IEEE P802.11
Wireless LANs

|  |
| --- |
| Resolution of SAR Related CIDs |
| Date: 2018-11-07 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Oren Kedem | Intel |  |  | oren.kedem@intel.com |
| Carlos Cordeiro  | Intel |  |  | carlos.cordeiro@intel.com |
| Solomon Trainin | Qualcomm |  |  | strainin@qti.qualcomm.com |
|  |  |  |  |  |

Abstract

This submission proposes resolutions to CIDs 3003, 3004, 3005, 3006, 3282, 3283, 3336, 3337, 3345,

3395, 3421, 3427, 3464, 3468, 3469, 3717 and 3718

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3003 | 10.26.5.3 | grammar issue reference "case segmentation and reassembly | Change " in case" to " if segmentation" | Accepted  |
| 3004 | 10.26.5.3 | grammar issue reference "case segmentation and reassembly | Change "in case" to " when segmentation" | Accepted |
| 3005 | 10.26.5.3 | grammar issue reference "case segmentation and reassembly | Change " in case" to " if segmentation" | Accepted |
| 3006 | 10.26.5.3 | grammar issue reference "case segmentation and reassembly | Change " in case" to " when segmentation" | Accepted |

**10.26.5.3 Scoreboard context control during full-state operation**

*Change text as follow*

b) For each received Data frame that is related with a specific full-state operation HT-immediate
block ack agreement, the block acknowledgment record for that agreement is modified as follows,
where SN is the value of the Sequence Number subfield of the received Data frame if
segmentation and reassembly is not used and is the value of the MPDU Sequence Number subfield
of the received Data frame when segmentation and reassembly is used:

c) For each received BlockAckReq frame that is related with a specific full-state operation
HTimmediate block ack agreement that is not a protected block ack agreement, the block
acknowledgment record for that agreement is modified as follows, where SSN is the value from
the Starting Sequence Number subfield of the received BlockAckReq frame if segmentation
and reassembly is not used and is the value from the MPDU Starting Sequence Number subfield
of the received BlockAckReq frame when segmentation and reassembly is used:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3282 | 5.1.5.1 | Figure 5-1 has been modified in 802.11ak-2018. Please incorporate changes from 802.11ak-2018 accordingly. | As in comment. | Accepted |
| 3283 | 5.1.5.1 | Figure 5-2 has been modified in 802.11ak-2018. Please incorporate changes from 802.11ak-2018 accordingly. | As in comment. | Accepted |

**5.1.5.1 General**

*Replace Figure 5-1 and Figure 5-2 with the following*





|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3336 | 10.72,1 | Figure 147 - Example of the segmentation and reassembly procedure is very helpful and does help clarify how segmentation and reassembly work. However, the figure could be clearer and seems to have some errors in it. 1) the use of the term "Originator" seems to be related to the BA Originator, if so please specifically state so it in the text. It is possible to interpit "originator" to be the high layer device which generated the PDU that is the source of the "large" MSDU, which results when the PDU crosses the MAC SAP in to the 802.11 MAC. This is further compounded by the use of "Upper Layer" which seems to be above the MAC, which means to me it is outside the scope of 802.11 and hence the use of "Originator" to refer to this is very confusing, if "Originator" is a BA Originator, which is an 802.11 entity. Also adding to the confusion is that an MSDU only exists on the 802.11 side of the MAC SAP, typically the packet is refered to as a PDU before it crosses the MAC SAP into the 802.11 MAC. This same confusion applies to the use of "Recipient:" 2) The figure provides no insite as to how the created MSDUs containing the segments are delt with in the PHY. Are they sent as individual PPDUs, can they be agragated into a single PPDU? 3) Showing the first N+2 segment in the Recipient Mack Layer Reordering Buffer is confusing as it was not received in the example, marking it as lost seem to be a poor way to show it was not receive, It would be better in my view to not show the first N-2 segment, simply adding a note to explain why it isn't there. As it is currently shown it looks like MSDU segment N+2 is received twice and then nothing is done with the first instance. 4) It would also improve clarity, to show the received MPDUs, from which the MSDU segments are taken from to fill the Reordering Buffer, which once full regenerates the "large" MSDU. | Improve the figure as described in the commnet | Revised  |

**Discussion**

The use of the term "Originator" seems to be related to the BA Originator, if so please specifically state so it in the text. It is possible to interpit "originator" to be the high layer device which generated the PDU that is the source of the "large" MSDU, which results when the PDU crosses the MAC SAP in to the 802.11 MAC. This is further compounded by the use of "Upper Layer" which seems to be above the MAC, which means to me it is outside the scope of 802.11 and hence the use of "Originator" to refer to this is very confusing, if "Originator" is a BA Originator, which is an 802.11 entity.

*[Revised]*

*The wording Originator/Responder was removed from the Upper Layer or MAC Layer and BA was added to the Originator/Responder were*

Also adding to the confusion is that an MSDU only exists on the 802.11 side of the MAC SAP, typically the packet is refered to as a PDU before it crosses the MAC SAP into the 802.11 MAC. This same confusion applies to the use of "Recipient:"

*[Revised]*

*Upper layer unitdata was changed from MSDU to PDU*

The figure provides no insite as to how the created MSDUs containing the segments are delt with in the PHY. Are they sent as individual PPDUs, can they be agragated into a single PPDU? 3)

*[Rejected]*

*The PHY handling is our of the scope of the figure which is describing a MAC mechanism. MPDU can be delivered by any form of allowed PPDU (PPDU, A-PPDU or any)*

Showing the first N+2 segment in the Recipient Mack Layer Reordering Buffer is confusing as it was not received in the example, marking it as lost seem to be a poor way to show it was not receive, It would be better in my view to not show the first N-2 segment, simply adding a note to explain why it isn't there. As it is currently shown it looks like MSDU segment N+2 is received twice and then nothing is done with the first instance. 4) It would also improve clarity, to show the received MPDUs, from which the MSDU segments are taken from to fill the Reordering Buffer, which once full regenerates the "large" MSDU.

*[Revised]*

*It is important to show non-received segment to demonstrate the retransmission and reordering procedure. Clarity was provided to better show that N+2 was not received by the responder*

**10.72.1 General**

*Replace Figure 150 with the following*



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3337 | 10.72,1 | The statement that a pair of STAs that use SAR shall not employ fragmentation is an awakward way to prevent the use of fragmentation. In my view it would be simpler to insure that the maximum MSDU size negotiated between communicating peers during SAR establishment is smaller than the fragmenation thresholds. (in both STAs). | As in the comment | Rejected Fragmentation and Segmentation cannot be utilized in parallel since both features use same fields for different purpose. If Segmentation is used, the Maximum MSDU Size is neggotiaed, the higher MSDU Size is the greater is the efficiency of the mechanism hence the expected size is greater than the common (7920B). There is no reason to set the maximum size smaller than fragmentation threshold which is likely to be smaller than default MSDU size (7920B).  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3345 | C.3 | Thre is no MIB variable associated with SAR feature. | Add MIB variable for SAR enablement | Rejected 802.11 includes many optional features which don’t have an associated MIB which is not necessary also in this case. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3395 | 9.4.2.265 | maximum supported MSDU size' should be maximum segmented MSDU size derived from table 11 | change to 'maximum segmented MSDU size derived from table 11' | Rejected Table 11 describes the configuration of the Segmentation and Reassembly Capability subfields which Maximum supported MSDU is one of them. Reference is the correct field, there is no necessity to point the reader to the specific table.  |

**9.4.2.265 Segmentation and Reassembly (SAR) Configuration element**

*Change text as follow*

The MSDU Buffer Size field indicates the number of buffers available for this particular TID. Each buffer

is capable of holding the number of octets equal to the maximum supported MSDU size as indicated in the

Segmentation and Reassembly Capability field of the STA’s EDMG Capabilities element.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3421 | 10.26.5.7.1 | The originator shall not transmit MSDUs that are lower than the current transmission window (i.e., SN < WinStartOJ).' Does this requirement apply to the case that MPDU modulo >9 (i.e. p189 L36)? The rx is really just using MPDU SN to tell old and new packets | add exception to this sentence in case MSDU\_modulo <5 | Accepted  |

**Discussion**

**10.26.2 Setup and modification of the block ack parameters**

When a block ack agreement with segmentation and reassembly is established, an EDMG originator may

change the size of its transmission window if the value in the MPDU Buffer Size or MSDU Buffer Size

fields within the SAR Configuration element contained in an ADDBA Response frame are larger than the

corresponding field values transmitted in the ADDBA Request frame. If the value in the MSDU Buffer

Size or the MPDU Buffer Size fields in the ADDBA Response frame is smaller than the corresponding

value in the ADDBA Request frame, the originator shall change the size of its transmission window

(WinSizeO or WinSizeOJ) so that it is not larger than the value of the field in the ADDBA Response frame

and not larger than 1024. In addition, the MPDU Buffer Size subfield shall not exceed the value of 2(MPDU

Modulo-2) and shall be equal or larger than the value of the resulting maximum segmented MSDU size as

indicated by the Maximum Segmented MSDU Exponent subfield in the EDMG Capabilities element

divided by the maximum MSDU size as indicated in Table 9-19 or as agreed between the peers via an

ADDTS Request and Response frame exchange for the respective TID. The originator may set the MSDU

Buffer Size subfield to a value greater than 2MSDU Modulo – 2 only if MPDU Modulo subfield is set to a value

greater than 9; in this case, the recipeint may receive multiple MSDUs with identical SNs into its receive

buffer.

**10.26.5.7 Originator’s behavior
10.26.5.7.1 General**

*Change the fifth paragraph as follows*

The originator may transmit QoS Data frames with a TID matching an established block ack agreement in

any order provided that their sequence numbers lie within the current transmission window. The originator

may transmit an MPDU with a sequence number that is beyond the current transmission window (SN >WinStartO + WinSizeO – 1), in which case the originator’s transmission window (and the recipient’s

window) is moved forward. The originator should not transmit MPDUs that are lower than (i.e., SN <

WinStartO) the current transmission window. Under a block ack agreement with segmentation and

reassembly, the originator may transmit QoS Data frames with a TID matching an established block ack

agreement in any order provided that their MPDU\_SN and MSDU\_SN lie within the current transmission

window. The originator may transmit an MPDU with a MSDU\_SN that is beyond the current transmission

window (MSDU\_SN > WinStartOJ + WinSizeOJ – 1), in which case the originator’s transmission window

(and the recipient’s window) is moved forward. The originator shall not transmit MSDUs that are lower

than the current transmission window (i.e., SN < WinStartOJ) in case MSDU Modulo subfield was set to a value smaller than 5.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3427 | 10.26.5.8 | If an MPDU with SN < MPDU\_SN with start of MSDUn set to 1, has not been ack'ed, can WinStart\_OJ change always? | reword the bullet to account for the missing MPDU before the start of the currently acked MSDU | Rejected MSDU SN is forwarded in case whole MSDU was received correctly as indicated by below bullets |

**10.26.5.8 Maintaining block ack state at the originator**Under a block ack agreement with segmentation and reassembly, the originator shall update WinStartOJ and
WinEndOJ at the arrival of a BlockAck frame. At each subsequent MPDU sent with End of MSDUn
subfield set to one in the Sequence Control field, WinStartOJ shall be set to MSDU\_SN+1 and WinEndOJ
shall be set to WinStartOJ + WinSizeOJ – 1 if following conditions are met:

⎯ The MPDU is indicated as acknowledged in the BlockAck bitmap; and
⎯ All preceding MPDUs starting from MPDU\_SN with Start of MSDUn subfield set to one in the Sequence Control field are indicated as successfully delivered.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3464 | 10.72.1 | While I appreciate the additional introductory text in 10.72.1 about the segmentation feature, it still doesn't explain why this feature is not simply an extension/enhancement to the existing fragmentation feature. The whole point is still to split a large MSDU into parts which are transmitted in individual MPDUs. The enhancements are good, but they could be done by extending the existing feature (which would possibly allow use by non-EDMG STAs, too). | Merge the new aspects of the EDMG segmentation and reassembly feature into the existing fragmentation/defragmentation feature, and consider whether it needs to be limited to EDMG STAs or could be generalized. | Rejected Section 10.72.1 explain well the usage of segmentation. The usage segmentation does with the Sequence Control field cant coexist with fragmentation. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3468 | 9.4.2.265 | "The MSDU Modulo subfield indicates the number of bits to be allocated to the length of the MSDU Modulo subfield." Huh? How can the subfield indicate the number of bits allocated to itself? |  | Revised  |

**9.4.2.265 Segmentation and Reassembly (SAR) Configuration element**

*Change text as follow*

The MSDU Modulo subfield indicates the number of bits to be allocated to the length of the MSDU Sequence Number subfield within the Sequence Control field.The MPDU Modulo subfield indicates the number of bits to be allocated to the length of the MPDU Sequence Number subfield within the Sequence Control field. The sum of the values of the MSDU Modulo subfield and the MPDU Modulo subfield is equal to fourteen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3469 | 10.26.1 | "Under a block ack agreement using segmentation and reassembly, operations on MSDU Sequence Number and MPDU Sequence Number are performed modulo MSDU\_Modulo and modulo MPDU\_Modulo respectively (see 10.72), where MSDU\_Modulo and MPDU\_Modulo are as defined in the SAR Configuration element. Operations on the MPDU sequence number and on the MSDU sequence number are performed modulo 2^MPDU\_Modulo and 2^MSDU\_Modulo, respectively." How can both be true? It is modulo the value, or modulo 2^the value? Also, the reference to 10.72 is confusing, as there is no mention of modulo operations in 10.72 | Align these sentences. It appears the latter is true,based on other text, later in 10.26. if that's correct, align these sentences to that operation, and update the reference to be to those sections. | Revised  |

**10.26.1 Introduction**

Under a block ack agreement using segmentation and reassembly, operations on MSDU Sequence Number

and MPDU Sequence Number are performed modulo 2MSDU\_Modulo and modulo 2MPDU\_Modulo

respectively (see 10.72), where MSDU\_Modulo and MPDU\_Modulo are as defined in the SAR

Configuration element. Operations on the MPDU sequence number and on the MSDU sequence number

are performed modulo 2MPDU\_Modulo and 2MSDU\_Modulo, respectively. Comparisons between MPDU sequence

number and MSDU sequence number are circular modulo 2MPDU\_Modulo and 2MSDU\_Modulo, respectively, i.e.,

the sequence number space is considered divided into two parts, one of which is “old” and one of which is

“new,” by means of a boundary created by adding half the sequence number range to the current start of

receive window (modulo 2MPDU\_Modulo and 2MSDU\_Modulo, respectively).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3717 | 9.2.4.4.2 | Why do you need the MSDU Sequence Number subfield when you have the Start of MSDUn bit and End of MSDUn bit? If the MPDU sequence numbers are assigned sequentially throughout MSDUs, the Start of MSDUn bit and the End of MSDUn bit will give enough information to reassemble and reorder the MPDUs into MSDUs. | Delete the MSDU Sequence Number subfield and simplify the reassembly mechanism. | Reject MSDU Sequence Number subfield allow the receiver to track lost MSDUs, (the higher MSDU\_Modulo is the tracking robustnace is increased). Hence there is motivation to preserve the MSDU Sequence Number and not replace it only with Start of MSDUn / End of MSDUn indications. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Comment** | **Proposed change** | **Resolution**  |
| 3718 | 9.2.4.4.2 | Why not unify the Sequence Control field structure among EDMG STAs? If the fragmentation is forbidden for EDMA STAs, that can be done. | Change the 9.2.4.4.1 subclause heading to "Sequence Control field structure for non-EDMG STAs". Change the 9.2.4.4.2 subcualse heading to "Sequence Control field structure for EDMG STAs". Change Figure 2 so that when SAR is not used, the Start of MSDUn and End of MSDUn bits are set to reserved. | **Rejected** Fragmentation is not forbidden for EDMG station and might be used by some of the implementations. |

**SP/M:** Do you accept the resolutions given in this document ?