IEEE 802.15  
Wireless Specialty Networks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IEEE P802.15.13  Updated text parts for clause 5 | | | | |
| Date: 2019-07-18 | | | | |
| Author: | | | | |
| Name | Affiliation | Address | Phone | Email |
| Kai Lennert Bober | Fraunhofer HHI |  |  | kai.lennert.bober@hhi.fraunhofer.de |
|  |  |  |  |  |
|  |  |  |  |  |

Abstract

# This document contains proposed text changes clause 5 of IEEE P802.15.13.

1. MAC functional specification

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**.

* 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.

* + 1. Addressing

Within the OWC MAC, each device shall be addressable through a 48-bit MAC address compatible with IEEE Std. 802-2014 (EUI-48).

In addition, a 16-bit short address is issued to each device as part of the association process. The allocation of short addresses to associated devices is at the discretion of the coordinator implementation. The short address 0x0000 shall always belong to the coordinator. Furthermore, the address 0xFFFF shall be regarded as the short broadcast address and hence received by all devices. It shall not be allocated to an associating device.

For each associated device, there is a 1:1 correspondence between short addresses and 48-bit MAC addresses. The two types of addresses may be used interchangeably where appropriate. The 48-bit broadcast address FF:FF:FF:FF:FF:FF shall correspond to the short broadcast address 0xFFFF;

Each OWPAN is identified through its OWPAN ID. The OWPAN ID corresponds to the MAC address of the coordinator.

OWPANs in broadcast mode shall make use of the OWPAN ID FF:FF:FF:FF:FF:FF.

* + 1. 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. The transmit process is illustrated in Figure 7.

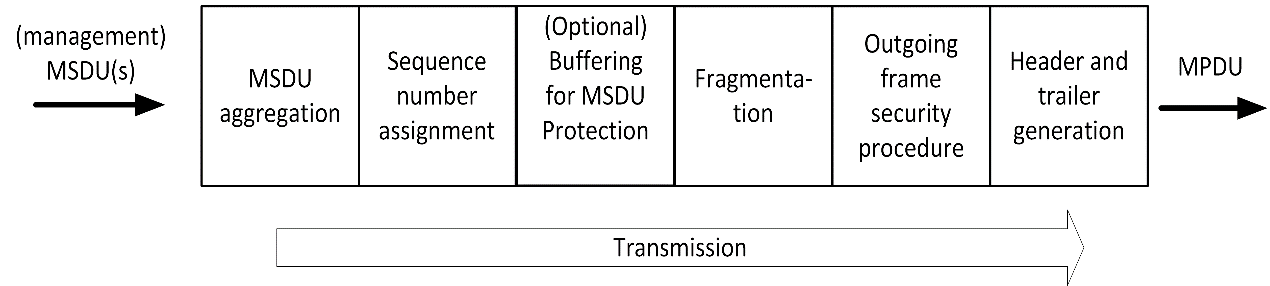


Figure 7: The MAC transmit process

A device prepares MPDUs for transmission in accordance with the requirements of the channel access mode. This includes generation of the approporiate header fields and limiting the MPDU size to the maximum available transmission time. In addition, a transmitting device shall ensure that the MPDU size does not exceed the maximum supported PSDU size of the used PHY.

MPDU generation process involves optional aggregation of multiple MSDUs (clause 5.6). The (A-)MSDUs may optionally be fragmented (clause 5.5). For protected transmission, sequence numbers may be assigned to MPDUs (clause 5.7).

Finally, the channel access mechanism applied in the OWPAN (see clauses 0. and 5.3) regulates when to hand the resulting MPDU to the PHY for transmission over the medium.

* + 1. The receive process

After a PSDU was successfully received by the PHY, the corresponding MPDU enters the MAC through the PHY-SAP. The receive process is illustrated in Figure 8.

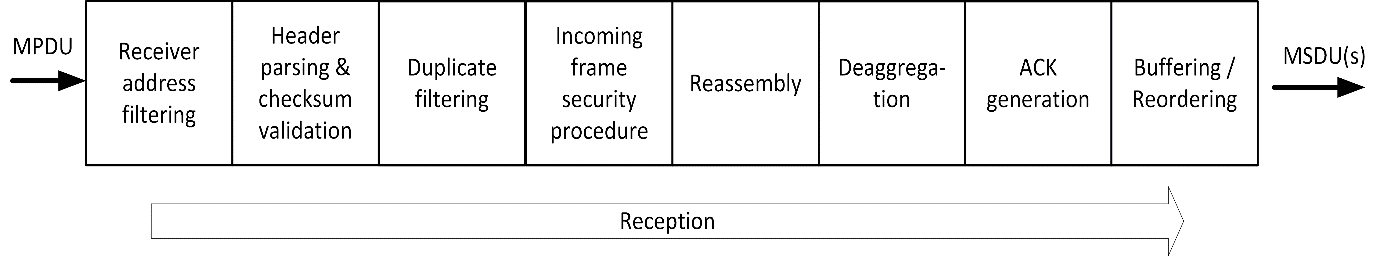


Figure 8: The MAC receive process

have a receiver address that does not belong to the device or the broadcast receiver address.its associated Each device shall receive MPDUs with the OWPAN ID and destination address set to the broadcast address FF:FF:FF:FF:FF:FF.

The MAC then shall check the integrity of the frame based on the FCS field included in the frame. If the frame contains errors, the MAC shall discard the frame.

The MAC shall discard frames that have an unsupported *Frame Version* in their *Frame Control* element (clause 6.2.1).

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 clause 8.2.3. 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 desaggregation according to 5.6.2 if the frame contains aggregated MSDUs.

The MSDUs of protected frames from a single transmitter shall be passed to the upper layers in order. The MAC shall discard duplicate protected MPDUs.

Finally, the MAC shall optionally generate an acknowledgment for each successfully received protedted MPDU according to clause 5.7.

* 1. Beacon-enabled channel access

If an OWPAN operates in beacon-enabled channel access mode, channel time is divided into subsequent superframes. Each superframe is composed of three major parts: a beacon transmission, a 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.

* + 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 6.6.4). 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 9. 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.

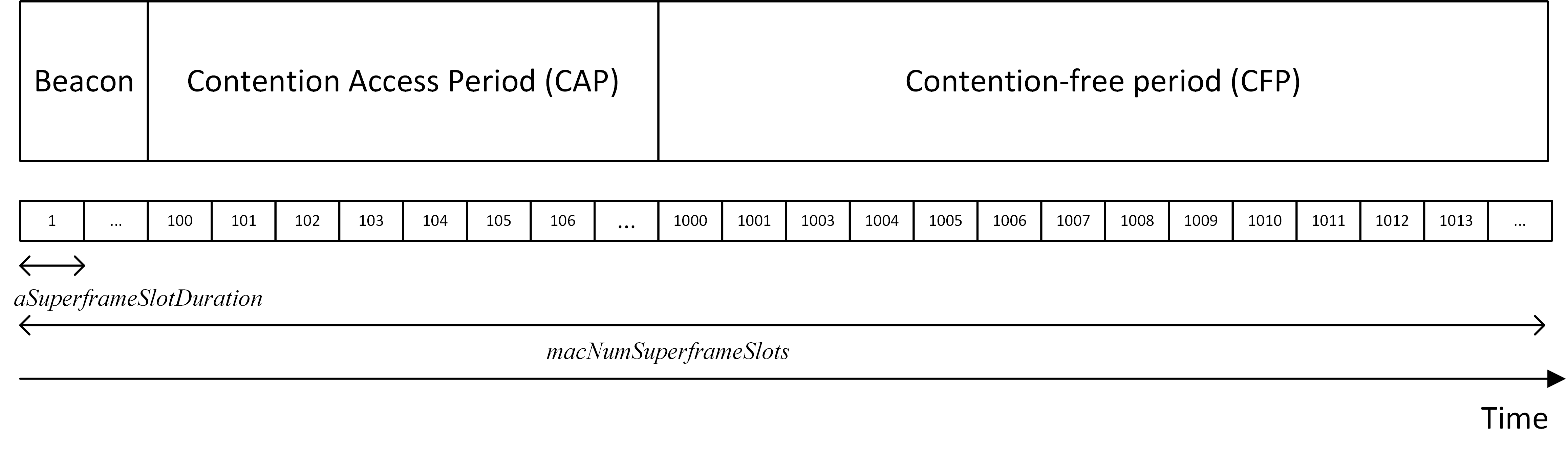


Figure 9: Superframe structure with exemplary number of superframe slots

* + 1. Beacon transmission

In the beacon-enabled channel access mode, the coordinator shall transmit a beacon at the beginning of the superframe. Beacons should be transmitted with a constant periodicity when possible.

The beacon frame is a control frame having the *Superframe Descriptor* element as payload (refer to clause 6.6.4). The *Superframe Descriptor* element may include additional elements in the embedded *Variable Element Container* element.

The coordinator shall maintain the *macBeaconNumber* PIB attribute and increment it by one for every started superframe and corresponding beacon transmission. The *macBeaconNumber* value wraps to 0 after reaching the maximum value given in Table 43.

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 multi-OFE pilots 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.

* + 1. Medium access in the CAP

The CAP shall only be used for frame transmissions in the

1. Association procedure (see 5.2.3.1)
2. 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 *macNumCapSlots* is advertised in the *CAP Slots* field of the *Superframe Descriptor* element contained in each Beacon frame (see 6.6.4). 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 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 each beacon frame (clause 6.6.4)*.*

A device willing to transmit a frame in the CAP shall choose a number of CAP slots **RS (“Random Slots”)** uniform randomly from [0, max(CW, CAP slots in current superframe - 1))], 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 *macCapMaxRetries.*

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 [0, max(CW, CAP slots in current superframe - 1))]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 10 and Figure 11 respectively.

* + - 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 10.

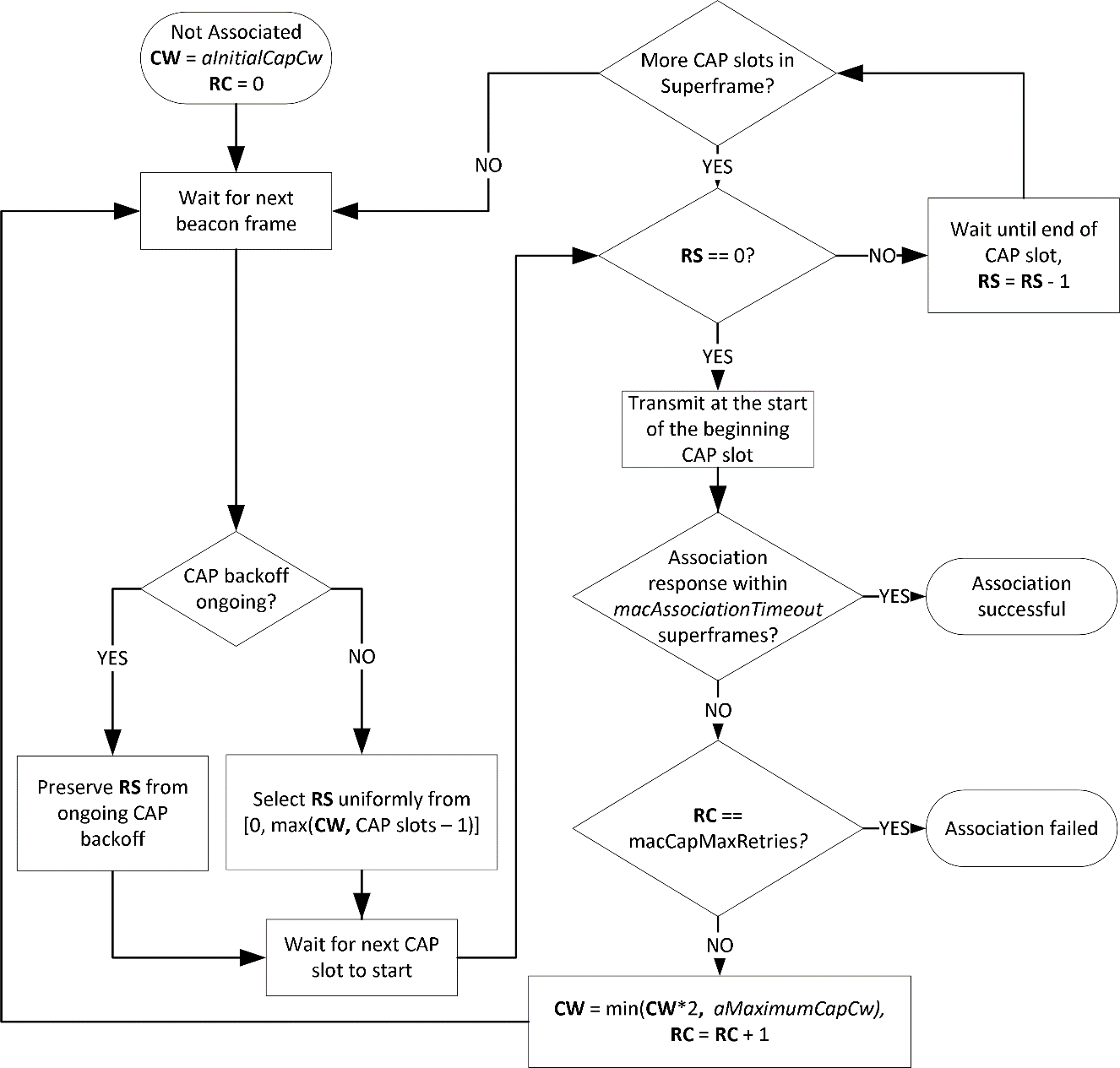


Figure 10: Flow chart of an association request in the CAP

After transmitting the *Association Request* element, the device shall continue to listen for an *Association Response* element from the coordinator for at least *macAssociationTimeout.*

If no *Association Response* element is received within this time, the device may reattempt association through sending the *Association Request* element again as described in 5.2.3. A device shall not attempt association more than *macCapMaxRetries* automatically.

* + - 1. Resource request procedure in the CAP

Typically, devices shall request additional resources in the CFP (see 5.2.5). However, when a device does not have any or only insufficient GTS time allocated for to perform a resource request, it may make use of the CAP to transmit a resource request to the coordinator. 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 control frame containing the *GTS Request* 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.

After transmitting the resource request, the device shall shall await updated GTS allocations from the coordinator. If the device does not receive sufficient GTS resources after 5 superframes, it shall consider the resource request failed. In that case, it may transmit a new resource request according to the rules in 5.2.3.

The flow chart for the resource request procedure is depicted in Figure 11.

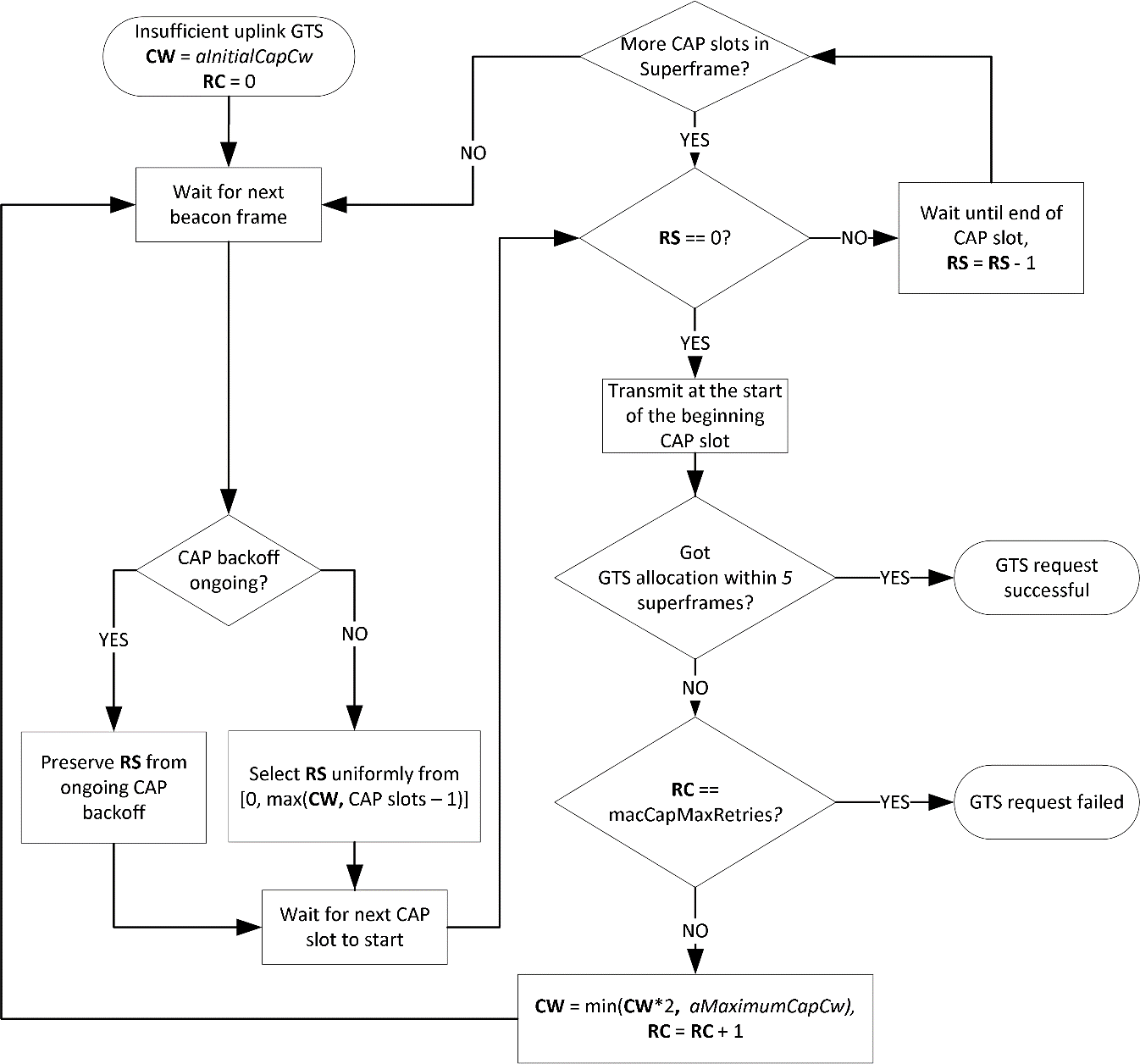


Figure 11: Flow chart of a GTS request in the CAP

* + 1. 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 6.6.7. 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 do not 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. If the *capFullDuplex* was negotiated during association, devices shall enable the receiver within the whole superframe. Otherwise, the connection is assumed to be half-duplex. In that case, devices shall enable their receiver outside their own GTSs.

* + 1. 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 and 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 *GTS Request* 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 signalled from the coordinator to the corresponding devices via control frames including *GTS Descriptor* elements or *GTS Descriptor List* element. 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 belongs to themselves based on that receiver address.

A GTS allocation is only valid for one superframe.

* + 1. 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 12. All superframe slots and hence timings within the superframe are thus relative to the start of the beacon preamble.

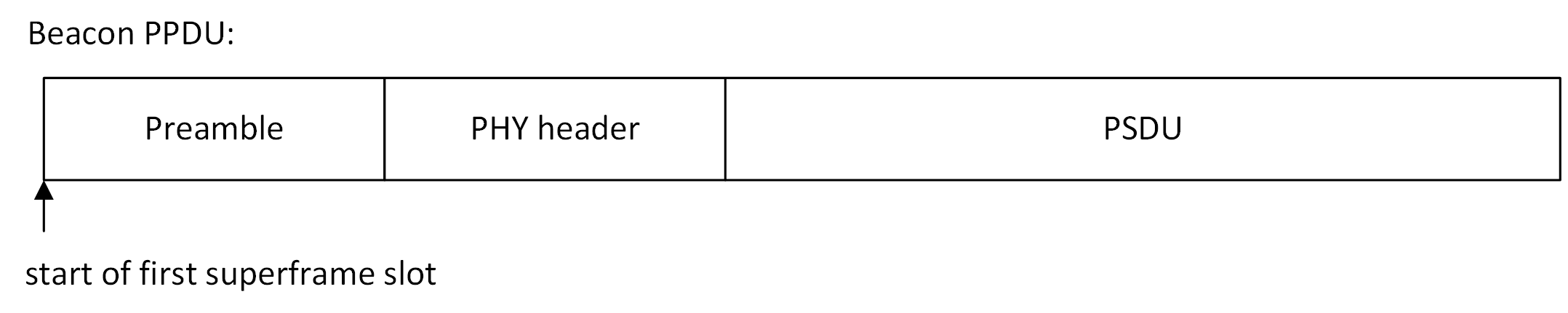


Figure 12: Timing relative to the beacon frame reception at every device

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

* + 1. Interfame spaces

The only IFS defined by this standard is the Turn Around Interframe Space (TAIFS). 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 utilize fully 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 and full-duplex operation was negotiated during association, it may disregard the requirement to finish transmissions at least a TAIFS before the end of its GTS. In that case, TAIFS may be set to 0.

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 subsequent transmissions without interruption.

* + 1. 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 Descriptor* element (see clause 6.6.6). Guard time is the time between the end of one GTS and the start of the next GTS. Guard time is counted in superframe slots. For that purpose, the guard time shall be rounded up to the next integer number of superframe slots.

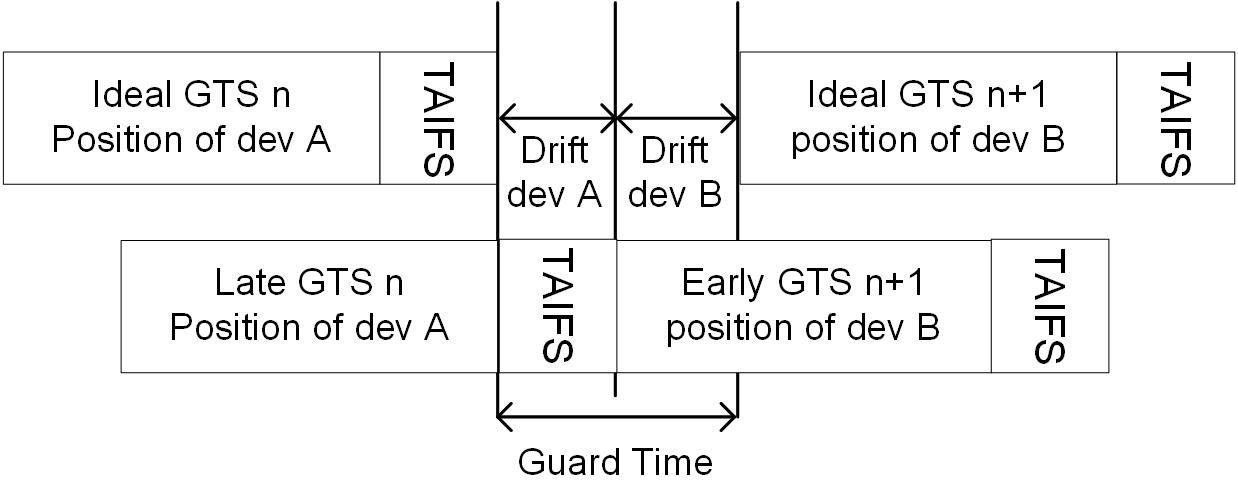


Figure 13: Application of the guard time and TAIFS between adjacent GTSs

Figure 13 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 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 can be calculated as follows:

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 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 lies between two subsequent GTS that are not orthogonal in space.

* 1. Non-beacon-enabled channel access

The MAC for non-beacon-enabled OWPAN starts with the transmission of a control frame (RA) from the coordinator, which carries the information required for new devices to initiate an association procedure. Following the control frame, a contention-based random access period starts. During the contention-based random access period, devices compete on the UL channel sending Poll Request frames to the coordinator to enable the coordinator to poll the device in the following polling period. In the contention-free period, the coordinator can poll one or multiple devices within one frame transmission. Devices are effectively polled by the coordinator and, as soon as they recognize the beginning of the polling frame (polling information is contained in the frame header at the beginning of the frame), they are allowed to transmit on UL channels. The flow chart in Figure 14 illustrates the logic for the non-beacon-enabled OWPAN MAC.

A close up of a sign

Description automatically generated

Figure 14 Flow chart of the non-beacon-enabled OWPAN MAC

The polling mechanism enables very low-complexity synchronization of the transmission slots and also enables the coordinator to dynamically adapt and assign the slots for transmissions. The specific polling algorithm and order is outside the scope of this specification and is left as a system designer's choice. The concept is presented in the example in Figure 15.

A screenshot of a cell phone

Description automatically generated

Figure 15 non-beacon-enabled structure example

* + 1. Contention-based Polling Period (CAPP)

Figure 16 presents a flow chart for the contention-based random access period process. The coordinator sends a control frame (RA) periodically at the beginning of a polling cycle. The first valid downlink frame (i.e., DATA, MAC Management or MAC Control frames) after the control frame (RA) marks the beginning of the contention period, i.e., the first valid frame after the control frame is interpreted as a poll to all stations that are not connected to transmit their Poll Request. All stations that have already been associated with the coordinator may ignore any polling during the contention period. If no stations have been connected to the coordinator, the first valid frame would be the subsequent control frame. After receiving the control frame, the device can send a Poll Request. The coordinator starts polling a station in the next polling round after receiving the Poll Request.

A close up of a map

Description automatically generated

Figure 16 Flow chart for contention-based random access period

In case several devices want to connect at the same time, a collision may occur. If it is not polled during the next polling round, a device assumes a collision has occurred and proceeds accordingly with a back-off and an attempt to reconnect. Then, the random backoff is introduced. For example, one device retries to establish a connection after 2 control frames (RA), another device retries after 5 control frames (RA). A device transmits a poll request frame immediately after it decodes a valid MAC frame header. Hence, as depicted in Figure 15, it will not wait for the entire frame on the downlink from the coordinator to be transmitted. Upon a successful polling request (the coordinator has successfully received the station's request to be polled and has indicated this by polling the station within the next transmission round), a station exchanges the necessary association and authentication information with the coordinator via association and authentication control frames. Upon successful authentication and association, the device is assigned a short address which will identify the device among all other devices connected to the same coordinator.

* + 1. Contention-Free Polling Period (CFPP)

During the contention-free polling period, each device is required to be able to respond to a Poll frame received from a coordinator. A device should also be able to request to be polled by an active coordinator. When polled by the coordinator, a device may transmit only one MPDU. The acknowledgments for any data frame can "piggyback" on the transmission of any management or data frame (including Null frames). If a frame is not acknowledged, retransmission of the frame will be rescheduled by the coordinator. If a polled device does not have any data or acknowledgement to transmit, it simply ignores the polling request. The coordinator will stop polling a device provided it does not receive any response from the device for 5 seconds. The concept is given in the flow chart in Figure 17.

A close up of a sign

Description automatically generated

Figure 17 Flow chart for Contention-free polling period

Any frames transmitted from the coordinator to the device are treated as polling frames, except for the control frame (RA). This can be assumed because every packet transmission contains a polled device number, which enables the stations to keep track of the polling order. A newly connected device is made aware of its queue number with the first valid packet it receives from the coordinator, which contains both its queue number and its MAC address. Furthermore, the coordinator will not start another polling request before the current polled device finishes it transmission on uplink. Hence, no collisions in the uplink are possible and each downlink packet transmission can function as a polling frame. In case the response to a poll request is not detected by the coordinator, the coordinator will attempt to poll the next device in the queue. Upon detection of a subsequent downlink packet by the device whose response was not detected, the device will stop transmission immediately in order to avoid any collisions in the uplink as illustrated in Figure 18. This functionality is necessary because the coordinator does not expect to always receive a response to its poll. If a device has no acknowledgment or information to transmit, it will simply ignore its possibility to transmit. However, a device will need to send a response even though it does not have acknowledgment or information to send after a while. The purpose of this is to keep the connection alive so that the coordinator will not consider the link is invalid. If there is no information to be transmitted to a device, which is next in the queue, a Data Null frame is used as a polling frame.

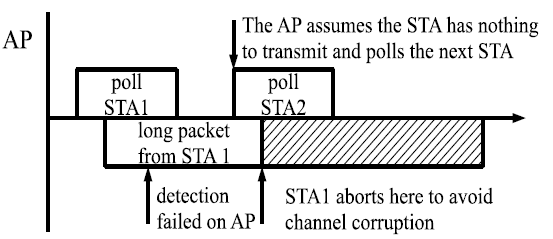


Figure 18 Consequences of the coordinator failing to recognize a poll response

* + 1. Acknowledgement and retransmission

The reception of every packet at the MAC layer level has to be acknowledged by the receiving side (coordinator or device). The flow chart in Figure 19 shows the logic of acknowledgement and retransmission processes.

A close up of text on a white background

Description automatically generated

Figure 19 Flow chart for acknowledgement and retransmission processes

If a packet is not acknowledged by the receiver (either the device or the coordinator) at the next polling round (i.e., when the receiver is polled next time), it is deemed that a retransmission is needed for the packet. Packets may arrive out of order. Hence, even if a device or a coordinator does not receive the expected acknowledgment during a given polling round, it can proceed with the transmission of the next packet. If packets are not acknowledged after 4 retransmissions, they are considered lost and are dropped. Figure 20 illustrates the acknowledgment procedure.

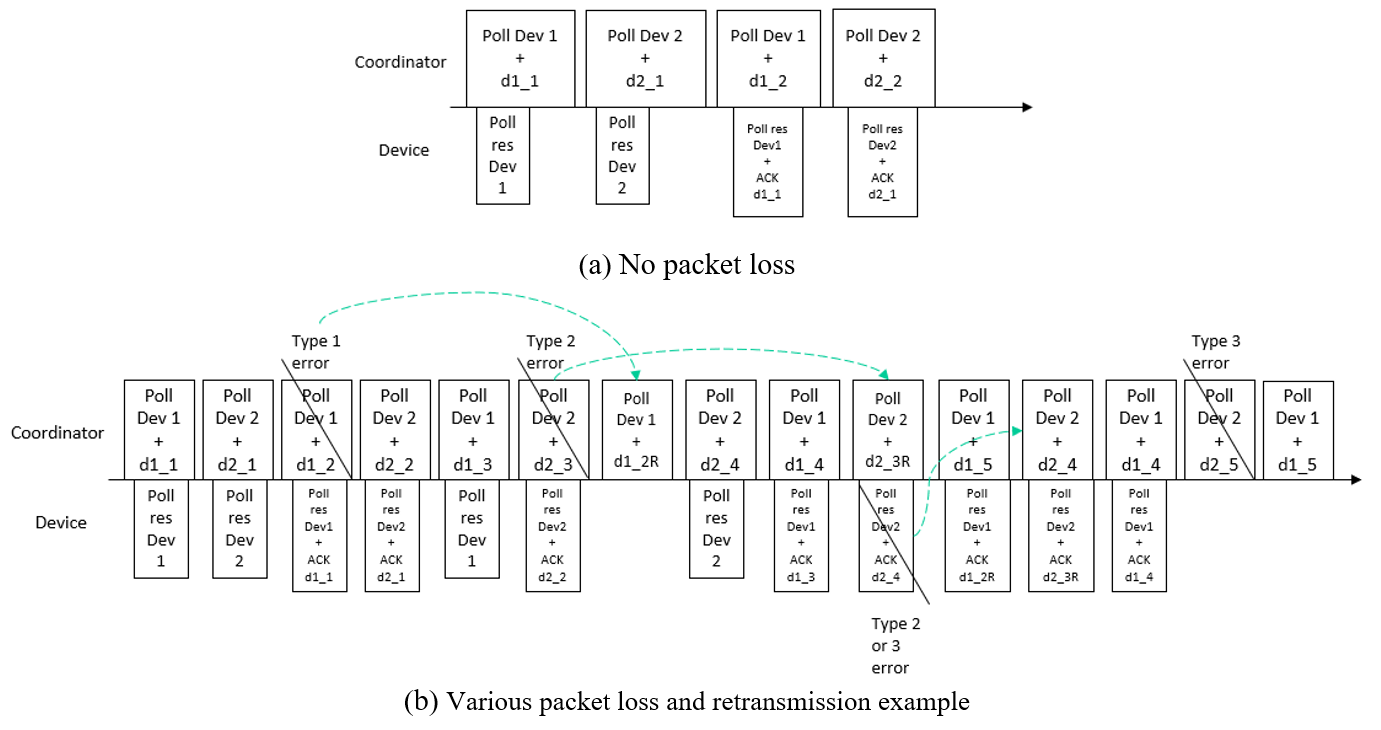


Figure 20 Downlink transmission example scenarios

Figure 20 shows an example for acknowledgment and retransmission. In this example, two devices are connected to the coordinator. The coordinator has a data frame (d1\_1) to transmit to the device 1, hence it polls the device and sends the data. Device 1 responds to the poll with a payload frame or a management frame as soon as it receives the PN sequence of the received frame. Note that if the high-reliability control header at the PHY is enabled, the device can respond as soon as this information is decoded at the PHY layer and the MLME is notified even before the entire DATA portion of the PHY packet is received and decoded. After both the downlink and the uplink packet transmissions are complete, the coordinator sends a data frame to Device 2. Device 2 replies to the poll with a payload frame as soon as the PN sequence is received, since the information of the current polled device was available from the high-reliability control header of the previous Poll. After completion of the transmission in both directions, the coordinator attempts to send a second data frame (d1\_2) to the device 1. Device 1 is polled again; hence it takes the opportunity to send the ACK for the first data frame d1\_1 to the coordinator. Device 2 acknowledges his reception once polled in another round. The Figure 20 (a) shows the above process when no packet loss exists.

Figure 20 (b) demonstrates a more complex example when three types of errors are considered.

1. Payload CRC error:
   * The error does not affect the decoding of the high-reliability control header.
   * The device which is supposed to be polled responds to the coordinator.
   * The device cannot acknowledge the coordinator for this frame due to the failure of decoding the DATA frame.
   * The lack of ACK results in a retransmission from the coordinator.
2. Header error:
   * The high-reliability control header cannot be decoded.
   * The device can still respond to the coordinator although the header is undecodable since each high-reliability header contains the information for the current polled device and the next device to poll.
   * Leads to no response for the next Poll frame as well as no reception of the frame d2\_3.
3. The packet is not detected successfully:
   * It has the lowest probability of occurrence.
   * It also affects the next Poll frame and receives no Poll response from the current polled device.
   1. 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.

* + 1. Scanning for OWPANs

A scan procedure is performed by a device to detect any OWPANs that are operating in its vicinity. In this standard, 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.

Devices shall support passive scanning for OWPANs. During a passive scan, the device listens for incoming frames and 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 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 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.

* + 1. 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 with observed OWPANs between multiple OWPAN coordinators can be provided through a coordinated topology.

The DME of the prospective coordinator shall select an OWPAN ID. The OWPAN ID shall also serve as the 48-bit MAC address of the coordinator. The DME shall provide the selected OWPAN ID 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.

* + 1. 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.

* + 1. 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.

* + 1. 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
   * + 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 variable PIB attributes 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 full 48-bit MAC addresses. The *Receiver Address* of the management frame shall be set to the coordinator’s address. The *Transmitter Address* of the frame shall be set to the 48-bit 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.

The requesting device shall transmit the management frame to the coordinator of the OWPAN. The channel access for association is detailed in clauses 0 and 5.3 respectively. The frame shall be transmitted unprotected (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. The coordinator shall transmit the *Association Response* to the requesting device.

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

A sequence chart of a successful association procedure is depicted in Figure 21.

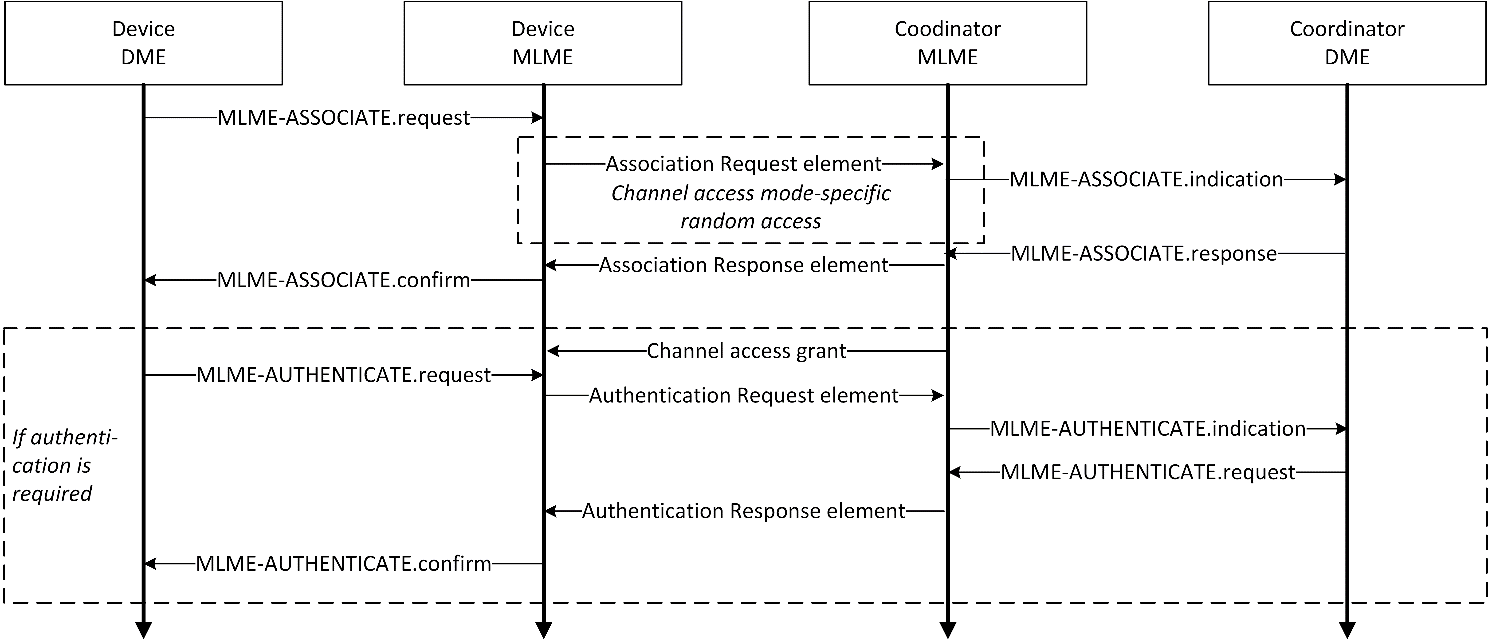


Figure 21: Association procedure message exchange

* + - 1. Authentication request

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. For that purpose, the coordinator shall grant channel access (e.g. GTS allocations or poll requests) to the authenticating device on a temporary basis for the transmission of an *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.

* + 1. 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.

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 22 a). If the device does not receive a corresponding acknowledgment frame, it shall retransmit the *Disassociation Notification* element up to *macMaxFrameRetries*. After reaching the maximum number of retransmissions, the device may be considered disassociated.

To disassociate a device from the OWPAN, the coordinator may transmit a management frame, containing the *Disassociation Notification* element, to the device to be disassociated as depicted in Figure 22 b). If the coordinator does not receive a corresponding acknowledgment frame, it shall retransmit the *Disassociation Notification* element up to *macMaxFrameRetries.* After reaching the maximum number of retransmissions, the device may be considered disassociated.

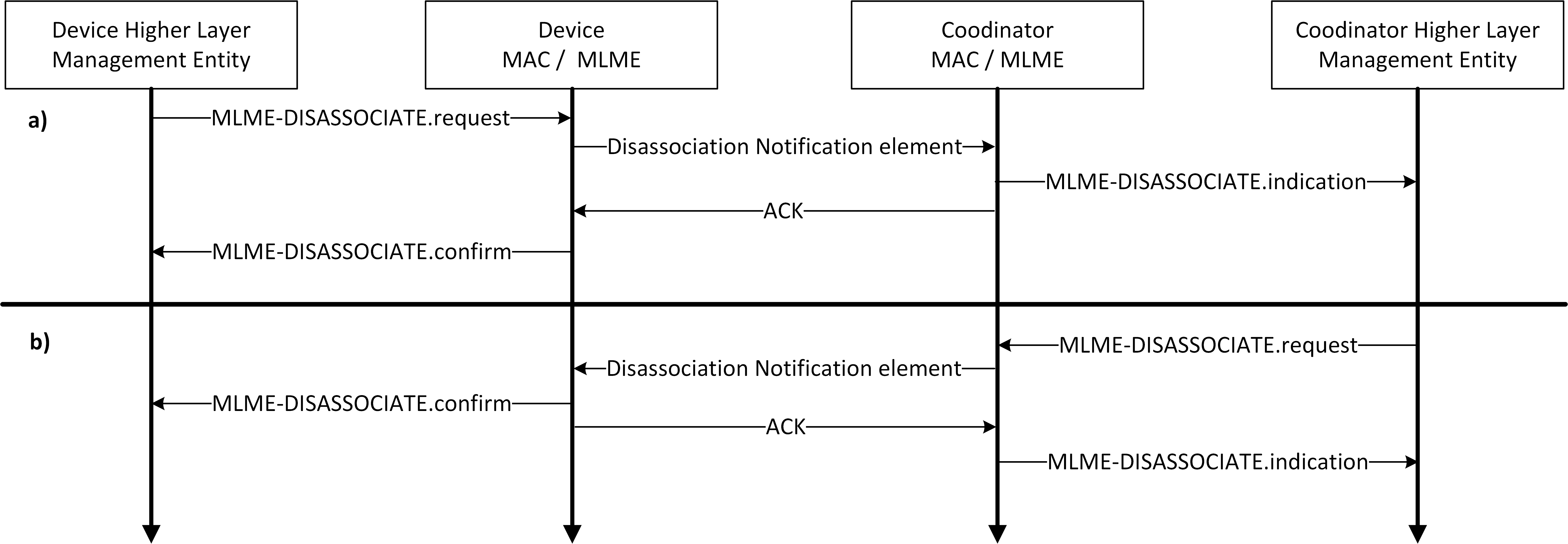


Figure 22: Disassociation initiated by the device (a) and the coordinator (b)

The lack of communication within a given time can be considered as loss of connection with a given device. If a coordinator does not receive any frames from a device for a duration of *macDeviceTimeout*, it may consider the device as disconnected.

* + 1. Interference detection

In response to the directional propagation characteristic of light, channel access mechanisms specified within this standard rely on centralized channel access control through the coordinator without the need for clear channel assessment befor each transmission. If a device not associated with the OWPAN enters the propagation domain of that OWPAN, it might interfere with ongoing transmissions because transmitters do not assess the channel at prospective receivers.

Devices supporting interference detection shall advertise the *capInterferenceDetection* capability. If the *capInterferenceDetection* capability was negotiated during association, devices may transmit an *Alien Signal Element* to the coordinator upon detection of a potentially interfering signal. When to transmit the *Alien Signal Element* is implementation-specific. The transmitting device shall fill in the fields of the *Alien Signal Element* as detailed in clause 6.6.17.

* 1. Fragmentation and reassembly

Fragmentation may be performed by the transmitting device on an MSDU or A-MSDU, together denoted as (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*.

(A-)MSDUAll fragments but the last fragment shall be sent with the *Last Fragment* field of the 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 by one. However, the *Fragment Number* field shall not be incremented when a fragment is retransmitted.

Fragments shall be transmitted in order of their F*ragment Numbers*. A device shall support concurrent reception of fragments of at least three (A-)MSDUs. A receiving device shall reassemble fragments with the same sequence number into the complete (A-)MSDU. The (A-)MSDU shall be completely reassembled before delivering it to the higher layer.

The receiving device may discard the fragments of an (A-)MSDU if it is not completely received within an implementation-specific timeout. The receiving device may also discard the oldest incomplete (A-)MSDU if otherwise a buffer overflow would occur upon reception of a new MPDU.

A device shall only acknowledge a sequence number if all fragments for that sequence number were successfully received and the corresponding (A-)MSDU could be reassembled. A device shall acknowledge successfully reassembled (A-)MSDUs if at least one MPDU containing a fragment of that (A-)MSDU had the Ack Request bit set to 1.

MPDUs containing fragments shall be acknowledged using Single ACK only. Block ACK shall only be used to ack reception of a whole successfully reassembled (A-)MSDU.

* 1. 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).

* + 1. Aggregation procedure

The optional aggregation procedure, as part of the transmit process detailed in Figure 7, is applied when a device decides to aggregate multiple MSDUs in a single MPDU. The combination of multiple aggregated MSDUs is referred to as A-MSDU.

A device shall only aggregate MSDUs that have the same receiver address. The total resulting MPDU size, resulting from all aggregated MSDUs and additional fields for aggregation 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 (see 6.6.9). The *MSDU Aggregation* element shall include the destination and source address of the MSDU, as well as the length of the wrapped MSDU.

MSDUs shall be contained in the A-MSDU in order of their reception through the MCPS-DATA.request. The first MSDU in the A-MSDU shall be the first MSDU received through the MCPS-SAP from all aggregated MSDUs.

An MPDU containing an A-MSDU shall be assigned a single sequence number like an MPDU containing only a single MSDU.

Figure 23 depicts the aggregation procedure during transmission of an MSDU with additional fragmentation of the A-MSDU for three MSDUs.

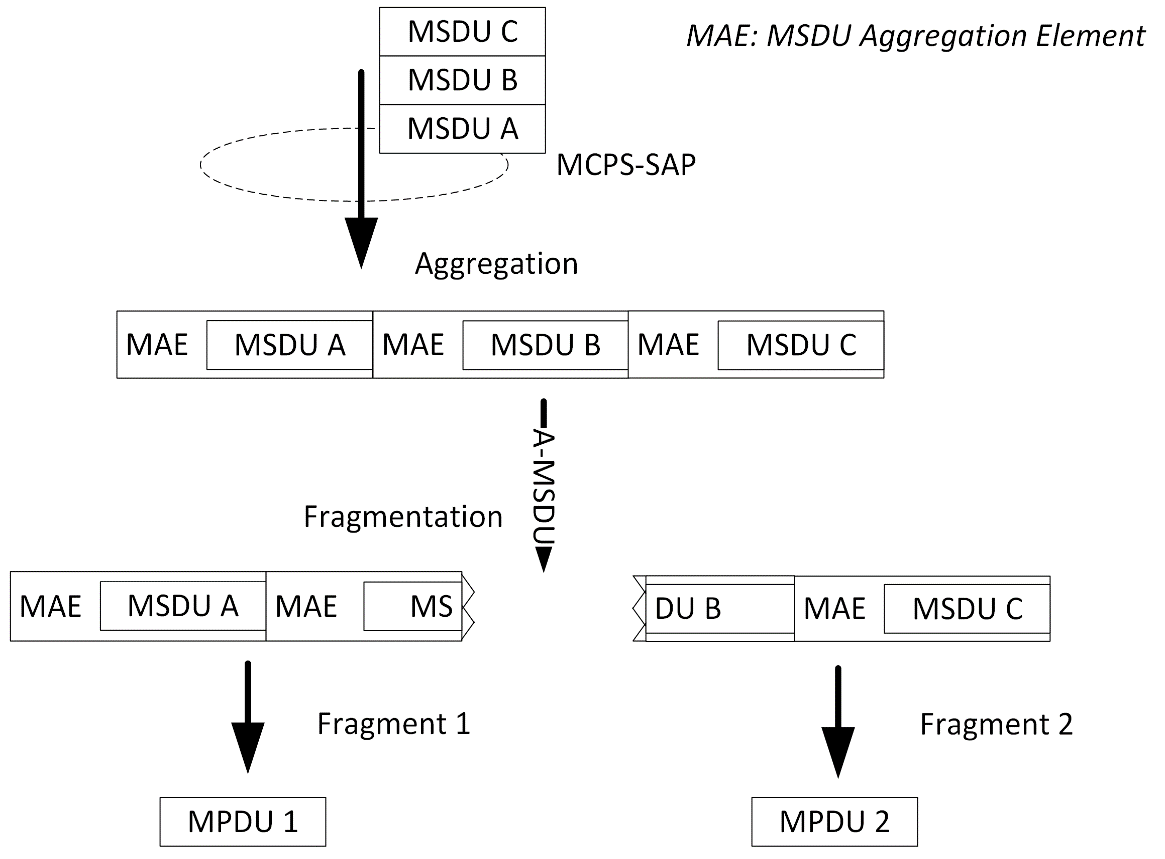


Figure 23: Aggregation and fragmentation

* + 1. Deaggregation procedure

If a device receives an A-MSDU data MPDU, 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. From the *MSDU Aggregation* elements, the original MSDUs can be recovered. The yielded MSDUs shall be delivered to the MCPS-SAP in their order within the A-MSDU. The first MSD U in the A-MSDU shall be delivered to the MCPS-SAP first. The destination and source address shall be obtained from the corresponding *MSDU Aggregation element*.

Figure 24an exemplarye in which a fragmented A-MSDU is reassembled prior to deaggregation.

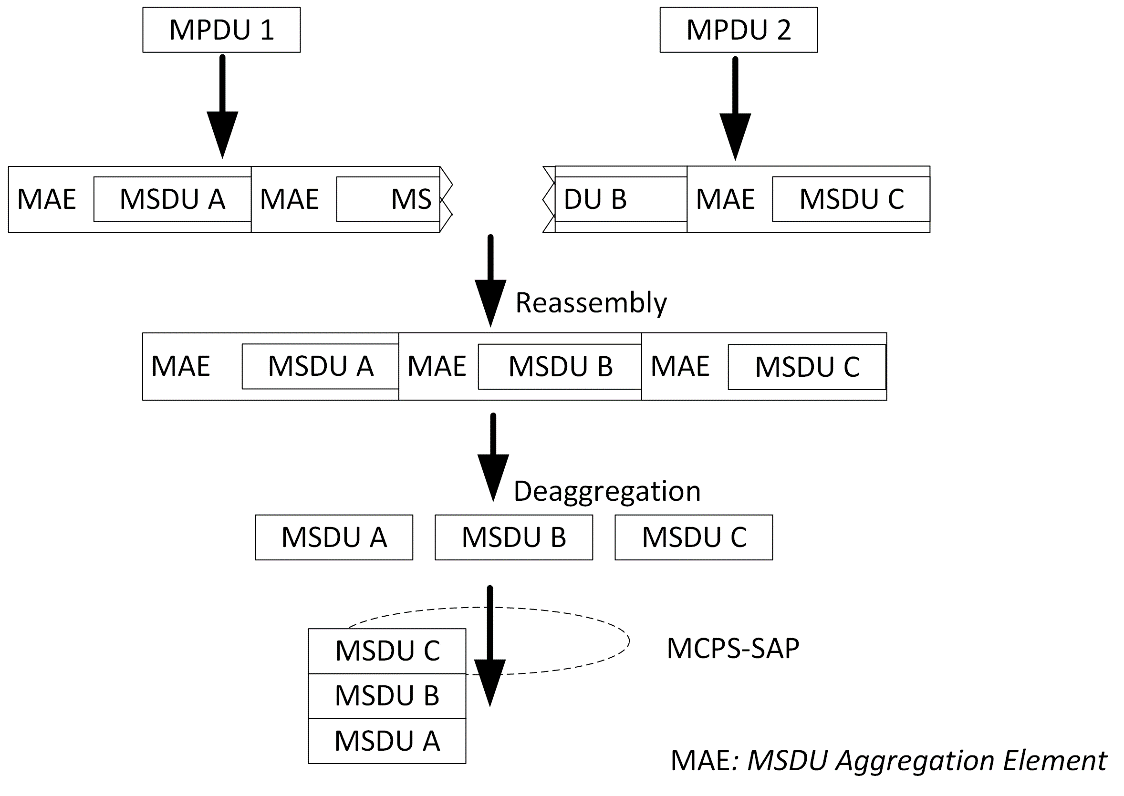


Figure 24: Reassembly and deaggregation

* 1. Protected transmission

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

If an MSDU was received through the MCPS-SAP with the *Protected* parameter set to TRUE, it shall be transmitted protected. MPDUs carrying protected MSDUs shall have the *Ack Request* bit set to 1. Protected MSDUs shall be buffered at the transmitter for retransmission if necessary.

Each device shall maintain an outgoing sequence number counter for transmissions towards every peer device with which it exchanges MPDUs. The sequence number is a 12 bit wide unsigned integer, which wraps to 0 after the highest possible value. Protected MPDUs shall be assigned a sequence number. The sequence number shall be incremented for each outgoing MPDU on a per receiver device basis. Rules for MPDUs carrying fragments are described in 5.5.

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

Each device shall maintain an incoming sequence number counter for receptions from each peer device with which it exchanges MPDUs.

MAC shall buffer MSDUs before handing it to the higher layer if any of the previous MSDUs is missing. The MAC shall pass MSDUs to higher layer in order.

Received MPDUs should be regarded as duplicates if they carry the same sequence and fragment number. Two received 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.

If the MPDU has the *ACK Request* bit set in the *Frame Control* element, 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.

* + 1. Single acknowledgement

The receiver of a protected MPDU (acknowledging device) shall acknowledge the successful reception of the protected MPDU by means of a single acknowledgement.

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

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

The acknowledging device shall ensure that the acknowledgement arrives at the transmitter of the acknowledged MPDU at most *macRetransmitTimeout* after the time of transmission of the acknowledged MPDU.

* + 1. Block acknowledgement

If the *capBlockAcknowledgment* was negotiated, a receiver may acknowledge successfully received MPDUs 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 a bitmap contained in a *Block Acknowledgment* element.

The source address in the frame containing the *Block Acknowledgment* element identifies the acknowledging device.

The receiver of multiple protected MPDUs from the same transmitting device may transmit a block acknowledgement via the *Block ACK Element* to that transmitting device instead of multiple single acknowledgments. The transmitter of a block acknowledgement shall ensure that the block acknowledgment arrives at the transmitter at most *macRetransmitTimeout* after the time of transmission of the first transmitted acknowledged MPDU.

The transmitter of a protected MPDU shall denote outstanding, i.e. unacknowledged MPDUs, as acknowledged after receiving a *Block ACK* element containing the sequence number of these MPDUs as acknowledged in the bitmap.

The transmitter of multiple MPDUs to the same device may request a block acknowledgment from the receiving device through transmitting a *Block Acknowledgment Request* element. Upon reception of a *Block Acknowledgement Request* element, the receiving device shall respond with a *Block Acknowledgment* element.

* + 1. Retransmission

A device shall retransmit a protected MPDU if its sequence number was not acknowledged after *macRetransmitTimeout*.The *macRetransmitTimeout* PIB attribute may be adjusted by the coordinator through the parameter management procedure described in 5.4.3. MPDUs of multiple fragments with one sequence number shall be retransmitted carrying the same fragments as before.

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.

* 1. Adaptive transmission

IEEE 802.15.13 PHYs are able to transmit frames under application of varying modulation and coding schemes (MCS). An MCS denotes some combination of modulation and error coding configuration. The specific definition of an MCS is dependent on the used PHY.

* + 1. Multi-rate MCS

A device may select the MCS and hence rate 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. Rate selection algorithms are out of scope of this standard.

For some frames, usage of specific modulation and coding schemes is mandatory (see 5.8.2).

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

* + 1. 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 shall be:

* *Association Request* and *Association Response* frames
* *Disassociation Notification* frames
* *Beacon* frames
* *Random Access* frames
* Control frames containing the *ACK* or *Block Ack Response* element
  + 1. MCS request feedback

An IEEE 802.15.13 device supporting the *capMcsRequest* 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.

* + - 1. Bitloading MCS request

If the *capHbPhy* was negotiated during association, a receiving 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 Bit Allocation Table (BAT) for future transmissions from the transmitter.

A BAT defines the modulation and coding formats to be used on each subcarrier and hence, how many bits are transported on the respective subcarrier. There are 32 distinct BAT IDs. IDs 0-7 are reserved for predefined BATs. The IDs 8-31 may be used for BATs defined at runtime (refer also to 12.3.2).

If an earlier requested BAT cannot be successfully decoded anymore due to channel changes, the receiving device shall request usage of a new and sufficiently robust BAT from the transmitter and signal the undecodable BAT as invalid. 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 future 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 that 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. The BAT ID corresponding to the *Updated BAT* field must be set to valid.

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 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 infer that runtime-defined BATs are consistent by deriving that fact from a successful frame reception.

Upon reception of a *BAT Request* element, a device shall not use BAT IDs that are marked as invalid in that *BAT Request* for transmissions to the source of the *BAT Request* element. The receiver of a *BAT Request* element should make use of the requested and latest BAT ID for transmissions towards the source of the *BAT Request* element until a newer *BAT Request* element is received from that device or the BAT ID is marked invalid.

* + 1. Multi-OFE channel feedback

Coordinators supporting the *capMultiOfeEstimation* capability shall be able to transmit multi-OFE pilots. All 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. Division numbering is defined in the respective PHY clauses per PHY. The transmission of multi-OFE pilots of a single PPDU transmitted over multiple OFEs shall happen synchronously at each OFE.

A device receiving a PPDU containing multi-OFE pilots shall be able to estimate the individual CSI between the transmitter of each pilot division and itself. The gathered CSI comprises time domain taps, which are described by the respective 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.

For each OFE, the device shall embed an *OFE Feedback Descriptor* element into the *Multi-OFE Feedback* element. Each *OFE Feedback Descriptor* shall in turn contain one or more *Tap Descriptor* elements. Each *Tap Descriptor* element shall correspond to a single identified receive tap in the time domain. For each tap, the signal strength and delay shall be calculated and quantized as described in 6.6.8.