Data Integrity in 802.16 MAC

IEEE 802.16 Presentation Submission Template (Rev. 8)

Document Number:

IEEE 802.16.3p-00/16

Date Submitted:

2001-01-23

Source:

Vladimir Yanover et al.
BreezeCOM
Atidim Technology Park, Bldg. 1
P.O. Box 13139, Tel-Aviv 61131, Israel

Voice: +972-36457834
Fax: +972-36456290
E-mail: vladimiry@breezecom.co.il

Venue:

To be considered at 802.16 Session #11, Jan 2001

Base Document:

This presentation illustrates IEEE 802.16.3c-00/16 .

Data Integrity in 802.16. MAC

Naftali Chayat, Vladimir Yanover,
Inbar Anson

BreezeCOM

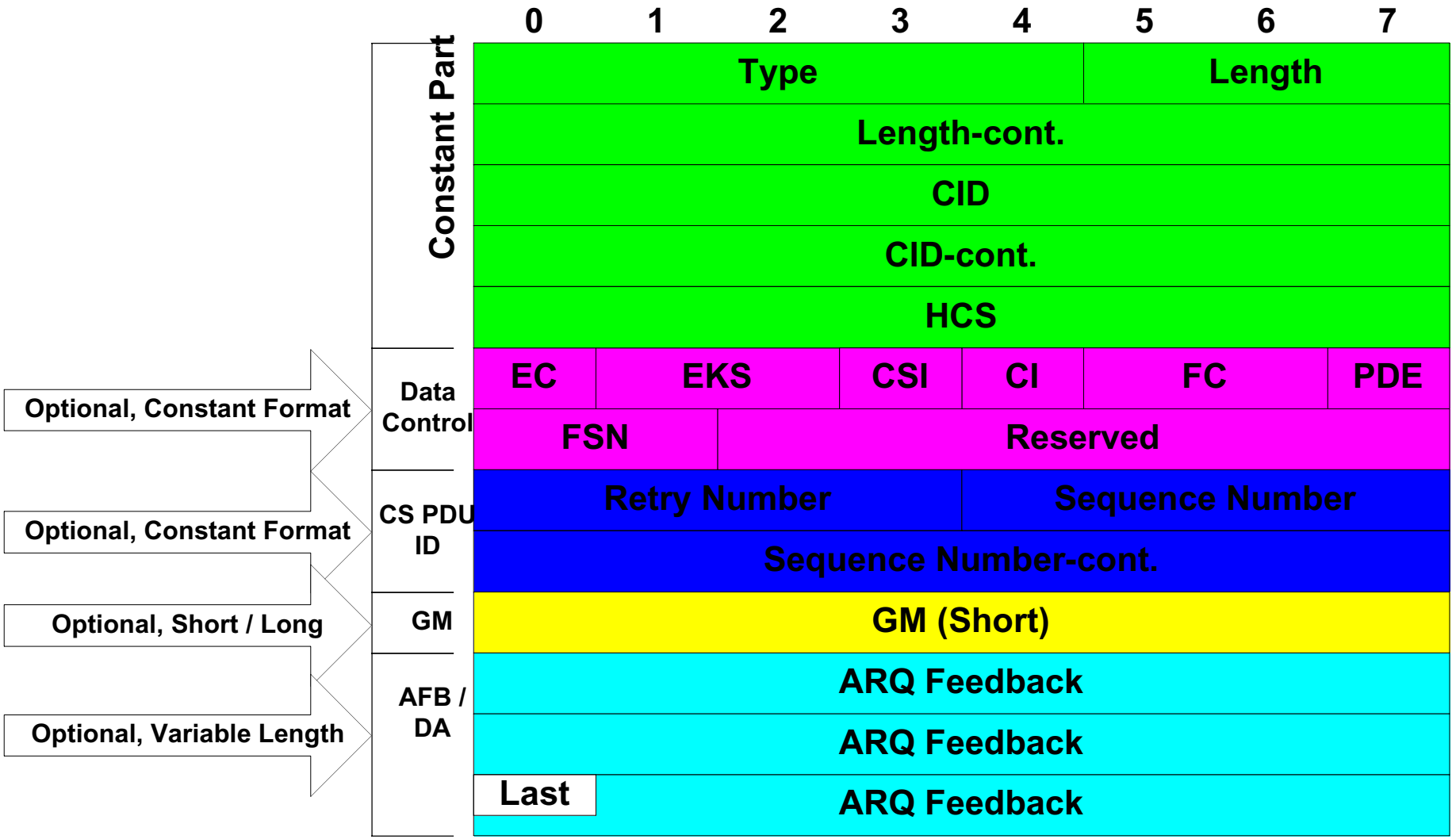
Data Integrity functionality

- ¥ Fragmentation / Assembling
- ¥ Concatenation
- ¥ Integrity check
- ¥ ARQ (Retransmissions)

Identification of MSDUs

- MAC assigns to each MSDU received from the Convergence Layer a sequence number (MPDU sequence Number = MSN) in the interval from 0 to 2^{12}-1
- The way the transmitter cares on the non-ambiguity of the sequence numbers is out of the standard s scope

General MAC Headers Format



Fragmentation

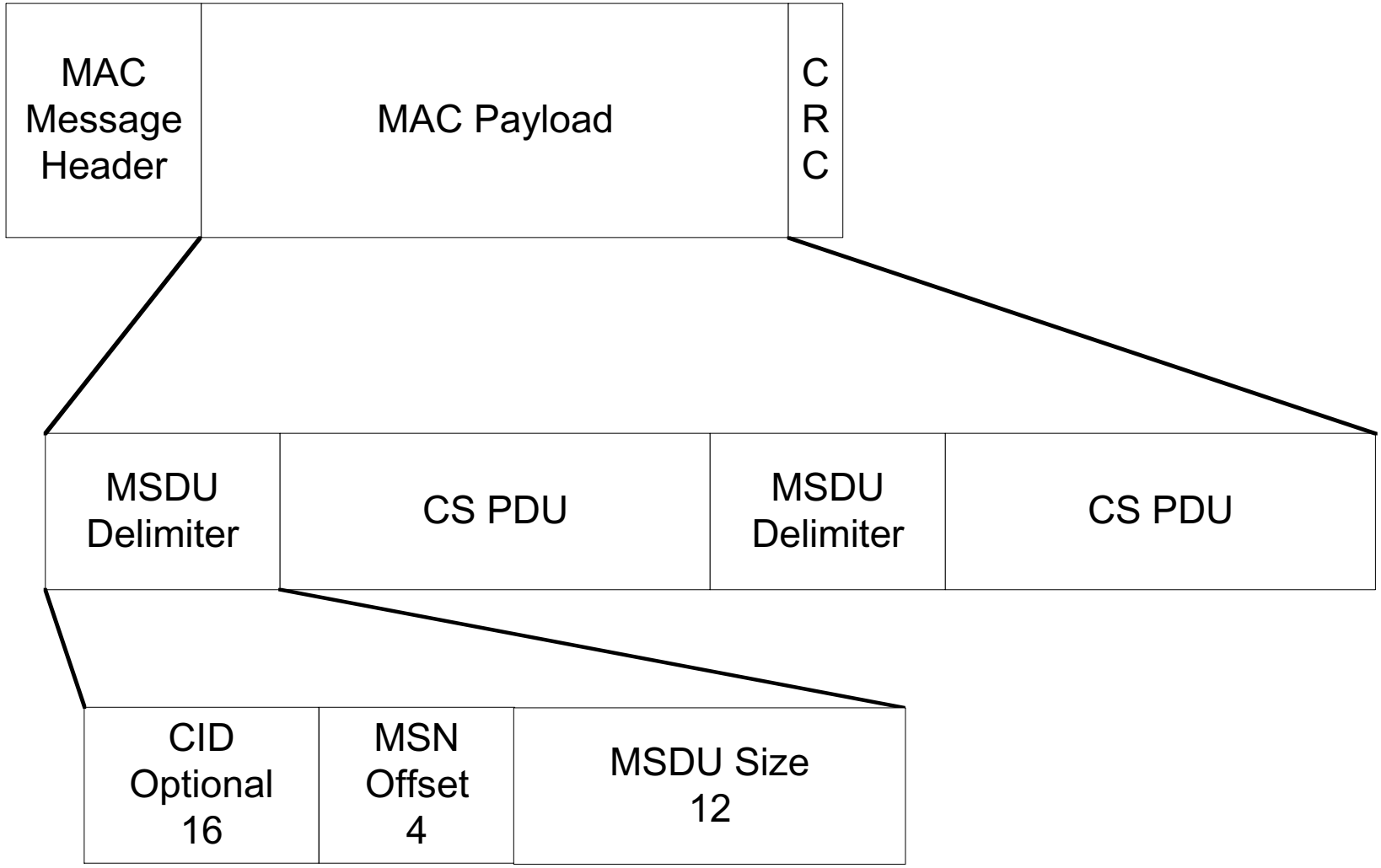
- Reasons
  - Lack of the frame time when allocating the air time to the given MSDU
  - High BER that requires employing integrity check for smaller data blocks
- Once applied, the fragmentation of the given MSDU never changes

Fragmentation-cont.

- Fragment Control code (FC):
  - 00 = non-fragmented MPDU
  - 01 = last fragment
  - 10 = first fragment
  - 11 = continuing (middle) fragment

Concatenation of Small CS PDUs into a Single MAC Message

Purpose: decrease PHY overhead



Concatenation-cont.

- Message Type = Data
- New flags in MAC header :
  - CONC = 1 if the given message includes concatenated CS PDUs
  - TDM = 1 if each MSDU delimiter contains CID
- MAC payload is always a non-fragmented CS PDU
- (If for this connection ARQ is enabled)
  - sequence number = (sequence No from MAC header) + (MSN Offset)

Concatenation-cont.

The MAC Header Fields

- EC, EKS —used for all CS PDUs
- FC = 00 (non-fragmented)
- CI = 1 means CRC presence at the end of MAC message
- PDE = 0 (N/A)

Requirements to ARQ

- Should be implemented at MAC layer, for both DL and UL
- A possibility to enable / disable ARQ function for each connection separately
- The tools used by the ARQ mechanism (like change in frame formats) have to add zero or negligible overhead to the connections with ARQ disabled
- ARQ on the level of CS PDU fragments should be supported

Requirements to ARQ

- ¥ Selective retransmissions should be employed
- ¥ Possibility for piggybacking the ARQ related info (e.g. ACK) onto the MAC messages
- ¥ Algorithm should provide group ACKs
- ¥ Discard algorithm should operate at the level of CS PDUs

Proposed ARQ Related Signaling Format

- ¥ Responds to all above requirements
- ¥ A part of MAC header (variable size)
- ¥ May be both standalone and piggybacked on the message of any type
- ¥ Flexible: Short .. Long

ARQ Feedback / Discard Info

AFB Short Format = 8 bits

Last	Mode	Reserved
1	3	4

Acknowledges all the MAC messages received in the current frame

AFB Medium Format = 32 bits

CID	Last	Mode	SerNo
16	1	3	12

For the given CID acknowledges all the CS PDUs with sequence numbers < SeqNo

AFB Long Format = 48 bits

CID	Last	Mode	SerNo	Mask
16	1	3	12	16

For the given CID acknowledges the fragments of the CS PDU messages with sequence numbers = SeqNo

Discard Info = 32 bits

CID	Last	Mode = 101	SerNo
16	1	3	12

(Sent by the transmitter). Informs the receiver that the CS PDU with the sequence numbers < SeqNo were discarded