MPCP FEC issues:
FEC_Overhead function

Jeff Mandin - PMC-Sierra
802.3av TF – New Orleans
January 2009

Deficit Idle Count Definition

46.3.1.4 Start control character alignment

On transmit, it may be necessary for the RS to modify the length of the <inter-frame> in order to align the Start control character (first octet of preamble) on lane 0. This shall be accomplished in one of the following two ways:

1) A MAC implementation may incorporate this RS function into its design and always insert additional idle characters to align the start of preamble on a four byte boundary. Note that this will reduce the effective data rate for certain packet sizes separated with minimum inter-frame spacing.

2) Alternatively, the RS may maintain the effective data rate by sometimes inserting and sometimes deleting idle characters to align the Start control character. When using this method the RS must maintain a Deficit Idle Count (DIC) that represents the cumulative count of idle characters deleted or inserted. The DIC is incremented for each idle character deleted, decremented for each idle character inserted, and the decision of whether to insert or delete idle characters is constrained by bounding the DIC to a minimum value of zero and maximum value of three. Note that this may result in inter-frame spacing observed on the transmit XGMII that is up to three octets shorter than the minimum transmitted inter-frame spacing specified in Clause 4; however, the frequency of shortened inter-frame spacing is constrained by the DIC rules. The DIC is only reset at initialization and is applied regardless of the size of the IPG transmitted by the MAC sublayer. An equivalent technique may be employed to control RS alignment of the Start control character provided that the result is the same as if the RS implemented DIC as described.

Operation of *Deficit Idle Count* in 10GEPON

- The *Deficit Idle Count* algorithm is part of the 10GBASE-R Reconciliation Sublayer and is incorporated in the 10GEPON RS (cf. 3av_1108_kramer_2.pdf slides 3-5)

- From the *Deficit Idle Count* definition we see that:

1. DIC enables the RS to *"maintain the effective data rate"*
   - So with DIC, the RS throughput over time is the same as when IPG is a constant 12

2. DIC sometimes inserts and sometimes deletes idle characters - in order *"to align the Start control character"*
   - So during End-of-frame processing when the next frame has not yet arrived, DIC is not active.

Issue: Extra IPG bytes added by *FEC_Overhead*()

- In draft 2.2, FEC Overhead() is defined as:

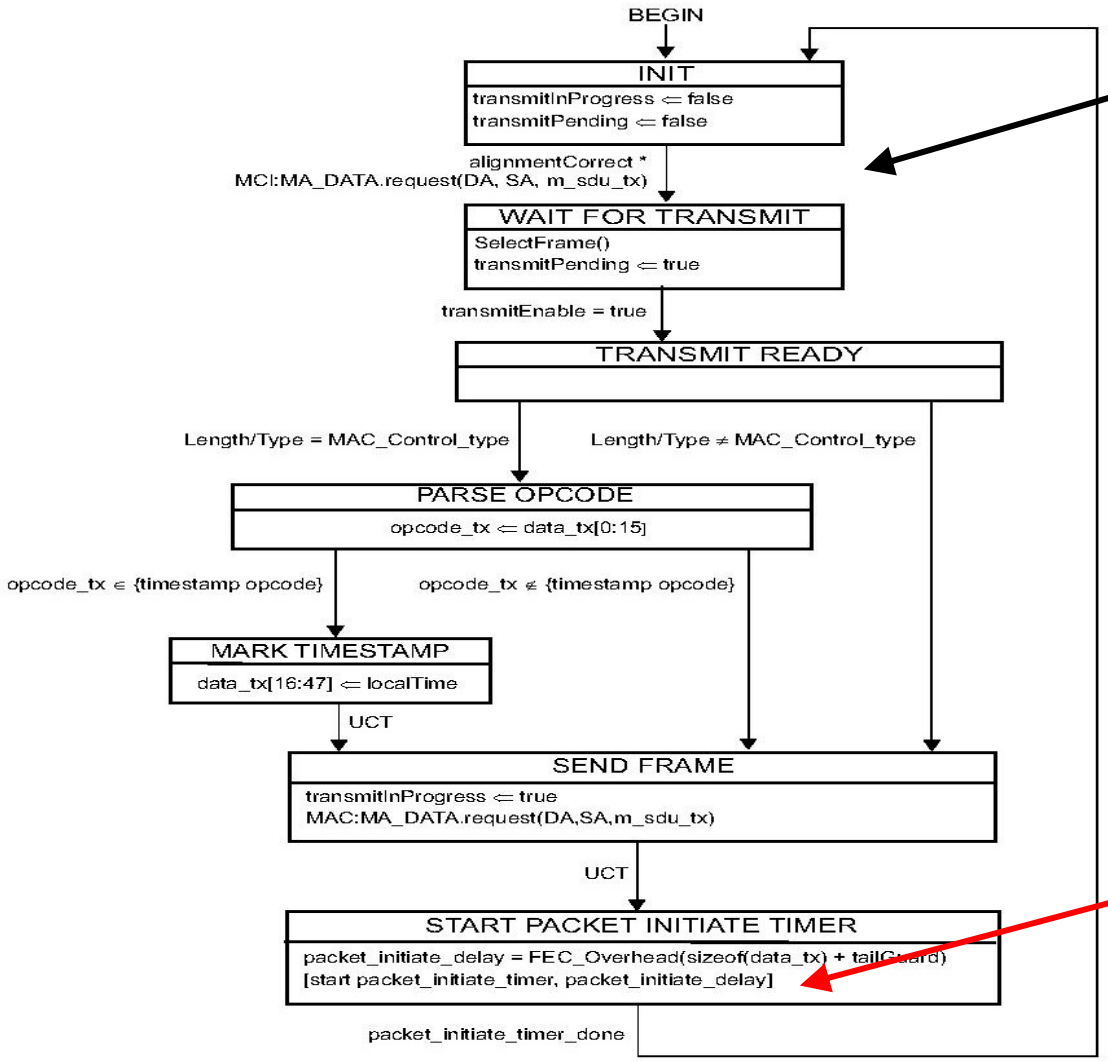
$$FEC\_Overhead(length) = L + \left\lfloor \frac{fecOffset + L}{FEC\_PAYLOAD\_SIZE} \right\rfloor \times FEC\_PARITY\_SIZE$$

where 'L' is the length of the frame rounded up to the nearest column boundary:

$$L = COLUMN\_SIZE \times \left\lceil \frac{length}{COLUMN\_SIZE} \right\rceil$$

- The delay of transmit completion to the column (ie. 4-byte) boundary in FEC_Overhead() is *not needed*:
  - DIC is not active at end-of-frame
  - The check of alignmentCorrect (after *WaitForTransmit* state in the OLT and after *TransmitReady* state in the ONU) *already guarantees* that the next frame will be sent during the PCS data transmission phase (rather than PCS parity transmission phase)

OLT control mux



2. If there is a real need for delay, it is done at this stage with alignmentCorrect – which ensures correct transmission each start of frame

1. **Currently, there is delay to column boundary at end of frame**

Problem and Remedy

- The MPCP logic for FEC handling should be clear and not include misleading or redundant steps

- Remedy:

  a) Restore the definition of FEC_Overhead() from draft 2.1 ie.

     - FEC_Overhead(length) = [(fecOffset/length)/FEC_PAYLOAD_SIZE] x FEC_PARITY_SIZE

  a) Restore the invocations of FEC_Overhead() from draft 2.1 ie.

     - Change the first line in *Start Packet Initiate Timer* State in figures 77-13 and 77-14 to:

packet_initiate_delay = FEC_Overhead(sizeof(data_tx) + tailGuard) + sizeof(data_tx) + tailGuard