Mechanisms for adding parity bytes

Eric Lynskey

November 2006 IEEE 802.3av meeting

Outline

- Problem statement
- How to delay MAC
- EPON mechanism
- Potential 10GEPON mechanism
- Conclusion

Problem

- FEC requires parity bytes
- MAC has no knowledge of FEC

- *MAC must be prevented from transmitting at full line rate in order to accommodate insertion of parity bytes*

Existing IEEE 802.3 methods

- 10G WAN uses MAC to insert extra idle
- EPON uses MAC Control to prevent frames from going to MAC
- EFM Copper uses Carrier Sense signal to prevent MAC from transmitting new frame

How MAC transmission is delayed

4A.2.3.2.1 Deference

When a ~~frame~~packet is submitted by the MAC client for transmission, the transmission is initiated as soon as possible, but in conformance with the following rules. The variable carrierSense is ignored in process Deference when the variable carrierSenseMode is FALSE.

The MAC sublayer monitors the transmitting variable, which indicates the MAC is transmitting data to the physical layer, as well as the carrierSense signal provided by the PLS, which indicates the physical layer is not ready for the next frame. When either transmitting or carrierSense is true, the MAC delays any pending transmission. When both are false, the MAC continues to defer for a proper ~~interFrameSpacing~~interPacket-Gap (see 4A.2.3.2.2).

If, at the end of the ~~interFrameSpacing~~interPacketGap, a ~~frame~~packet is waiting to be transmitted, transmission is initiated. When transmission has completed (or immediately, if there was nothing to transmit) the MAC sublayer resumes its original monitoring of transmitting and carrierSense.

Deference process

deferring: Boolean; {Implies any pending transmission must wait for the physical layer to be ready for the next packet and for the ~~interframe spacing~~interpacket gap}

deferenceMode: Boolean; {Indicates the desired mode of operation, and enables waiting for ~~interframe spacing~~interpacket gap during the deference process}

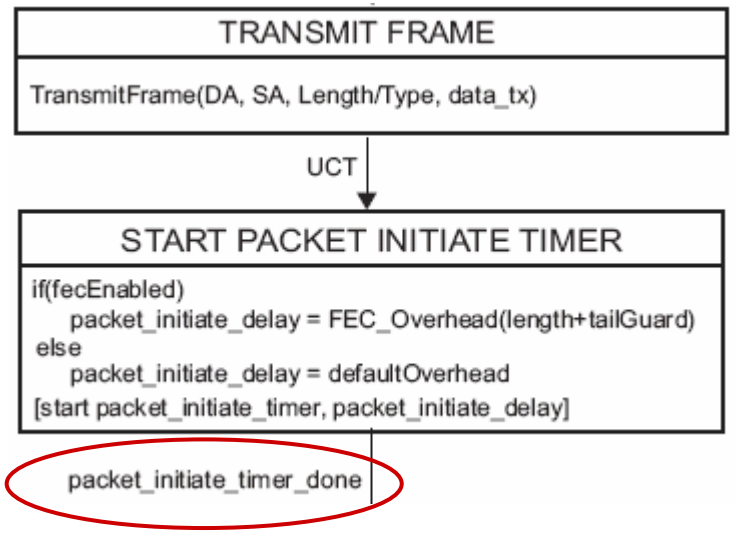
carrierSenseMode: Boolean; {Indicates the desired mode of operation, and enables using carrierSense to extend deference due to congestion in the PHY}

The Deference process runs asynchronously to continuously compute the proper value for the variable deferring:

```
process Deference;
begin
   cycle {Main loop}
      while (not transmitting and not (carrierSenseMode and carrierSense)) do nothing; {Wait for the start
                                                                                          of transmission or congestion}
      deferring := true; {Inhibit future transmissions}
      while (transmitting or (carrierSenseMode and carrierSense)) do nothing; {Wait for the end of
                                                                                transmission and congestion}
      if deferenceMode then Wait(interPacketGap~~interFrameSpacing~~);
            {Time out entire ~~interframe~~interpacket gap if enabled}
      deferring := false {Don't inhibit transmission}
   end {Main loop}
end; {Deference}
```

EPON mechanism

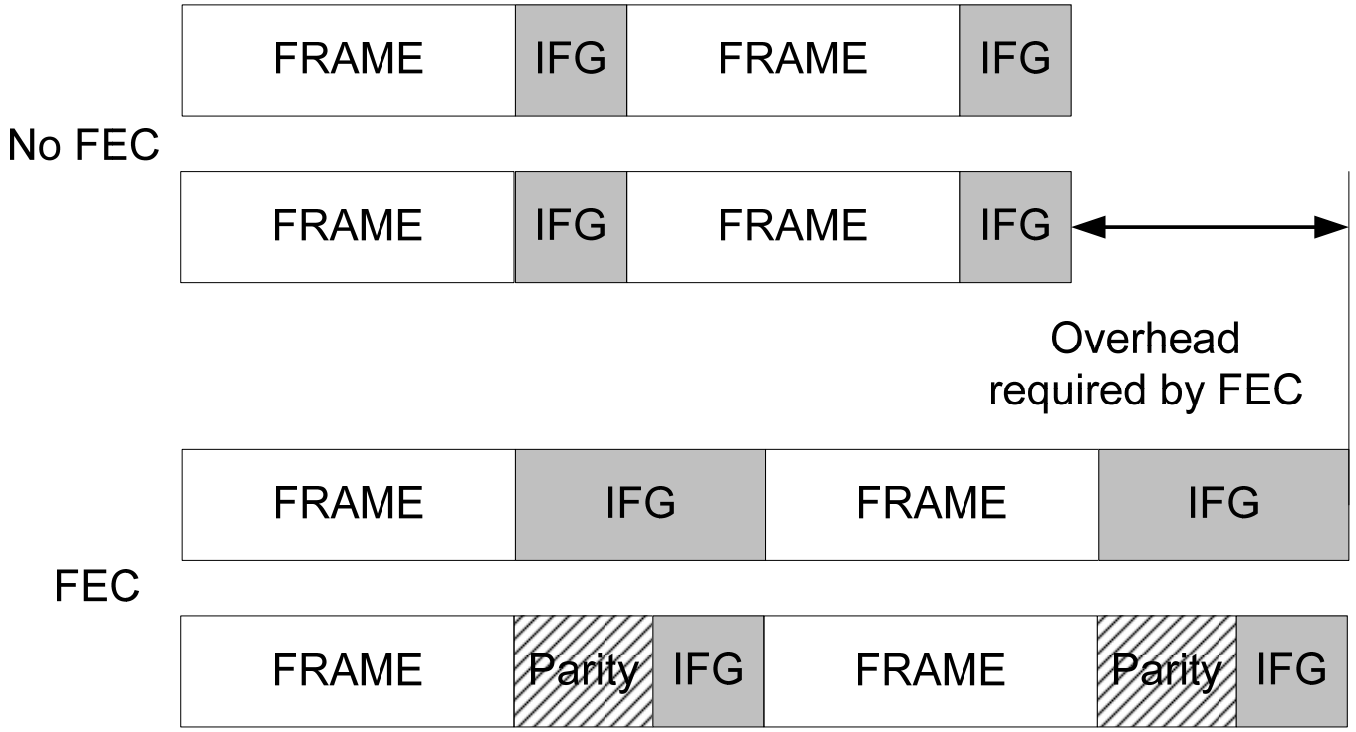
- In EPON, MAC Control sublayer controls MAC transmissions. OLT and ONU use same mechanism (Figures 64-12 and 64-13)



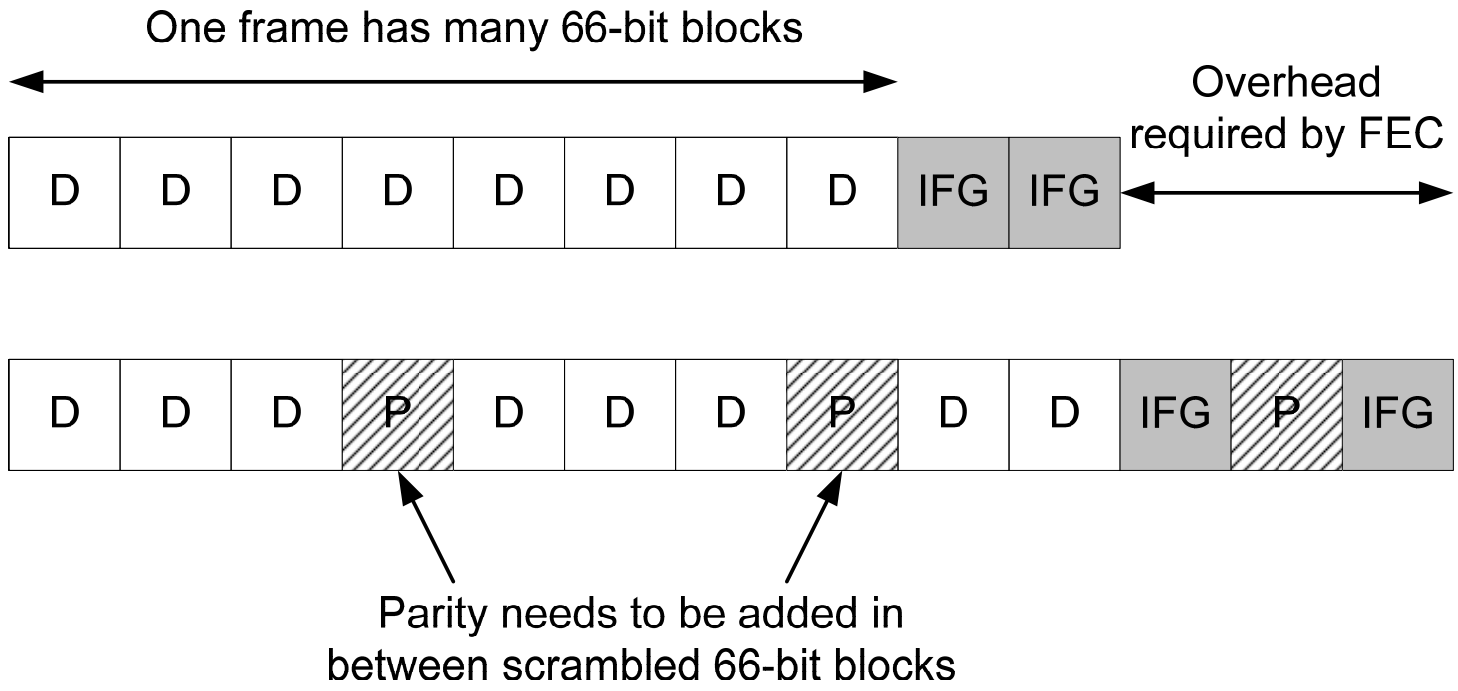
Default value is 6 time quanta,

Increased as necessary for FEC

EPON with frame based FEC



10GEPON with stream based FEC

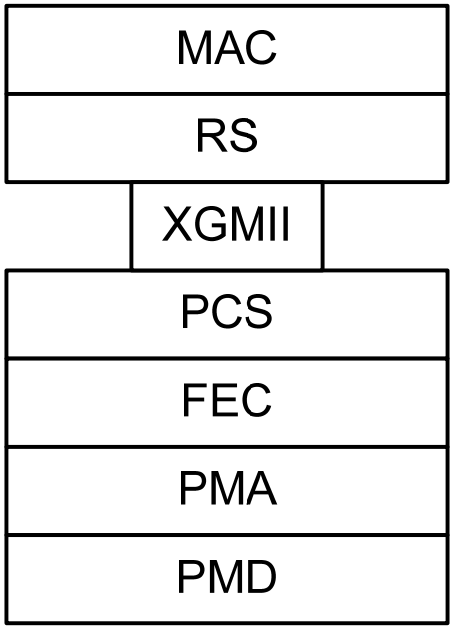


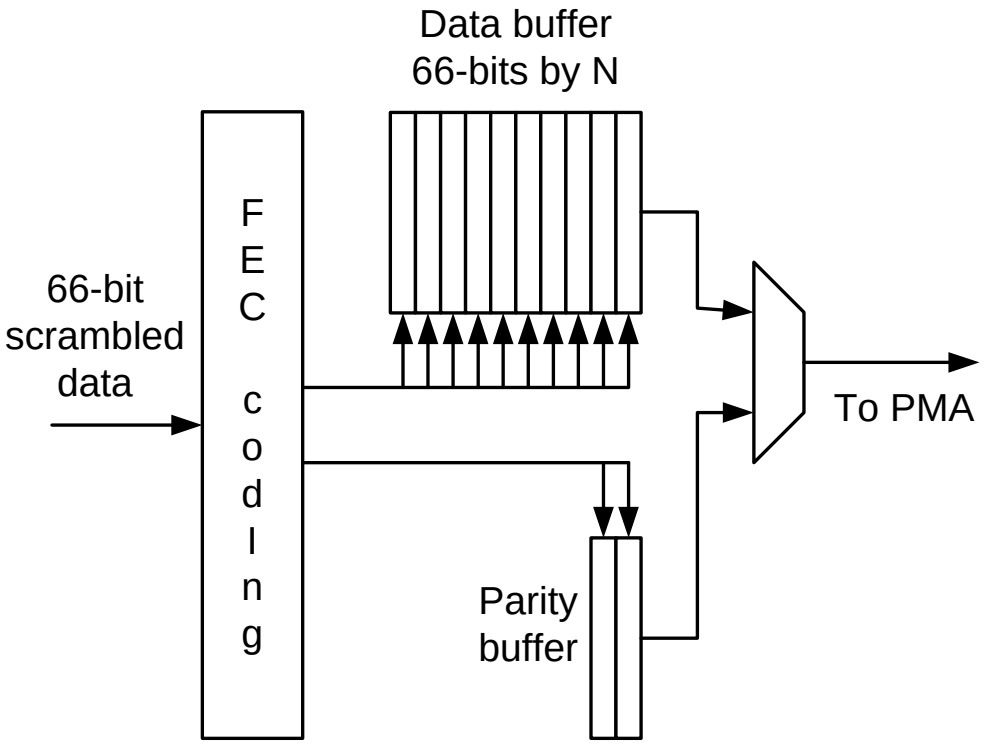
Possible 10GEPON mechanism

- MAC Control or PHY reduces throughput by increasing the size of IFG
- PCS can delete Idle codes to account for insertion of parity
- FEC sublayer maintains buffer
  - Holds 66-bit scrambled blocks
  - Large enough to account for the amount of parity necessary for max size frame
- When parity bytes need to be inserted
  - Start filling FEC buffer
  - Insert parity bytes

Buffer in FEC sublayer

- Need buffer for data while parity is inserted
- Buffer holds 66-bit scrambled blocks
- Buffer size must account for max size frame





Potential problem with using CarrierSense

- As soon as TransmitFrame function completes, MAC will service request from client for new frame transmission
- This new frame could be delayed in the MAC for the insertion of parity bytes if CarrierSense is used
- This delay can have potential impact on timestamp integrity

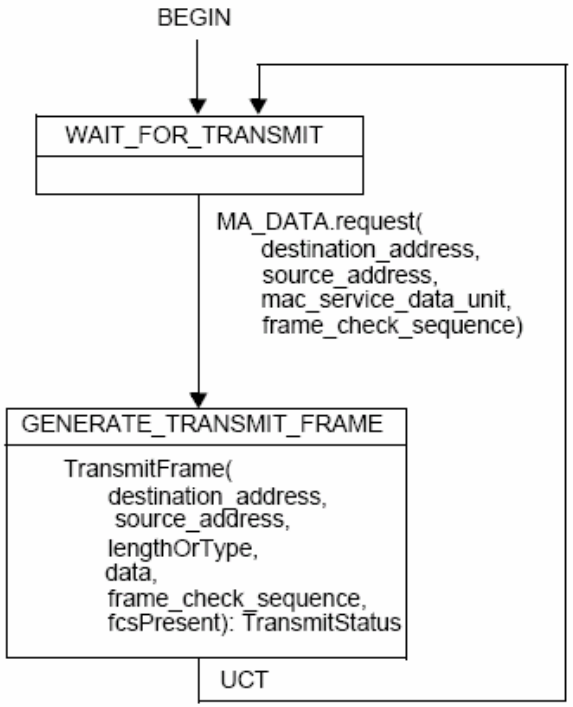


Figure 4A–3— MAC client transmit interface state diagram

Potential issue with using deference and MPCP

- The amount of parity to be added per frame in stream based FEC is “fixed”
  - Depending on boundaries extra parity may be needed
- MAC Control can calculate amount of extra IFG needed per frame (same mechanism as in EPON)
- To calculate number of parity bytes
  - knowledge of when parity is inserted OR
  - always add extra idle (more inefficient)

Conclusion

- Start of MAC frame transmissions can be controlled by both MAC Control and PHY sublayers
- With FEC, MAC throughput needs to be throttled
- Some type of mechanism needs to be defined for how to handle the insertion of parity into the scrambled data stream