IEEE P802.11
Wireless LANs

|  |
| --- |
| TGac SB 01 Comment Resolution |
| Date: 2013-07-11 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Jonathan Segev | Intel | 94 Em Hamoshavot rd. PTK, Israel | +972-54-2403587 | jonathan.segev@intel.com  |
| Robert Stacey | Intel |  | +1-503-724-0893 | robert.stacey@intel.com |

Abstract

This submission included proposed resolutions to CIDs 10071, 10072, 10073, 10074 and 10075.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CID |  |  | Comment | Proposed Resolution |
| 10071 | 139.3 | 9.7.10 | I assume that non-VHT STAs should not include CH\_BANDWIDTH\_IN\_NON\_HT and DYN\_BANDWIDTH\_IN\_NON\_HT in RXVECTOR. | Add sentence after first paragraph on page 139: "Non-VHT STAs shall not include CH\_BANDWIDTH\_IN\_NON\_HT or DYN\_BANDWIDTH\_IN\_NON\_HT in the Clause 18 RXVECTOR" |

**Resolution**:

REJECTED – The commenter is correct, a non-VHT STA should not include CH\_BANDWIDTH\_IN\_NON\_HT and DYN\_BANDWIDTH\_IN\_NON\_HT in RXVECTOR as is stated by the cited sentence.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CID |  |  | Comment | Proposed Resolution |
| 10072 | 140.25 | 9.7.11.3 | The min() operation in min(VHT-MCS,7) is redundant, since VHT-MCS is 0, 1, 2 or 3. | Remove min() operation |

**Discussion:**

Clause 9.7.11.3 deals with additional rate selection constraints when transmitting using specific BW and in a manner that is backwards compatible with an HT STA as HT STA uses NSS that is less than or equal to 4.

***“9.7.11.3 Additional rate selection constraints for VHT PPDUs***

*The following apply for a STA that transmits a VHT PPDU with a number of spatial streams (NSS) less than*

*or equal to 4:*

*— If the channel width of the PPDU is equal to CBW20 or CBW40, then the STA should not use a*

*<VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0, 1, 2 or 3 and the HT MCS with value min(VHT-MCS, 7) + 8(NSS – 1) is marked as unsupported in the Rx MCS bitmask of the HT capabilities element of the receiver STA. “*

The MCS indices for HT and VHT were created using different indexing methods, for HT MCSs the indices were created such that each NSS continues from the highest index of the previous NSS with the highest QAM such that the NSS is essentially encoded into the HT MCS index whereas for VHT the method used was to create a VHT tuple and have the VHT MCS describe the modulation and coding and the NSS describe separately.

The highest modulation used for VHT which is also used by HT STAs is QAM 64 which takes the highest VHT MCS of 7, to conversion from VHT MCS tuple to HT MCS is therefore :

min(VHT MCS, 7)+8(NSS-1)

Indeed removing the ‘min’ will cause no harm.

**Recommendation**:

REVISED – Since VHT-MCS is restricted to 0 to 3, the min() function always evaluates to the VHT-MCS term. Remove the min() function using the editing instructions in <this document> under CID 10072.

Modify text of section 9.7.11.3 as follows:

**“9.7.11.3 Additional rate selection constraints for VHT PPDUs**

The following apply for a STA that transmits a VHT PPDU with a number of spatial streams (NSS) less than or equal to 4:

* If the channel width of the PPDU is equal to CBW20 or CBW40, then the STA should not use a <VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0, 1, 2 or 3 and the HT MCS with value ~~min(~~VHT-MCS~~)~~ + 8(*NSS* – 1) is marked as unsupported in the Rx MCS bitmask of the HT capabilities element of the receiver STA.”
* If the channel width of the PPDU is equal to CBW80, CBW160 or CBW80+80, then the STA should not use a <VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0 or 1 and both the HT MCS values min(2(VHT-MCS),7) + 8(NSS – 1) and min(2(VHT-MCS) + 1, 7) + 8(NSS – 1) are marked as unsupported in the Rx MCS bitmask of the HT capabilities element of the receiver STA.”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CID |  |  | Comment | Proposed Resolution |
| 10073 | 140.25 | 9.7.11.3 | In <VHT-MCS,NSS>, no subscript is used. In min(VHT-MCS,7)+8(N\_SS-1), SS is in subscript | Use same notation in both places |

**Discussion:**

Prologue:

**“9.7.11.3 Additional rate selection constraints for VHT PPDUs**

The following apply for a STA that transmits a VHT PPDU with a number of spatial streams (NSS) less than

or equal to 4:

— If the channel width of the PPDU is equal to CBW20 or CBW40, then the STA should not use a

<VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0, 1, 2 or 3 and the HT MCS with value min(VHT-MCS,7) + 8(*NSS* – 1) is marked as unsupported in the Rx MCS bitmask of the HT capabilities

element of the receiver STA.”

— If the channel width of the PPDU is equal to CBW80, CBW160 or CBW80+80, then the STA should

not use a <VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0 or 1 and both the HT MCS values

min(2(VHT-MCS),7) + 8(NSS – 1) and min(2(VHT-MCS) + 1, 7) + 8(NSS – 1) are marked as

unsupported in the Rx MCS bitmask of the HT capabilities element of the receiver STA.”

Note:

The commenter points out the inconsistency of using the NSS parameter within this paragraph. However note that throughout D5.1 inconsistent usage of it exists.

**Resolution**:

REVISED – Change N\_SS to NSS.

Modify text of 9.7.11.3 as follows:

— If the channel width of the PPDU is equal to CBW20 or CBW40, then the STA should not use a

<VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0, 1, 2 or 3 and the HT MCS with value min(VHT-MCS,7) + 8(*~~NSS~~* NSS - 1) is marked as unsupported in the Rx MCS bitmask of the HT capabilities element of the receiver STA.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CID |  |  | Comment | Proposed Resolution |
| 10074 | 140.31 | 9.7.11.3 | The min() operation in min(2(VHT-MCS),7) is redundant since VHT-MCS is 0 or 1 | Remove min() operation |

**Resolution:**

REVISED – The min() function is not needed. Remove with editing instruction in <this document>.

**“9.7.11.3 Additional rate selection constraints for VHT PPDUs**

The following apply for a STA that transmits a VHT PPDU with a number of spatial streams (NSS) less than

or equal to 4:

— If the channel width of the PPDU is equal to CBW20 or CBW40, then the STA should not use a

<VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0, 1, 2 or 3 and the HT MCS with value min(VHT-MCS,7) + 8(*NSS* – 1) is marked as unsupported in the Rx MCS bitmask of the HT capabilities

element of the receiver STA.”

— If the channel width of the PPDU is equal to CBW80, CBW160 or CBW80+80, then the STA should

not use a <VHT-MCS, NSS> tuple if the VHT-MCS is equal to 0 or 1 and both the HT MCS values

~~min(~~2(VHT-MCS)~~,7)~~ + 8(NSS – 1) and ~~min(~~2(VHT-MCS) + 1~~, 7)~~ + 8(NSS – 1) are marked as

unsupported in the Rx MCS bitmask of the HT capabilities element of the receiver STA.”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CID |  |  | Comment | Proposed Resolution |
| 10075 | 140.57 | 9.7.11.3 | Why are the last two rows of the second column blank? | If HT MCS 0 and 1 are indicated as unsupported, shouldn't VHT-MCS 0 and 1 in 40 MHz be listed as not used?Likewise, If HT MCS 2 and 3 are indicated as unsupported, shouldn't VHT-MCS 2 and 3 in 40 MHz be listed as not used? |

**Discussion**:

Table 9-4a brings a tabulated example of the behaviour described in 9.7.11.3 for both the rules applied to CBW20 & CBW40 and the rules applied to CBW80, CBW160 and CBW80+80. The first 4 lines refer to the former while the last 2 refers to the latter case.

There is no specifying for the CBW20 and CBW40 for last two rows as this case was described in the first 4 rows of the table. The commenter is correct however that this is somewhat misleading.



**Resolution:**

REVISED – In the first column, the HT MCSs included is ambiguous since they are listed modulo 8. Modify the figure so that each row is an unambiguous list of unsupported HT MCSs. Replace the Table 9-4a with that provided in <this document> for CID 10075.

**Table 9.4a – Examples of rate selection for VHT PPDUs**

|  |  |  |
| --- | --- | --- |
| **HT MCSs that are marked as unsupported** | **<VHT-MCS, NSS> tuples that are not used for CBW20 and CBW40** | **<VHT-MCS, NSS> tuples that are not used for CBW80, CBW160 and CBW80+80** |
| 0, 8, 16 | <0, 1>, <0, 2>, <0, 3> | - |
| 1, 9 | <1, 1>, <1, 2> | - |
| 10 | <2, 2> | - |
| 3 | <3, 1> | - |
| 0, 1 | <0, 1>, <1, 1> | <0, 1> |
| 2, 3 | <2, 1>, <3, 1> | <1, 1> |
| 0, 1, 8, 9 | <0, 1>, <1, 1>, <0, 2>, <1, 2> | <0, 1>, <0, 2> |