IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Unsolicited Block ACK Extension Draft Text | | | | |
| Date: 2017-07-17 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Joe Andonieh | Peraso Technologies |  |  | joe@perasotech.com |
| Brad Lynch | Peraso Technologies |  |  | brad@perasotech.com |
| Gary Cheng | Peraso Technologies |  |  | gary@perasotech.com |
| Chris Hansen | Peraso Technologies |  |  | chris@covariantcorp.rom |
| Oren Kedem | Intel |  |  | Oren.kedem@intel.com |
| Solomon Trainin | Qualcomm |  |  | strainin@qti.qualcomm.com |
|  |  |  |  |  |
|  |  |  |  |  |

Abstract

Draft text to cover the Implicit Block ACK feature in 802.11ay

*Editor's Note: Modify this section as shown*

*Instruct the Editor to Modify the following section as follows:*

#### Block Ack Timeout Value field

The Block Ack Timeout Value field is used in the ADDBA Request, Response frames, and in the Unsolicited Block ACK Extension Element to indicate the timeout value for Block Ack. The length of the Block Ack Timeout Value field is 2 octets. The Block Ack Timeout Value field is illustrated in Figure 9-80.

*Editor's Note: Modify this section as shown*

*Instruct the Editor to add the following text:*

### 9.4.2.XXX Unsolicited Block ACK Extension element

The Unsolicited Block ACK Extension element includes information necessary to set up an unsolicited Block extension agreement between a STA and AP/PCP throughout association phase or between non-AP/non-PCP STAs using Information Request/Response mechanism. The IE can be carried in Probe Request / Response, Association Request/Response, Re-association Request/Response. or Information Response frames. The format of the element is shown in Figure 9-ZZZ.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  | Element ID | Length | Element ID Extension | Unsolicited Block ACK Extension Parameters Field | Block Ack Timeout Value |
| Octets: | 1 | 1 | 1 | 2 | 2 |

### Figure 9-ZZZ - Unsolicited Block ACK Extension Element

The Element ID, Length and Element ID Extension fields are defined in 9.4.2.1.

The Unsolicited Block ACK Extension Parameters Field is given below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | B0 B7 | B8 | B9 B15 | B16 B26 | B27 B31 |
|  | Reserved | A-MSDU Supported | Reserved | Buffer Size | Reserved |
| Bits: | 8 | 1 | 6 | 11 | 5 |

### Figure 9-ZZZ - Unsolicited Block ACK Extension Parameter Field

The A-MSDU Supported sub-field is set to 1 by a STA to indicate that it supports an A-MSDU carried in a QoS Data frame sent under the Unsolicited Block ACK extension. It is set to zero otherwise.

The Buffer Size subfield is an integer from 1 to 1024 that indicates the number of buffers available for each Unsolicited Block ack Extension context.

The Block Ack Timeout Value field is defined in 9.4.1.15.

*Editor's Note: Modify this section as shown*

*Instruct the Editor to incorporate the following text:*

#### ADDBA Request frame format

*>>>> Modify the 4th paragraph as follows*

The Dialog Token field is set to a nonzero value chosen by the STA~~.,~~ or it is set to zero for the Unsolicited Block ack Extension Agreement.

### 9.7.3 A-MPDU Contents

Modify Tables 9-245 and 9-246 as shown below

### 10.

|  |  |  |  |
| --- | --- | --- | --- |
| * Table 9-425 A-MPDU contents in the data enabled  immediate response context | | | |
| MPDU Description | Conditions | | |
| (#1198)Ack(#190) | If the preceding PPDU contains an MPDU that requires an (#1198)Ack frame(#2069) response, a single (#190)(#1198)Ack frame at the start of the A‑MPDU. | | In a non-DMG STA:(11ad) at most one of these MPDUs is present.  In a DMG STA: at most one (#1198)Ack (Ed)frame is present, and zero or more HT-immediate BlockAck frames(#6575) are present.(11ad) |
| HT-immediate BlockAck | In a non-DMG STA:(11ad) if the preceding PPDU contains an implicit or explicit block ack(#2069) request for a TID for which an HT-immediate (#2353)block ack agreement or an Unsolicited BlockAck extension agreement exists, at most one (#192)BlockAck frame for this TID, in which case it occurs at the start of the A-MPDU.  In a DMG STA: if the preceding PPDU contains an implicit or explicit block ack(#2069) request for a TID for which an HT-immediate (#2353)block ack agreement or an Unsolicited BlockAck extension agreement exists, one or more copies of the same BlockAck for this TID.(11ad) | |
| Delayed BlockAcks | BlockAck frames with the BA Ack Policy subfield equal to No Acknowledgment with a TID for which an HT-delayed (#2353)block ack agreement exists. | | |
| Delayed block ack(#2069) data | QoS Data (Ed)frames with a TID that corresponds to a Delayed or HT-delayed (#2353)block ack agreement.  These have the Ack Policy field equal to Block Ack. | | |
| Action No Ack | Action No Ack frames.(MDR) | | |
| Delayed BlockAckReqs | (#193)BlockAckReq frames with a TID that corresponds to an HT-delayed (#2353)block ack agreement in which the BA Ack Policy subfield is equal to No Acknowledgment. | | |
| Data (Ed)frames sent under an HT-immediate (#2353)block ack agreement or Unsolicited Block ACK extension agreement (M34) | QoS Data (Ed)frames with the same TID, which corresponds to an HT-immediate (#2353)block ack agreement  (#202)See NOTE. | Of these, at most one of the following is present in a non-DMG BSS:(M34)   * One or more QoS Data (Ed)frames with the Ack Policy field equal to Implicit Block Ack Request * A BlockAckReq frame   Of these, at most one of the following is present in a DMG BSS:(M34)   * One or more QoS Data (Ed)frames with the Ack Policy field equal to Implicit Block Ack Request * QoS Null MPDU with Ack Policy set to No Ack(#7713) * A BlockAckReq frame with an optional QoS Null MPDU with Ack Policy set to No Ack(#7713) | |
| QoS Null MPDUs with Ack Policy set to No Ack(#7713)(#2095) | In a DMG BSS, QoS Null MPDUs with Ack Policy set to No Ack(#7713). |
| Immediate BlockAckReq | At most one BlockAckReq frame with a TID that corresponds to an HT-immediate (#2353)block ack or Unsolicited Block ACK extension agreement.  This is the last MPDU in the A-MPDU.  It is not present if any QoS (#100)Data frames for that TID are present. |
| (#202)NOTE—These MPDUs all have the Ack Policy field equal to the same value, which is either Implicit Block Ack Request or Block Ack. | | | |

## Block acknowledgment (block ack)

### Introduction

>>>> Modify the second paragraph of this section as follows:

The block ack mechanism is initialized explicitly by an exchange of ADDBA Request/Response frames or by using the unsolicited block ack extension mechanism. After initialization, blocks of QoS Data frames may be transmitted from the originator to the recipient. A block may be started within a polled TXOP, within an SP, or by winning EDCA contention. The number of frames in the block is limited, and the amount of state that is to be kept by the recipient is bounded. The MPDUs within the block of frames are acknowledged by a BlockAck frame, which is requested by a BlockAckReq frame.

### Setup and modification of the block ack parameters

>>>> add the following at paragraph at the end of this section before last paragraph “Once the block ack exchange has been set up, Data and Ack frames are transferred using the procedure described in 10.24.3”

Support of an Unsolicited Block Ack extension is a capability of a recipient. Support of the Unsolicited Block Ack extension by the recipient is advertised by insertion of the Unsolicited Block ACK Extension Element in the management and action frames. A PCP and AP shall advertise the capability in the Association Response frame. A non-PCP and non-AP STA shall advertise the capability in the Association Request frame. An Originator may use the Unsolicited Block Ack extension in case the recipient is capable of it.

An Unsolicited Block Ack extension agreement is established per TA, RA, TID triple. The Unsolicited Block Ack extension agreement has been set up if the following conditions apply:

* the recipient transmits and the originator receives a BlockAck frame in response to MPDU with Ack Policy field set to Implicit Block Ack Request sent by the originator and
* the originator does not receive the ADBBA Response frame with Status Code equal to SUCCESS prior to receive of the BlockAck frame or the triple of the ADBBA Response frame with Status Code equal to SUCCESS received prior to receive of the BlockAck frame is different of the BlockAck frame triple.

Parameters of the established Unsolicited Block Ack extension agreement are equal to advertised by the recipient in the Unsolicited Block ACK Extension Element.

The Unsolicited Block Ack extension agreement may be torn down following rules defined in 10.24.5.

### Data and acknowledgment transfer using immediate block ack policy and delayed block ack policy

>>>> No Change

### Receive buffer operation

>>>> Append the following to the first paragraph as follows

For each block ack agreement, the recipient maintains a MAC variable NextExpectedSequenceNumber. The

NextExpectedSequenceNumber is initialized to the value of the Block Ack Starting Sequence Control field of

the ADDBA Request frame of the accepted block ack agreement. Under unsolicited block ack extension the NextExpectedSequenceNumber is initialized to the value zero at successful association establishment.

### Unsolicited Block ACK Extension

#### Introduction

An Unsolicited extension to the block ack feature (defined in 10.24.1 through 10.24.5), called Unsolicited block ack, is defined in 10.24.12.2 through 10.24.12.8. The unsolicited extension to the HT Immediate block ack feature allows sending of aggregated mpdu (A-MPDUs) without the need to exchange of ADDBA request and response frames for setup of the feature and optimizes the buffering requirement needed at the recipient side with the ability to share the recipient resources among all originators.

A STA that indicates a support of the Unsolicited Block ACK Extension is ready to support the extension to an originator STA since an association or peer-to-peer connection is established between the recipient and the originator STAs.

A station indicates that it can support Unsolicited Block Ack extensions by adding an Unsolicited Block ACK Extension element, described in 9.4.2.XXX, to its Probe Request, Probe Response, Association Request, or Association Response frames. Unsolicited Block ACK Extension Element can also be shared via Information Request and Information Response frames when a requesting STA adds the Extended Request Element in its Information Request frame to the point coordinator.

#### Unsolicited Block Ack Agreement Extension Architecture

The unsolicited extension to the HT-immediate block ack architecture is identical to the HT immediate block ack extension architecture described in section 10.24.7.2 with the following rules.

1. WinSizeO is the value of the Buffer Size field advertised in the recipient unsolicited block ack extension parameter set part of the unsolicited block ack extension information element.
2. WinStartO is equal to the sequence number of the first MPDU inside the first A-MPDU sent with a specific triple <TA, RA, TID>
3. The recipient can only use full state operation of the scoreboard context control.

#### Scoreboard context control during full-state operation

The scoreboard context control within the unsolicited block ack extension shall follow the rules of full-state operation defined in section 10.24.7.3 with the following rule added.

When the recipient flushes buffers in the reordering buffer control ones in scoreboard of SN that are higher than lowest SN of unsuccessful MPDU and lower than higher SN of successful MPDU are reverted to 0. WinStartR, WinSizeR and WinEndR remain unchanged.

#### Generation and transmission of BlockAck frames

Under this extension the recipient follows the same rules defined in section 10.24.7.5 Generation and transmission of BlockAck frames by an HT STA or DMG STA.

#### Receive reordering buffer control operation

Under this extension the receive reordering buffer control behave the same as in the HT Immediate block ack extension defined in section 19.24.7.5 with the ability of the recipient to flush the buffers when it starts receiving from a different originator or TID.

Under the unsolicited BlockAck extension the parameters are initialised as follows:

WinSizeB is set to the value of the Buffer size field advertised in the unsolicited block ack extension parameter set.

When the unsolicited block ack extension agreement is established and activated, WinEndB is initialized to the sequence number of the first MPDU inside the first A-MPDU sent with a specific triple <TA, RA, TID>, and WinStartB is set to WinEndB – WinSizeB + 1.

The Buffered MSDUs in the receive reordering buffer control can be flushed when the station stops receiving from the same <TA, TID> pair.

#### Originator Behaviour and block ack state maintenance.

In addition to the rules defined for the HT immediate block ack extension defined in section 10.24.7.7 and 19.24.7.8 the originator that utilized the Unsolicted BlockAck extension

1. First to transmit first A-MPDU of specific triple <TA, RA, TID>, the originator should transmit few MPDUs of the same triple out A-MPDU to synchronize WinStartR and WinStartB at the recipient\*
2. Shall not release transmitted buffers (MPDUs) with SN equal or higher than the first unsuccessful MPDU.
3. At the end of its TXOP or SP shall set to zero the acknowledgement status of MPDUs with SN equal or higher than the first unsuccessful MPDU even if they have been already positively acknowledged.
4. At the start of its next TXOP/SP may transmit all the MPDUs with SN equal to the first unsuccessful MPDU and above or may send a BlockAckReq frame with SSN equal to the SN of the first unsuccessful MPDU to determine the state of the buffers at the recipient and retransmit MPDUs that are not received/or flushed by the recipient.

\*Note: In case the originator is not aware of the currect value of WinStarB, and WinStartR It cn use this option.