FEC_Overhead considerations

Eric Lynskey

IEEE 802.3av Interim Meeting May 13-15, 2008

Munich, Germany

Current situation in D1.3

- FEC_Overhead equation
  - Note stating equation is broken (bad).
  - It does not properly account for overhead (bad).
  - Count in units of time_quanta (inefficient).

- Proposal for new equation
  - Return best case and worst case values
  - Return value in units of bytes

What is FEC_Overhead used for?

- OLT
  - Used to increase gap between frames to account for parity to be added by PCS (*packet_initiate_delay*).
- ONU
  - Used to increase gap between frames to account for parity to be added by PCS (*packet_initiate_delay*).
  - Used to determine if next frame will fit in the current burst (*nextTxTime*).
  - Used to determine maximum amount of time you can wait in a Discovery window before attempting transmission of REGISTER_REQ (*maxDelay*).

packet_initiate_delay

- Timer is seeded with packet_initiate_delay variable. This variable currently has units of time_quanta.
- Propose to change units to bytes.
- Propose to assume best case FEC overhead and allow some possible delay in MAC. Do not allow wasted bandwidth.

$$packet\_initiate\_delay = FEC\_Overhead\_Min\left(sizeof\left(data\_tx\right) + tailGuard\right)$$

NextTxTime

- Determines whether or not final frame will fit in burst.
- Currently has units of time_quanta.
- Propose to leave existing units.
- Propose to use worst case overhead calculation.

$$nextTxTime = \frac{\left(sizeof\left(data\_tx\right) + tailGuard\right) + FEC\_Overhead\_Max\left(sizeof\left(data\_tx\right) + tailGuard\right)}{tqSize}$$

maxDelay

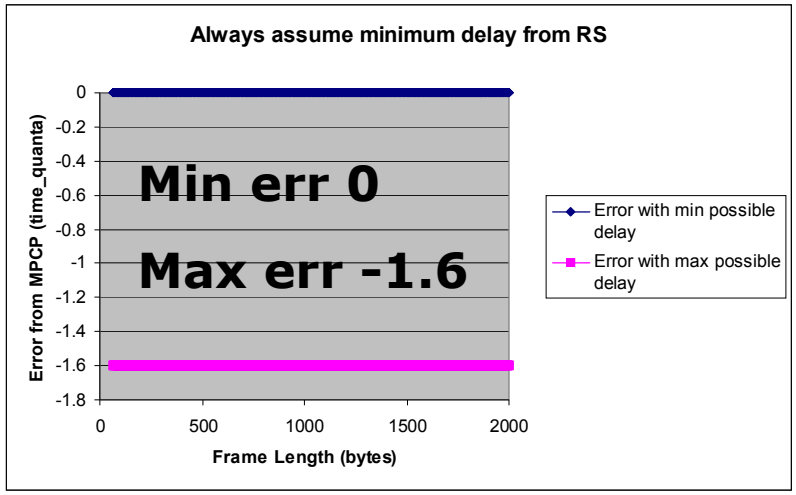
- Timer is seeded with value of maxDelay.
- There are no units associated with this variable.
- Propose to use worst case overhead calculation.

$$\max Delay = currentGrant.length - laserOn - syncTime - laserOff - discoveryGrantLength$$

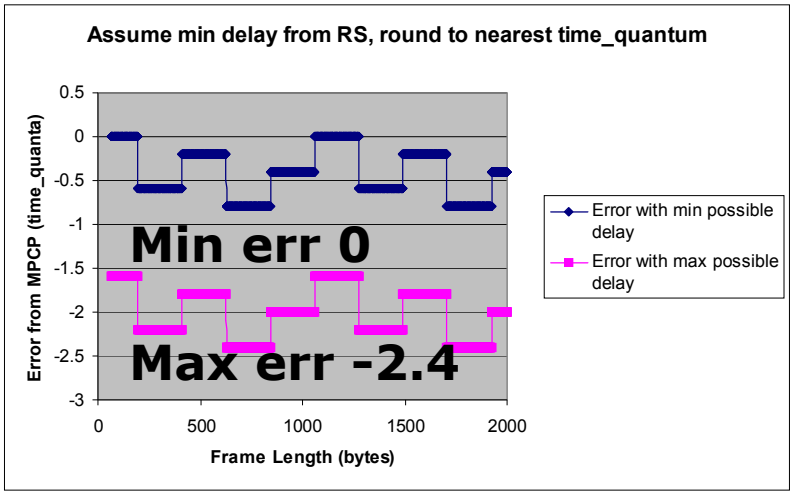
$$\max Delay = \max Delay - \frac{FEC\_Overhead\_Max(discoveryGrantLength \times tqSize)}{tqSize}$$

Four different options for calculating overhead

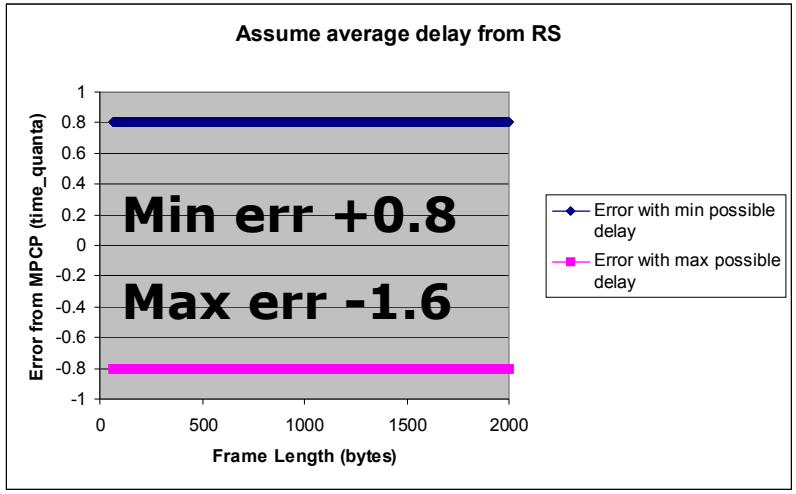
A



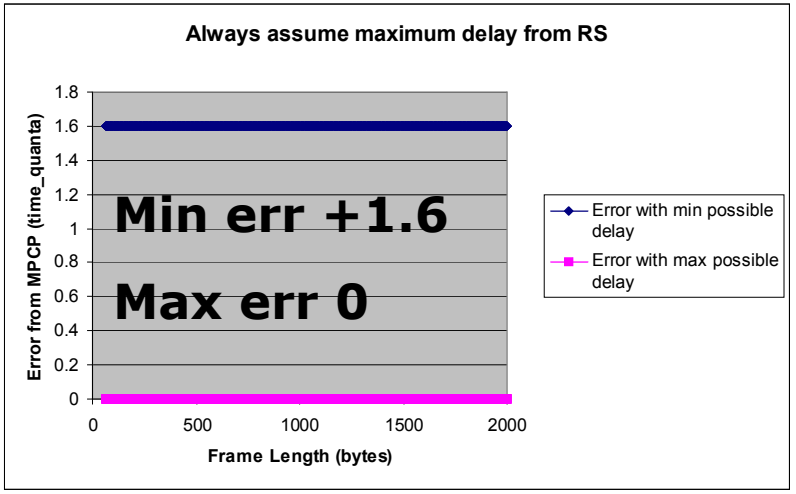
B



C



D



TEKNOVUS

Proposal for Option A (change units to bytes)

- FEC_Overhead equations for 10G-EPON become:

$$FEC\_Overhead\_Min = \left\lfloor \frac{frameLen + preLen + ipgLen}{colSize} \times \frac{1}{blockSize} \right\rfloor \times parityRatio$$

$$FEC\_Overhead\_Max = FEC\_Overhead\_Min + (parityRatio \times colSize)$$

Frame Length		FEC Overhead	
Min	**Max**	**Min (bytes)**	**Max (bytes)**
64	195	0	32
196	411	32	64
412	627	64	96
628	843	96	128
844	1059	128	160
1060	1275	160	192
1276	1491	192	224
1492	1707	224	256
1708	1923	256	288
1924	2000	288	320

```
frameLen   = bytes in frame
preLen     = bytes of preamble,       8
ipgLen     = bytes of IPG,           12
colSize    = bytes per column,        4
blockSize  = columns per FEC block,  54
parityRatio = columns of parity,     8
```


TEKNOVUS

Proposal for Option B (stick with time_quanta)

- FEC_Overhead equation for 10G-EPON becomes:

$$FEC\_Overhead = \left\lfloor \left\lfloor \frac{frameLen + preLen + ipgLen}{payloadLen} \right\rfloor \times \frac{colSize \times parityRatio}{tqSize} \right\rfloor$$

Frame Length		
Min	**Max**	**Overhead**
64	195	0
196	411	1
412	627	3
628	843	4
844	1059	6
1060	1275	8
1276	1491	9
1492	1707	11
1708	1923	12
1924	2000	14

```
frameLen   = bytes in frame
preLen     = bytes of preamble,       8
ipgLen     = bytes of IPG,           12
colSize    = bytes per column,        4
payloadLen = bytes per FEC block,    54
parityRatio = columns of parity,     8
tqSize   = bytes per time_quantum,   20
```


TEKNOVUS

Conclusion

- Go with Option A if group is willing to change units.
- Create FEC_Overhead_Min() and FEC_Overhead_Max() functions.
- The min overhead is used by OLT and ONU when adding IPG between frames.
- The max overhead is used by the ONU for determining if frame fits within burst and amount of time to wait during discovery.
- Modify equations as necessary.