Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**	
Title	**Recirculation Letter Ballot 13a Comment 520 Reply**	
Date Submitted	**2004-01-13**	
Source(s)	Robert Nelson MacPhy Technologies 1104 Pittsburgh Landing Richardson, TX 75080	Voice: 972-239-9224 Fax: 972-671-1455 bob_nelson@ieee.org
Re:	Letter Ballot 13a on document ***IEEE P802.16-REVd/D2-2003***	
Abstract	Discussion and suggested resolution to recirculation ballot 13a comment 520	
Purpose	Reply comment for comment 520 in LB13a comment database	
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.	
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.	
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures <http://ieee802.org/16/ipr/patents/policy.html>, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair <mailto:chair@wirelessman.org> as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site <http://ieee802.org/16/ipr/patents/notices>.	

Letter Ballot #13a
Comment 520 Reply

Bob Nelson
MacPhy Technologies

ARQ Bulk ACK/NACK
The suggested concept appears to provide an effective efficient mechanism for reporting feedback information when the size of transmitted fragments is large compared to the size of the component ARQ blocks.

Regarding the suggested implementation, I have the following observations:

a) The term *bulk* is used to refer to each collection of blocks associated with each *bulk* map descriptor. The term represents the effect of the operation (ACKing blocks en masse) but is not a good fit for the block sequences themselves. In place of bulk, the term *block sequence*. or just *sequence* is suggested. Alternately, a definition of a *bulk* would be appropriate.
b) The ACK type (=1) for Cumulative ACK is overloaded with the new functionality. While the current Cumulative Ack only functionality is not lost, the max number of ACK Maps that may be included is 3 instead of 4. Since an unused reserved value is available, it is suggested that the new functionality be assigned to this value.
c) Personal preference, include all the map information in Table 88 instead of creating a second subtable.
d) The definition of the BSN values associated with a *bulk* seems larger (by one) than it should be. Also, it is not completely clear whether the BSN of the first bock in the first *bulk* descriptor appearing in an IE has the value of the Cumulative ACK or the Cumulative Ack plus one. I believe it should be CACK plus one.

The following amended text takes these observations into account and is suggested for adoption:

Page 129, Line 1:
Replace Table 88 and all text following the table to the end of section 6.4.4.2 (page 129, line 54) with the following:

Table 88—ARQ Feedback IE

Syntax	Size	Notes
ARQ_feedback_IE (LAST) {	variable	
CID	16 bits	The ID of the connection being referenced.
LAST	1 bit	0 = More ARQ feedback IE in the list. 1 = Last ARQ feedback IE in the list.
ACK Type	2 bits	0x0 = Selective ACK entry 0x1 = Cumulative ACK entry 0x2 = Cumulative with Selective ACK entry 0x3 = Cumulative ACK with Block Sequence Ack entry
BSN	11 bits	
Number of ACK Maps	2 bits	If ACK Type == 01, the field is reserved and set to 00. Otherwise the field indicates the number of ACK maps: 0x0 = 1, 0x1 = 2, 0x2 = 3, 0x3= 4
if (ACK Type != 1) {		

Table 88—ARQ Feedback IE (Continued)

Syntax	Size	Notes
for (i=0; i< Number of ACK Maps + 1; ++i) {		
if (ACK Type != 3) {		
Selective ACK Map	16 bits	
}		
else {		Start of Block Sequence ACK Map definition (16 bits)
Sequence Format	1 bit	Number of block sequences associated with descriptor 0: 2 block sequences 1: 3 block sequences
if (Sequence Format = 0) {		
Sequence ACK Map	2 bits	
Sequence 1 Length	6 bits	
Sequence 2 Length	6 bits	
Reserved	1 bit	
}		
else {		
Sequence ACK Map	3 bits	
Sequence 1 Length	4 bits	
Sequence 2 Length	4 bits	
Sequence 3 Length	4 bits	
}		
}		End of Block Sequence ACK Map definition
}		
}		
}		

BSN

If (ACK Type == 0x0): BSN value corresponds to the most significant bit of the first 16-bit ARQ ACK map.

If (ACK Type == 0x1): BSN value indicates that its corresponding block and all blocks with lesser (see 6.4.4.6.1) values within the transmission window have been successfully received.

If (ACK Type == 0x2): Combines the functionality of types 0x0 and 0x1.

If (ACK Type == 0x3): BSN value indicates that its corresponding block and all blocks with lesser (see 6.4.4.6.1) values within the transmission window have been successfully received.

Selective ACK Map

Each bit set to one indicates the corresponding ARQ block has been received without errors. The bit corresponding to the BSN value in the IE, is the most significant bit of the first map entry. The bits for succeeding

block numbers are assigned left-to-right (MSB to LSB) within the map entry. If the ACK Type is 0x2, then the most significant bit of the first map entry shall be set to one and the IE shall be interpreted as a cumulative ACK for the BSN value in the IE. The rest of the bitmap shall be interpreted similar to ACK Type 0x0.

Sequence ACK Map

Each bit set to one indicates the corresponding block sequence has been received without error. The MSB of the field corresponds to the first sequence length field in the descriptor. The bits for succeeding length fields are assigned left-to-right within the map entry.

Since the block sequence described by the the first descripter of the first map entry of the IE corresponds to the sequence of blocks immediately after the Cumulative ACK, the ACK map bit for this sequence shall be zero indicating this sequence has not yet been reeived.

Sequence Length

This value indicates the number of blocks that are members of the associated sequence.

The BSN of the first block of the block sequence described by the first descriptor of the first IE map entry is the value of the Cumulative ACK plus one. The BSN of the first block of each block sequence is determined by adding the BSN of the first block of the previous block sequence to the length of that sequence. Within a map entry, Sequence Map/Length ordering follows the rule specified in the definition of Sequence ACK Map. Across map entries, ordering moves from the first map entry (i=0) to the last map entry (i=**Number of ACK Maps)**

Tranmit Block State Diagram

Accept suggested change. However, an error occurred during transcription of the diagram. The following figure includes the suggested change without the transcription error.

Page 132, Line 3: Replace Figure 34 with the following:

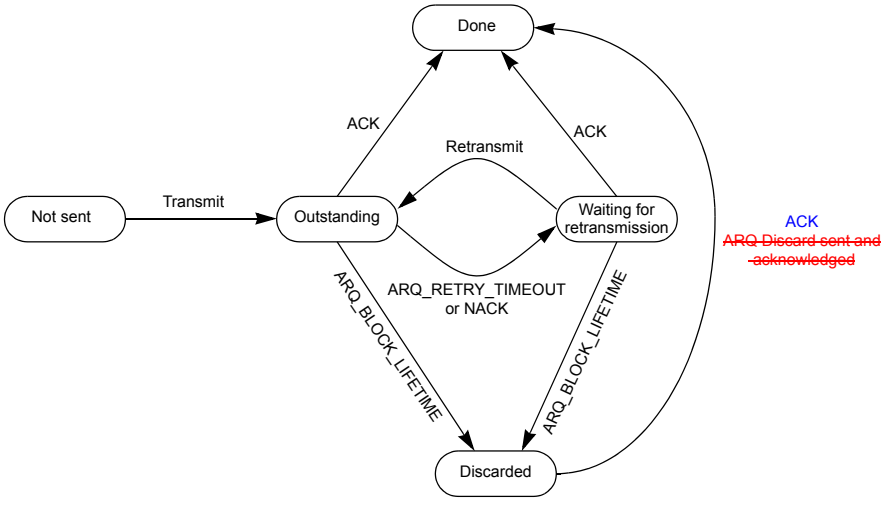

Figure 34—ARQ transmit block states

ARQ Block Receiption Diagram

Assertions are made that there are problems with Figure 34 that are corrected by the submission:

The first issue is essentially that handling of *ARQ_RX_HIGHEST_BSN* is not included in the figure.
This is correct, the current diagram does not present the appropriate logic for updating this value.

The second issue is that the diagram improperly depicts handling of updates to *ARQ_RX_WINDOW_START*.
The assertion is correct. The diagram does match the corresponding text that properly describes the operationl.

Also, an enhancement is proposed concerning handling of duplicate blocks.
The enhancement appears to be to short-circuit processing if a duplicate is detected and only take one action (resetting the purge timer for the block). This change is incorrect. Doing so results in no stimulus being provided to the receiver to initiate transmission of feedback ACK information. The fact that a duplicate was received indicates that a previous ACK transmission was lost, and the transmitter requries a new feedback IE transmission. Implementing the suggested logic suppresses possibly the only stimulus that could result in that transmission, potentially causing the state machine to lock-up.

The following diagram implements logic for managing for *ARQ_RX_HIGHEST_BSN*, corrected handling for *ARQ_RX_WINDOW_START* and appropriate handling on duplicate blocks.

Page 135, Line 13: Replace Figure 34 with the following:

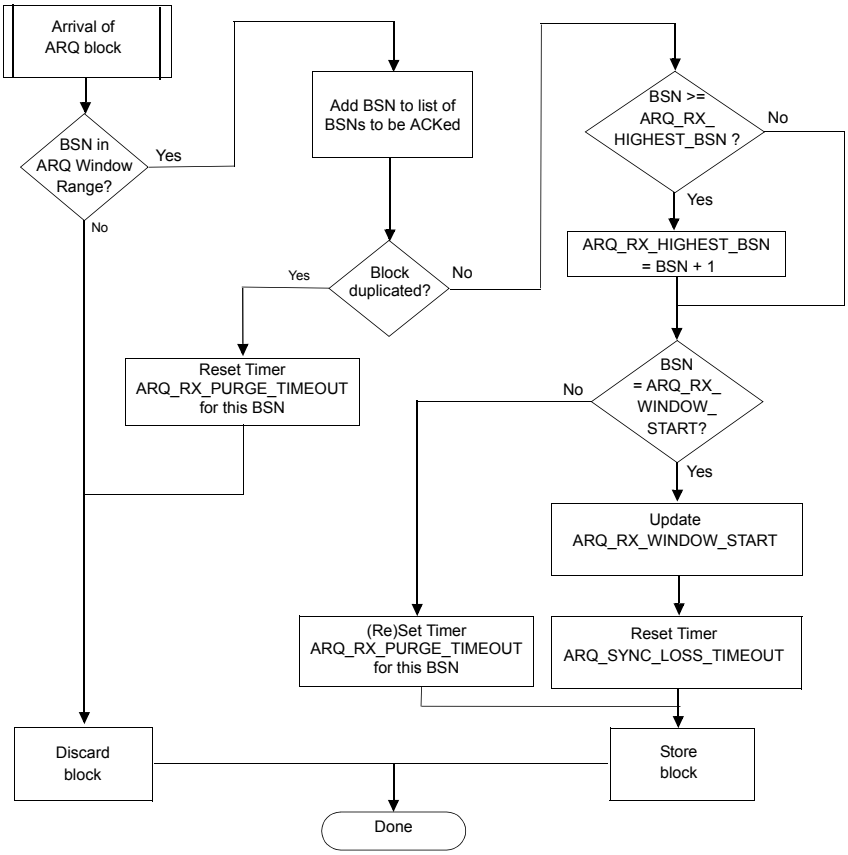

Figure 37—ARQ block reception