IEEE 802.15
Wireless Specialty Networks

|  |
| --- |
| IEEE P802.15.13Updated text for clause 6 |
| Date: 2019-07-15 |
| Author: |
| Name | Affiliation | Address | Phone | Email |
| Kai Lennert Bober | Fraunhofer HHI |  |  | kai.lennert.bober@hhi.fraunhofer.de |
|  |  |  |  |  |
|  |  |  |  |  |

Abstract

# This document contains updated text for clause 6 of IEEE P802.15.13.

1. MAC frame formats

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

* 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.

* + 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 25.

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 and presented leftmost.



Figure : 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.

Some values for certain fields may be reserved. Values of fields that are “reserved” shall not be used in outgoing frames. If a MAC frame is received with one or more fields carrying a reserved value, that MAC frame shall be dropped.

* + 1. 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 26.

|  |  |
| --- | --- |
| **width** | **width** |
| field description | field description |
| **optional alternate description** |

Figure : 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 : 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 27.

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 27.

* 1. 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 6.4.1 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 28.

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

Figure : General MAC frame (MPDU) format

Individual fields are explained in the subsequent clauses.

* + 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 | LastFragment | Reserved |

Figure : 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 10. 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 6.4.1).

**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.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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. I.e. coordinator to device or device to coordinator. | Address of the designated receiver.*(= MSDU destination address).**Optionally short* | Address of the transmitting device.*(= MSDU source address).**Optionally short* | Coordinator address *(= OWPAN ID)*if short addressing is used. Otherwise, the field does not exist. |
| 1 | 0 | An MSDU originates from a device and is destined to a peer in the integrated LAN | Coordinator address *(= OWPAN ID)* | Address of the transmitting device.*(= 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. *(= MSDU destination address).* | Coordinator address *(= OWPAN ID)* | MSDU source address. |
| 1 | 1 | Reserved | - | - | - |

Table : 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 *Poll ACK* 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 *Receiver* and *Transmitter Address* fields of the MAC header. Short addressing is only allowed if the *To* and *From Backhaul* fields are both set to 0. If short addresses are used in the header, this field shall set to 1. Otherwise, it shall be set to 0.

Short addresses shall only be used if the frame does not carry a MSDU. For MSDUs, the corresponding address fields need to contain the full MAC addresses for identification of the source or the destination of the MSDU.

**Last Fragment:** In a data MPDU, this filed shall be set to 1 if the payload contains the last fragment of a fragmented (A-)MSDU. For MPDUs that do not contain fragments, the field should always be set to 1. For other frames than data frames, the field is reserved to be 0.

* + 1. Poll ACK Field

The *Poll ACK* 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).

|  |  |  |
| --- | --- | --- |
| **Bits: 0-6** | **7-13** | **14-15** |
| DeviceCompressedAddress | CompressedSequenceNumber | ACK |

Figure : Poll ACK field

**Device Compressed Address:** Bits 0 to 6 contain the compressed address of the device, which transmitted the packet. The device with such address is to be acknowledged by this *Poll ACK* 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 7 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.

* + 1. Address Fields (Receiver, Transmitter and Auxiliary Address)

The address fields indicate multiple addresses to the receiver of a MAC frame. These fields may comprise 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.

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 or the OWPAN ID.

The usage of short addresses is only allowed if the *To Backhaul* and *From Backhaul* bits are set to 1 each, i.e. in case a frame does not have to be bridged with the integrated LAN by the coordinator. For example, this is the case for control frames.

* + 1. Sequence Control Field

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

|  |  |
| --- | --- |
| **Bits: 0-3** | **4-15** |
| FragmentNumber | SequenceNumber |

Figure : 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.

* + 1. Auxiliary Security Header

This field contains security information. It is only present if the *Security Enabled* field of the frame control is set to 1. Otherwise it is not contained in the frame. It is further defined in clause 8

* + 1. Payload

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

* + 1. 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:

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

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

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

and

1. 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:

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

* 1. 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 32.

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

Figure : Data frame structure

The header fields contained in data MPDUs are described further in 6.2.

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 4.

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

Table : Data frame subtypes

For data frames with the *Subtype* 0000 (Null frames), the payload has a length of 0. These Null-Frames may be used to transmit MPDUs or corresponding PPDUs for various reasons. Null frames shall not be delivered to the MCPS-SAP but discarded instead after the receive procedure.

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 6.2.5.

* 1. 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 33.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Octets: 2** | **0/2** | **2/6** | **2/6** | **6** | **2** | **variable** | **variable** | **4** |
| Frame Control | PollACK | Receiver Address | Transmitter Address | Auxiliary Address | Sequence Control | AuxiliarySecurityHeader | Management Information | **FCS** |
| **MAC frame header (MHR)** | **Payload** |

Figure : Management frame structure

The header fields contained in management MPDUs are described further in 6.2.

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 5.

|  |  |  |
| --- | --- | --- |
| **Management frame** | **Subtype** | **Payload** |
| Association Request | 0000 | Association Request Element (6.6.1) |
| Association Response | 0001 | Association Response Element (6.6.2) |
| Disassociation Notification | 0010 | Disassociation Notification Element (6.6.3) |
| Alien Signal | 0011 | Alien Signal Element (6.6.17) |
| Poll  | 0100 | none (see clause 6.4.1) |
| Poll Request | 0101 | none (see clause 6.4.2) |
| Poll Response | 0110 | none (see clause 6.4.3) |
| Variable Element Container | 0111 | Variable Element Container Element (6.6.24) |
| Probe Request | 1000 | Probe Request Element (6.6.20) |
| Probe Response | 1001 | Probe Response Element (6.6.21) |
| Attribute Change Request | 1010 | Attribute Change Request Element (6.6.22) |
| Attribute Change Response | 1011 | Attribute Change Response Element (6.6.23) |
| *Reserved* | 1100 | - |
| *Reserved* | 1101 | - |
| *Reserved* | 1110 | - |
| *Reserved* | 1111 | - |

Table : 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 6.2.5.

* + 1. Poll Frame

The Poll frame is a lower MAC frame, which is sent by the coordinators to poll devices on downlink. The Poll frame does not have any field except the common management information.

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

* + 1. Poll Request Frame

The Poll request frame is a lower MAC frame, which is sent by the devices to establish the connection with the coordinator via the uplink. The Poll request frame does not have any field except the common management information.

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

* + 1. Poll Response Frame

The Poll response frame is a lower MAC frame, which is sent by the devices to response to the Poll frame from coordinators. The Poll response frame does not have any field except the common management information.

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

* 1. Control frames

Control frames aid the MAC and PHY at their operation. The MPDU structure of control frames is depicted in Figure 34.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Octets: 2** | **0/2** | **2/6** | **2/6** | **6** | **variable** | **4** |
| FrameControl | ACKInformation | ReceiverAddress | TransmitterAddress | AuxiliaryAddress | ControlInformation | **FCS** |
| **MAC frame header (MHR)** | **Payload** |

Figure : Control frame structure

The header fields of control MPDUs are a subset of the header fields in the general MAC frame format, described in clause 6.2.

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** | **Payload** |
| ACK | 0000 | Ack Element (6.6.10) |
| Block ACK | 0001 | Block Acknowledgment Element (6.6.12) |
| Block ACK Request | 0010 | Block Acknowledgment Request Element (6.6.11) |
| MCS Request | 0011 | MCS Request Element (6.6.13) |
| GTS Request  | 0100 | GTS Request Element (6.6.15) |
| GTS Allocation | 0101 | GTS Descriptor Element (6.6.7) |
| GTS Allocation List | 0110 | GTS Descriptor List Element (6.6.6) |
| Variable Element Container | 0111 | Variable Element Container Element (6.6.24) |
| Beacon | 1000 | Superframe Descriptor Element (6.6.4) |
| Random Access | 1001 | Random Access Element (6.6.19) |
| BAT Request | 1010 | BAT Request Element (6.6.14) |
| Multi-OFE Feedback | 1011 | Multi-OFE Feedback Element (6.6.8) |
| HCM Allocation | 1100 | HCM Allocation Element (6.6.16) |
| *Reserved* | 1101 | - |
| *Reserved* | 1110 | - |
| *Reserved* | 1111 | - |

Table : Control frame subtypes

* 1. 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 the variable element container. Table 7 lists the elements defined within this standard and their corresponding ID and definition clause.

* + 1. Association Request Element

Element ID: 0x0001

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

|  |  |
| --- | --- |
| **variable**  | **1 + a/b/c octets** |
| CapabilityList | SupportedMCS |

Figure : Association Request element

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

**Supported MCS:** *Supported MCS* element, indicating which MCS are supported by the requesting device.

* + 1. Association Response Element

Element ID: 0x0002

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

|  |  |  |  |
| --- | --- | --- | --- |
| **1 octet** | **2 octets** | **variable** | **1 + a/b/c octets** |
| StatusCode | Short Address | CapabilityList | Supported MCS |

Figure : 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 : Status codes of the Association Response element

**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.

**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 MCS:** The MCS supported by the coordinator.

* + 1. Disassociation Notification Element

Element ID: 0x0003

The *Disassociation Notification* element conveys information about the disassociation of a device from an OWPAN. It may be generated bei either a coordinator or an OWPAN member device in order to initiate disassociation of that device from the OWPAN.

|  |
| --- |
| **Octets: 1** |
| Reason Code |

Figure : 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 : Reason codes of the Disassociation Notification element

* + 1. Superframe Descriptor Element

Element ID: 0x0004

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Octets: 2** | **2** | **1** | **1** | **variable** |
| BeaconNumber | TotalSuperframeSlots | CAPSlotWidth | CAPSlots | VariableElementContainer |

Figure : Superframe Descriptor element

**Beacon Number:** Continuous number ideitifying each beacon frame and corresponding (hence current) superframe. The number is a wrapping integer as described in 5.2.2 and corresponds to the *macBeaconNumber* PIB attribute.

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

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

**CAP Slots:** The number of CAP slots included in the CAP of the current superframe.

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

* + 1. Capability List Element

Element ID: 0x0005

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** |
| BitmapWidth | CapabilityBitmap |

Figure : Capability List element

**Bitmap Width:** Contains the width in octets of the subsequent *Capability Bitmap* field. The *Capability Bitmap field* can thus include at most the capability with the ID *Bitmap Width* \* 8 – 1 since capabilitiy IDs start with 0 (see 7.4). The *Bitmap Width* 0 may be used to indicate an empty list of capabilities where needed.

**Capability Bitmap:** A bitmap indicating a set of capabilities as given in Table 45. In the bitmap each bit represents the capability corresponding to the capability ID given by the bit’s offset from the leftmost bit. Hence 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 set, 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).

* + 1. GTS Descriptor List Element

Element ID: 0x0006

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Bits: 0-7** | **4 ocets** | … | **4 octets** |
| GTSDescriptorCount (N) | GTSDescriptor 1 | GTSDescriptor N |

Figure : GTS Descriptor List element

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

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

* + 1. GTS Descriptor Element

Element ID: 0x0007

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

|  |  |  |
| --- | --- | --- |
| **2 octets** | **1 bit** | **15 bits** |
| GTSStart Slot | ImmediatelyValid | GTSLength |

Figure : GTS Descriptor element

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

**Immediately Valid:** If the field is set to 1 the GTS becomes effective in the same superframe, the GTS Descriptor element was received. Otherwise, the GTS becomes effective in the following superframe.

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

* + 1. Multi-OFE Feedback Element

Element ID: 0x0008

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** | **2 octet** | **Variable** |
| Number of OFEs (N) | TAP format | NoiseFloor | OFE feedbackdescriptor element 1 | … | OFE feedbackdescriptor element N |

Figure : 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.

**Noise Floor**: The measured noise floor [dBmV]

|  |  |  |
| --- | --- | --- |
| **Value** | **Strength**  | **Delay** |
| 0000 |

|  |  |
| --- | --- |
| Bits: | 10 |
| Signal overnoise floor: | -20 dB |
| step: | 0.1 dB |

 |

|  |  |
| --- | --- |
| Bits: | 14 |
| First tap: | 0 ps |
| step: | 100 ps |

 |
|  |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

 |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

 |
|  |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

 |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

 |
|  |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

 |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

 |
| 0010-1111 | Reserved | Reserved |

Table 10: 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

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. Its structure is given in Figure 44.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Bits: 0-2** | **4-7** | **8-13** | **14-15** | **variable** |
| PilotSymbolNumber | DivisionId | NumberofTaps (M) | *Reserved* | TapDescriptor1 | … | TapDescriptorM |

Figure : 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 Id:** Specifies the pilot division. This is for example the Hadamard coding or the subcarrier spacing and shift as indicated in the PPDU header. The division id can take values from 0 to 31 and is defined in each respective PHY clause.

**Number of Taps:** Specifies the number of subsequent *Tap Descriptor* elements, referred to by M.

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

The *Tap Descriptor* element includes the information about a single tap. Its structure is given in Figure 45.

|  |  |
| --- | --- |
| **variable** | **variable** |
| Strength | Delay |

Figure : Tap Descriptor element

**Strength:** Signal over noise ratio of the 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 arriving tap in the whole *Multi-OFE Feedback* element shall be 0.

* + 1. MSDU Aggregation Element

Element ID: 0x0009

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** |
| DestinationMACAddress | SourceMACAddress | MSDULength | MSDU | VariablePadding |

Figure : 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 number of padded octets can be derived based on the *MSDU Length* field. 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.

* + 1. ACK Element

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

|  |  |
| --- | --- |
| **bits 0-11** | **bits 12-15** |
| SequenceNumber | Reserved |

Figure : ACK element

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

* + 1. 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** |
| FirstSequenceNumber | reserved |

Figure : Block Ack Request element

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

* + 1. Block ACK Element

The *Block ACK* element may be used by ta receiving device to signal the successful reception of multiple MPDUs to the transmitter in a single frame.

|  |  |  |
| --- | --- | --- |
| **4 Bits** | **12 Bits** | **1-16 octets** |
| BitmapWidth | FirstSequenceNumber | ACKBitmap |

Figure : Block Ack element

**Bitmap Width:** This field determines the width of the *ACK Bitmap* field in integer octets and hence the maximum number of included acknowledgments. The actual width of the bitmap in octets 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* + 1octets 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 leftmost bit, i.e. the bit to be processed first by the definition given in 6.1.1, corresponds to the first sequence number, as given in the *First Sequence Number* field. The rightmost 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* = 1 and the *First Sequence Number* = 320 would look as follows if the sequence numbers 320, 321, 322, 324, 325, 326, 327, 328, 329, 330, 332, 333, 334, 335 were successfully received:

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

* + 1. 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 : 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 the respectice PHY.

* + 1. BAT Request Element

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Bits: 0-23** | **24-28** | **29-31** | **32-34** | **35-39** | **Variable** |
| ValidBAT Bitmap | UpdatedBAT | FECBlockSize | FECCodeRate | Reserved | BATGroup 1 | … | BAT Group N |

Figure : BAT Request element

**Valid BAT Bitmap:** Specifies the BATs requested to be valid. The first bit of the bitmap corresponds to the BAT ID 8, while the last (i.e.. rightmost) bit corresponds to the BAT ID 31. A bit set to 1 indicates that the BAT is valid and 0 indicates that the BAT is not invalid, i.e. shall not be used by the transmitter anymore.

**Updated BAT:** Specifies the ID of the runtime-defined BAT to be updated. Only values 8-31 are allowed. The value 0 indicates that no new BAT is updated. This may be the case if only validity information is signalled in the *BAT Request* element. Values 1-7 are reserved.

**FEC Block Size:**

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

Table : FEC block sizes for the HB-PHY

**FEC Code Rate:** Specifies the requested FEC coding rate. Valid values and corresponding code rates are listen in Table 12.

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

Table : 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.

A *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 | LoadedBits |

Figure : BAT Group element

The structure of the *BAT Group* element is shown in Figure 52. It has the following fields:

**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

* + 1. GTS Request Element

The *GTS Request* element is transmitted by a device in order to inform the coordinator about the state of its MSDU queues.

**Queue Size Priority 0 … 7:** The number of octets currently queued in the corresponding outbound queue of the device. The actual value shall be rouded to the closes possible value.

* + 1. HCM Allocation Element

The *HCM Allocation* element is sent by the coordinator to allocate one or more HCM rows to a device.

|  |
| --- |
| **2 octets** |
| HCM Mask |

Figure 55: HCM Allocation element

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

* + 1. 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** | **0/6 octets** | **0/6 octets** |
| SignalPower | Decodable | SameMACMode | OWPAN IDClash | reserved | ForeignOWPANID | ForeignDevice Address |

Figure : 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 0 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 Address of the foreign transmitting device. If the address is unknown, the field shall be set to the broadcast short address.

* + 1. Supported MCS Element

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1 octet** | **1 octet** | **a/b/c** | **…** | **a/b/c** |
| PHYID | MCSCount (N) | MCS 1 | MCS N |

Figure : Supported MCS element

**PHY ID:** The ID of the PHY for which the *PHY MCS e*lement is defined.

|  |  |
| --- | --- |
| **Value** | **PHY** |
| 0 | PM-PHY (clause 10) |
| 1 | LB-PHY (clause 11) |
|  |  |
| 2-255 | *Reserved* |

Table : PHY IDs

**MCS 1…N:** A PHY-specific supported MCS. The format depends on the value of the *PHY ID* field.

* + 1. 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 | RandomAccessInterval | CapabilityInformation | OWPANID | SupportedRates | Country | ExtendedSupportedRates |

Figure : Random Access element

**Timestamp:** The Timestamp field allows synchronization between the devices in an OWPAN. When coordinators prepare to transmit a Random access frame, the coordinator timer is copied into the frame’s timestamp field. Devices associated with a coordinator accept the timing value in any received random access frames, 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.

* + 1. 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 device. Coordinators that receive Probe Requests use the information to determine whether the device can join the network. To make a successful connection, the 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.

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

|  |  |  |
| --- | --- | --- |
| **2** | **variable** | **variable** |
| OWPANID | SupportedRates | Extended Supported Rates |

Figure : 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. The format of supported rates is illustrated as follows:



Figure Supported rates format

**Extended supported rates:** Extended Supported Rates was standardized to handle more than eight data rates. The format of extended supported rates is illustrated as follows:



Figure Extended supported rates format.

* + 1. Probe Response Element

If a Probe Request encounters a network with compatible parameters, the coordinator 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 Figure 41.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Octets: 2** | **?** | **2** | **2** | **variable** | **variable** |
| Timestamp | BeaconInterval | Capability Information | OWPANID | SupportedRates | ExtendedSupportedRates |

Figure : 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 device, while others are optional.

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

* + 1. 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.

|  |  |
| --- | --- |
| **2 octets** | **variable** |
| AttributeID | NewValue |

Figure : 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 43.

**New Value:** The new value to assign to the attribute. The field format is to be deducted from the Table 43.

* + 1. 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** |
| AttributeID | NewValue | Status |

Figure : 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 43.

**New Value:** The new value assigned to the attribute. The field format is to be deducted from the Table 43.

**Status:** The result of the former attribute change request. Possible values are described in Table 14.

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

Table : Status codes for the attribute change request result.

* + 1. 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. The element ends with a 2-octet zero value enabling receivers to detect the end of the type length value group of sub-elements.

|  |  |
| --- | --- |
| **N\*(4+variable) octets** | **2 octets** |
| Type1 | Length1 | Element1 | … | TypeN | LengthN | ElementN | 0x0000 |

Figure : Variable Element Container element

**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 7. The type field after the last contained element shall have the value 0x0000, indicating that the list has ended.

**Length 1 … N:** This field contains the length of the subsequent element in octets and is 2 octets wide.

**Element 1 … N:** The contained element indicated by the *Type* field as defined in the respective clause.

**0x0000:** Termination type.