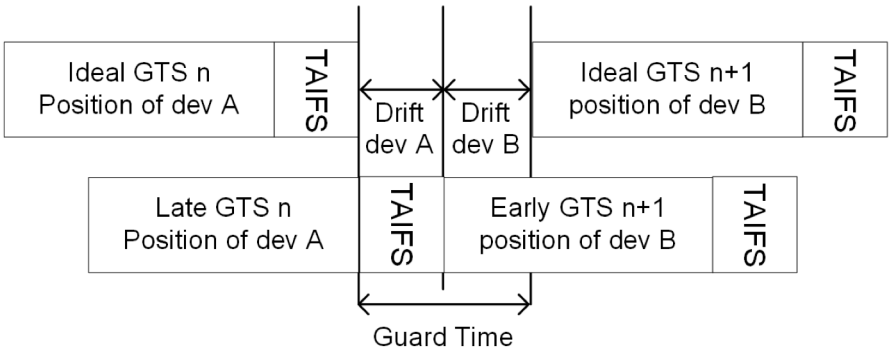

Figure 5-7: Application of the guard time and TAIFS between adjacent GTSs

Figure 5-7 depicts an illustration of the guard time such that consecutive transmissions are always separated by at least a TAIFS if the owners of adjacent GTS have drift towards the other GTSs.

The required guard time depends on the maximum drift $MaxDrift$ between a device's local time and the ideal time. This drift is a function of the time elapsed since a synchronizing reference event, i.e. the beacon reception, and the precision of local oscillators in OFEs and devices defining the local sampling clock. In an IEEE 802.15.13 OWPAN, the synchronizing event is the start of the preamble of a beacon. The maximum drift $MaxDrift$ can be calculated as follows:

$$MaxDrift = clock\ accuracy / superframe\ duration$$

The clock accuracy depends on the device implementation but shall not be worse than the value given by the *aClockAccuracy* PIB attribute. The superframe duration is the current duration of the superframe and hence periodicity of the synchronizing event.

The synchronization accuracy $SyncAccuracy$ describes how accurately the devices can be synchronized to the coordinator's clock. This value depends on the coordinator implementation and shall be determined by the vendor. The value shall also include the uncertainty introduced through the varying propagation time of the beacon frame, based on which the synchronization is performed.

The coordinator shall ensure that a guard time of at least $2 \cdot (MaxDrift + SyncAccuracy)$ lies between two subsequent GTS that are not orthogonal in space.

5.3 Non-beacon-enabled channel access

[Rename to polling-based channel access?]

[see document 15-18-0488-01-0013]

5.4 OWPAN management

This clause describes the scanning for existing OWPANs, starting of new OWPANs as well as the association and disassociation of devices with / from and existing OWPANs.

5.4.1 Scanning for OWPANs

A scan procedure is performed by a device to detect any OWPANs that are operating in its vicinity. In light communication, a single frequency range in the baseband is utilized for all transmissions. Hence, scanning for

existing OWPANs reduces to the scanning of a single frequency channel. However, multiple OWPANs may be coordinated by a master coordinator and share the total available channel time.

IEEE 802.15.13 devices shall support passive scanning for OWPANs. During a passive scan, the device listens for incoming frames but also non-decodable signals whose received power exceeds the threshold of *macEdScanThreshold*. If a device makes use of multiple optical frontends, it shall listen on all frontends and try to decode receptions for each frontend individually.

The scan is started upon request by the DME through the MLME-SCAN.request primitive or by the MLME itself. A device instructed to scan for OWPANs shall listen for received beacon or RA frames during the scan period. During a scan, the MAC sublayer shall discard all other received frames.

For every successfully decoded beacon or RA frame in the scan period, the device shall add the corresponding OWPAN ID and OWPAN name to the scan result list. It shall furthermore add the received electrical SNR and security type as indicated in the frame to the result list. The returned list shall not contain duplicate entries.

If a device detects at least one non-decodable signal that has a received power of more than *macEdScanThreshold* during the scan time, the device shall add an entry with OWPAN ID = 0xFFFF, OWPAN name = “Unknown” and the received power level of the strongest received signal to the scan result list.

If the scan was initiated through the MLME-SCAN.request primitive, the results of the scan shall be returned via the MLME-SCAN.confirm primitive.

[Sequence graph?]

5.4.2 Starting an OWPAN

The process of starting a new OWPAN is initiated after a coordinator-capable device was instructed to do so through the MLME-START.request primitive of the MLME-SAP. This subclause describes the steps involved in starting and maintaining the OWPAN. If the prospective coordinator maintained an OWPAN before, the DME shall stop the OWPAN, according to 5.4.4, prior to starting a new OWPAN in order to reset all MAC and PHY state and disassociated potentially associated devices.

The DME shall issue a scan immediately before attempting to start a new OWPAN. The DME shall only issue the MLME-START.request primitive, if the corresponding scan reported an empty result list or if resource coordination between multiple OWPAN coordinators can be provided through a coordinated topology.

The DME of the prospective coordinator shall select an OWPAN ID and OWPAN name. If the coordinator implements the *capShortAddressing* capability, it shall adopt the selected OWPAN ID as its short address. The DME shall provide the selected OWPAN ID, OWPAN name and its short address as a parameter of the MLME-START.request. The MAC shall set the *macSecurityType* attribute to the security type conveyed via the MLME-START.request primitive.

NOTE - The OWPAN ID may be allocated by a master coordinator. Two neighboring OWPANs shall not use the same OWPAN ID. Two OWPANs may use the same OWPAN name.

On receipt of the MLME-START.request, the MLME of the prospective coordinator shall prepare operation as a coordinator and subsequently start transmitting frames in accordance with the configured channel access mode.

5.4.3 Maintaining an OWPAN

After successfully starting an OWPAN, the coordinator and associated devices shall support the primitives of the MCPS-SAP and the corresponding MAC data path functionality as well as the primitives of the MLME-SAP that implemented and part of the supported capabilities.

A coordinator may change parameters of a running OWPAN such that devices that are associated with the OWPAN need to modify their respective PIB attributes. To control PIB attributes of associated devices, the coordinator may transmit an *Attribute Change Request* element to the concerned device. The *Attribute Change Request* element shall contain the corresponding PIB attribute name and the new value to be set.

A device receiving the *Attribute Change Request* shall modify the value of the indicated attribute to reflect the requested change. Subsequently, it shall respond to the coordinator with an *Attribute Change Response*, indicating the result of the attempted attribute change.

5.4.4 Stopping an OWPAN

To stop an existing OWPAN, the DME of a coordinator shall issue the MLME-STOP.request through the MLME-SAP. Upon reception of the primitive, the coordinator should disassociate all associated devices with an appropriate reason code. Successively, it shall purge all state that was introduced during the up time of the OWPAN.

5.4.5 Associating with an OWPAN

The association procedure involves multiple steps:

1. Request association with the goal to obtain (temporary) channel access
2. Optionally request authentication if required by the OWPAN

5.4.5.1 Association request

A device MLME is instructed to attempt association with an existing OWPAN by the DME through the MLME-ASSOCIATE.request primitive. Before starting the association procedure, a device shall reset all state including queues and variables of its MAC.

After receiving the MLME-ASSOCIATE.request, the device shall prepare a management frame to be transmitted to the OWPAN coordinator. The management frame shall include the *Association Request* element by either being a dedicated *Association Request* frame or having the *Association Request* element included by other means such as being contained in the *Variable Element Container*.

The management frame shall make use of 6 octet MAC addresses. The *Receiver Address* of the management frame shall be set to the coordinator's address [in beacon-enabled mode, we need the beacon to use the full 6 octet MAC address therefore]. The *Transmitter Address* of the frame shall be set to the 6 octet MAC address of the device seeking association.

The *Association Request* element shall include the capabilities supported by the device for the desired association. Furthermore, the request shall include the necessary information as detailed in clause 6.6.1.1.

The requesting device shall transmit the management frame to the coordinator of the OWPAN in accordance with the channel access rules for association as detailed in clauses 5.2 and 5.3 respectively. The frame shall be transmitted unprotected (refer to clause 5.7).

If the coordinator MLME decides to pursue association, it shall prepare a management frame containing the *Association Response* element. The *Association Response* element shall include a set of capabilities to be used during the prospective association. The set of capabilities shall include no other capabilities than previously

indicated by the device in the *Association Request* element. The precise set of capabilities may be selected by the coordinator.

If the coordinator decides, not to pursue association, it shall an *Association Response* element with the appropriate *Status Code* set.

If the OWPAN is secured and requires further authentication, the *Association Response* element shall contain further information required for the subsequent authentication of the device as detailed in clause 8. After successful reception of an *Association Response* element, the device shall perform authentication with the OWPAN coordinator if necessary.

A sequence chart of a successful association procedure is depicted in Figure 5-8.

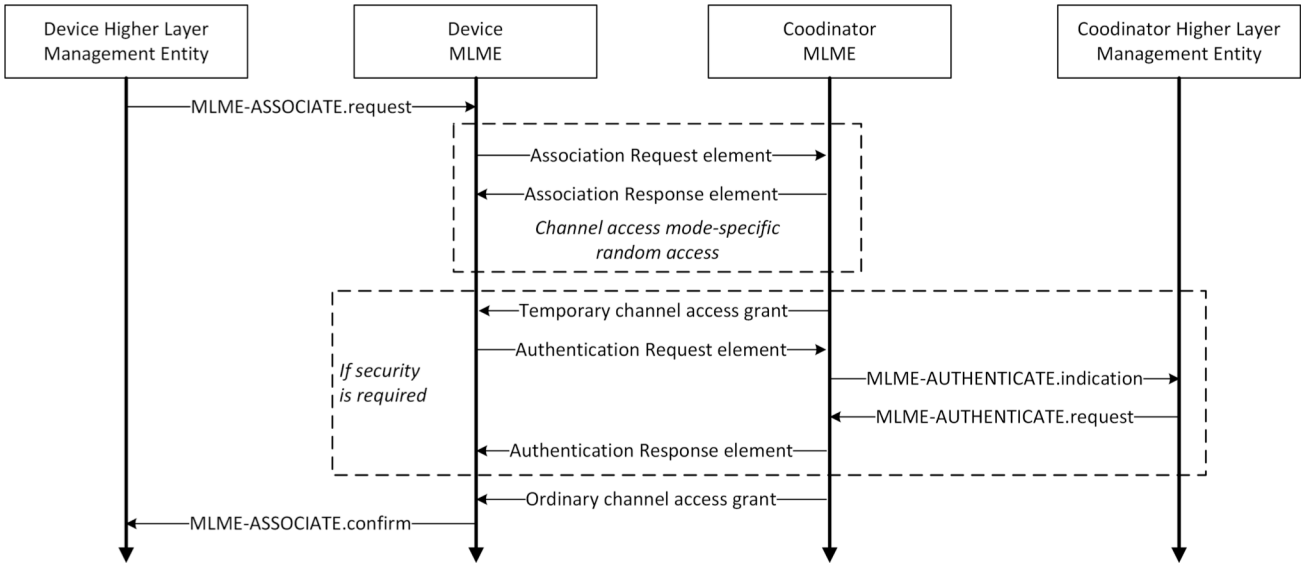

Figure 5-8: Association procedure message exchange

5.4.5.2 Authentication request

If the *Association Response* element received by the device indicates that further authentication is required, the device shall process the authentication material included in the *Association Response* element in correspondence with the applied security type. The resulting authentication data shall then be included in an *Authentication Request* element and transmitted via management frame to the coordinator.

For transmission of the *Authentication Request* element, the coordinator may grant temporary channel access to the associating device. If not, the device may transmit the *Authentication Request* element analogously to the association request in the CAP.

After receiving the Authentication Request element from the associating device, the coordinator MLME shall indicate to the DME that a device seeks authentication via the MLME-AUTHENTICATE.indication. The DME shall then authenticate the device and provide the result to the MLME via the MLME-AUTHENTICATE.request. The DME shall respond to the MLME-AUTHENTICATE.indication within 30 seconds.

The MLME shall transmit an *Authentication Response* element to the device attempting association. If the authentication was successful, the device shall consider being associated with the OWPAN. For the duration of the ongoing association, it shall make use of the security, i.e. encryption, integrity assurance and replay protection required by the OWPAN and detailed in the respective security type clause.

The coordinator may consider the device successfully associated after receiving an acknowledgment for the frame containing the *Association Response* element or *Authentication Response* element respectively.

5.4.6 Disassociating from an OWPAN

The disassociation of a single device from an OWPAN may be initiated by either the coordinator of the OWPAN or the affected device itself through the MLME-DISASSOCIATE.request primitive.

To disassociate a device from the OWPAN, the coordinator shall transmit a management frame, containing the *Disassociation Notification* element, to the device to be disassociated as depicted in Figure 5-9 a). If the coordinator does not receive a corresponding acknowledgment frame, it shall retransmit the *Disassociation Notification* element up to *macMax*…. After reaching the maximum number of retransmissions, the device can be considered disassociated.

A device that wants to disassociate from the OWPAN shall transmit a management frame containing the *Disassociation Notification* element to the coordinator of the OWPAN as depicted in Figure 5-9 b). If the device does not receive a corresponding acknowledgment frame, it shall retransmit the *Disassociation Notification* element up to *macMax*…. After reaching the maximum number of retransmissions, the device can be considered disassociated.

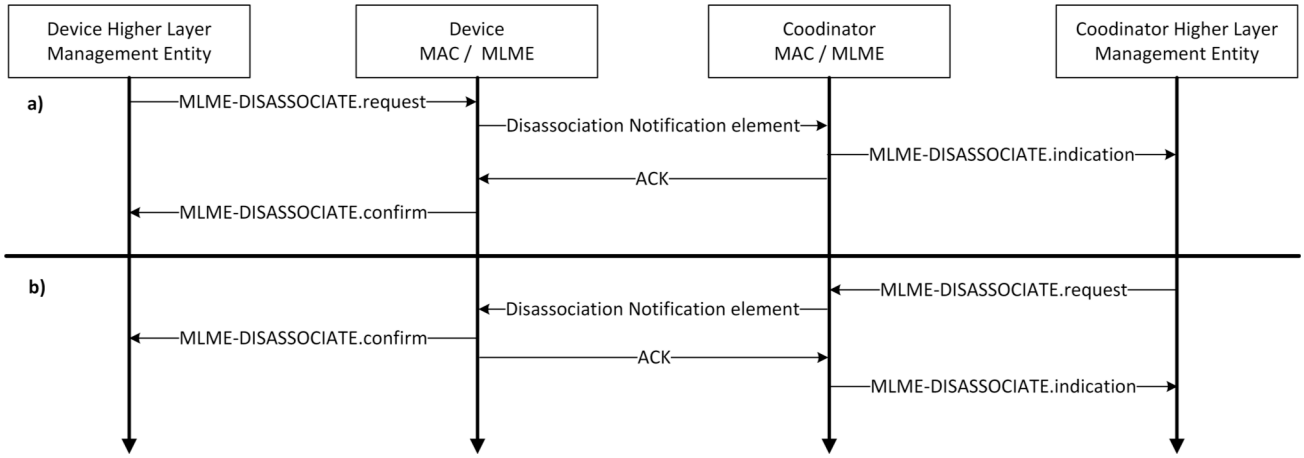

Figure 5-9: Disassociation initiated by the device (a) and the coordinator (b)

If a coordinator does not receive any management or data frames from a device for a duration of XXX,…

5.5 Fragmentation and reassembly

Fragmentation may be performed by the transmitting device on a MSDU or A-MSDU. An (A-) MSDU shall be fragmented into 16 fragments at most. All fragments shall contain an even number of octets, except the last fragment, which may contain an odd number of octets. Once the (A-) MSDU is fragmented and a transmission attempted, it shall not be fragmented again. The smallest size of a fragment, excluding the last fragment, shall be at least a*MinFragmentSize*.

An MPDU containing a fragmented MSDU or A-MSDU shall have the *Sequence Control* element present. All fragments but the last fragment shall be sent with the *Last Fragment* field of the data MPDU set to 0. The last fragment shall have the *Last Fragment* field set to 1. Each subsequent fragment shall be sent with the *Fragment Number* field incremented. However, the *Fragment Number* field shall not be incremented when a fragment is retransmitted.

All fragments of the same (A-) MSDU shall have the same sequence number in the MPDU header. Defragmentation of an (A-) MSDU is the reassembly of the received fragments into the complete (A-) MSDU. The (A-) MSDU shall be completely reassembled in the correct order before delivering it to the higher layer.

The receiving device may discard the fragments of an MSDU if it is not completely received within a timeout determined by the receiving device. The destination device may also discard the oldest incomplete MSDU if otherwise a buffer overflow would occur. Fragments shall be transmitted in order of their fragment numbers. If the no-ACK policy is used, the destination device shall discard an MSDU immediately if a fragment is missing. A device shall support concurrent reception of fragments of at least three MSDUs.

5.6 Aggregation

A device may aggregate multiple MSDUs in a single MPDU in order to avoid the overhead of transmitting multiple MPDUs and corresponding PPDUs. Aggregated MSDUs (A-MSDUs) are transmitted in the payload of data frames of the A-MSDU subtype (see 6.3).

5.6.1 Aggregation procedure

The optional aggregation procedure, as part of the transmit process detailed in Figure 5-1, is applied when a device MAC decides to aggregate multiple MSDUs in a single MPDU.

A device shall only transmit multiple MSDUs in a single MPDU if all MSDUs have the same receiver address. [What about limitations for aggregation? Priorities? Flows?].The aggregated MSDUs are denotes as a single A-MSDU. All MSDUs in an A-MSDU shall be either protected or unprotected. Protected MSDUs shall not be mixed with unprotected MSDUs. The total resulting MPDU size, resulting from all aggregated MSDUs and additional fields for aggregation, in octets shall not exceed *phyMaxPsduSize* of the used PHY.

Each MSDU to be part of an A-MSDU shall be wrapped in an *MSDU Aggregation* element. The *MSDU Aggregation* element shall include the total length of the wrapped MSDU in octets.

[Cover order of aggregation and delivery to the MCPS SAP]

Furthermore, the *MSDU Aggregation* element shall include the sequence number assigned to the MSDU. The MPDU having the resulting A-MSDU as payload shall have the same address fields as every single MSDU would have in a non-aggregated transmission.

[Priorities?]

If the MPDU containing aggregated MSDUs is transmitted reliably, it shall be assigned a single sequence number like an MPDU containing only a single MSDU. Upon reception of an acknowledgment by the receiver of the MPDU, all MSDUs contained in the MPDU shall be considered acknowledged and hence successfully transmitted.

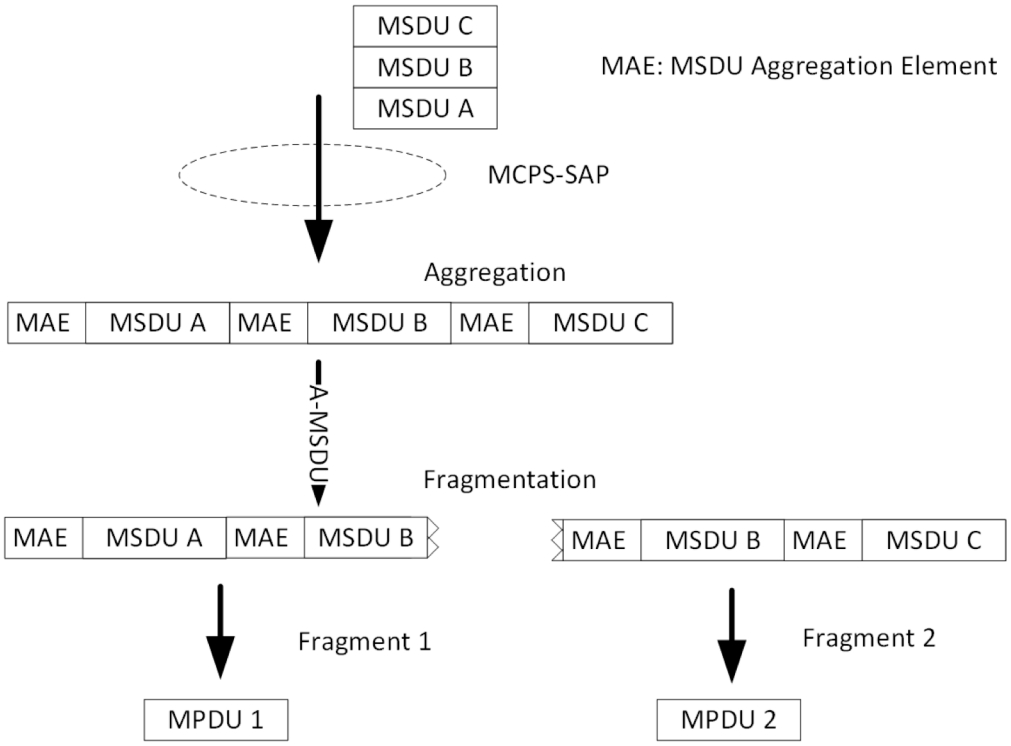

Figure 5-10: Aggregation and fragmentation

Figure 5-10 depicts the aggregation and fragmentation during transmission of a frame. Three MSDUs, arriving at the MAC through the MCPS-SAP, are aggregated by being included in a *MSDU Aggregation* element (abbreviated as MAE).

An A-MSDU may optionally be fragmented. In the example, the A-MSDU, consisting of the three MSDUs A, B and C are divided into two fragments and wrapped in an MPDU each. The MPDU includes a new sequence number, serving reassembly of the A-MSDU at the receiver. That sequence number does not have to be acknowledged by the recipient device. However, each MSDU in an A-MSDU has its sequence number associated in the MSDU Aggregation element. These sequence numbers must be acknowledged by the receiver if the *Frame Control* element of the MPDU has the *Ack Request* bit set.

5.6.2 Disaggregation procedure

If a device receives an A-MSDU data frame, it shall first check integrity of the whole MPDU based on the MPDU FCS field. If the MPDU was received without errors, the device shall acknowledge the corresponding sequence number of the MPDU.

Subsequently, the receiving device shall separate the payload of the A-MSDU frame into separate *MSDU Aggregation* elements based on the size given in the *MSDU Aggregation* element. The MAC shall then check the integrity of each MSDU based on the FCS included in the corresponding *MSDU Aggregation* element. If the MSDU was received without errors, the MAC shall acknowledge the corresponding sequence number included in the *MSDU Aggregation* element.

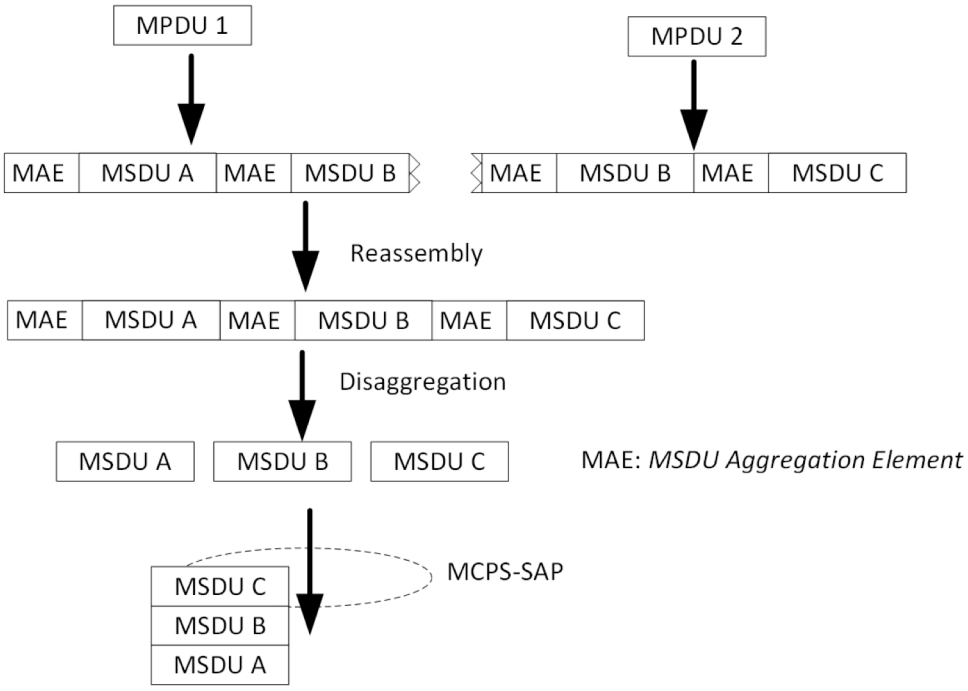

Figure 5-11: Reassembly and disaggregation

Figure 5-11 shows the reassembly and disaggregation procedure. Two MPDUs 1 and 2 are received from the PHY. Both

5.7 Protected transmission

Transmissions between IEEE 802.15.13 devices may be protected. The protection ensures that MSDUs are neither duplicated nor changed in order during transmission between two MACs. Moreover, the protection prevents frame losses by an acknowledgment- and retransmission mechanism.

Each device shall maintain an individual sequence number counter for transmissions towards every peer device it communicates with. The sequence number is a 12 bit wide unsigned integer, which wraps to 0 after the highest possible value.

If an MSDU was received through the MCPS-SAP with the *Protected* parameter set to TRUE, it shall be transmitted in an MPDU having the *Ack Request* bit set to 1 and hence include the *Sequence Control* field. The *Sequence Control* element shall contain a sequence number ensuring delivery of the contained MSDU.

For A-MSDUs, the *Sequence Control* field of the MPDU shall contain a sequence number, if the A-MSDU includes at least one MSDU that was received with the *Protected* parameter set to TRUE.

Received MPDUs should be regarded as duplicates if they are bitwise equal, i.e. also carry the same sequence and fragment number. Two received bitwise equal MPDUs shall not be regarded as duplicates if the receiver received more than *aProtectedWindow* unique sequence numbers since the reception of the first of the potential duplicate MPDUs. If a duplicate is detected, the receiver shall discard the last received duplicate.

A transmitting device shall not have transmitted more than *aProtectedWindow* unacknowledged MPDUs outstanding.

If the MPDU indicates acknowledged transmission and has the *ACK Request* bit set in the *Frame Control* element as well as the *Sequence Control* element present, the receiver of that MPDU shall acknowledge successful transmissions with either of the following acknowledgment types:

- Single acknowledgment (clause 5.7.1)
- Block acknowledgment (clause 5.7.2)

Otherwise, it shall not transmit an acknowledgment.

5.7.1 Single acknowledgement

The receiver of an MSDU may decide to acknowledge the successful reception by means of a single acknowledgement. Hence, the information returned to the transmitter contains solely information about the successful reception of a single MSDU.

The acknowledgment information shall be embedded in the *ACK Information* element as part of either of the following:

- In a dedicated *Acknowledgement* control frame, containing only the *ACK Information* element in its payload.
- In any frame including the *Variable Element Container* element, containing the *ACK Information* element.

5.7.2 Block acknowledgement

A receiver may acknowledge successfully received MSDUs by means of a cumulative acknowledgement. The corresponding block acknowledgment frame contains information about one or multiple successfully received MPDUs (i.e. their sequence numbers) in an aggregated way through including the *Block Acknowledgment* element.

The receiver may transmit a block acknowledgment either unsolicited or upon request by the transmitter of received MPDUs through a *Block Acknowledgment Request* element.

The Block Acknowledgment element shall only be transmitted in frames having a unique source address. The source address of the frame, containing the *Block Acknowledgment* element identifies the acknowledging device.

[TODO: explain block ACK further]

5.7.3 Retransmission

A device shall retransmit a protected MSDU after it was not acknowledged after at least *macRetransmitTimeout*. The *macRetransmitTimeout* PIB attribute may be adjusted by the coordinator through the parameter management procedure described in 5.4.3.

A device shall not attempt more than *macMaxFrameRetries* of the same MPDU. After the last retransmission attempt failed, the device shall consider the transmission of all MSDUs in the MPDU as failed and indicate the result to the higher layers through the MCPS-SAP with the corresponding reason.

A device shall consider all MSDUs of a previously transmitted MSDU or A-MSDU as successfully received if it receives an acknowledgment for the sequence number of the corresponding MPDU.

5.8 Adaptive transmission

A device may select the rate, e.g. through choosing modulation and coding, for each outgoing PPDU based on available information about the channel between itself and the receiver. The information is typically obtained from each designated receiver via a feedback mechanism or inferred by the transmitter by other means.

5.8.1 Multi-rate MCS

IEEE 802.15.13 PHYs are able to transmit frames under application of varying modulation and coding schemes (MCS). The specific definition of an MCS is dependent on the used PHY. It may contain details regarding the error coding, and modulation.

By default, a device may select the MCS for transmissions to another device freely. Rate selection algorithms are out of scope of this standard. However, for some frames, usage of specific modulation and coding schemes is mandatory (see 5.8.2).

If two devices support the *capEffectiveChannelFeedback* capability, a receiver of frames may request the usage of a specific MCS from the prospective transmitter (see 5.8.3).

5.8.2 Transmission of essential frames

Some frames shall be transmitted at the base rate specific to the used PHY. The frames to be transmitted with base rate are listed in Table 5-1.

Association Request and *Association Response* frames
Disassociation Notification frames
Beacon frames
Random Access frames
Control frames containing the *ACK Information* or *Block Ack Response* element

Table 5-1: Frames to be transmitted at base rate

5.8.3 MCS request feedback

An IEEE 802.15.13 device supporting the *capEffectiveChannelFeedback* capability is able to measure the quality of signals received from other devices. Moreover, it shall be able to transmit *MCS Request* control frames and process received modulation request control frames as follows.

MCS Request control frames are transmitted from the prospective receiver of frames to the prospective transmitter. The prospective receiver may transmit an *MCS Request* element if it detects that the previously requested MCS may not be successfully decodable or that an MCS with a higher rate could be used.

If a device receives a *MCS Request* control frame from another device, it shall make use of the requested modulation and coding schemes for subsequent transmissions if transmissions do not comprise frames that require special modulation and coding as defined in 5.8.2.

5.8.3.1 Bitloading MCS request

If the *capHbPhy* was negotiated during association, each device may measure the effective channel at receptions of unicast frames. Based on the channel measurement result, the receiving device may subsequently request the usage of a certain BAT for future transmissions from transmitter.

If the channel measurement result indicates that an earlier requested BAT cannot successfully be decoded, the device shall request usage of a new and sufficiently robust BAT from the transmitter. Also, a new BAT may be requested if the device could not decode the payload of a frame with the BAT indicated in the header of that frame. A device may also request usage of a new, BAT in order to increase throughput, for example because the channel quality improved.

For the request, the device shall prepare a *BAT Request* element as follows:

The *Valid Bat Bitmap* field shall indicate the set of BATs that may be used for transmissions to the device. The bitmap shall only indicate the BATs as valid for which the device knows that the prospective transmitter assumes the same configuration as the device. For BATs which cannot be known to have the same configurations at the device and the prospective transmitter, the device shall set the bit in the bitmap to 0. How to infer that the prospective transmitter has the same configuration for a BAT is described further in the text.

The *Updated BAT* field shall indicate a new and previously invalid BAT ID for which a new configuration is requested. The *FEC Block Size* field shall contain a block size for error coding and the *FEC Code Rate* field shall contain the code rate which is rested to be used for subsequent transmissions.

The device shall fill the *BAT Group 1 ... N* fields with the requested bits per subcarrier. It may form multiple groups, containing varying number of subcarriers, to have the same modulation format. The total number of groups shall cover all available subcarriers of the PHY. The total number of subcarriers covered by the groups may be larger than the actual number of subcarriers. In that case, the excess subcarriers, contained in the last BAT Group shall be ignored.

The device shall transmit the *BAT Request* element in a control or management frame. In case the device transmits the element via a control frame, it cannot expect an acknowledgment and hence does not know whether the prospective transmitter has received the request.

A device can know that the BATs are consistent either for predefined BATs, or by deriving that fact from a frame reception. This is the case, if the BAT Request element was transmitted via a management frame. Alternatively, the device shall

TBD

5.8.4 Multi-OFE channel feedback

Coordinators supporting the *capMultiOfeEstimation* capability shall be able to transmit multi-OFE pilots whereas non-coordinator devices supporting the *capMultiOfeEstimation* capability shall be able to receive multi-OFE pilots and subsequently estimate the channels between each transmitter of multi-OFE pilots.

If a coordinator makes use of multiple OFEs, it may embed different divisions of the multi-OFE pilot symbol in the PPDU for every OFE as defined in clause 10 and 12. If the coordinator is part of a coordinated topology, the divisions and time slots to be used for multi-OFE pilot embedding shall be coordinated by the master coordinator for all coordinators.

The receiver of multi-OFE pilots is able to estimate the individual CSI between the transmitter of each pilot symbol and itself, although the signals of multiple transmitters may overlap in time. The gathered CSI comprises time domain taps, which are described by the respective optical signal power and delays relative to the very first received tap.

Upon reception of a PPDU containing multi-OFE pilot symbols, a device shall estimate the individual channels. The device shall then transmit a *Multi-OFE Feedback* element, containing the measured CSI for each identified transmitting OFE of orthogonal pilots, to the coordinator of the OWPAN.

A device shall not use a format that provides only smaller values for quantization than the actual strength value.

5.9 MIMO Communication

The use of adaptive multiple input multiple output (MIMO) communication for High rate PD is optional. **Figure 1** illustrates the general overview of the MIMO communication.

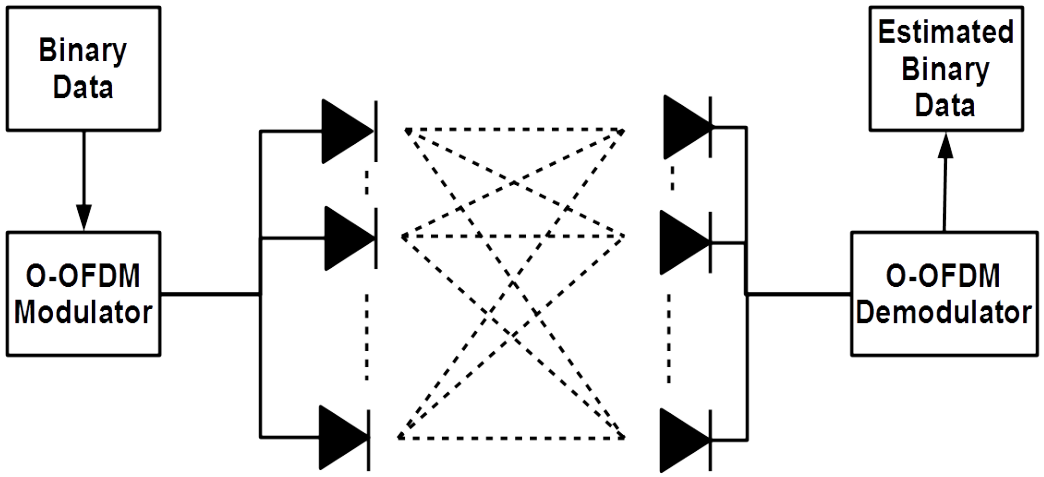

Figure 18 General Overview of High-Rate PD MIMO Communication

The information is transmitted from multiple LEDs to multiple PDs through optical wireless channels. The maximum number of LED arrays and PD arrays, which can be supported by the High rate PD is set to 16.

5.9.1 MIMO Communication Setup

To setup the MIMO communication, it is assumed that the association is realized in SISO mode. Afterwards VLC receiver sends MIMO info request the start the setup. Transmitter provides the number of Transmit elements and its MIMO capabilities. The receiver sends channel info request to start the channel estimation process. The transmitter sends in SISO mode information elements, which include the channel estimation sequence. The channel state information (i.e., channel coefficients, channel correlation, signal-to-noise ratio etc.) is estimated by the receiver. Based on channel conditions, the receiver selects the optimal transmission mode, which includes modulation type, modulation order, MIMO configuration and MIMO type. The selected transmission mode is provided to the transmitter via a feedback channel.

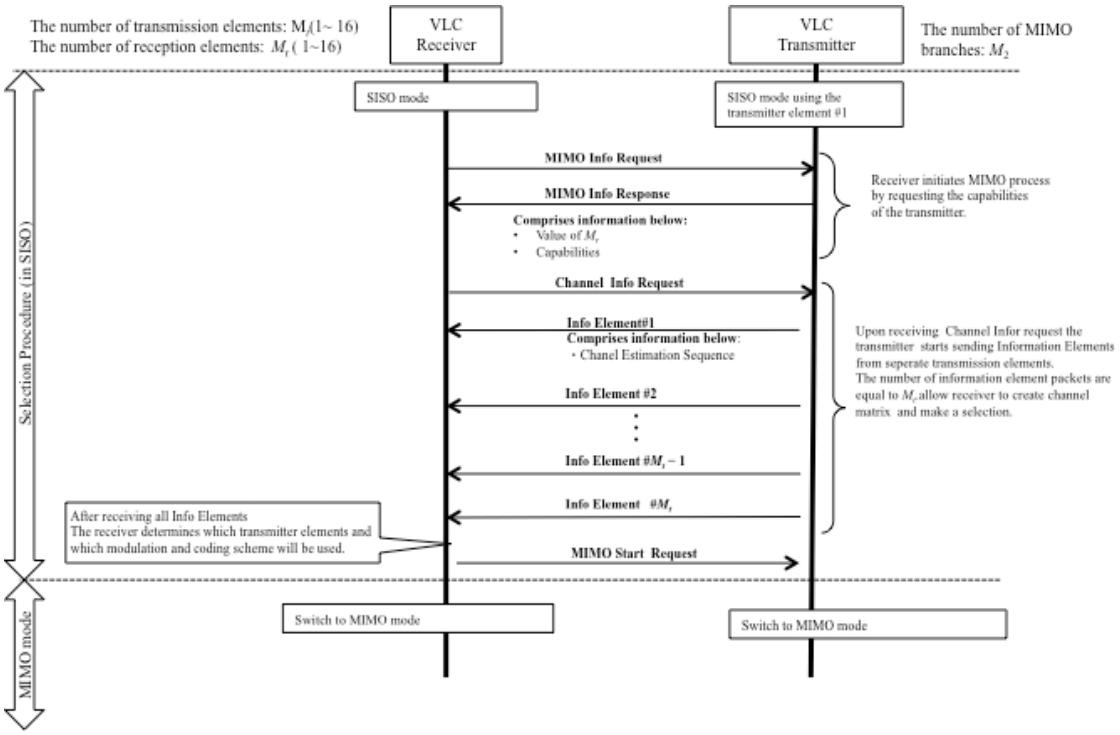

Figure 19 MIMO Communication Setup

All possible communication modes should be set on a lookup table for selection.

The selection algorithm is provided below. The performance metric is pre-determined values according to the quality-of-service (QoS) requirements for targeted applications, such as highest date rate under a specific BER threshold.

Two main MIMO schemes can be supported.

1) Repetition Coding:

In the repetition coding, the same information is transmitted from all transmit elements.

2) Spatial Multiplexing

In the spatial multipexing case every transmit element sends independent information.

6 MAC frame formats

This clause provides specifications of frame formats that are used by the MAC.

6.1 Bit order and representation

Figures in clause 6 may represent the information contained in MAC frames. Figures may depict whole MAC frames, elements or fields. Elements are groups of fields for common usage. Elements aid the readability of the standard. MAC frames are described by the fields and elements it contains.

6.1.1 Bit order

The relationship between processing (that means transmitting or interpreting) of MAC frames and their representation in this standard is as follows: bits, fields and elements are processed in their order of representation in figures from left to right. This relationship is depicted in Figure 6-1.

If a field contains a numeric value, represented by a combination of multiple bits, bits are processed in MSBit first order. Hence, the bit with the highest value is processed first. If the numeric value is specified in binary representation within this standard, MSBit representation is used.

If a field's numeric value exceeds the length of an octet, it is stored within the field in big endian representation. Hence, the octet containing the MSBit of the numeric value is processed first.

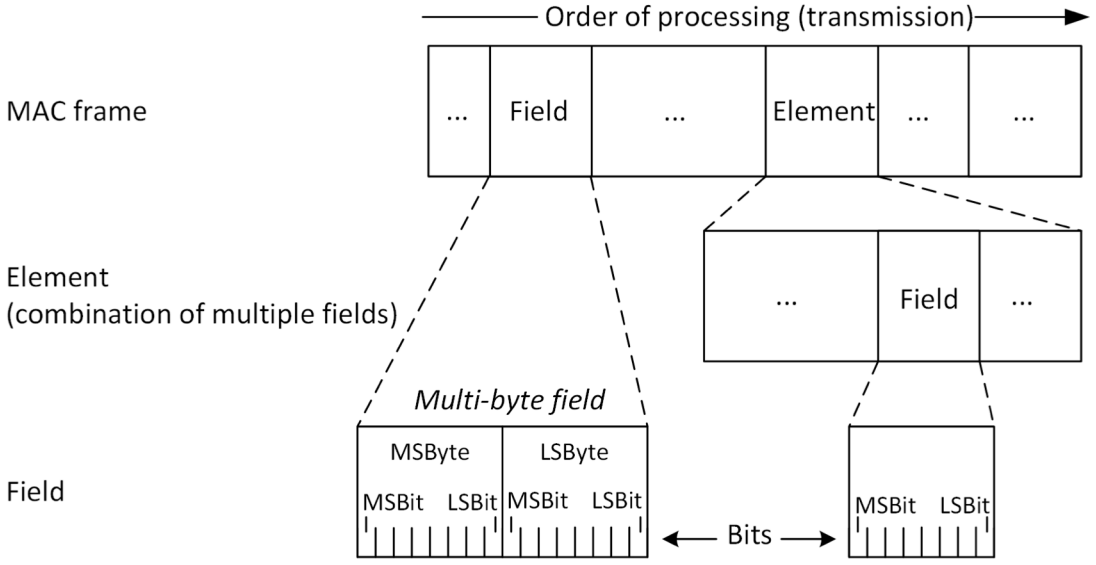

Figure 6-1: Fields and elements in the MAC frame

Fields that are "reserved" do not carry meaningful information in the current version of the standard. This may be changed in later versions. Fields that are reserved shall be set to all zeros for transmission and shall be ignored on reception. The values of reserved fields shall not influence the behavior of devices.

6.1.2 Representation

MAC frames or elements in this standard are represented as figures in table format. The top line specifies the width of fields or elements. The second (middle) line provides a description of the field or a reference to an

element specified elsewhere in the standard. The third (bottom) line is optional and may provide an alternate description of the fields or elements corresponding to its columns. The scheme is represented in Figure 6-2.

width	**width**
field description	field description
optional alternate description	

Figure 6-2: MAC frame or element representation example

Widths of fields are specified in both numbers of bits or numbers of octets if the total number of bits is representable by an integer number of octets.

Bit 0-3	**4 Bits**	**2 Octets**	**Variable**	**5 Octets**
Field 1	Field 2	Field 3	Field 4	Field 5

Figure 6-3: Width specification example

In consecutive fields that include no variable widths from the start of the parent frame or element, fields can be described by the first and last bit in the field. The corresponding notion reads the word “Bit” and successively the specification of the first and last denoted bit. This is demonstrated for Field 1 in Figure 6-3.

If an element or set of consecutive fields has variable width, its width is specified by the word “variable”. If between the field and the start of the MAC frame lies a variable width field, the absolute bit specification cannot be used.

NOTE – To allow correct processing of MAC frames, the width of variable width elements must be deductible from other fields.

Widths can be given by their number of bits or octets. The corresponding notion includes the number of bits or octets followed by the word “Bits” or “Octets” as shown in Field 2, 3 and 5 in Figure 6-3.

6.2 General MAC frame format

This standard defines a single general MAC frame format, occurring in multiple variants depending on what information is carried in the frame. For discrimination and subsequent interpretation, each MAC frame starts with a *Frame Control* element, indicating a *Type* and *Subtype* of frame. Currently, three basic frame *Types* for the transmission of data, management and control information are supported.

Data, management and control frames have distinct MAC headers, detailed in clauses 6.3, 6.4 and 6.5 respectively.

The payload in turn differs for different *Subtypes* of data-, management- or control frames. It contains the actual information to be conveyed via the MAC frame. For data frames, this may be one or multiple MSDUs received via the MCPS-SAP for transmission. For management frames, the payload constitutes of management information. Analogously, the payload of control frame comprises control information, aiding the MAC in its operation.

Each MAC frame shall end with the FCS field, containing a 32-bit CRC sum over all preceding information bits of the MAC frame.

The general MAC frame structure is depicted in Figure 6-4.

Octets: 2	**0/2**	**2/6**	**2/6**	**0/2/6**	**0/2**	**variable**	**variable**	**4**
Frame Control	ACK Information	Receiver Address	Transmitter Address	Auxiliary Address	Sequence Control	Auxiliary Security Header	**Payload**	**FCS**
MAC frame header (MHR)								

Figure 6-4: General MAC frame (MPDU) format

Individual fields are explained in the subsequent clauses.

6.2.1.1 Frame Control Field

The *Frame Control* field comprises multiple bits that serve the determination of the further MAC frame structure or indicate properties of the payload. The *Frame Control* field is present at the beginning of each MAC frame.

Bits: 0-1	**2-3**	**4-7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Frame Version	Type	Subtype	To Backhaul	From Backhaul	Security Enabled	ACK Request	Non-beacon-enabled	Short Addressing	Last Fragment	Reserved

Figure 6-5: Frame Control element

Frame Version: The *Frame Version* subfield specifies the version number corresponding to the frame. This subfield shall be set to '00' to indicate a frame compatible with IEEE 802.15.13. All other values shall be reserved for future use.

Type: For management frames, the Type field shall be set to 00. For control frames, the field is 01. For data frames, the field is set to 00. The value 11 is reserved.

Subtype: Indicates the subtype of the frame, i.e. the contents of the payload. Subtypes are listed in the clauses for data management and control frames (6.3, 6.4 and 6.5).

To Backhaul / From Backhaul: These fields are needed for the correct interpretation of the addressing fields of data frames in a topology, where the OWPAN is integrated into a logical LAN. For example, this may be the case in the coordinated topology.

A data frame (MSDU) within a LAN has a 48 bit source and destination address, used for bridging and LAN integration. In contrast, the receiver and transmitter address are the addresses of 802.15.13 devices to receive or transmit a MPDU respectively.

[NOTE: move to extra chapter to describe topology / bridging / data service. Maybe 5.1?]

To Backhaul	From Backhaul	Description	Receiver address	Transmitter address	Auxiliary address
0	0	The frame originates from a device and is destined to another device.	Address of the designated receiver. *Same as MSDU destination address.* *Optionally short*	Address of the transmitting device. *Same as MSDU source address.* *Optionally short*	-
1	0	An MSDU originates from a device and is destined to a peer in the integrated LAN	Coordinator address	Address of the transmitting device. *Same as MSDU source address.*	MSDU destination address.
0	1	An MSDU originates from a peer in the integrated LAN and is destined to a device.	Address of the designated receiver. *Same as MSDU destination address.*	Coordinator address	MSDU source address.
1	1	Reserved	-	-	-

Table 6-1: To Backhaul and From Backhaul field description

For control frames, the *To Backhaul* and *From Backhaul* fields shall be 0.

Security Enabled: The *Security Enabled* field shall be set to 1 if the frame is secured by the MAC sublayer and shall be set to 0 otherwise. The *Auxiliary Security Header* field of the MHR shall be present only if the *Security Enabled* subfield is set to 1.

ACK Request: The *Acknowledgment Request* field specifies whether an acknowledgment is required from the recipient device on receipt of a data or MAC management frame. If this subfield is set to '1', the recipient device shall send an acknowledgment frame. If this subfield is set to '0', the recipient device shall not send an acknowledgment frame.

For control frames, the *ACK Request* field is reserved to be 0.

Non-beacon-enabled: Specifies whether the transmitting device operates in the non-beacon-enabled mode and hence whether the *ACK Information* element is present in the remaining MAC header. If the transmitting device operates in non-beacon-enabled mode, this field shall be set to 1. Otherwise, it shall be set to 0.

Short Addressing: Indicates whether short addresses are used in the address fields of the MAC header. If short addresses are used in the header, this field shall set to 1. Otherwise, it shall be set to 0. Some addresses, such as the OWPAN ID, are always in short representation.

Short addresses shall only be used, if the frame does not carry a MSDU so that the corresponding address field needs to contain the full MAC address for identification of the source or destination of the MSDU. For example, short addresses may be used in control frames that are solely transmitted over the wireless medium.

Last Fragment: In a data frame, this field shall be set to 1 if the payload contains a fragment of an (A-) MSDU, which is not the last fragment. It shall be set to 0 otherwise.

6.2.1.2 ACK Information field

The *ACK Information* field contains acknowledgment information specific to the non-beacon-enabled mode. It is only present in frames originating from devices operating in the non-beacon-enabled channel access mode. This is indicated through the *Non-beacon-enabled* bit in the *Frame Control* field (see 6.2.1.1).

Bits: 0-4	5-13	14-15
Device Compressed Address	Compressed Sequence Number	ACK

Figure 6-6: ACK Information field

Device Compressed Address: Bits 0 to 4 contain the short address of the device which transmitted the packet. The device with such address is to be acknowledged by this ACK information field. In the uplink transmission, these bits identify the device transmitting the current packet as the acknowledgment can be only for packets transmitted by the coordinator.

Compressed Sequence Number: Bits 5 to 13 identify the sequence number of the packet which is being acknowledged.

ACK: Bit 14 is set to '1' when a packet is being acknowledged with the current frame, and set to '0' otherwise. Bit 15 is set to '1' when the last Beacon frame reception is being acknowledged, and set to '0' otherwise.

6.2.1.3 Address Fields (Receiver, Transmitter and Auxiliary Address)

The address fields indicate multiple addresses to the receiver of a MAC frame. These fields may comprise a either a 16 bit short MAC address or 48 bit full MAC address. The address format is indicated by the *Short Addressing* field in the *Frame Control* element as described in 6.2.1.1.

The *Transmitter Address* shall identify the device which transmitted the frame over the wireless medium.

The *Receiver Address* shall identify the designated receiver of the MAC frame.

The *Auxiliary Address* is used to include additional information about the source or destination of the frame.

6.2.1.4 Sequence Control Field

The *Sequence Control* element contains information for fragmentation and reliable transmission of a frame.

Bits: 0-3	4-15
Fragment Number	Sequence Number

Figure 6-7: Sequence Control field

Fragment Number: If the MPDU contains a fragment of an (A-) MSDU, the field contains the respective fragment number.

Sequence Number: This field contains the assigned sequence number of the MPDU.

6.2.1.5 Auxiliary Security Header

This field contains security information and is further defined in clause 8.

6.2.1.6 Payload

The payload of MAC frames consists of information specific to the subtype of each frame.

6.2.1.7 The FCS field

The FCS field is a 32-bit field containing a 32-bit CRC. The FCS is calculated over all the fields of the MAC header and the frame body field. These are referred to as the calculation fields. The FCS is calculated using the following standard generator polynomial of degree 32:

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

The FCS is the ones complement of the sum (modulo 2) of the following:

a) The remainder of $x^k(x^{31} + x^{30} + x^{29} + \cdots + x^2 + x + 1)$ divided (modulo 2) by G(x), where k is the number of bits in the calculation fields,

and

b) The remainder after multiplication of the contents (treated as a polynomial) of the calculation fields by x^{32} and then divided by G(x).

The FCS field is transmitted in order of the coefficient of the highest-order term first.

As a typical implementation, at the transmitter, the initial remainder of the division is pre-set to all ones and is then modified by division of the calculation fields by the generator polynomial G(x). The ones complement of this remainder is transmitted, with the highest-order bit first, as the FCS field. At the receiver, the initial remainder is pre-set to all ones and the serial incoming bits of the calculation fields and FCS, when divided by G(x), results in the absence of transmission errors, in a unique nonzero remainder value. The unique remainder value is the polynomial:

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

6.3 Data frames

Data frames serve the transmission of MSDUs that are received via the MCPS-SAP to a peer device. The MPDU structure of data frames is depicted in Figure 6-8.

Octets: 2	**0/2**	**2/6**	**2/6**	**0/6**	**0/2**	**variable**	**variable**	**4**
Frame Control	ACK Information	Receiver Address	Transmitter Address	Auxiliary Address	Sequence Control	Auxiliary Security Header	MSDU / A-MSDU	**FCS**
MAC frame header (MHR)							**Payload**	

Figure 6-8: Data frame structure

Data frame include the header fields as depicted in Figure 6-8.

Data frames may be protected by a sequence number and retransmitted upon loss. The *Sequence Control* element is present in data frames that have the *ACK Request* bit set to 1 in the *Frame Control* element.

If the payload is secure, the data frame must have the *Auxiliary Security Header* present. The presence of the *Auxiliary Security Header* is indicated by the *Security Enabled* bit in the *Frame Control* element (see 6.2.1.1). The format of the *Auxiliary Security Header* is variable and depends on the details of the applied security type, as defined in clause 8.

The payload content of data frames is described by the subtype field of the *Frame Control* element. Currently, the payload may contain different formats, as listed in Table 6-2.

Data frame Subtype	Payload
0000	Null (zero length)
0001	MSDU
0010	A-MSDU
1001-1111	-

Table 6-2: Data frame subtypes

For data frames with the *Subtype* 0000, the payload has a length of 0. These Null-Frames may be used to transmit MPDUs or corresponding PPDUs for various reasons.

The *Subtype* 0001 indicates that the payload of the data frame contains a single MSDU.

The *Subtype* 0010 indicates an A-MSDU in the payload of the data frame. The format of A-MSDUs is detailed in 5.6.

The Subtypes 1001-1111 are reserved. The payload for these subtypes is undefined. The frames with reserved subtypes should be ignored upon reception.

The *FCS* field of data frames contains the frame check sequence as defined in 6.2.1.5.

6.4 Management frames

Management frames convey management information, aiding the communication of two MLMEs in different protocol exchange procedures. The MPDU format of management frames is depicted in Figure 6-9.

<table>
<tr><th>Octets: 2</th><th>0/2</th><th>2/6</th><th>2/6</th><th>0/6</th><th>0/2</th><th>variable</th><th>variable</th><th>4</th></tr>
<tr><td>Frame Control</td><td>ACK Information</td><td>Receiver Address</td><td>Transmitter Address</td><td>Auxiliary address</td><td>Sequence Control</td><td>Auxiliary Security Header</td><td>Management Information</td><td rowspan="2">FCS</td></tr>
<tr><td colspan="7">MAC frame header (MHR)</td><td>Payload</td></tr>
</table>

Figure 6-9: Management frame structure

Management frames may carry an *ACK Information* element in their header. The presence of the *ACK Information* element is indicated by the *ACK Info* field in the *Frame Control* element of the frame. The *ACK Information* element may be used by the transmitter to acknowledge the successful reception of an earlier frame.

Each management frame has a *Receiver Address* and *Transmitter Address* field. These fields may comprise a either a 16 bit short MAC address or 48 bit full MAC address. The address format is indicated by the *Short Addressing* field in the *Frame Control* element as described in 6.2.1.1.

Management frames may be protected by a sequence number and retransmitted upon loss. The *Sequence Control* element is present in management frames that have the *ACK Request* bit set to 1 in the *Frame Control* element.

The payload of management frames contains one or multiple elements defined in clause 6.6. The subtype describes which elements reside in the payload field. For simple management frames, the payload consists solely of a single element. The element to be present for which subtype can be derived from Table 6-3.

Management frame Subtype	Element in Payload	Payload Element ID	Payload Element Clause
0000	Association Request		
0001	Association Response		
0010	Disassociation Notification		
0011	Authentication Request		
0100	Authentication Response		
0101	Poll Frame	-	-
0110	Poll Request Frame	-	-
0111	Poll Response Frame	-	-
1000	Variable Element Container		
1001-1111	-		

Table 6-3: Management frame subtypes

By having the *Variable Element Container* element present in the payload, a single management frame is able to include more than a single element.

The *FCS* field of management frames contains the frame check sequence as defined in 6.2.1.5.

6.5 Control frames

Control frames aid the lower MAC and PHY at their operation. The MPDU structure of control frames is depicted in Figure 6-10.

Octets: 2	0/3	2/6	2/6	variable	variable	4

Control frames contains a subset of general MAC frame format header fields as defined in 6.2. Especially, the *Auxiliary Address* field is not required, as control frames are exchanged solely between the coordinator and devices.

Information conveyed via control frames is of ephemeral nature and quickly outdated. Hence, control frames are not retransmitted upon loss. Rather, a new control frame containing the most recent control information may be transmitted. Due to their nature, control frames do not carry sequence numbers.

Control frames may optionally be secured. In that case, the *Auxiliary Security Header* shall be included in the frame header and the corresponding bit set in the *Frame Control* field.

Control Frame Subtype	Element in Payload	Payload Element ID	Payload Element Clause
0000	Waveform Control		
0001	Advanced Modulation Control		
0010	Acknowledgment		
0011	Modulation Request		
0100			
0101			
0110			
0111	Superframe Descriptor		
1000	Variable Element Container		
1001-1111	-		

Table 6-4: Control frame subtypes

e

6.6 Elements

Elements are collections of related fields that serve a common MAC functionality as defined in clause 6.1. Elements may be used to define the content of certain frames and aid the readability of the document through defining semantics of certain frames in one place without redundancy.

If an element contains a variable number of fields or other elements, the total length of the element must be deductible from its field contents in order to allow parsing.

Each element has an ID assigned, which identifies it in some frames. Table 6-5 lists the elements defined within this standard and their corresponding ID and definition clause.

Element Name	ID	Clause
reserved	0	
Association Request	1	
Association Response	2	
Disassociation Notification	3	
Authentication Request	4	
Authentication Response	5	
	6	
Superframe Descriptor	7	
OWPAN Descriptor	8	
Variable Element Container	9	

Element Name	ID	Clause
Clock Rate Change		
Scan Over Backhaul Request		
Scan Over Backhaul Confirm		

Table 6-5: Elements and their IDs

6.6.1.1 Association Request Element

The *Association Response* element is transmitted by a device to the coordinator of an OWPAN in order to request association.

2/6 octets?	variable	1 + a/b/c octets
OWPAN ID	Capability List	Supported Rates

Figure 6-11: Association Request element

OWPAN ID: The OWPAN ID of the OWPAN the device requests to associate with.

Capability List: *Capability List* element, describing the supported capabilities of the device requesting association.

Supported Rates: *Supported Rates* element.

6.6.1.2 Association Response Element

The *Association Response* element is transmitted by a coordinator to a device requesting association.

1 octet	2 octets	2 octets	variable	1 + a/b/c octets
Status Code	OWPAN ID	Short Address	Capability List	Supported Rates

Figure 6-12: Association Response element

Status Code: The status code indicates the result of the preceding association request.

Value	Description
0	reserved
1	Denied
2	Success
3	Require further authentication
4-255	reserved

Table 6-6: Status codes of the Association Response element

OWPAN ID: The OWPAN ID of the OWPAN the device requested to associate with.

Short Address: The short address assigned to the device if the association was not denied. If the association was denied, the field shall be ignored.

[NOTE – A field for authentication information may be added later while adding security to the standard]

Capability List: This field contains a *Capability List* element, describing the set of capabilities to be used for further channel access if the association was not denied. If the association was denied, the field shall be ignored.

Supported Rates: The rates supported by the coordinator.

6.6.1.3 Disassociation Notification Element

The *Disassociation Notification* element conveys information about the disassociation of a device from an OWPAN.

Octets: 1
Reason Code

Figure 6-13: Disassociation Notification element

Reason Code: The reason code indicates a reason for disassociation.

Value	Description
0	reserved
1	Other
2	Handover
3	Lack of resources
4	Poor channel
5	Unreliable connection
6-255	reserved

Table 6-7: Reason codes of the Disassociation Notification element

6.6.1.4 Authentication Request Element

The *Authentication Request* element is transmitted by a device to the coordinator of an OWPAN in order to request authentication if required to successfully associate with that OWPAN.

Octets: 1	2	1	0/128
Authentication Algorithm	Authentication Transaction Token	Status Code	Challenge

Figure 6-14: Authentication Request element

Authentication Algorithm: The authentication algorithm used for authentication.

Authentication Transaction Token: The authentication algorithm used for authentication.

Status Code: The authentication algorithm used for authentication.

Challenge: The authentication algorithm used for authentication.

6.6.1.5 Authentication Response Element

6.6.1.6 Superframe Descriptor Element

The *Superframe Descriptor* element conveys information about the beginning superframe.

Octets: 2	2	1	1	1	variable
Superframe Number	Total Superframe Slots	Superframe Slot Duration	CAP Slot Width	CAP Slots	Variable Element Container

Figure 6-15: Superframe Descriptor element

Superframe Number: The number of the subsequent superframe. Wrapping integer as described in 5.2.2.

Total Superframe Slots: The number of superframe slots in the subsequent superframe. Devices associated with the OWPAN or attempting association shall set their *macNumSuperframeSlots* PIB attribute to the value contained in this field.

Superframe Slot Duration: The duration of a single superframe slot.

CAP Slot Width: The number of superframe slots per CAP slot.

CAP Slots: The number of CAP slots included in the subsequent CAP.

Variable Element Container: A *Variable Element Container* element, containing one or more elements.

6.6.1.7 Capability List Element

The *Capability List* element is used to transfer information about capabilities as described in clause 7.4 between two devices.

1 octet	0-255 octets
Bitmap Width	Capability Bitmap

Figure 6-16: Capability List element

Bitmap Width: Specifies the subsequent *Capability Bitmap* field in octets. The *Capability Bitmap field* can thus include at most the capability with the ID *Bitmap Width* * 8 – 1. The *Bitmap Width* 0 may be used to indicate an empty list of capabilities where needed.

Capability Bitmap: A bitmap indicating a subset of capabilities as given in Table 7-28. In the bitmap, the leftmost bit, i.e. the bit to be processed first, corresponds to the ID 0. The rightmost bit, i.e. the bit to be processed last by the definition given in 6.1.1, corresponds to the ID *Bitmap Width * 8* – 1. If a capability is included in the subset, the bit corresponding to the ID of the capability shall be set to 1. Otherwise, the bit shall be set to 0.

For example, a bitmap with a width of 1 octet (8 bits), indicating the presence of the capabilities with the IDs 1, 4 and 7 would be **01001001** (processing from left to right).

6.6.1.8 GTS Descriptor List Element

The *GTS Descriptor List* element holds multiple *GTS Descriptor* elements for a device in the beacon-enabled channel access mode.

Bits: 0-7	8	9	10	11-15	variable	variable		
GTS Descriptor Count	Validity Present	Device Address Present	Directions Present	Reserved	GTS Directions	GTS Descriptor 1	…	GTS Descriptor N

Figure 6-17: GTS Descriptor List element

GTS Descriptor Count: This field includes the number of GTS descriptors that are subsequently included.

Validity Present: If set to 1, child GTS descriptors have the validity field present. Otherwise, it is set to 0.

Device Address Present: If set to 1, each child *GTS Descriptor* shall have the *Device Short Address* field present.

GTS Directions: If the *Directions Present* field is set to 1, this field indicates the directions of the subsequently included GTSs.

GTS Descriptor 1 … N: These fields contain one or multiple *GTS Descriptor* elements

6.6.1.8.1 GTS Descriptor Element

This element describes a single GTS in the CFP of the beacon-enable channel access mode.

2 octets	2 octets
GTS Start Slot	GTS Length

Figure 6-18: GTS Descriptor element

GTS Start Slot: This field specifies the first slot of the allocated GTS.

GTS Length: This field specifies the duration of the GTS in superframe slots.

6.6.1.9 Multi-OFE Feedback Element

Needs sequence number or beacon number?

The *Multi-OFE Feedback* element is used to transfer multi-OFE channel feedback from a device to the coordinator of the OWPAN.

Bits: 0-3	4-7	Variable		
Number of OFEs (N)	TAP format	OFE feedback descriptor element 1	…	OFE feedback descriptor element N

Figure 6-19: Multi-OFE Feedback element

Number of OFEs: The number of distinct recognized OFEs. This determines the number of totally included *OFE Feedback Descriptor* elements.

Tap format: This field describes the format for taps included in the child *Tap Descriptor* elements.

Value	Strength	Delay
0000	Bits: 8 0 value: 0 dBm step: 0.15 dBm	Bits: 16 0 value: 0 ps step: 30 ps
0001	Bits: 4 0 value: 0 dBm step: 2 dBm	Bits: 8 0 value: 0 ns step: 4 ns
0000	Bits: 8 0 value: 0 dBm step: 0.15 dBm	Bits: 8 0 value: 0 ns step: 4 ns
0000	Bits: 6 0 value: 0 dBm step: 0.625 dBm	Bits: 10 0 value: 0 ns step: 1 ns
0100-1111	Reserved	Reserved

Table 6-8: Tap formats in the Multi-OFE Feedback element

OFE Feedback Descriptor Element 1 … N: *OFE Feedback Descriptor* elements containing CSI for the channels between the device and each transmitting OFE. The number of elements N is equal to the *Number of OFEs* field

6.6.1.9.1 OFE Feedback Descriptor Element

The *OFE Feedback Descriptor* element contains channel state information about a received signal from a given transmitter, i.e. a single multi-OFE pilot division.

Bits: 0-2	4-7	8-15	variable		
Pilot Symbol Number	Division	Number of Taps (N)	Tap Descriptor 1	…	Tap Descriptor N

Figure 6-20: OFE Feedback Descriptor element

Pilot Symbol Number: Specifies the position (temporal) of the pilot symbol within the PPDU, from which the included feedback was measured, within the respective received PPDU. Values 1-7, 0 reserved.

Division: Specifies the pilot division. This is for example the Hadamard coding or the subcarrier spacing and shift as indicated in the PPDU header.

Number of Taps: Specifies the number of subsequent *Tap Descriptor* elements, also denoted by N.

Tap Descriptor 1 … N: the *Tap Descriptor* elements for the respective taps. The first *Tap Descriptor* element shall correspond to the first received tap from that OFE.

6.6.1.9.2 Tap Descriptor Element

The *Tap Descriptor* element includes the information about a single tap.

variable	variable
Strength	Delay

Figure 6-21: Tap Descriptor element

Strength: Optical signal strength of the given tap. The format is specified in the *Tap Format* field of the parent *Multi-OFE Feedback* element.

Delay: Integer delay in the format specified in the *Tap Format* field of the parent *Multi-OFE Feedback* element. The delay is relative to the first received tap of all OFEs. The delay for the first tap shall be 0.

6.6.1.10 MSDU Aggregation Element

The *MSDU Aggregation* element serves the aggregation of multiple MSDUs in one A-MSDU data frame.

6 octets	6 octets	2 octets	variable	0-3 octets
Destination MAC Address	Source MAC Address	MSDU Length	MSDU	Variable Padding

Figure 6-22: MSDU Aggregation element

Destination MAC Address: The destination address of the MSDU.

Source MAC Address: The source address of the MSDU.

MSDU Length: The field contains the length of the subsequent MSDU in octets.

MSDU: This field contains the MSDU to be aggregated.

Variable Padding: This field contains 0, 1, 2 or 3 octets in order to make the total length of the element a multiple of 4 octets. The padded octets shall have the value 0. The padded octets shall not be interpreted as information. The actual value of the padded octets shall not have influence on the protocol procedures. Receivers shall discard the padding exceeding the actual length of the MSDU.

6.6.1.11 ACK Information Element

The *ACK Information* element is used by the receiver of an MPDU to signal successful reception of that MPDU to its transmitter. The receiver of an *ACK Information* element shall infer the identity of the acknowledging device based on the transmitter address of the frame containing the *ACK Information* element.

bits 0-11	bits 12-15
Sequence Number	Reserved

Figure 6-23: ACK Information element

Sequence Number: The sequence number of the MPDU to be acknowledged.

6.6.1.12 Block ACK Request Element

The *Block ACK Request* element is used by the transmitter of MPDU(s) to request an acknowledgment for the successful reception from the receiver.

bits 0-11	12-15 bits
First Sequence Number	reserved

Figure 6-24: Block Ack Request element

First Sequence Number: The sequence number of the first MPDU to be acknowledged.

6.6.1.13 Block ACK Element

The *Block ACK* element is used by the receiver of multiple MPDUs to signal their successful reception to the transmitter in a collected manner. This element shall only be transmitted in frames having a source address, which is neither a multicast nor a broadcast address.

5 Bits	11 Bits	1-31 octets
Bitmap Width	First Sequence Number	ACK Bitmap

Figure 6-25: Block Ack element

Bitmap Width: The maximum number of included acknowledgments. This field determines the width of the *ACK Bitmap* field in integer octets. The actual width of the bitmap is the integer contained in the *Bitmap Width* field plus one.

First Sequence Number: The sequence number corresponding to the first bit in the subsequent *ACK Bitmap* field.

ACK Bitmap: The actual acknowledgment information. The bitmap is *Bitmap Width* + 1 octets wide. The transmitter of a *Block ACK* element shall select the width of the bitmap such that it can hold the desired number of acknowledgments.
In the bitmap, the rightmost bit, i.e. the bit to be processed last by the definition given in 6.1.1, corresponds to the first sequence number, as given in the *First Sequence Number* field. The leftmost bit, i.e. the bit to be processed last, corresponds to the sequence number

$$First\ Sequence\ Number\ +\ (Bitmap\ Width\ +\ 1)\ *\ 8\ -\ 1.$$

For every successfully received MPDU, the transmitter of a *Block ACK* element shall set the bit corresponding to its sequence number to 1. All other bits shall be set to 0.

An *ACK Bitmap* field with for the *Bitmap Width* of 00001 (1) and the first sequence number of 321 would look as follows if the sequence numbers 320, 321, 322, 324, 325, 326, 327, 328, 329, 330, 332, 333 were successfully received:

processed first (left)																processed last(right)
bits in the bitmap:	1	1	1	0	1	1	1	1	1	1	1	0	1	1	0	0
sequence number:	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335

6.6.1.14 MCS Request Element

The *MCS Request* element is used by the prospective receiver of a transmission to request the usage of a certain MCS by the prospective transmitter. The *MCS Request* element may be used with the PM-PHY.

Bits: 0-7
Requested MCS ID

Figure 6-26: MCS Request element

Requested MCS ID: The ID of the requested MCS. The MCS ID shall be a valid MCS for the PM-PHY as indicated in TABLE XXX.

6.6.1.15 BAT Request Element

The *BAT Request* element may be used by a receiving device using the HB-PHY to request a transmitter to use a certain bitloading and error coding scheme.

Bits: 0-23	24-28	29-31	32-34	35-39	Variable		
Valid BAT Bitmap	Updated BAT	FEC Block Size	FEC Code Rate	Reserved	BAT Group 1	…	BAT Group N

Figure 6-27: BAT Request element

Valid BAT Bitmap: Specifies the BATs requested to be valid.

Updated BAT: Specifies the ID of the BAT to be updated.

FEC Block Size:

[Maybe the block size should be selected by the transmitter to avoid excessive padding for small frames if a block size is prescribed. It would then be indicated in the PHY frame header.]

Value	Block Size (Bits)
001	168
010	960
011	4320
100-111	Reserved

Table 6-9: FEC code rates for the HB-PHY

FEC Code Rate: Specifies the requested FEC coding rate. Valid values and corresponding code rates are listen in Table 6-10.

Value	Code rate
001	1/2
010	2/3
011	5/6
100	16/18
101	20/21
110-111	Reserved

Table 6-10: FEC code rates for the HB-PHY

BAT Group 1 … N: *BAT Group* elements describing the modulation for the nth group of subcarriers. There shall be enough groups to cover all subcarriers. The last group may be wider than the remaining number of subcarriers. The requested modulation for those excess subcarriers shall be ignored.

6.6.1.15.1 BAT Group Element

The *BAT Group* element contains information about a group of adjacent subcarriers, having the same number of bits loaded in a bit-loading capable PHY transmission.

Bits: 0-3	4-7
Grouping	Loaded Bits

Figure 6-28: BAT Group element

Grouping: This field contains the number of subcarriers in this group. Valid values are:

1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096

Loaded Bits: The number of bits loaded on each subcarrier of the group. Valid values are:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

6.6.1.16 Queue State Element

The Queue State element is transmitted by a device in order to inform other devices about the state of its MSDU queues.

5 bits	2 octets
Queue ID	Queued Bytes

Figure 6-29: Queue State element

XXX:

6.6.1.17 HCM Allocation Element

The *HCM Allocation* element is used to allocate one or more HCM rows to a device.

1 octet
HCM Mask

Figure 6-30: HCM Allocation element

HCM Mask: The HCM rows assigned to the device. Each bit corresponds to a HCM row. The MSBit, i.e. the leftmost bit, corresponds to row 0, while the rightmost bit corresponds to row 7.

6.6.1.18 Alien Signal Element

The *Alien Signal* element contains information about a signal that was received but identified as not originating from a device that is a member of the same OWPAN. The *Alien Signal* element shall be transmitted in a unicast frame, having unique transmitter and receiver addresses.

1 octet	1 bit	1 bit	1 bit	5 bits	0/2 octets	0/2 octets
Signal Power	Decodable	Same MAC Mode	OWPAN ID Clash	reserved	Foreign OWPAN ID	Foreign Device Address

Figure 6-31: Alien Signal element

Signal Power: The optical power in dBm of the alien signal.

Decodable: This bit shall only be set to one, if the alien signal is decodable by the PHY and MAC. This should be the case if the signal originates from another IEEE 802.15.13 device.

Same MAC Mode: This bit shall only be set to 1 if the received frame originates from an IEEE 802.15.13 OWPAN that uses the same channel access mode as defined in 5.2 and 5.3 respectively.

OWPAN ID Clash: This bit shall be set to one if the received frame originates from an OWPAN that has the same OWPAN ID.

Foreign OWPAN ID: This field shall only be present, if the *OWPAN ID Clash* field was set to 0. This field contains the OWPAN ID of the foreign network, from which the alien frame was received.

Foreign Device Address: This field shall only be present, if the *Decodable* field was set to 1. This field contains the Device Short Address of the foreign transmitting device. If the address is unknown, the field shall be set to the broadcast short address.

6.6.1.19 Supported MCS Element

The *Supported MCS* element may be used to convey a set of supported rates of a device. The possible included values depend on the used PHY.

1 octet	a/b/c
PHY ID	PHY Rates Element

Figure 6-32: Supported MCS element

PHY ID: The ID of the PHY that the following *PHY Rates* element is specific to.

PHY Rates Element: A PHY-specific *PHY Rates* element as defined in the respective PHY clauses. The *PHY Rates Element* format should be specified in the respective clause of each PHY.

6.6.1.20 OFE Selection Element

The *OFE Selection* element contains sequential field that serve the measurement of channels between multiple OFEs and a device.

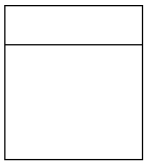

Figure 6-33: OFE Selection element

XXX:

6.6.1.21 Waveform Control Element

The waveform control frame relates to the adaptive OFDM technique. Multiple waveforms such as eU-OFDM and RPO-OFDM may exist in the network. The control frame could be used when adaptive adjustment of the waveform is required.

8 octets	6	8	1
Timestamp	OWPAN ID	Switching Time	New Waveform

Figure 6-34: Waveform Control element

Timestamp: The Timestamp field allows synchronization between the devices in an OWPAN. The master timekeeper for an OWPAN periodically transmits the number of microseconds it has been active. When the counter reaches its maximum value, it wraps around.

OWPAN ID: OWPAN ID field gives the ID for the OWPAN.

Switching Time: [TBD]

New Waveform: The 8 bits are to indicate the waveform to switch to among the supported waveforms by IEEE 802.15.13.

6.6.1.22 Advanced Modulation Control Element

Advanced modulation control frame indicates the advanced modulation capabilities of a communication node.

1 bit	4	1	4	1	4
Adaptive Loading	eU	RPO	Relaying	MIMO	MIMO Channel Number

Figure 6-35: Advanced Modulation Control element

Adaptive Loading: A single bit indicates whether the communication node transmitting the advanced modulation control frame supports adaptive bit and energy loading:

"1" indicates: Adaptive bit and energy loading is supported.
"0" indicates: Adaptive bit and energy loading is not supported.

eU: The 4 bits indicate if the node supports eU-OFDM. The bit value at a given position out of the four positions indicates whether eU-OFDM implementation with the same number of streams as the bit position is supported. Positions are counted from left to right. For example:

"1000" indicates eU-OFDM with one stream only is supported.
"1100" indicates eU-OFDM with one and two streams only is supported.
"1010" indicates eU-OFDM with one and three streams only is supported.
"1111" indicates eU-OFDM with all possible streams is supported.
"0000" indicates eU-OFDM is not supported.

RPO: A single bit indicates whether the communication node transmitting the advanced modulation control frame supports RPO-OFDM:
"1" indicates: RPO-OFDM is supported.
"0" indicates: RPO-OFDM is not supported.

Relaying: The 4 bits indicate the types of relaying operations the communication node transmitting the advanced modulation control frame supports.

The first bit indicates whether relaying in FD is supported:
"1" indicates: Relaying in FD is supported.
"0" indicates: Relaying in FD is not supported.

The second bit indicates whether relaying in HD is supported:
"1" indicates: Relaying in HD is supported.
"0" indicates: Relaying in HD is not supported.

The third bit indicates whether AF relaying is supported:
"1" indicates: AF relaying is supported.
"0" indicates: AF relaying is not supported.

The fourth bit indicates whether DF relaying is supported:
"1" indicates: DF relaying is supported.
"0" indicates: DF relaying is not supported.

MIMO: A single bit indicates whether the communication node transmitting the advanced modulation control frame supports MIMO communication:

“1” indicates: MIMO is supported.
“0” indicates: MIMO is not supported.

MIMO Channel Number: The 4 bits indicate the maximum number of MIMO communication channels which the communication node transmitting the advanced modulation control frame supports.

A value of '0000' corresponds to 1 channel, and a value of '1111' corresponds to 16 channels.

6.6.1.23 Random Access Element

The *Random Access* element contains information used to trigger the random access procedure in the non-beacon-enabled channel access mode.

Furthermore, Random Access frames announce the existence of a non-beacon-enabled network. They are transmitted at regular intervals (i.e., each random access interval) by coordinators to allow devices to find and identify a network and possibly join it. Random access frames are supposed to be transmitted exactly as the random access interval ends, at the so-called target Random Access transmission time (TBTT). In an infrastructure network, the coordinator is responsible for transmitting Random Access frames with information such as timestamp, OWPAN ID, and other parameters regarding the coordinator to devices that are within range.

8 octets	2	2	6	variable	variable	variable
Timestamp	Random Access Interval	Capability Information	OWPAN ID	Supported Rates	Country	Extended Supported Rates

Figure 6-36: Random Access element

Timestamp: The Timestamp field allows synchronization between the devices in an OWPAN. When coordinators prepare to transmit a Beacon frame, the coordinator timer is copied into the Beacon’s timestamp field. Devices associated with a coordinator accept the timing value in any received Beacons, but they may add a small offset to the received timing value to account for local processing by the antenna and transceiver.

Random Access Interval: Each OWPAN can transmit *Random Access* frames at its own specific interval.

Capability Information: The 16-bit Capability Information field is used to advertise the network’s capabilities. In this field, each bit is used as a flag to advertise a particular function of the network. Devices use the capability advertisement to determine whether they can support all the features in the OWPAN. Devices that do not implement all the features in the capability advertisement are not allowed to join.

OWPAN ID: OWPAN ID field gives the ID for the OWPAN.

Supported Rates: Several data rates have been standardized for each PHY in IEEE 802.15.13. When mobile devices attempt to join the network, they check the data rates used in the network. Some rates are mandatory and must be supported by the mobile device, while others are optional.

Country: The initial specifications were designed around the existing regulatory constraints in place in the major industrialized countries. Rather than continue to revise the specification each time a new country was added, a new specification was added that provides a way for networks to describe regulatory constraints to new stations. Maximum transmission power is specified using the country element in beacon frames. The information is available to any station wishing to associate to a network. The Country element specifies the regulatory maximum

power, and the Power Constraint element can be used to specify a lower maximum transmission power specific to the network.

Extended Supported Rates: Extended Supported Rates element was standardized to handle more than eight data rates.

6.6.1.24 Probe Request Element

The probe request allows a device to send a request with information to a target coordinator in order to scan an area for existing IEEE 802.15.13 networks. A Probe Request frame contains two fields: the OWPAN ID and the rates supported by the mobile device. Coordinators that receive Probe Requests use the information to determine whether the mobile device can join the network. To make a happy union, the mobile device must support all the data rates required by the network and must want to join the network identified by the OWPAN ID. This may be set to the OWPAN ID of a specific network or set to join any compatible network. Drivers that allow cards to join any network use the broadcast OWPAN ID in Probe Requests.

All devices shall be capable of transmitting this command, although a device is not required to be capable of receiving it.

The probe request frame shall be formatted as illustrated in

6	**variable**	
OWPAN ID	Supported Rates	Extended Supported Rates

Figure 6-37: The probe request element

OWPAN ID: OWPAN ID field gives the ID for the requested OWPAN, hence the corresponding coordinator processes the request.

Supported rates: Several data rates have been standardized for each PHY in IEEE 802.15.13. When mobile devices attempt to join the network, they check the data rates used in the network. Some rates are mandatory and must be supported by the mobile device, while others are optional.

Extended supported rates: Extended Supported Rates element was standardized to handle more than eight data rates.

6.6.1.25 Probe Response Element

If a Probe Request encounters a network with compatible parameters, the network sends a Probe Response frame. The coordinator that sent the last Beacon is responsible for responding to incoming probes. After a coordinator transmits a Beacon, it assumes responsibility for sending Probe Response frames for the next Beacon interval.

This response shall only be sent by the coordinator or a coordinator to a device that is currently trying to associate.

All devices shall be capable of receiving this frame, although a device is not required to be capable of transmitting it.

The probe response frame shall be formatted as illustrated in table xxx.

Timestamp	Beacon Interval	Capabilities	OWPAN ID	Supported Rates	Extended Supported Rates

Figure 6-38: The probe response element

Timestamp: The Timestamp field allows synchronization between the devices in an OWPAN. The master timekeeper for an OWPAN periodically transmits the number of microseconds it has been active. When the counter reaches its maximum value, it wraps around.

Beacon interval: Each OWPAN can transmit Beacon frames at its own specific interval.

Capability Information: The 16-bit Capability Information field is used to advertise the network's capabilities. In this field, each bit is used as a flag to advertise a particular function of the network. Devices use the capability advertisement to determine whether they can support all the features in the OWPAN. Devices that do not implement all the features in the capability advertisement are not allowed to join.

OWPAN ID: OWPAN ID field gives the ID for the OWPAN.

Supported rates: Several data rates have been standardized for each PHY in IEEE 802.15.13. When mobile devices attempt to join the network, they check the data rates used in the network. Some rates are mandatory and must be supported by the mobile device, while others are optional.

Extended supported rates: Extended Supported Rates element was standardized to handle more than eight data rates.

6.6.1.26 Attribute Change Request Element

The *Attribute Change Request* element may be used by the coordinator of an OWPAN to change the PIB attribute value of an associated device.

1-32 octets	variable
Attribute ID	New Value

Figure 6-39: Attribute Change Request element

Attribute ID: This field indicates the attribute to be updated. The ID for a given attribute can be found in Table 7-26.

New Value: The new value to assign to the attribute. The field format is to be deducted from the Table 7-26.

6.6.1.27 Attribute Change Response Element

The *Attribute Change Response* element is transmitted from a device to the coordinator as a response to the *Attribute Change Request* element to indicate whether the attribute change was successful.

2 octets	variable	1 octet
Attribute ID	New Value	Status

Figure 6-40: Attribute Change Response element

Attribute ID: This field indicates the attribute to be updated. The ID for a given attribute can be found in Table 7-26.

New Value: The new value assigned to the attribute. The field format is to be deducted from the Table 7-26.

Status: The result of the former attribute change request. Possible values are described in Table 6-11.

Value	Description
0	Success
1	Invalid attribute name
2	Invalid new value
3	Read-only
4	Other error
5-255	Reserved

Table 6-11: Status codes for the attribute change request result.

6.6.1.28 Variable Element Container Element

The *Variable Element Container* element comprises one or multiple other elements. For each element, a type, an optional length and the actual element are included.

bit 0	bits 1-7	2 octets	0/2 octets	variable	variable				2 octets
Explicit Length Present	Reserved	Type 1	Length 1	Contained Element 1	…	Type N	Length N	Contained Element N	0x0000

Figure 6-41: Variable Element Container element

Explicit Size Prefix: If set to 1, the *Length* field shall be present for every included element after the *Type* field. Otherwise, the field shall be set to 0. The size prefix may be used to increase parsing speed.

Type 1 … N: The type of the subsequent element. This field has 2 octets width. The value shall be a valid ID as taken from Table 6-5. The type field after the last contained element shall have the value 0x0000, indicating that the list has ended.

Length 1 … N: This field is only present for every contained element if the *General Size Prefix* field is set to 1. It contains the length of the subsequent element in octets and is 2 octets wide.

Contained Element 1 … N: The contained element as defined in the respective clause.

0x0000: Termination type

7 MAC services

The IEEE 802.15.13 MAC offers its service to the higher protocol layers and DME through the MCPS-SAP and MLME-SAP respectively. The MCPS-SAP includes primitives that support the integration of IEEE 802.15.13 networks in bridged LANs in accordance with IEEE 802.1AC. The MLME-SAP exposes basic management functions and further advanced functionality to the DME.

A primitive invocation originating from the service user, i.e. higher layer, carries the suffix **.request**. Hence, it requests the start of a service, i.e. action, on the service provider, which is the next lower layer. The immediate response to a **.request** is a **.confirm** primitive, returned by the service provider, i.e. the MAC or MLME.

Externally caused events at the MAC or MLME are indicated to the higher layer through primitives carrying the **.indication** suffix. Such events may originate from unrequested actions, e.g. the reception of a specific management frame, or as an asynchronous response to a finished service invocation through a preceding service request.

A number of PIB attributes defines the behavior of the MAC and reflects the current system state.

Moreover, capabilities indicate subparts of functionality in this standard that are supported by a given device implementation. Those capabilities are used to negotiate functionality that can be used while a device is associated with a given OWPAN.

7.1 MCPS-SAP

The MCPS-SAP supports the transport of MSDUs between the MACs of peer IEEE 802.15.13 devices through the primitives listed in Table 7-1.

MCPS-SAP primitive	Request	Confirm	Indication
MCPS-DATA	7.1.1	7.1.2	7.1.2

Table 7-1: MCPS-SAP primitives

7.1.1 MCPS-DATA.request

[This primitive should support the 802.1AC ISS:

M_UNITDATA.indication (
destination_address,
source_address,
mac_service_data_unit,
priority,
drop_eligible,
frame_check_sequence,
service_access_point_identifier,
connection_identifier
)
]

The MCPS-DATA.request primitive is used by the higher layer to request the transfer of data to another device.

The parameters of the primitive are listed in Table 7-2.

Parameter name	Range (Steps)	Parameter description
DestinationAddress	48 bit MAC addresses	The destination address of the MSDU. MAC-48 format.
SourceAddress	48 bit MAC addresses	The source address of the MSDU. MAC-48 format.
Msdu	octet sequence	The actual MSDU.
Priority	[0, 7] (1)	The priority of the MSDU.
Protected	true, false	Whether the associated MSDU shall be transmitted protected.
MsduHandle	[0, 0xFFFF] (1)	The handle of the MSDU whose transmission is requested.

Table 7-2: Parameters of the MCPS-DATA.request primitive

7.1.2 MCPS-DATA.confirm

The MCPS-DATA.confirm primitive reports the result of a previous MCPS-DATA.request invocation to the higher layer.

Parameter name	Range (Steps)	Parameter description
MsduHandle	[0, 0xFFFF] (1)	The handle of the MSDU whose transmission is being confirmed.
Status	48 bit MAC addresses	The result of the transmission request.

Table 7-3: Parameters of the MCPS-DATA.confirm primitive

7.1.3 MCPS-DATA.indication

MCPS-DATA.indication primitive is issued by the MAC of a device upon reception of a MSDU from a peer device.

The parameters of the primitive are listed in Table 7-4.

Parameter name	Range (Steps)	Parameter description
DestinationAddress	48 bit MAC addresses	The destination address of the MSDU. MAC-48 format.
SourceAddress	48 bit MAC addresses	The source address of the MSDU. MAC-48 format.
Msdu	octet sequence	The actual MSDU.
Priority	[0, 7] (1)	The priority of the MSDU

Table 7-4: Parameters of the MCPS-DATA.indication primitive

7.2 MLME-SAP

The MLME-SAP supports the management and usage of a device's MLME functionality through the DME.

MLME-SAP primitive	Request	Indication	Response	Confirm
MLME-ASSOCIATE	X	X		
MLME-AUTHENTICATE	X	X		
MLME-DISASSOCIATE	X	X		
MLME-GET	X		X	
MLME-SET	X		X	
MLME-SCAN	X		X	
MLME-START	X		X	
MLME-STOP	X		X	

Table 7-5 MLME primitives

7.2.1 MLME-ASSOCIATE

The MLME-ASSOCIATE primitive serves the association process of a device with an OWPAN as described in clause 5.4.5.

7.2.1.1 Request

The MLME-ASSOCIATE.request is issued by the DME to the device MAC to initiate the association process with a given OWPAN. Upon reception of the primitive, the MLME shall start the association procedure as detailed in 5.4.5.

If the MLME of a device receives multiple MLME-ASSOCIATE.request primitives for different target OWPAN IDs, it shall discard all but the first request and wait for its completion or timeout before accepting another request.

The parameters of the primitive are listed in Table 7-6.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	integer	[1, 65534] (1)	The OWPAN ID as indicated in the beacon or RA frames of the target OWPAN.
SecurityType	integer	IDs as defined in Table 8-1	The type of security as specified in clause 8.
AuthenticationDetails	variable	variable	The type-specific security information. Format depends on the SecurityType parameter.
Timeout	milliseconds	[1, 65535] (1)	Maximum time in ms until a corresponding indication primitive is expected. If none is received by the DME, the association is assumed to have failed.

Table 7-6: Parameters of the MLME-ASSOCIATE.request primitive

7.2.1.2 Confirm

The MLME-ASSOCIATE.confirm primitive is issued by the MAC layer of a device to report the result of the previously requested association attempt to the DME:

The parameters of the primitive are listed in Table 7-7.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	integer	[1, 65534] (1)	The OWPAN ID as indicated in the beacon or RA frames of the target OWPAN.
Status	enumeration		

Table 7-7: Parameters of the MLME-ASSOCIATE.confirm primitive

7.2.1.3 Indication

MLME-ASSOCIATE.indication primitive is issued by the MLME to report the result of a preceding MLME-ASSOCIATE.request primitive.

The parameters of the primitive are listed in Table 7-8.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	integer	[1, 65534] (1)	The OWPAN ID for which the association was requested earlier.
Status	enumeration	SUCCESS, FAILURE	The result of the association process.
FailureReason	enumeration	UNAUTHORIZED, IS_COORDINATOR, OTHER	The reason for failure, if the association status is FAILURE.

Table 7-8: Parameters of the MLME-ASSOCIATE.indication primitive

7.2.1.4 Response

The MLME-ASSOCIATE.response primitive is used by a coordinator DME to respond to a MLME-ASSOCIATE.indication after deciding how to proceed with the requested association.

The parameters of the primitive are listed in Table 7-9.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	integer	[1, 65534] (1)	The OWPAN ID as indicated in the beacon or RA frames of the target OWPAN.
DeviceId	integer	[1, 65534] (1)	The short address of the device to be disassociated.
Reason	enumeration	USER_REQUEST, AUTHENTICATION_END	The reason for disassociation

Table 7-9: Parameters of the MLME-ASSOCIATE.response primitive

7.2.2 MLME-AUTHENTICATE

The MLME-AUTHENTICATE primitive allows the DME of an OWPAN coordinator to authenticate a device that previously requested authentication.

7.2.2.1 Request

The MLME-AUTHENTICATE.request primitive is issued by the coordinator DME in order to allow or deny authentication of a requesting device. The primitive is invoked following a preceding MLME-AUTHENTICATE.indication primitive.

The parameters of the primitive are listed in Table 7-10.

Parameter name	Type	Value range (Steps)	Parameter description
Mac48Address	MAC-48 Address	valid addresses	The MAC-48 address of the device requesting to be authenticated.
DeviceId	device short address	[1, 245] (1)	The short address of the device requesting to be authenticated.
Status	enumeration	ACCEPT, DENY	The result of the authentication process.

Table 7-10: Parameters of the MLME-AUTHENTICATE.request primitive

7.2.2.2 Indication

The MLME-AUTHENTICATE.indication primitive is issued to the DME by the coordinator MAC upon reception of an *Authentication Request* element from a device attempting association.

The parameters of the primitive are listed in Table 7-11.

Parameter name	Type	Value range (Steps)	Parameter description
Mac48Address	MAC-48 Address	valid addresses	The MAC-48 address of the device requesting to be authenticated.
DeviceId	device short address	[1, 65534] (1)	The short address of the device requesting to be authenticated.
SecurityType	integer	IDs as defined in Table 8-1	The type of security as specified in clause 8.
AuthenticationDetails	variable	variable	The type-specific security information. Format depends on the SecurityType parameter.

Table 7-11: Parameters of the MLME-AUTHENTICATE.indication primitive

7.2.3 MLME-DISASSOCIATE

The MLME-DISASSOCIATE primitive is invoked in order to disassociate a given device from an OWPAN. The primitive may be invoked by a participant device or the OWPAN coordinator, as described in 5.4.6.

7.2.3.1 Request

The MLME-DISASSOCIATE.request indicates to the MLME to begin with the disassociation procedure as described in 5.4.6.

The parameters of the primitive are listed in Table 7-12.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	integer	[1, 65534] (1)	The OWPAN ID from which the device is supposed to be disassociated.
DeviceId	integer	[1, 65534] (1)	The device short address of the device to be disassociated from the OWPAN.
Reason	enumeration	USER_REQUEST, AUTHENTICATION_END	The reason for disassociation

Table 7-12: Parameters of the MLME-DISASSOCIATE.request primitive

7.2.3.2 Confirm

The MLME-DISASSOCIATE.confirm is …

The parameters of the primitive are listed in Table 7-14.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	integer	[1, 65534] (1)	The OWPAN ID as indicated in the beacon or RA frames of the target OWPAN.
DeviceId	integer	[1, 65534] (1)	The short address of the device to be disassociated.
Status	enumeration	ACKNOWLEDGED, TIMEOUT	The status of disassociation

Table 7-13: Parameters of the MLME-DISASSOCIATE.confirm primitive

7.2.3.3 Indication

The MLME-DISASSOCIATE.indication is invoked by the MAC to indicate the disassociation of a device from an OWPAN. It may be used by the MLME of a coordinator or participant device of an OWPAN.

The parameters of the primitive are listed in Table 7-14.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	integer	[1, 65534] (1)	The OWPAN ID as indicated in the beacon or RA frames of the target OWPAN.
DeviceId	integer	[1, 65534] (1)	The short address of the device to be disassociated.
Reason	enumeration	USER_REQUEST, AUTHENTICATION_END	The reason for disassociation

Table 7-14: Parameters of the MLME-DISASSOCIATE.indication primitive

7.2.4 MLME-GET

The MLME-GET primitive allows the DME to obtain the value of certain readable MAC and PHY PIB attributes.

7.2.4.1 Request

Upon reception of a MLME-GET.request primitive, the MLME shall read the requested MAC or PHY PIB attribute from its information storage.

The parameters of the primitive are listed in Table 7-15.

Parameter name	Type	Value range (Steps)	Parameter description
AttributeId	integer	Valid attribute IDs as listed in **Error! Reference source not found.**.	The ID of the attribute to get.

Table 7-15: Parameters of the MLME-GET.request primitive

7.2.4.2 Confirm

The MLME-GET.confirm primitive is issued by the MLME as a response to a preceding MLME-GET.request primitive.

The parameters of the primitive are listed in Table 7-16.

Parameter name	Type	Value range (Steps)	Parameter description
Status	enumeration	SUCCESS, FAILURE	Indicates whether the preceding MLME-GET.request primitive was successful or not.
FailureReason	enumeration	NON_EXISTENT, OTHER_ERROR	Indicates the reason for failure if the Status is FAILURE.
AttributeId	integer	Valid attribute IDs as listed in **Error! Reference source not found.**.	The ID of the attribute to get.
AttributeValue	variable	attribute-specific	The value of the attribute to get.

7.2.5 MLME-SET

The MLME-SET primitive allows the DME to modify the value of certain writable MAC and PHY PIB attributes.

7.2.5.1 Request

Upon reception of a MLME-GET.request primitive, the MLME shall set the requested MAC or PHY PIB attribute to have the value provided with the AttributeValue parameter.

If a PIB attribute is set by the coordinator in accordance with the OWPAN operation configuration, it shall not be writable via the MLME-SET.request primitive.

If setting a read-only attribute is attempted, the MLME shall respond in the corresponding confirm with the FailureReason parameter set to READ_ONLY.

The parameters of the primitive are listed in Table 7-19.

Parameter name	Type	Value range (Steps)	Parameter description
AttributeId	integer	Valid attribute IDs as listed in **Error! Reference source not found.**.	The ID of the attribute to get.
AttributeValue	variable	attribute-specific	The value of the attribute to set.

7.2.5.2 Confirm

Through issuing the MLME-SET.confirm primitive, the MLME responds to a previous MLME-SET.request.

The parameters of the primitive are listed in Table 7-18.

Parameter name	Type	Value range (Steps)	Parameter description
AttributeId	integer	Valid attribute IDs as listed in **Error! Reference source not found.**.	The ID of the attribute to set.
AttributeValue	variable	attribute-specific	The value of the attribute to set.
Status	enumeration	SUCCESS, FAILURE	Indicates whether setting the PIB attribute was successful or not.
FailureReason	enumeration	READ_ONLY, NON_EXISTENT	Indicates the reason for failure if the Status is FAILURE.

Table 7-18: Parameters of the MLME-SET.confirm primitive

7.2.6 MLME-SCAN

The MLME-SCAN primitive supports the DME in requesting the MLME to issue a scan for existing OWPANs.

7.2.6.1 Request

The MLME-SCAN.request is issued by the DME in order to initiate the scanning procedure.

The parameters of the primitive are listed in Table 7-19.

Parameter name	Type	Value range (Steps)	Parameter description
ScanDuration	milliseconds	[1, 65535] (1)	Specifies for how long the device shall listen for incoming frames.

Table 7-19: Parameters of the MLME-SCAN.request primitive

7.2.6.2 Confirm

The MLME-SCAN.confirm primitive is used by the MLME to report the results of a scan to the DME.

The parameters of the primitive are listed in Table 7-20.

Parameter name	Type	Value range (Steps)	Parameter description
ResultList	a list of result entries	entries as specified in Table 7-21	The set of observed OWPANs.
EntryCount	integer	implementation-specific	The number of entries in the result list.
Status	enumeration	SUCCESS, FAILURE	The result of the association process.
FailureReason	enumeration	SCAN_IN_PROGRESS, OTHER	The reason for failure if the Status parameter is FAILURE.

Table 7-20: Parameters of the MLME-SCAN.confirm primitive

The *ResultList* parameter shall contain a list in which every entry has the elements listed in Table 7-21.

Detail	Description
OWPAN ID	The ID of the observed OWPAN
OWPAN Name	The name of the observed OWPAN
Electrical SNR	The SNR during the reception of the OWPAN's frame
Security Type	The security required by the observed OWPAN

Table 7-21: Scan result entry elements

7.2.7 MLME-START

The MLME-START primitive is used to instruct a device MAC to serve as a coordinator and start operation of a new OWPAN.

7.2.7.1 Request

The MLME-START.request primitive is issued by the DME and received by the MLME and triggers the procedure to start an OWPAN.

The MLME-START.request primitive shall be confirmed by the MLME through a subsequent MLME-START.confirm primitive invocation.

The parameters of the primitive are listed in Table 7-22.

Parameter name	Type	Value range (Steps)	Parameter description
OwpanId	16 bit integer	[1, 65534]	The OWPAN ID as indicated in the beacon or RA frames of the target OWPAN.
OwpanName	6-24 octet ASCII string	any	The OWPAN name serving as a human-readable ID for the OWPAN.
SecurityType	integer	IDs as defined in Table 8-1	The type of security as specified in clause 8.

Table 7-22: Parameters of the MLME-START.request primitive

7.2.7.2 Confirm

The MLME-START.confirm primitive is issued by the coordinator MLME to report the result of the preceding request to start a new OWPAN.

The parameters of the primitive are listed in Table 7-23.

Parameter name	Type	Value range (Steps)	Parameter description
Status	enumeration	SUCCESS, FAILURE	Whether the preceding MLME-START.request primitive was successful or failed.
FailureReason	enumeration	PARAMETER_CONTRADICT, OTHER	If stopping the OWPAN was not successful, the parameter indicates the reason.

Table 7-23: Parameters of the MLME-START.confirm primitive

7.2.8 MLME-STOP

The MLME-STOP primitive is issued by the DME of a coordinator in order to cease operation of a running OWPAN.

7.2.8.1 Request

The MLME-STOP.request primitive is issued by the DME of an active coordinator to the MLME in order to stop a running OWPAN.

The parameters of the primitive are listed in Table 7-24.

Parameter name	Type	Value range (Steps)	Parameter description
Timeout	16 bit integer	[1, 65535]	The time in microseconds after which the OWPAN shall be stopped. If the given time has passes since invocation of the primitive, the MLME shall respond with the corresponding MLME-STOP.confirm primitive, indicating success or failure.
Force	Boolean	true, false	Whether to stop an OWPAN forcefully. If set to true, the coordinator MLME shall not wait for successfully disassociation of the associated devices.

Table 7-24: Parameters of the MLME-STOP.request primitive

7.2.8.2 Confirm

The MLME-STOP.confirm primitive is issued by the MLME of a coordinator as a response to a preceding MLME-STOP.request primitive.

The parameters of the primitive are listed in Table 7-25.

Parameter name	Type	Value range (Steps)	Parameter description
Status	enumeration	SUCCESS, FAILURE	The result of the preceding MLME-STOP.request primitive.
FailureReason	enumeration	DISASSOCIATION_FAILED, OTHER	If stopping the OWPAN was not successful, the parameter indicates the reason.

Table 7-25: Parameters of the MLME-STOP.confirm primitive

7.3 MAC PIB Attributes

The MAC comprises variables and constants that define its behavior. In this standards, these are referred to as "PIB attributes".

Table 7-26 and Table 7-27 list variable and constant PIB attributes. It provides the attribute name, a description and information about the constant or space of possible values and associated units. Some variables shall be readable and writable via the MLME-GET.request and MLME-SET.request respectively. Whether a variable can be read or written is indicated by a get for read or set for write in the get/set column. Attributes that are purely internal to the MAC are neither readable nor writable.

[NOTE: There should be default values for the PIB variables]

Variable attributes					
Name	**ID**	**Description**	**get / set**	**Bits**	**Unit / Range**
Association and OWPAN membership					
macOwpanId	x	The ID of the OWPAN with which the device is associated.	get	16	integer [1, 65534]
macCoordShortAddress	x	The 16-bit short address assigned to the coordinator through which the device is associated. A value of 0x0000 indicates that this value is unknown.	get	16	integer [0, 65534]
macSecurityType	x	The security type used by the OWPAN.	get	8	IDs from Table 8-1
macDevShortAddress	x	The short address assigned to the dev during association.	get	16	integer [1, 65534]
macEdScanThreshold	x	The threshold for energy detection during a passive scan.	get set	[?]	dBm optical power
macDeviceTimeout	x	The duration after which a coordinator assumes a device to be disassociated if it does not receive frames from that device.	get set	16	[1, 65545] milliseconds
Beacon-enabled channel access					
macBeaconNumber	x	The number of the current superframe, embedded by the coordinator in the beacon frame.	get	16	integer [1, 65535]
macNumSuperframeSlots	x	The total number of superframe slots in a superframe	get	16	integer [1, 65535] superframe slots
macCapMaxRetries	x	The maximum retransmission attempts for CAP transmissions.	get set	8	integer [1, 255]
macCapSlotLength	x	The number of superframe slots that form a single CAP slot for the slotted ALOHA access in the CAP.	get	8	integer [1, 255] superframe slots
macBsn	x	The sequence number added to the last transmitted beacon frame.	get		

Table 7-26: Variable MAC PIB attributes

Variable attributes (continued)					
Name	**ID**	**Description**	**get / set**	**Bits**	**Unit / Range**
Protected Transmission					
macRetransmitTimeout	x	The duration after which an ACK is required for a transmitted frame. Upon expiration, a MPDU is typically retransmitted.	get	16	unsigned integer [1, 65535] µs
macMaxFrameRetries	x	The maximum number of attempted retransmissions, before the transmission of an MPDU is ultimately considered to be failed.	get set	8	TRUE (1), FALSE (0)

Table 7-26: Variable MAC PIB attributes (continued)

Constant attributes			
Name	**Description**	**Value**	**Unit**
aSuperframeSlotDuration	The duration of a single superframe slot.	**1**	µs
aInitialCapCw	The value to select for the back off window for the first retransmission in the CAP.	**1**	CAP slots
aMaximumCapCw	The maximum value for CW in the CAP.	**i**mplementation-specific	
aClockAccuracy	The accuracy of the device system-clock.	**30**	ppm
aMinFragmentSize	The minimum size of a MSDU fragment.	**64**	octets
aProtectedWindow	The maximum number of unacknowledged MPDUs to be in-flight.	**1024**	MPDUs
aMaxAssocRetries	The maximum number of retries for the association through random channel access.	**10**	retransmissions
aMac48Address	The device's MAC-48 address.	**any valid MAC-48 address**	MAC-48 address
aGtsTaifs	The Turn Around Interframe Space used in the beacon-enabled mode.	-	-

Table 7-27: Constant MAC PIB attributes

7.4 Capabilities

Capabilities formally indicate functionality that are supported, i.e. implemented, by a device. Each capability has a name and a numeric ID with a width of 16 bits. Some capabilities may require other capabilities to be implemented through the device. Capabilities are listen in Table 7-28.

Name	ID	Description	Required capabilities
capHbPhy	1	The device supports the usage of the HB-PHY.	*capMultipleInputFeedback, capEffectiveChannelFeedback, capVariableElements*
capMultiOfeEstimation	2	The device supports orthogonal pilot channel estimation and feedback.	
capMcsRequest	3	The device supports requesting a modulation and a coding scheme to be used for transmission towards it.	
capShortAddressing	4	The device supports the use of short addresses.	
capFullDuplex	5	The device supports simultaneous transmission and reception.	
capBlockAcknowledgment	6	The device supports the block acknowledgment mechanism.	
capVariableElements	7	The device supports parsing and transmitting of MAC frames containing a variable number of elements.	
capCoordinator	8	The device supports acting as a coordinator.	
capSecurity	9	The device supports security	
reserved	10-2040		

Table 7-28: MAC capabilities

8 Security

The MAC sublayer is responsible for providing security services on specified incoming and outgoing frames. This standard supports the following security services:

- Data confidentiality
- Data authenticity
- Replay protection

…

This standard supports the usage of different security types. For this purpose, every distinct security type has an ID assigned in order to facilitate differentiation between the approaches in the MAC protocol. Table 8-1 lists the different security types and their corresponding ID as well as the clause in which they are described.

Security Type	ID	Clause
Open System	0	8.2
AES-256 PSK	1	8.3
AES-256 EAPOL	2	8.4
…	?	?

Table 8-1: Security types

Except the *Open Systems* security, each security type is denoted by the applied encryption algorithm and authentication method. Each security type may make use of the existing protocol procedures. For example, during association, generic fields for the authentication handshake are included in the exchanged frames.

8.1 Generic security description

In the IEEE 802.15.13 standard, security mainly affects two processes:

1. The association with an OWPAN if the latter is secured.
2. The transmission and reception of MSDUs and management MPDUs

[Maximum MPDU size including security!]

8.2 Open Systems

8.3 AES-256 PSK

8.4 AES-256 EAPOL

9 PHY services

This clause specifies the services provided by the PHY to the MAC layer. Implementations are expected to tightly integrate the MAC and PHY. Hence, the interfaces to the PHY, i.e. the PHY- and PLME-SAP are anticipated to be not exposed by a standards-compliant chip.

9.1 PHY-SAP

The PLME-SAP constitutes a logical interface for requesting PPDU transmissions by the PHY from the MAC layer and indicating PPDU receptions by the PHY to the MAC layer. Due to the aforementioned monolithic implementation of chipsets, the PHY-SAP is not explicitly specified within this standard.
However, in order to refer to different generic functions of PHYs, clause 9.3 describes specific functions and services provided by one or multiple PHYs within this standard.

9.2 PLME-SAP

The PLME-SAP constitutes a logical interface to invoke management functions on the PHY from the MAC layer. However, due to the aforementioned monolithic implementation of chipsets the PLME-SAP is expected to be no external interface and is thus also not specified within this standard.
However, in order to refer to different generic functions of PHYs, clause 9.3 describes specific functions and services provided by one or multiple PHYs within this standard.

9.3 Generic PHY functions

This clause describes the generic PHY functionality, common to all PHYs included in the 802.15.13 standard. This is required to regard for generic functionality of different PHYs on the MAC layer without knowing the PHY's specifics.

9.3.1 Base rate

Each PHY defines a base rate, which is used to transmit specific frames such as the beacon or RA control frames.

9.3.2 Turnaround time

If a PHY operates in time division half-duplex mode, it may require a certain time to switch between transmit and receive mode.

9.3.3 Multi-OFE channel estimation

Some PHYs support concurrent channel estimation between a multiple transmitters and one or more receivers. This is achieved by simultaneously transmitting signals from the respective transmitters which overlap in space and time. However, distinct channel estimation is still possible through pilot symbol design and assigning the transmitters different orthogonal “divisions” of the pilot signal.

Multi-OFE pilots have, in contrast to conventional pilots, more than one division. Supporting PHYs in this standard support up to 32 orthogonal divisions. Furthermore, a single PPDU may include up to 7 multi-OFE pilot symbols.

A coordinator supporting the *capMultiOfeEstimation* capability shall support the transmission of Multi-OFE channel estimation pilots. For a given PPDU, which is designated by the MAC to contain multi-OFE pilots, a PHY receives through the PHY-SAP the division and symbol position of the requested pilot.

A device supporting the *capMultiOfeEstimation* capability shall support measuring the channel from multiple simultaneous transmitters based on multi-OFE pilots. The measured CSI shall comprise the received signal strength from every transmitter (OFE) for all relevant taps, as well as the delay for all taps.

9.4 PHY PIB Attributes

PHY PIB attributes determine the behavior of the MAC, analogously to what MAC PIB attributes do for the MAC. As the PLME-SAP is unspecified, the management of PHY PIB attributes are left to the implementer. However, in order to make attributes accessible from the DME where necessary, some PHY PIB attributes can be read or written via the MLME-GET and MLME-SET primitives. The **r/w** column indicates whether an attribute is readable or writable this way.

Name	Description	get/set	Value [Range] (Step)	Unit
phyTxDelay	The exact time between starting the transmission of a PSDU on the MAC sublayer and transmission of the first PPDU signal sample at the point of transmission.	get	[1, 65535] (1)	µs
phyRxDelay	The exact time between receiving the first sample of the PPDU preamble and the instant in time when the complete PSDU is available to the MAC sublayer.	get	[1, 65535] (1)	µs
phyMaxPsduSize	The maximum supported PSDU size. This attribute is PHY-specific.	get	65535	octets
phyMultiOfeDivisions	The number of orthogonal pilot divisions (e.g. subcarrier spacings or Hadamard codes). This attribute shall be present if the device implements the *capMultiOfeEstimation* capability.	get	[1, 32]	distinct orthogonal pilots
phyMultiOfeSymbols	The number of consecutive additional channel estimation symbols supported by the PHY. This attribute shall be present if the device implements the *capMultiOfeEstimation* capability.	get	[0, 7]	symbols

Table 9-1: PHY PIB attributes

10 PM-PHY

[see document 15-18-0003-07-0013]

11 LB-PHY

[see document 15-18-0267-05-0013]

12 HB-PHY

[see document 15-18-0273-02-0013]

Acknowledgements

Tuncer Baykas acknowledges the support TUBA-Gebip award program.