Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**ARQ for TG3 and TG4 Systems**
Date Submitted	**2000-03-08**
Source(s)	Subbu Ponnuswamy Malibu Networks 1035 Suncast Lane El Dorado Hills, CA 95762 Voice: 916-941-8815 Fax: 916-941-8850 mailto:subbu@malibunetworks.com Jacob Jorgensen Malibu Networks 1035 Suncast Lane El Dorado Hills, CA 95762 Voice: 916-941-8810 Fax: 916-941-8850 mailto:jacob@malibunetworks.com
Re:	In response to "802.16 Task Group 3 Call For Contributions: Proposed MAC Enhancements, Session #12", 802.16.3-01/06, and "Call for Comments on IEEE 802.16.4-01/05 and 802.16.4-01_06: Strawman MAC and PHY proposals for the 802.16.4 Air Interface Standard", IEEE 802.16.4-01/07.
Abstract	In this contribution, we propose ARQ algorithms for TG3 and TG4 systems. We present a variation of Selective Repeat ARQ and describe specific changes to the 802.16 base-line MAC. Although the focus of this submission is on Selective Repeat ARQ, we also show that the Go-back-N ARQ scheme can also be implemented with the proposed sequence numbering scheme, if desired.
Purpose	Approve portions of this document described in section "Specific comments on IEEE 802.16/D2-2001" as additions to IEEE 802.16 Standard Air Interface for fixed Broadband Wireless Access Systems.
Notice	
Release	
Patent Policy and Procedures	

ARQ for TG3 and TG4 Systems

Subbu Ponnuswamy
Jacob Jorgensen
Malibu Networks

Introduction

In this submission, we present an ARQ mechanism for TG3 and TG4 MAC. The additions to the 802.16 MAC and the formats of new messages to be added to the existing 802.16 MAC draft, are discussed. Our proposal includes a variation of the Selective Repeat ARQ that has been shown to outperform other well-known ARQ schemes in terms of higher channel efficiency and low mean delay. The proposed approach not only optimizes the path from transmitter to the receiver by selectively retransmitting only the lost PDU, but also provides an efficient bitmap-based ACK management mechanism to optimize the ACK traffic flowing in the reverse direction. Similar bitmap-based ACK management for Selective Repeat has been used by many, including HIPERLAN/2 [2], GPRS [4] and Wireless ATM [8].

In summary:

- Each MPDU is given a sequence number. This sequence number is used for re-assembly as well as ARQ
- Selective Repeat ARQ algorithm is used to selectively retransmit only lost MPDUs
- Cumulative ACK and bitmap-based ACK are used to manage ACK traffic
- ARQ ACKs are allowed to piggyback with data as well as use an ACK connection.
- ARQ works efficiently with various combinations of optional features such as fragmentation and packing.
- Incorporating ARQ into the standard does not increase overhead for connections that do not want to use ARQ.

The following acronyms are used in this document:

ACK	Acknowledgement(s)
ACKC	ACK Congestion
ARQ	Automatic Repeat reQuest
BBN	Bitmap Block Number
BM	BitMap
CACK	Cumulative Acknowledgement(s)
MPDU	MAC Protocol Data Unit
MPDU-SN	MPDU-Sequence Number
MSDU	MAC Service Data Unit (Higher Layer Packet)
SR	Selective Repeat

Summary

In this section we present the summary of ARQ algorithms, retransmission units and message formats defined in this submission. The following sections describe specific text to be added to the 802.16 MAC standard.

Sequence numbering

The 802.16 MAC [1] defines two fields, Fragmentation Control (FC) and Fragment Sequence Number (FSN) to control fragmentation and re-assembly. These fields are currently being made optional sub-headers in 802.16.1. The 3-bit FSN not only restricts the number of allowed fragments per-MSDU to eight, but also creates ambiguity in re-assembly, when some MPDUs are lost, and when ARQ is not used. Therefore, we propose that a MPDU Sequence number be used instead of the FSN to unambiguously re-assemble MPDUs. Since this MPDU is not a per-MSDU sequence number, gaps in MPDU-SN can be used to re-assemble correctly received MSDUs and to discard MSDUs with missing MPDUs. Since the Fragmentation sub-header (FC and FSN) requires at least one additional byte to the generic MAC header, the (per-MSDU FSN) + FC can be replaced by the MPDU-SN (6 bits) + FC (2 bits). With the help of 2-bit FC fields and MSDU-SN, the receiver can unambiguously determine the first and last fragments of MSDUs and missing MSDUs respectively.

MSDU-SN and Retransmission Unit

Since the ARQ is implemented at the MAC layer, the natural choice for the retransmission unit is the MPDU. As described in the previous subsection, the MPDUs are given a sequence number when they are created [9]. The receivers and transmitters use this sequence number to track and retransmit lost MPDUs and to advance their receiving and sending windows based on positive acknowledgements. The MPDU-SN-based ARQ makes no assumptions about any optional features of 802.16 MAC such as packing and fragmentation. This MPDU-SN is assigned to all MPDUs irrespective of whether ARQ is enabled or not. When fragmentation is allowed for a connection, the MSDU-SN can be used to re-assemble packets, even if ARQ is not enabled.

This also allows us to implement ARQ (with small window size) with no additional overhead. The same sub-header that is used for re-assembly can be used by the ARQ. If bigger window size is desired, an additional byte is added to the sub-header. The TYPE field in 802.16 Generic MAC Header can be used to control the number of additional bytes present.

Packing Sub-header

Another option in 802.16 that can be used with ARQ and/or fragmentation is packing. When packing is allowed for a connection, a single MPDU may have one or more full or fragmented MSDUs. Since only the MPDUs are given a sequence number, no additional overhead is introduced for such packed MPDUs. 802.16 MAC defines a packing sub-header similar to the one shown in Figure 2. The only difference between this and the one in the draft is the TYPE field. Since FSN is not needed, the three FSN bits are used for the TYPE field in Figure 2. As shown in a later section, the TYPE field is used to piggyback ARQ messages (ACK, Discard and Reset Messages).

Selective Repeat ARQ

Selective Repeat (SR) has been proposed for many wireless and mobile standards such as HIPERLAN/2, IMT-2000. Since wireless resources are scarce and SR provides higher channel efficiency, SR has been the preferred algorithm for many wireless systems compared to other ARQ protocols such as Go-Back-N (GBN). Unlike GBN, the SR ARQ algorithm selectively retransmits only the lost PDUs. There are many variations of SR. However, in this submission we consider a variation of SR that uses CACK and bitmap-based ACK. A bitmap-based ACK combines the benefits of a cumulative ACK and a set of positive and negative ACKs into one message. The MPDU-SN described in a previous subsection is used to implement SR ARQ.

Go-back-N ARQ

The focus of this submission is on Selective Repeat ARQ. Since the channel efficiency and mean delay of go-back-N or any flavor of go-back-N are much worse than the Selective Repeat ARQ in wireless environments [2], Go-back-N ARQ is not considered in this submission. However, if desired, the MPDU-SN can be used to implement g-back-N ARQ as well. This MPDU-SN scheme provides all the flexibility required by go-back-N such as the ability to change MPDU size between retransmissions.

Adaptive Modulation and ARQ

Based on link conditions, the PHY modulation may have to be changed dynamically. When the modulation changes, the capacity of the channel also changes on a per-frame basis. However, it is possible to keep the MPDU size the same, by varying the number of PHY slots requires to transmit a given MPDU, when modulation changes. With ARQ, if the MPDU size is kept the same between retransmissions, it does not have an effect on the correctness of the SR ARQ described in this submission. However, if changing the MPDU size between retransmissions is desired, an alternative scheme proposed in [10] may be used as an optional feature. It is also possible to have other hierarchical numbering schemes with additional overhead to implement such an ARQ mechanism.

The following sections describe the specific texts to be added to the 802.16 MAC standard.

MPDU Sequence Number

Each MPDU is assigned a sequence number. The sub-header for a connection with fragmentation enabled is shown in Figure 1(b). The MPDU-SN along with the FC bits is sufficient for unambiguous re-assembly.

Table 1: TG3/TG4 options with different sub-headers (Ref: Figure 1)

	NO ARQ	**ARQ with window size <= 32**	**ARQ with window size <= 128**	**ARQ with window size <= 1024**
1(a)	No fragmentation	No fragmentation	No fragmentation	N/A
1(b)	With fragmentation	With fragmentation	N/A	N/A
1(c)	N/A	N/A		With or without fragmentation

MPDU-SN (8)

(a) Fragmentation not enabled and
Window sizes of <= 128

FC (2)	MPDU-SN (6)

(b) With Fragmentation and
Window sizes of <= 32

FC (2)	RSVD (3)	MPDU-SN (11)

(c) With or without fragmentation and
Window sizes of <= 1024

Figure 1: ARQ Sub-headers for 802.16 MAC

TYPE (3)	FC (2)	Length (11)

Figure 2: Packing Sub-header for 802.16 MAC

Automatic Repeat Request Mechanism

Selective Repeat Algorithm

Selective Repeat with bitmap-based Cumulative Acknowledgments is specified as the ARQ mechanism for 802.16. To avoid ambiguity, the window size of the SR ARQ must be less than half the size of the maximum possible sequence number. The transmitters are required to retransmit only the MPDUs that are lost. The following subsections describe various options and the actions to be taken at the transmitter and receiver for an ARQ enabled connection. The ACK, Discard and Reset message formats and the methods for handling ACK messages are described.

ARQ Options and Sub-headers

Three MAC sub-headers are defined to support ARQ. Since it is important to optimize the use of optional sub-headers, the options that are most likely to be used together are combined to define sub-headers. Figure 1 shows the supported options. 1(a) is used by connections that do not use fragmentation, but use ARQ. This option allows ARQ window sizes up to 128. If fragmentation is used by a connection, option 1(b) is used. This option is used by connections whether ARQ is enabled or not. If ARQ is enabled, the window size is limited by 32. When ARQ is not used by a connection that supports fragmentation, 1(b) is used to unambiguously re-assemble MSDUs. The third option 2(c) is used by connections that require window sizes up to 1024. Since there are sufficient bits for large sequence numbers, two bits are always allocated to FC. If fragmentation is not supported by a connection that uses this option, the FC bits should be ignored. The TYPE field of the Generic Header is used to control these options. The appropriate TYPE values are TBD. Table 1 summarizes these options and their applications.

Acknowledgement Format

The ACK message is specified as a bitmap as shown in Figure 3. A variable length ACK message has the following fields:

- **Connection ID:** The connection ID of the connection for which this ACK message is being generated.
- **Type:** Indicates the type of the message. TYPE = 1 for ARQ bitmap.
- **CACK Flag:** Indicates if the first ACK MAP entry in the message is a Cumulative Acknowledgement. It is not necessary for all ACK messages to have this flag set. However, cumulative ACK must be sent frequently enough to advance the transmitter's sliding window. Specifying a frequency for CACK is beyond the scope of the standard, however the frequency should be considerably smaller than the window size.
- **ACKC Flag:** This is valid only in the upstream direction. The SS sets this flag if there is ACK message backlog and the messages cannot be piggybacked on to other data MPDUs going upstream. The exact behavior of the BS scheduler upon receipt of a message with this flag set is not defined here, as it is beyond the scope of the standard. The BS scheduler may allocate slots for the management connection for that CPE to carry these ACKS.
- **Length:** The total length of the ACK message in number of ACK MAPs, where each ACK MAP consumes two bytes.
- **ACK MAP (BBN + BM):** The BBN and BM together describe a one-byte bitmap. A single ACK message can have more than one BBN + BM combination. The BBN and BM fields are described in detail below.

An ACK MAP consists of a Bitmap Block Number (BBN) and a Bitmap (BM). The basic idea behind the BBN + BM representation is to create bitmaps in 8-bit increments to support efficient hardware implementations. Since the maximum possible sequence number is 2047, the sequence number space is divided into 256 8-bit blocks. A BBN indicates the Bitmap Block Number within this 256 possible blocks and the 8-bit BM indicates the 8 sequence numbers within a BBN. For example, a BBN of 25 and a BM of 11100101 indicate the status of eight MPDUs stating from sequence number 200 (= 8 * 25). This ACK MAP indicates that the receiver has correctly received MPDUs 200, 201, 202, 205, 207 correctly, whereas MPDUs 203, 204 and 206 were lost. If the CACK bit was set in this ACK message and this ACK MAP is the first MAP in the message, then this also serves as a Cumulative Acknowledgement for MPDU with sequence number 202.

The number of ACK MAPS in a single ACK Message is variable subject to a vendor-specified limit. An ACK PDU is a collection of one or more ACK messages. In order to reduce the impact of ACK traffic on the reverse direction, the ACK PDUs are allowed to piggyback on data flowing towards the transmitter. The scheduler is also allowed to allocate specific slots for the ACK PDUs in both upstream and downstream, in case piggybacking is not possible.

No assumptions should be made about the correctness of packets that fall between two ACK MAPS of the same message. Only the first ACK MAP in an ACK message with CACK bit set should be considered as a positive acknowledgment. When the CACK bit is set for an ACK message, the first ACK MAP's BM portion should have the first (MSB) bit set. A receiver can send bitmaps only for BBNs that address the last correctly received block or before.

ARQ ACK messages can be piggybacked with transport MPDUs if there is unused space. The TYPE field in the packing sub-header (Figure 2) indicates whether the PDU that follows the packing sub-header is another full or fragmented MSDU or an ACK PDU. A separate ACK connection is also defined to carry ACK traffic. If the ACK message is transported over this separate connection, a CRC-16 should be appended.

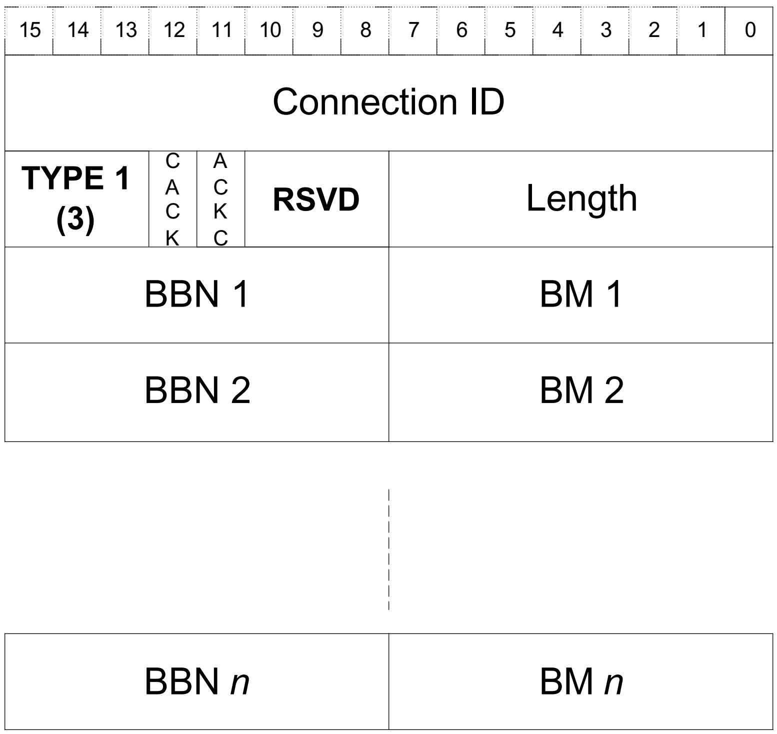

Figure 3: ARQ ACK Message Format with Cumulative ACKS and Bitmap NACKS

Discard and Reset Messages

Discard messages contain the MPDU sequence number up to which the transmitter requests the receiver to skip. When a receiver receives a discard message, it advances its window to the sequence number specified in the discard message. This message informs the receiver that the transmitter wants to discard MPDUs with sequences numbers up to but not including the discard MPDU-SN specified in the discard message. The conditions for generating discard messages are not specified here.

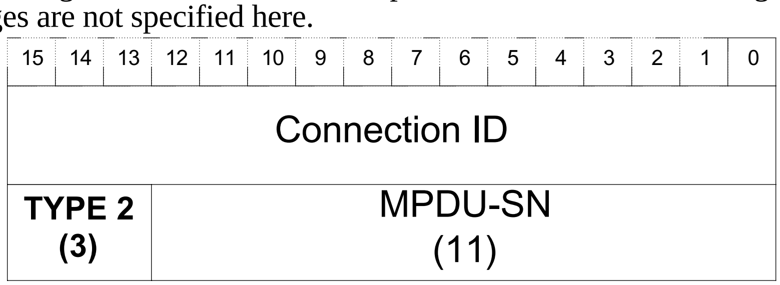

Figure 4: ARQ Discard Message Format

Reset messages are used to reset the ARQ state of a connection. The receiver should acknowledge a reset message and the bottom of windows should be set to zero on both sides upon synchronization. All state information should be cleared and the receiver should discard all data in its buffer. Specifying the exact conditions for generating these messages is beyond the scope of a standard.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Connection ID															
TYPE 3 (3)			RSVD (5)												

Figure 5: ARQ Reset Message Format

Receiver Actions

- The bottom of the receiver's window is the lowest MPDU-SN (l-MPDU-SN) not yet correctly received. The receiver should maintain this value as an 11-bit MPDU-SN.
- Should check if the MPDU is received error free (CRC Check)
- If the MPDU is corrupted, discard the MPDU and generate an ACK MAP
- If the MPDU is not corrupt and if the MPDU-SN is within the receiver's window - keep the MPDU and update appropriate variables.
- Must send periodic CACKs even when the channel is error free. While the exact frequency of this message is not specified in the standard, the frequency should be considerably smaller than the window size of the connection.
- A timer should be used to send duplicate CACK messages.
- If a discard message is received and it is within the receiver's window, advance the window to the sequence number specified in the discard message. The receiver may deliver correctly received packets from its current start of the window up to the discard sequence number to the Convergence Sub-layer.
- If the receiver is an SS, set the ACKC bit in the outgoing ACK messages, if there is a backlog of ACK messages.

Transmitter Actions

- Hold copies of transmitted MPDUs until cumulatively acknowledged by the receiver.
- Set a timer every time the transmitter's window is advanced. This timer should be set to 5 seconds. Since this is a fairly large timeout value, if the timer expires before any CACK is received, transmitted should send a reset.
- If the transmitter is a BS, and ACKC bit is set fro ACK messages from an SS, the scheduler may allocate slots for the ACK connection of that SS.

ARQ Parameters

The following ARQ parameters need to be specified for connections that have ARQ enabled:

- **Window size:** The window size to be used for this connection. Should be less than or equal to 1024
- **Maximum retry limit:** The maximum number of retransmissions allowed for an MPDU of this connection. The actual number of retries can be made adaptive based on the QoS and other properties of the connection, subject to this maximum limit.

These parameters are communicated to the SS during connection setup.

Specific Comments on IEEE 802.16/D2-2001

We suggest that the following changes be made to the “Draft Standard Air Interface for Fixed Broadband Wireless Access Systems”, IEEE 802.16/D2-2001 [1].

1. Sections “Automatic Repeat Request Mechanism“ and all subsections under the same be included under section 6.2.3.4 of [1]
2. References to FSN be removed from [1], including Table 1, sections 6.2.3.2 and 6.2.3.3.2
3. In section 6.2.3.3.2 of [1], the packing sub-header of Figure 61 be replaced with Figure 2 of this document.
4. Section “MPDU Sequence Number” of this document be included under section 6.2.3.2.

References

1) “Draft Standard Air Interface for Fixed Broadband Wireless Access Systems”, *IEEE 802.16/D2-2001*

2) “Automatic Repeat Request (ARQ) Mechanism in HIPERLAN/2”, by H. Li, J. Lindskog, G. Malmgren, G. Miklos, F. Nilsson, and G. Rydnell, *The IEEE Semiannual Vehicular Technology Conference* – VTC2000-Spring, 2000

3) “Broadband Radio Access Networks (BRAN): HIPERLAN Type 2 – Data Link Control (DLC) Layer – Part 1: Basic Data Transport Functions”, *ETSI TS 101 761-1 v1.2.1 (2000-11)*

4) “Digital cellular telecommunication system (Phase 2+): General Packet Radio Service (GPRS) – Overall description of the GPRS Radio Interface Stage 2”, *ETSI TS 101 350 v6.1.0 (1998-10)*

5) “Performance Evaluation of Retransmission Mechanisms in GPRS Networks”, by Q. Pang, A. Bigloo, V. C. M. Leung, and C. Scholefield, *IEEE WCNC'00*, Sep. 2000

6) “Improved Selective Repeat ARQ Scheme for Mobile Multimedia Communications”, by W. S. Jeon, and D. G. Jeong, *IEEE Communications Letters*, Vol. 4, No. 2, 2000

7) “A selective repeat ARQ scheme for point-to-multipoint communications and its throughput analysis”, by S. R. Chandran, *ACM SIGCOMM conference on Communications architecture & protocols*, 1986

8) “Wireless ATM: Limits, Challenges, and Proposals”, by E. Ayanoglu, K. Y. Eng, and M. J. Karol, *IEEE Personal Communications*, Aug 1996

9) “A Proposal for the Enhancement of the 802.16.1 MAC to Provide a per-Service Flow Reliable Connection Service”, *IEEE 802.16 Session 11 (802.16.4c-01/10)*, by Jacob Jorgenson, Ken Pierce and Subbu Ponnuswamy, Malibu Networks

10) Data Integrity in 802.16.4 MAC, *IEEE 802.16 Session 11 (802.16.4c-01/06)*, by Naftali Chayat, Vladimir Yanover, and Inbar Anson, BreezeCOM Ltd