IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Comment resolution | | | | |
| Date: 2018-02-04 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Solomon Trainin | Qualcomm |  |  | strainin@qti.qualcomm.com |
|  |  |  |  |  |

Resolution of CIDs 1069, 1072, 1081, 1083, 1101, 1112, 1163, 1164, 1188, 1189, 1217, 1254, 1255, 1256, 1257, 1258, 1259, 1260, 1285, 1380, 1381, 1464, 1656, 1692, 1722, 1858, 1859, 1867, 1940, 1943, 2113, 2114, 2135, 2173, 2174, 2201, 2202, 2203, 2204, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2231, 2235, 2244, 2260, 2265, 2400 is presented

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1069 |  | 10.6.2.1 | It is not clear whether the MAC level fragmentation is allowed in combination with the EDMG segmentation. From the frame formats, it is likely to be not allowed. | Clarify whether the MAC level fragmentation is allowed in combination with the EDMG segmentation. |

**Proposal: Revised**

Discussion:

There is no subclause 10.6.2.1 as in the comment and no page number is presented. I expect that the comment is related to 10.62.1 of 10.62 (EDMG segmentation and reassembly operation). The suggested clarification already exists in the subclause 10.62.1 “A pair of STAs that use segmentation and reassembly for a particular TID shall not employ MSDU fragmentation for this TID.” Suggest modifying to add reference.

***TGay Editor: modify (Draft 1.1)***

A pair of STAs that use segmentation and reassembly for a particular TID shall not employ MSDU fragmentation defined in 10.5 for this TID.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1072 | 34.17 | 9.3.1.9.7 | "... A-MPDUs with a length that is not less than indicated by Maximum A-MPDU Length Exponent (Table 3)" Can A-MPDUs exceed Maximum A-MPDU Length? | Change to "... A-MPDUs with a length equal to the length indicated by Maximum A-MPDU Length Exponent (Table 3)" |

**Proposal: Revised**

Discussion:

Proposed text is better than existing.

***TGay Editor: modify in* Table 1 — RBUFCAP encoding for the EDMG Compressed BlockAck variant *P53 (Draft 1.1) as follows:***

Indicates that the recipient’s memory has enough space to receive A-MPDUs with a length equal to the length indicated in Maximum A-MPDU Length Exponent (Table 5)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1081 | 112.23 | 10.13.7 | 802.11ah introduced "S-MPDU" as a general term. So, there is no need to add a new terminology for EDMG case. | Use S-MPDU for the whole case. |
| 2400 | 35.00 | 9.3.1.9.8 | S-MPDU is introduced in 11ah, is defined in REVmd\_D 0.5. 11ax also replaces single MPDU to S-MPDU | Change "single MPDU" to "S-MPDU". Do it through the whole draft. |

**Proposal: Accept**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1083 | 196.12 | 10.62.1 | It looks like the MPDU sequence number is the same with the original sequence number. The relation between the original sequence number should be explained somewhere in section 9. | As in comment. |

**Proposal: Reject**

Discussion:

Sequence Control field that covers the relevant sequence number is defined in 9.2.4.4.1 (Sequence Control field)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1112 | 31.07 | 9.2.4.4.1 | The Start of MSDUn subfield is set to one to indicate that the MPDU contains the first segment of an MSDU. It is set to zero otherwise. | Either change the logic to "The Start of MSDUn subfield is set to zero to indicate..." or remove the n from the subfield name |

**Proposal: Reject**

Discussion:

The comment and the proposal are unclear, and the suggested change does not have any technical impact.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1163 | 196.19 | 10.62.2 | "until it is successfully delivered to the recipient STA" -- this cannot be guaranteed | append "or the entire MSDU is discarded" |

**Proposal: Accept**

It is about P236L21 in the Draft 1.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1164 | 196.16 |  | Segmentation looks like the existing fragmentation mechanism. | Either explain what segmentation offers that fragmentation does not, or remove it entirely.  If kept, it might help readers understand which mechanism to use by adding an introductory para explaining when it is appropriate to use the mechanism. |
| 1722 | 195.29 | 10.62 | The Segmentation feature appears to be an enhancement to the existing (very similar) Fragmentation feature. The enhancement is the negotiation of the size of the 'segments', using ADDTS. This enhnacement could be added to the existing Fragmentation, without creating a whole new, parallel, and nearly identical facility. This would also help with numerous small technical errors around wording like "partitioning an MSDU into a sequence of MPDUs." | Remove the new EDMG segmentation facility, and add the negotiation of maximum 'segment/fragment' size as a negotiated extension of the Fragmentation facility. |

**Proposal: Revised**

Discussion: There are few issues in the current spec that SAR resolves.

* MPDU size is limited by max A-MSDU size of 7935 octets that the size is limited by existent FCS of 4 octets.
* The fragmentation is not supported by HT Immediate BlockAck

SAR keeps max MPDU size of 7935 octets while max MSDU is no more limited by this size. The MSDU size negotioated over the SAR establishment. SAR is also an extension of HT Immediate BlockAck. There are substantial changes that are not related to the existent Fragmentation definition.

***TGay Editor: modify in P235L37 (Draft 1.1)***

The segmentation and reassembly mechanism allows a STA to receive at the MAC SAP a MSDU of a size that is optimal for upper layers and application and is not limited to protocol data unit size (MTU). The MSDU is delivered to the MAC SAP of the destination STA through MSDU segments carried within MPDUs transmitted over the wireless link using HT Immediate BlockAck mechanism. The max MSDU size is negotiated between communicating peers over the SAR establishment.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1188 | 35.05 | 9.3.1.9.8 | A two bit field should be enumerated as integers when its values are defined, for stylistic similarity to the baseline, and to avoid ambiguity of endianness. | Flip a coin and replace AckType subfield values 00, 10, 11, 01 with 0,1,3,2 or 0,2,3,1.  Order by increasing values.  Consider not introducing a reserved value in the middle of the enumeration. |

**Proposal: Revised**

Discussion: Implement as per comment

***TGay Editor: modify Table 2 —AckType subfield definition (Draft 1.1) as follows***

|  |  |  |  |
| --- | --- | --- | --- |
| **AckType subfield value** | **TID subfield value** | **Presence of Block Ack Starting Sequence Control subfield and Block Ack Bitmap subfields** | **Context of a Per TID Info subfield in a Multi-STA BlockAck frame** |
| 0 | 0-15 | Present | Block acknowledgment context:  Sent as a response to MPDUs in an A-MPDU that solicit an immediate block acknowledgement or to a BlockAckReq frame. |
| 1 | 0-15 | Not present | Acknowledgment context:  Sent as a response to an MPDU or EDMG single MPDU that solicits an immediate acknowledgment. |
| 2 | 0-15 | Not present | All-ack context:  Sent as a response to an A-MPDU that solicits an immediate response and all MPDUs contained in the A-MPDU are received successfully. |
| 3 | N/A | N/A | Reserved |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1189 | 35.08 | 9.3.1.9.8 | "The bitmap length is equal to 2(3+ BlockAck Bitmap Subfield Length)."    Units should be provided because the name is poorly chosen. Normally one would expect the units of "bitmap length" to be bits. | Add "octets" before the period. |

**Proposal: Accept**

***TGay Editor: modify in P54L8 (Draft 1.1)***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1217 | 119.25 |  | "EDMG Multi-TID BlockAck variant and Multi-TID BlockAckReq variant" -- what is this supposed to mean? Both frames are valid? I think this text assumes that "variant" refers to the encoding of the variant, and not the format of the frame itself - which is wrong. | If this is not supposed to allow the non-EDMG Multi-TID BlockAckReq, delete "and Multi-TID BlockAckReq variant""    Same possible issue in the previous line with the compressed variant. |

**Proposal: Reject**

Discussion:

I think that the commenter didn’t pay attention that the sentence is about BlockAck variant and BlockAckReq variant so there is no contradiction at all. There is no EDMG Multi-TID BlockAckReq variant defined and the existent Multi-TID BlockAckReq variant is in use. Both mentioned frames are valid for EDMG STA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1254 | 30.15 | 9.2.4.4.1 | There is no figure to illustrate of the sequence control field (when segmentation and reassebmly is not supported) | Include the figure for sequence control field (when segmentation and reassebmly is not supported) |

**Proposal: Reject**

Discussion:

As already defined in the text “If segmentation and reassembly is not supported, the Sequence Control field consists of two subfields, the Sequence Number and the Fragment Number, and its format is illustrated in Figure 9-4. If segmentation and reassembly is supported, the Sequence Control field has the format illustrated in Figure 4. See the Figure 9-4 in the basic spec.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1255 | 30.04 | 9.2.4.4.1 | The bit position of the MSDU Sequence Number field in Figure 4 should be B(MSDU Modulo+2) if the starting bit position of this field is B2 | Modify the diagram with correct bit position |
| 1256 | 30.04 | 9.2.4.4.1 | The starting bit position of the MPDU Sequence Number field in Figure 4 should be B(MPDU Modulo+3). | Modify the diagram with correct bit position |
| 1257 | 30.04 | 9.2.4.4.1 | the length of the MPDU Sequence Number field in Figure 4 is illustrated as B(MSDU Modulo+2), should it be B(MPDU Modulo+3) | Correct the diagram |

**Proposal: Reject**

Discussion:

Let’s take an example of MSDU Modulo = 5. In this case the MSDU SN field is [B2 – B6], it is 5 bits allocated for MSDU Modulo. The MPDU SN field is [B7-B15] – it is 9 bits allocated for MPDU SN. 9+5 = 14 equal to the MSDU modulo + MPDU modulo. So, no issues to fix are found.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1285 | 31.07 | 9.2.4.4.1 | "The Start of MSDUn subfield is set to one to indicate that......"  Change "set to one to " to " set to 1 to" for keeping the same style as REVmd | As in comment.  Much places need to change, if this comment is approved. |

**Proposal: Reject**

Discussion:

Propose keeping on the editor’s responsibility to resolve the style issues.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1380 | 31.04 | 9.2.4.4.1 | It is not clear what is the structure of this field (reserved bits), when MSDU Modulo+MPDU Modulo<14 | Add reserved bits |
| 1381 | 32.14 | 9.3.1.8.3 | It is not clear what is the structure of this field (reserved bits), when MSDU Modulo+MPDU Modulo<14 | Add reserved bits |

**Proposal: Revised**

Discussion: The commented issue is resolved per CID 1735 (IEEE 802.11-18/0278r2) that defined “The sum of the values of the MSDU Modulo subfield and the MPDU Modulo subfield shall be equal to fourteen.”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1464 | 30.12 | 9.2.4.4.1 | Are there ever going to be cases when a transmitting and receiving STA will support segmentation and reassembly, but they will not employ it? If, so then the text here needs to change and we need a way to signal when it is employed. Also, do we need text to say segmentation and reassembly is not used for broadcast frames? | Clarify how the sequence control field is used if necessary. |

**Proposal: Reject**

Discussion:

Segmentation and reassembly is established for a particular TID using an ADDBA Request and ADDBA Response frame exchange that includes the SAR Configuration element and the support of SAR may be rejected as defined in 10.25.2 Setup and modification of the block ack parameters, with addition of CID 2245 (IEEE 802.11-18/0336r2)

SAR exploits the HT Immediate Block Ack that does not provide support for broadcast frames.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1656 | 197.30 | 10.63 | Limiting to TIDs that correspond to the AC that gains the access essentially make Multi-TID A-MPDU the 2-TID A-MPDU for SU (without considering TSIDs), which are very constrained. It is reasonable to allow the 11ax-style Multi-TID aggregation rule which is to allow any higher AC MPDU as long as the TXOP boundary is maintained. | Please revise |

**Proposal: Revised**

Discussion:

It is already resolved as proposed by the commenter in resolution to CID’s 1956, 2272 (IEEE 802.11-18/0137r2)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1692 | 196.08 | 10.62.1 | "of different MSDUs" cannot be because is the segmentation of a single MSDU | replace "different MSDUs" with "different MPDUs" |

**Proposal: Reject**

Discussion:

It is allowed to transmit multiple MSDUs and the reordering buffer may contain multiple MSDU’s. There is no place in the definition that limits the flow to single MSDU. Different MSDU’s are identified by MSDU SN.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1858 | 31.13 | 9.2.4.4.1 | As described in 10.24.2 the block Ack agreement is between the originating STA and the intended receiving STA. It should be noted in the text. | Change "the recipient's" to "the intended recipient's" |
| 1859 | 31.17 | 9.2.4.4.1 | As described in 10.24.2 the block Ack agreement is between the originating STA and the intended receiving STA. It should be noted in the text. | Change "the recipient's" to "the intended recipient's" |

**Proposal: Revised**

***TGay Editor: modify as in the Proposed Change in P49L13, P49L17, P51L19, and P51L23 (Draft 1.1)***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Line** | **Clause** | **Comment** | **Proposed Change** |
| 1867 | 99.24 | 24 | 9.71 | The A-MPDU format for call rules on EOF padding subframes. VHT PPDUs and PPDUs set the maximum length of the A-MPDU based on pre-EOF padding. The maximum length of an A-MPDU in an EDMG PPDU should be based on the same criteria. | Change "A-MPDU in an EDMG PPDU" to "A-MPDU pre-EOF padding in an EDMG PPDU" |

**Proposal: Reject**

Discussion:

1. The PPDU length is the same for all users
2. Each user gets a length field in header-B covering the length of the actual data of the PSDU (Table-43)
3. Per each user, there is PHY padding to the maximum length of all users (30.5.9.4.4 MU PPDU padding and space-time streams mapping
4. From a MAC perspective, there is no need for any EOF padding to compensate for MU PPDU size, all is done at the PHY level.
5. Max A-MPDU length is per user (indicated by Maximum A-MPDU Length Exponent). A-MPDU length includes EOF padding, so the RX buffer can accommodate entire A-MPDU of maximal length. Header B contains PSDU length that is equal to the A-MPDU length.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1940 | 86.01 | 9.4.2.268 | Actual transmission or receiving in a TDD slot the STA is assigned to depends on the STA role. Existing language is that AP STA transmits in TX Slot and non-AP STA transmits in RX Slot. The existing definition requires separate TX/RX rules for AP and non-AP STA that misleads implementation of the lower MAC. Propose to clarify definition of the TDD slot to unify the behavioral rule. | Modify definition of TDD slots that instead of TX TDD and RX TDD make it clear that each slot is TX for one STA and RX slot for another STA thus unify definition in 10.36.6.2.2 to avoid double rules covered AP STA and non-AP STA. Submission is ready to present |

**Proposal: Revised**

Discussion:

Definition in 9.4.2.268 and behavioral rules in 10.36.6.2.2 in the IEEE P802.11ay/D1.1, February 2018 are modified following the Proposed Change, still the 10.3.2.10 Acknowledgement procedure is not aligned with the new definition of 9.4.2.268. The modification below resolves the issue

***TGay Editor: modify as follows (Draft 1.1)***

**10.3.2.10 Acknowledgement procedure**

P132L38

A STA operating in an SP with TDD channel access (see 10.37.6.2.2), upon reception of a frame that requires acknowledgement shall transmit an Ack or BlockAck frame at the start of the earliest occurring TDD slot the STA is assigned to, with Bitmap and Access Type Schedule field (Table 17)

of the TDD slot set to TX, and with slot category of the TDD slot set to Basic TDD slot, as indicated in the TDD Slot Schedule element (see 9.4.2.268).

In addition a DMG AP shall transmit an Ack or BlockAck frame upon reception of a frame from a non-AP and non-PCP STA that requires acknowledgement and with the To DS subfield equal to 1.,

P133L6

A STA operating in an SP with TDD channel access (see 10.37.6.2.2) has AckTimeout interval value equal to the duration from the PHY-TXEND.confirm primitive of the current frame to the end of the earliest occurring TDD slot the addressed recipient of the MPDU is assigned to, with Bitmap and Access Type Schedule field (Table 17) of the TDD slot set to RX, and with slot category of the TDD slot set to Basic TDD slot, as indicated in the TDD Slot Schedule element (see 9.4.2.268).

***TGay Editor remove next paragraph***

**10.25.7.5 Generation and transmission of BlockAck frames by an HT STA or DMG STA**

P150L7

A STA operating in an SP with TDD channel access (see 10.37.6.2.2), that receives a PPDU that contains a BlockAckReq frame in which the Address 1 field matches its MAC address during either full-state or partial-state operation shall transmit a PPDU containing a BlockAck frame starting at the earliest occurring TDD slot the STA is assigned to, with Bitmap and Access Type Schedule field (Table 17) of the TDD slot set to TX, and with slot category of the TDD slot set to Basic TDD slot, as indicated in the TDD Slot Schedule element.

***TGay Editor remove next paragraph***

P150L19

A STA operating in an SP with TDD channel access (see 10.37.6.2.2), that receives an A-MPDU that contains one or more MPDUs in which the Address 1 field matches its MAC address with the Ack Policy field equal to Normal Ack (i.e., implicit block ack request) during either full-state operation or partial-state operation shall transmit a PPDU containing a BlockAck frame starting at the earliest occurring TDD slot the STA is assigned to, with Bitmap and Access Type Schedule field (Table 17) of the TDD slot set to TX, and with slot category of the TDD slot set to Basic TDD slot, as indicated in the TDD Slot Schedule element.

***TGay Editor remove next paragraph***

**10.25.7.7 Originator’s behaviour**

P154L10

A DMG STA originator operating in an SP with TDD channel access (see 10.37.6.2.2) shall not transmit more than one MPDU or A-MPDU that has an Ack Policy of Normal Ack to another STA per each occurrence of a TDD slot the STA is assigned to, with Bitmap and Access Type Schedule field (Table 17) of the TDD slot set to TX, and with slot category of the TDD slot set to Basic TDD slot, as indicated in the TDD Slot Schedule element.

***TGay Editor remove next paragraph***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 1943 | 28.01 | 8.3.5.12.2 | Terms of Primary Channel Offset, secondary, secondary1, and secondary2 are illustrated in Figure 1 and Figure 2 but are not defined. | Provide definition of the terms. Definition is provided as a table in separate submission |

**Proposal: Revised**

Discussion:

The CID is resolved in IEEE 802.11-18/0377r2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2113 | 30.12 | 9.2.4.4.1 | "The Sequence Control field is 16 bits in length and its format depends whether segmentation and reassembly (see 10.62) is supported between the transmitting and receiving STA. If segmentation and reassembly is not supported, the Sequence Control field consists of two subfields, the Sequence Number and the Fragment Number, and its. The format of the Sequence Control field is illustrated in Figure 9-4. If segmentation and reassembly is supported, the Sequence Control field has the format illustrated in Figure 4. The Sequence Control field is not present in Control frame" The The Sequence Control field is not stricly dependent on whether seg and reassmebly is supported or not. It should be rephrased to say the The Sequence Control field formta in case of segmenataion and reassmebly is supported | as suggested |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2202 | 30.12 | 12 | 9.2.4.4.1 | It seems unnecessary to complicate the Sequence control field structure with the changes necessary for segmentation and reassembly. Add a clause for Sequence Control field structure for segmentation and reassembly. This would simplify the specification and not complicate the legacy text. | Do not added these changes to 9.2.4.4.1, create a new clause 9.2.4.4.x for the segmentation and reassembly case. |

**CID’s 2113 2202**

**Proposal: Revised**

Discussion:

Agree to add new subclause as follows:

***TGay Editor: modify as follows (Draft 1.1)***

P48L11

The Sequence Control field is 16 bits in length and consists of two subfields, the Sequence Number and the Fragment Number, and it is illustrated in Figure 9-4. The Sequence Control field is not present in Control frames.

*Insert at P49L4*

**9.2.4.4.x Sequence Control field in SAR**

If SAR is supported as defined in 10.62, the Sequence Control filed consists of four subfields Start of MSDUn, End of MSDUn, MSDU Sequence Number, and MPDU Sequence Number if segmentation and reassembly is supported (10.62)., The format is illustrated in Figure 4.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2114 | 30.12 | 9.2.4.4.1 | Is Segmentation and Reassmebly feature applied to HT, VHT, HE? The frame format here does not indicate this is applied to EDMG only but the reference is made to EDMG section | clarify |

**Proposal: Reject**

Discussion:

Provided reference to 10.62 completely resolves the commented issue. Conditions to support SAR are defined there that the format is part of it. The optional feature as defined is EDMG specific and capability to support it is provided in the EDMG capabilities that are EDMG specific and not applicable to other amendments.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2173 | 30.16 | 9.2.4.4.1 | "Figure 4" is not the correct format | Change "Figure 4" to the correct format |
| 2174 | 31.05 | 9.2.4.4.1 | "Figure 4" is not the correct format | Change "Figure 4" to the correct format |

**Proposal: Reject**

Discussion:

It is editor’s decision to keep global enumeration of new added figures in the draft, it is not specific to the Figure 4. Should be resolved of the entire draft at the time decided by the editor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2201 | 32.06 | 9.3.2.8.3 | It seems unnecessary to complicate the Compressed BlockAckReq variant with the changes necessary for segmentation and reassembly. Added a clause for Compressed BlockAckReq variant for segmentation and reassembly, would simplify the specification and not complicate the legacy text. | Do not added these changes to 9.3.1.8.3, create a new clause 9.3.1.8.x for the segmentation and reassembly case. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1258 | 32.04 | 9.3.1.8.3 | The bit position of the MPDU Starting Sequence Number filed in Figure 5 -Bar Information field format should be B(MSDU Modulo+2) if the starting bit position of this field is B2 | Modify the diagram with correct bit position |
| 1259 | 32.04 | 9.3.1.8.3 | The starting bit position of t MPDU Starting Sequence Number filed in Figure 5 -Bar Information field formatshould be B(MPDU Modulo+3). | Modify the diagram with correct bit position |
| 1260 | 32.04 | 9.3.1.8.3 | the length of the MPDU Starting Sequence Number filed in Figure 5 -Bar Information field format is illustrated as B(MSDU Modulo+2), should it be B(MPDU Modulo+3) | Correct the diagram |

**Proposal: Revised**

Discussion:

*There is no need to change the BA Starting Sequence Control field format of the Compressed BlockAckReq variant. The existent BA Starting Sequence Control field of the Compressed BlockAckReq variant is fine to be used to shift MSDU under SAR. The SAR format modifications should be removed.*

***TGay Editor: modify as follows (Draft 1.1)***

P51L6

*Remove sentence:* In case of a block ack agreement that does not use segmentation and reassembly (see 10.62),

P51L13

*Remove text that starts with* “In case of a block ack agreement that uses segmentation and reassembly…” and ends by … element contained in the recipient’s ADDBA Response frame (see 10.25.2).”

*Remove* **Figure 5 —BAR Information field format**

P143L40

Remove:

“and the size of the MSDU Starting Sequence Number and MPDU Starting Sequence Number subfields of the BAR Information field (see Figure 5)”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2203 | 93.10 | 9.6.5.2 | It seems unnecessary to complicate the ADDBA Request frame format with the changes necessary for segmentation and reassembly. Add a clause for the ADDBA request frame format for segmentation and reassembly. This would simplify the specification and not complicate the legacy text. | Do not added these changes to 9.6.5.2, create a new clause 9.6.5.x for the segmentation and reassembly case. |

**Proposal: Revised**

Discussion:

*There is no need to change the BA Starting Sequence Control field format of the ADDBA Request frame. The existent BA Starting Sequence Control field of the ADDBA Request frame is fine to be used to set the MSDU SN under SAR. Reference to the Figure 5 should be removed.*

***TGay Editor: modify as follows (Draft 1.1)***

P143L38

***TGay Editor remove:*** “which serves to confirm the size of the MPDU Sequence Number and MSDU Sequence Number subfields in the Sequence Control field and the size of the MSDU Starting Sequence Number and MPDU Starting Sequence Number subfields of the BAR Information field (see Figure 5).”

P120L6

***TGay Editor modify:***

The Starting Sequence Number subfield of the Block Ack Starting Sequence Control field (see Figure 9-27) contains the sequence number of the first or next (in the case of a renegotiation of a block ack agreement) MSDU to be sent under this block ack agreement. The Fragment Number subfield is set to 0.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2204 | 107.07 | 10.3.2.11.2 | Add a reference to the normative section where this is defined - 9.2.4.4.1 or is it 9.4.2.250.7 table 5 | Add a reference to the normative text. |

**Proposal: Revised**

Discussion:

*The commented text is a NOTE, the reference is added as suggested*

***TGay Editor: modify as follows (Draft 1.1)***

P134L14

NOTE—Under a block ack agreement using segmentation and reassembly (10.62), the MPDU sequence number is represented by a 2MPDU Modulo counter and the MSDU sequence number is represented by

2MSDU Modulo counter (9.2.4.4.1), where MPDU Modulo and MSDU Modulo are as defined in the SAR Configuration element (9.4.2.266).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2212 | 195.31 | 10.62.1 | Intro need to be clearer. The flow of MSDU to MSDU segments, the transmission of each MSDU segment in a MPDU, the reception of each of the MPDU, from which each MSDU segment is received, and then the reassembly of the MSDU segments into the original MSDU which is then delivered to the MAC must be clear. The use of the term MSDU to refer to an MSDU segment is very confusing and should be avoided. | Rework the paragraph so that the manner of moving the MSDU into MSDU segments and then back again to the original MSDU is clear. |
| 2215 | 196.17 | 10.62.2 | There is no statement the MSDUs are segmented such that the segmented MSDU will be transmitted in a maximum sized MPDU, except for the final MDDU segment which can be any size equal to or less than the max MPDU. | Clearly state how an MSDU is segmented in the MSDU segments. Clearly state that the size of an MSDU segment is chosen so that when it is carried by the MPDU the MPDU size is at the maximum MPDU size, except for the MPDU which contains the final MSDU segment which can be smaller. |

**Proposal: Revised**

Discussion:

*The required text to clarify the behavior of segmentation and reassemble already exists in 10.62.2 Segmentation operation and in 10.62.3 Reassembly operation respectively. Add references to the commented text.*

***TGay Editor: modify as follows (Draft 1.1)***

P235L34

EDMG STA that supports segmentation and reassembly may segment large MSDUs received at the MAC SAP into MSDU segments that are transmitted into MPDUs (10.62.2). These MSDU segments are reassemble at the receiving STA to recreate the original MSDU (10.62.3).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2213 | 196.03 | 10.62.1 | How a particular TID chosen? This seems unclear. Who does the choosing and how is it checked that the chosen TID is not already in use on one of the STAs. I think more definition of this process is necessary. | Please define how a TID is assigned to a particularly MSDU fragment stream. How are they reused? Who keeps track so different MSDU fragment streams don't use the same TID and hence mix the data. |

**Proposal: Reject**

Discussion:

SAR does not change the TID assignment and the TID assignment is not part of the SAR. The TID assignment is covered in the basic spec 5.1.1.2 Determination of UP (IEEE Std 802.11-2016)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2214 | 196.12 | 10.62.1 | Figure 110 is very confusing, it appears the x-axis is time but reassembly is shown before reception of all segments, also acks are not shown. The concept of transmission/retransmission is not clear and there is nothing about out of order reception/reordering | Rework the figure so that the procedure is clearly defined. |

**Proposal: Revised**

Discussion:

*Unitdata primitives should be appended to the Figure 110 to address commenter concerns.*

*Reference to the HT Immediate BA should be provided in the subclause 10.62.2 Segmentation operation to address commenter concern about transmission rules.*

***TGay Editor: modify as follows (Draft 1.1)***

*In the Figure 127 —Example of the segmentation and reassembly procedure*

*on the Originator Upper Layer each box of MSDU modify as follows:*



*on the Recipient Upper Layer each box of the MSDU and box of the MSDU Segment 4 modify as follows:*



***TGay Editor: modify as follows (Draft 1.1)***

P235L39

… carried within MPDUs transmitted over the wireless link (10.24.7 HT-immediate block ack extensions)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2216 | 196.17 | 10.62.2 | It does not make sense that an MSDU segment can be a complete MSDU, as if it can be one there is no reason to segment the MSDU. Hence an MSDU segment may not comprise all or a portion of an MSDU. | Change the sentence:  An MSDU segment may comprise all or a portion of an MSDU.  To be:  An MSDU segment comprises a portion of an MSDU. |

**Proposal: Reject**

Discussion:

*MAC establishes SAR enabled BA agreement per TID. MAC is not responsible for MSDU sizes arrived at MAC SAP. Any size of MSDU can arrive at the established BA even of single MPDU size.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2217 | 197.05 | 10.62.3 | All the discussion of reception of the MSDU fragments which refer to them as MSDUs is very confusing. There has got to be a better way to describe the removal of the fragment MSDU's header and then the concatenation of the fragments MSDU's date with the previously received fragments to form the original MSDU, without calling each fragment an MSDU. | Rework the text so that it is clear that each MDPU contains a fragment MSDU and that the fragment MSDUs are reassembled to create the original MSDU and once the MSDU has been reassembled it is passed to the MAC. The current description is lacking. |
| 2218 | 197.08 | 10.62.3 | The discussion about de-encapsulated and decryption is not very clear. I assume that this is done to the MSDU fragment before it is inserted into the MPDU, and hence these operations need to be reversed prior to obtaining the MSDU fragment which is used to be concatenated with the other MSDU fragments to form the original MDSU. | Clarify the process, so that it is clear that an MSDU fragment is treated by the PHY in the same manner that the PHY treats an MSDU, hence the MDSU fragment may be encapsulated or encrypted prior to insertion into the MPDU and then once the MPDU is received the PHY must de-encapsulated and decrypt the MSDU fragment received in the MPDU to obtain the original MSDU fragment and then that these original MSDU fragments must then be assembled to form the original MSDU. |
| 2219 | 196.19 | 10.62.2 | There is no discussion in the Segmentation operation section about encapsulation or encryption. This needs to be added or a statement made that the PHY can conduct additional operation on the MSDU fragment, just as it would with a regular MSDU. | Clarify the process, so that it is clear that an MSDU fragment is treated by the PHY as if it is an MSDU and may be encapsulated or encrypted. |

**Proposal: Revised**

Discussion:

*There is some misconception presented in the comments in relation to the MAC and PHY layering. The PHY has nothing to do with the MSDU that appears only on the MAC SAP. The existent text clearly states that “An MSDU segment is carried within an MPDU”.*

*As defined, the SAR is applicable for EDMG STA only (10.62.1). The EDMG STA uses the GCMP encryption that is defined per MPDU, so no need for any additional explanation in this relation.*

*There are few places in the commented text that I suggest editing:*

***TGay Editor: modify as follows (Draft 1.1)***

P237L2

A recipient STA reassembles an MSDU comprised of one or more MSDU segments according to the Start of MSDUn, End of MSDUn, MSDU Sequence Number, MPDU Sequence Number indications of the MPDU the segment is contained as described in 10.25.7.6.2. The recipient MAC shall reassemble the MSDU by concatenating MSDU fragments contained in decrypted frame body of the MPDUs with equal MSDU sequence numbers starting from MPDU with Start of MSDUn subfield equal to one and ending with MPDU End of MSDUn subfield equal to one in increasing order of MPDU SN. The MA-UNITDATA.indication is used by the recipient MAC to release the reassembled MSDU at MAC SAP.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2231 | 31.05 | 9.2.4.4.1 | SAR could be supported but BA agreement with SAR is not enabled (e.g. rejected by recipient) | change supported to enabled |

**Proposal: Accept**

***TGay Editor: modify in P49L5 (Draft 1.1) Figure 4***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Line** | **Clause** | **Comment** | **Proposed Change** |
| 2235 | 61.08 | 8 | 9.4.2.250.6 | If a STA can rx n TIDs in an AMPDU it should be able to generate multi-TID BA for this number of TIDs. Not clear why 2 separated capabilities.    Is the intention of EDMG Multi-TID BA agreement subfield to support the case that a STA does not support rx of Multi-TID AMPDU but can receive/ack A-PPDU with multi-TIDs?    The EDMG Multi-TID Aggrement Support subfield should be a receiver capability | remove 'aggregate' on L11. Clarify the intention of the EDMG Multi-TID BlockAck Support subfield |

**Proposal: Revised**

*Discussion: It is solved in resolution of CIDs 1956, 2272 in doc 18/0137r2 that only EDMG Multi-TID Aggregation Support is used to indicate the capability of supporting multi-TID aggregation and the relevant BA and BAR formats*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2244 | 85.12 | 9.4.2.266 | The MPDU buffer size field was also included in the mandatory Block Ack Parameter Set    The requirement in 10.24.2 indicates that with SAR enabled the MPDU buffer size cannot be greater than 1024 so the Buffer Size in Block Ack Parameter set should be sufficient | Remove this field |

**Proposal: Revised**

Discussion:

There is a reason to keep the filed that the buffer size subfield in the Block Ack Parameter Set  
may contain different value. Resolution of the CID requires more changes to resolution of CID2245 already resolved as part of 18/0336r2 See below the complete resolution.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2245 | 116.03 | 10.24.2 | It is not clear whether a recipient can accept a BA agreement by setting SAR enabled to 0 (or SAR config element not present) and accepts/modifies the Block Ack Parameter set in ADDBA response | should not allow such behavior  change to "A recipient may reject the BA agreement ..." |

Proposal: **Revised**

Discussion:

A recipient may reject the ADDBA request by setting the Status code in the ADDBA response frame to anything but SUCCESS. From the other side the originator may reject the Block Ack agreement by issuing DELBA frame to the recipient. There are multiple parameters that may be negotiated over Block Ack agreement so, no need is seeing to require rejection of Block Ack agreement establishment in case of recipient rejection support of SAR. Some clarification is needed to the case the agreement is established w/o SAR support however the SAR configuration element is present in the ADDBA request frame.

***TGay editor (Draft 1.1)***

***In P143L43 append***

A recipient may reject the SAR configuration by setting the SAR Enabled field within the SAR Configuration element in the ADDBA Response to zero or by not including the SAR Configuration element in the ADDBA Response frame.

Size of the originator transmission window shall be not greater than value in the Buffer Size subfield of the Block Ack Parameter Set field of the ADDBA Response frame in case that the Recipient rejects the SAR configuration and responds with Status code set to SUCCESS in the ADDBA response frame.

The originator shall set to zero Sequence Number field in the MPDU it transmits first under the established Block Ack agreement in case that the Recipient rejects the SAR configuration and responds with Status code set to SUCCESS in the ADDBA response frame,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2260 | 196.12 | 10.62.1 | The Start/End of MSDUn flag is set differently from the definition in 9.2.4.4 | change the flag in the figure |

Proposal: **Accept**

Discussion: the commenter is right, polarities of the field shall be opposite.

***TGay editor (Draft 1.1)***

In P236 Figure 127 in all appearances of Start of MSDUn and End of MSDUn replace 0 by 1 and vice versa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Comment** | **Proposed Change** |
| 2265 | 116.13 | 10.24.2 | Not clear why it is 2^(MPDU\_Modulo-2) instead of 2^(MPDU\_Modulo-1) | change to 2^(MPDU\_Modulo-1) |

Proposal: **Reject**

Discussion The buffer size shall be less than half of the SN range to allow shifting of the window start for a window size. It is important to keep the opportunity to shift the window as part of the recovery mechanism defined in 10.24.7.3

**References:**

1. IEEE P802.11ay/D1.1
2. IEEE Std 802.11-2016