Delay Variability Through MAC and PHY

Glen Kramer

Preface

- In July, we updated the FEC_Overhead() formula to more precisely calculate the overhead.

- Simulation revealed that delay variability can be made close to zero, but two additional problems need to be resolved first.

Updated FEC_Overhead()

- The following formula for calculating FEC overhead is proposed:

```
FEC_Overhead( length )
{
    byte_time ⇐ byte_time Mod FEC_CODEWORD_SIZE
    return FEC_PARITY_SIZE * ⌊(byte_time + length) / FEC_PAYLOAD_SIZE⌋
}
```

Where:

FEC_PAYLOAD_SIZE = payload size in octets
Value: 216 (FEC_DSize * 8)

FEC_PARITY_SIZE = parity size in octets
Value: 32 (FEC_PSize * 8)

FEC_CODEWORD_SIZE = FEC codeword size in octets
Value: FEC_PAYLOAD_SIZE + FEC_PARITY_SIZE

Problem #1

- **Idle Deletion process cannot remove the necessary number of IDLE blocks**

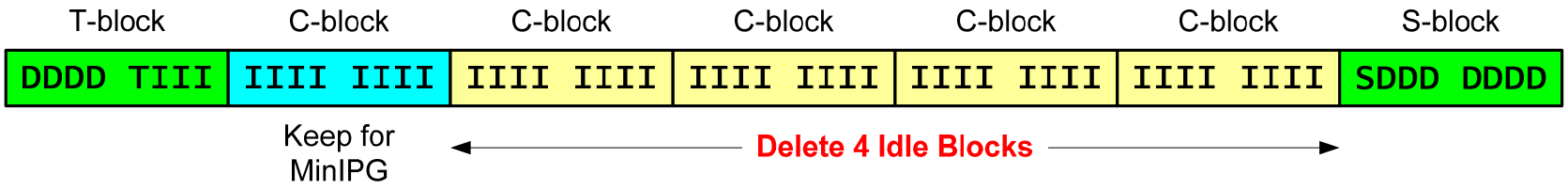
- What happens?
  - Idle deletion keeps one block full of Idles characters as min. ipg and tries to delete additional blocks full of Idles. Since Idle characters may occupy portions of T and S blocks, Idle Deletion process may not be able to remove the necessary number of Idles.
- Why is this bad?
  - If Idle Deletion does not remove the sufficient number of Idles, the FIFO_DD buffer in Data Detector quickly overflows and loses data.

Example of Problem #1

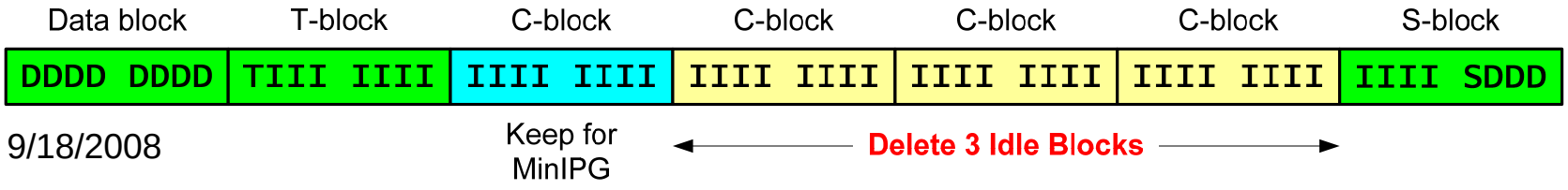
- MinIPG = 12 bytes
- FEC Overhead = 32 bytes
- Total gap = 44 bytes

- In this example, Idle Deletion is supposed to delete 4 blocks (32 bytes) of Idles

(a) IPG = 44 Idle Characters. Idle deletion process can remove 4 idle blocks.



(b) Same number of Idle Characters. Idle deletion can remove only 3 blocks.



Solution to Problem #1

- Modify MPCP calculation such that IPG is extended to guarantee that the sufficient number of idle blocks can always be removed.

- How?
  - Round up the (frame_length + min_ipg) to 72-bit block boundary

```
int16s length = 8 * ⌈(sizeof(data_tx) + TAIL_GUARD )/8⌉

packet_initiate_timer = length + FEC_Overhead(length)
```

Problem #2

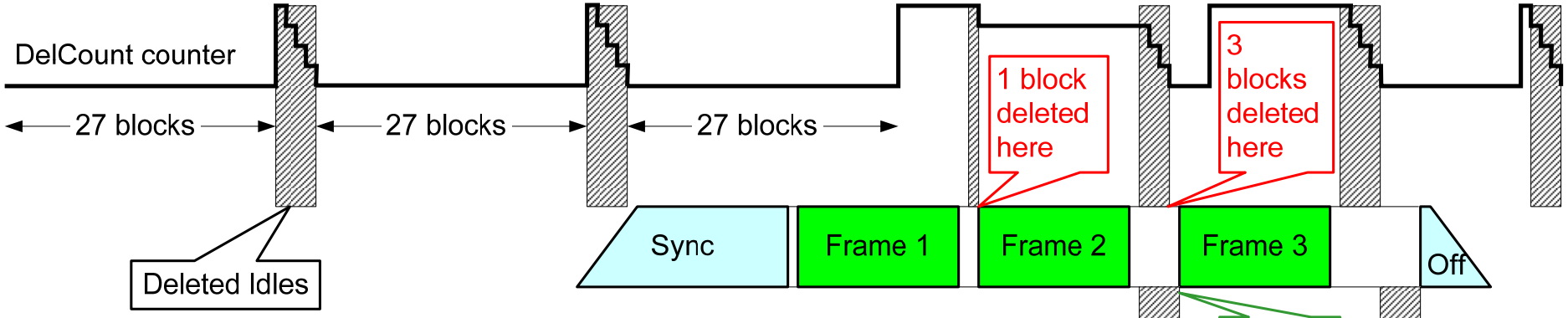
- **Increased delay variability (~4 TQ) is observed in the upstream.**

- Analysis has revealed the reason is the misalignment between where Idles are removed by Idle Deletion process and where parity is inserted by the Data Detector process.

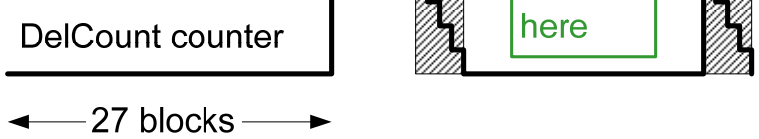
Details of Problem #2

- Counters **VectorCount** and **DelCount** determine when the idles are deleted to allow parity insertion.
- In current spec, VectorCount and DelCount run continuously from the initialization, but the FEC codeword boundary is reset at the beginning of every burst.
- This can lead to increased delay variability for some frames, as shown below.

(a) Current situation



(b) Corrected situation



Simulation results: MPCPDU delay

- 100K packets; size 64…2000 bytes

Function	Downstream			Upstream		
	Min delay (TQ)	Max delay (TQ)	Max drift (TQ)	Min delay (TQ)	Max delay (TQ)	Max drift (TQ)
MPCP_TX	0.0	0.0	0.0	0.0	0.0	0.0
MAC_TX	**0.2**	**0.2**	0.0	0.0	0.0	0.0
XGMII_TX	**0.2**	**0.2**	0.0	0.0	0.0	0.0
IDLE_DEL	0.0	0.0	0.0	**0.4**	**0.4**	0.0
66B_ENCODER	0.0	0.0	0.0	0.0	0.0	0.0
SCRAMBLER	0.0	0.0	0.0	0.0	0.0	0.0
DATA_DET	0.0	0.0	0.0	**24.4**	**28.4**	**4.0**
FEC_DECODER	**12.4**	**12.4**	0.0	**12.4**	**12.4**	0.0
DESCRAMBLER	0.0	0.0	0.0	0.0	0.0	0.0
66B_DECODER	0.0	0.0	0.0	0.0	0.0	0.0
IDLE_INS	**16.0**	**16.0**	0.0	**16.0**	**16.0**	0.0
XGMII_RX	0.0	0.0	0.0	0.0	0.0	0.0
MAC_RX	**3.6**	**3.6**	0.0	**3.6**	**3.6**	0.0
MPCP_RX	0.0	0.0	0.0	0.0	0.0	0.0
TOTAL	**32.4**	**32.4**	**0.0**	**56.8**	**60.8**	**4.0**

Solution to Problem #2

Modify Code in state CLASSIFY_VECTOR_TYPE

```
if( IdleCount > DelayBound )
{
   HalfShift   ⇐ FALSE
   VectorCount ⇐ 2
   DeleteCount ⇐ 0

   if( T_TYPE(tx_raw<71:36>) = S )
   {
        tx_next   ⇐ IDLE
        HalfShift ⇐ TRUE
        IdleCount ⇐ 0
   }
}

if( HalfShift )
{
   tx_temp<35:0> ⇐ tx_raw<71:36>
   tx_raw<71:36> ⇐ tx_raw<35:0>
   tx_raw<35:0>  ⇐ tx_next<35:0>
   tx_next<35:0> ⇐ tx_temp<35:0>
}
```

Between bursts keep resetting the starting position of FEC Codeword in anticipation that a new burst will start at the next clock

New simulation results

- 100K packets; size 64…2000 bytes

Function	Upstream		
	Min delay (TQ)	Max delay (TQ)	Max drift (TQ)
MPCP_TX	0.0	0.0	0.0
MAC_TX	0.0	0.0	0.0
XGMII_TX	0.0	0.0	0.0
IDLE_DEL	**0.4**	**0.4**	0.0
66B_ENCODER	0.0	0.0	0.0
SCRAMBLER	0.0	0.0	0.0
DATA_DET	**26.0**	**26.0**	0.0
FEC_DECODER	**12.4**	**12.4**	0.0
DESCRAMBLER	0.0	0.0	0.0
66B_DECODER	0.0	0.0	0.0
IDLE_INS	**16.0**	**16.0**	0.0
XGMII_RX	0.0	0.0	0.0
MAC_RX	**3.6**	**3.6**	0.0
MPCP_RX	0.0	0.0	0.0
TOTAL	**58.4**	**58.4**	**0.0**