Start-of-Frame Alignment Within 66-bit Block

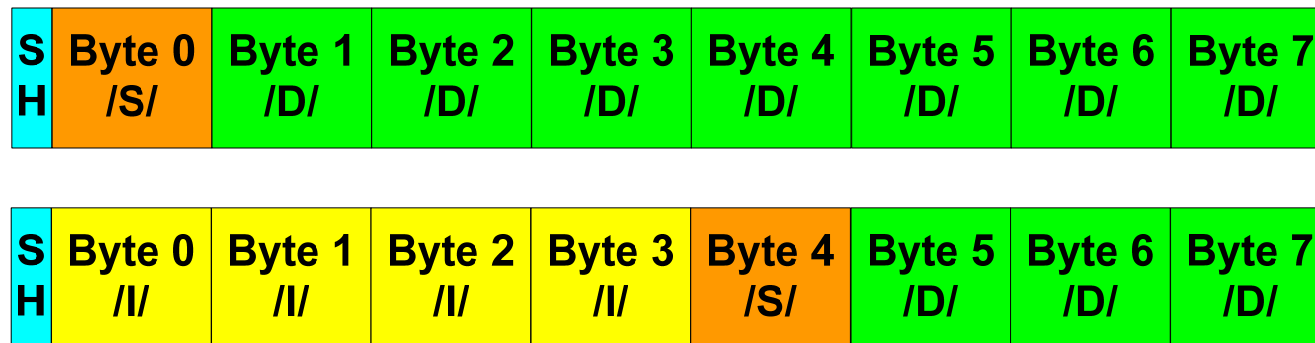Glen Kramer, glen.kramer@teknovus.com
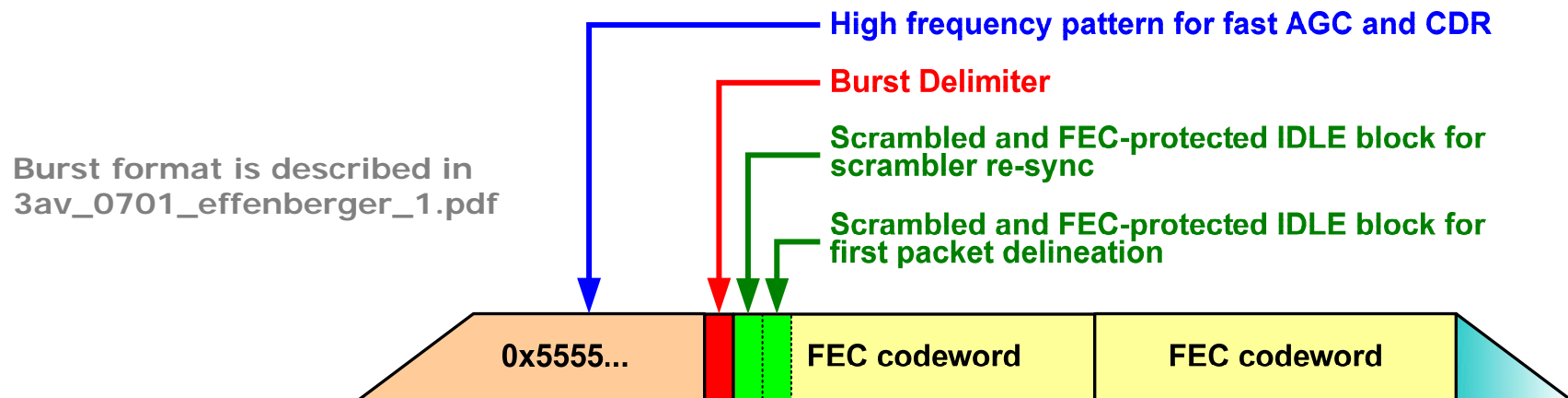
Overview and Definitions

- In PCS, two 36-bit XGMII words are combined into one 72-bit vector (tx_raw<71:0>).

- tx_raw<71:0> is then encoded into 66-bit block

- In XGMII word, /S/ is always aligned to lane 0.

- Within 66-bit block, /S/ may appear in byte 0 or byte 4.

66-bit block

S H	Byte 0 /S/	Byte 1 /D/	Byte 2 /D/	Byte 3 /D/	Byte 4 /D/	Byte 5 /D/	Byte 6 /D/	Byte 7 /D/

S H	Byte 0 /I/	Byte 1 /I/	Byte 2 /I/	Byte 3 /I/	Byte 4 /S/	Byte 5 /D/	Byte 6 /D/	Byte 7 /D/

Position of /S/ Affects Grant Size

- FEC coverage starts two 66-bit clocks ahead of the block containing /S/
  - First block of IDLEs is to synchronize the scrambler
  - Second block IDLEs is required for proper SOP detection

High frequency pattern for fast AGC and CDR

Burst Delimiter

Scrambled and FEC-protected IDLE block for scrambler re-sync

Burst format is described in 3av_0701_effenberger_1.pdf

Scrambled and FEC-protected IDLE block for first packet delineation

| 0x5555... | | | FEC codeword | FEC codeword | |

- When /S/ is located in position 4, the transmission length increases by 4 bytes
- MPCP does not know if /S/ will happen to be in position 0 or in position 4 when it is time to transmit, so MPCP must always report 4 extra bytes
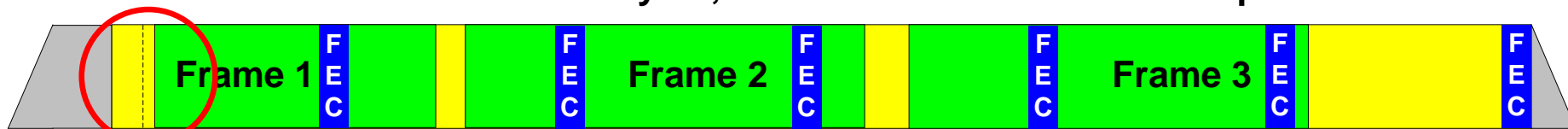
Effect of Reporting Extra Bytes (1)

- If queued frames fit in **N** FEC codewords with less then 4 spare bytes left, then adding 4 extra bytes would require the OLT to additionally allocate full FEC codeword

- In reality, the first /S/ may end up in byte 0, and the ONU would still only send **N** codewords, however the OLT would allocate **N+1** codewords according to previous REPORT

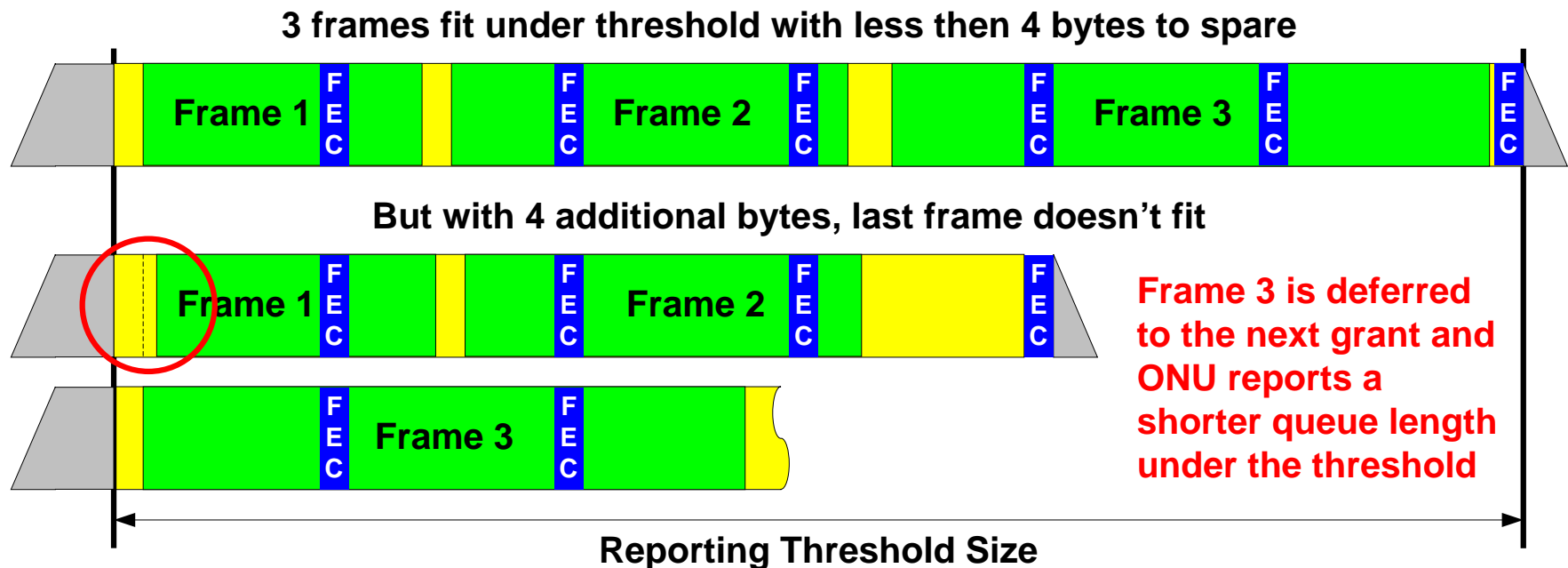**3 frames fit in N FEC codewords with less then 4 bytes to spare**

But with 4 additional bytes, an additional codeword is required

Extra (and almost empty) codeword is transmitted

Effect of Reporting Extra Bytes (2)

- If queued frames fit in **N** FEC codewords with less then 4 spare bytes left, and these **N** codewords add up exactly to the reporting threshold, then adding 4 extra bytes would make the last frame not fit in the current grant

- The ONU will not include the last frame in its REPORT and the OLT will grant a smaller slot. The last frame will wait for the next grant.

3 frames fit under threshold with less then 4 bytes to spare

But with 4 additional bytes, last frame doesn't fit

Frame 3 is deferred to the next grant and ONU reports a shorter queue length under the threshold

Reporting Threshold Size

Different Alignment Strategies

- **No Alignment**
  - All /S/ characters can be in either position 0 or 4.
  - The state machine at the top of PCS treats all frames identically.
  - Some bandwidth maybe lost.
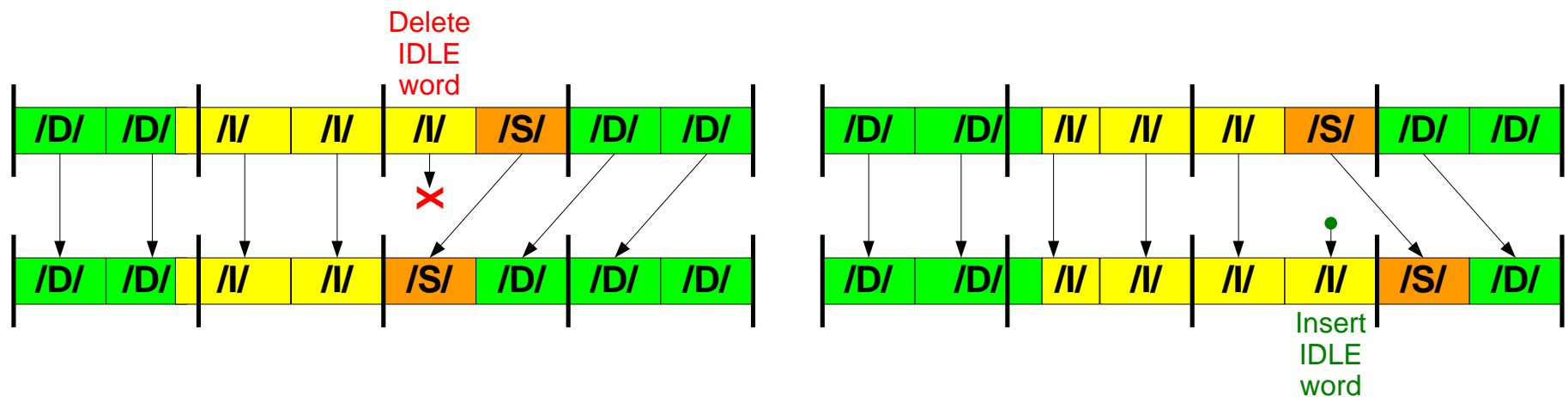
- **Aligning First Frame**
  - Only the first /S/ character is aligned to the 0-position. All consecutive frames may start in either position 0 or position 4.
  - The state machine at the top of PCS should distinguish some frames from the others.

- **Aligning All Frames**
  - All /S/ characters are aligned to the 0-position.
  - The state machine at the top of PCS treats all frames identically.

Aligning All Frames

- Due to operation of Deficit Idle Counter (DIC), IPG includes 2 or 3 full XGMII words plus possibly a fraction of a word.

- If /S/ is in byte 4 and should be aligned to position 0, an XGMII IDLE word should either be inserted or deleted, while maintaining a minimum of 2 full XGMII words of IDLE characters
    - If IPG has only 2 full XGMII words, must insert IDLE word
    - IF IPG has 3 full XGMII words, may insert or delete an additional IDLE word.
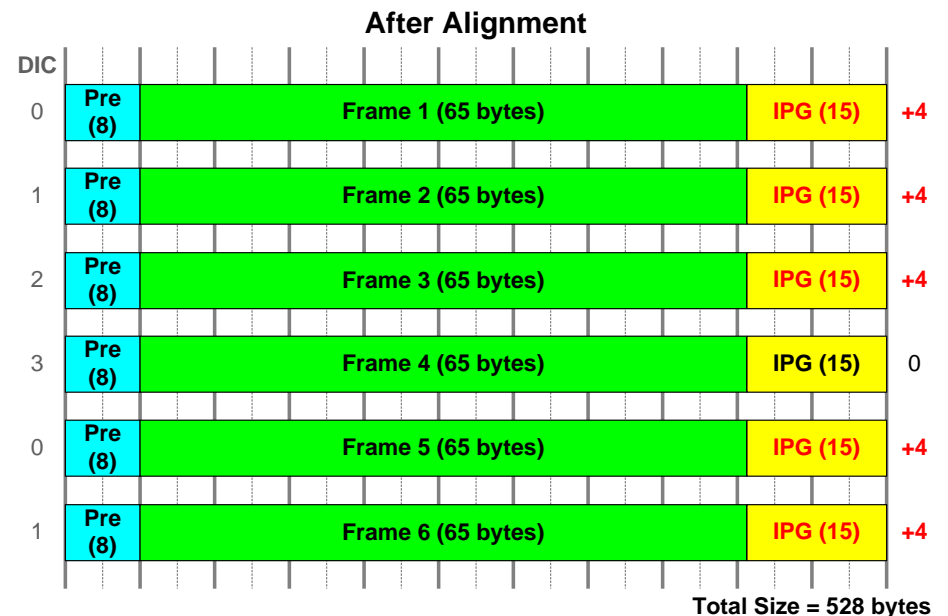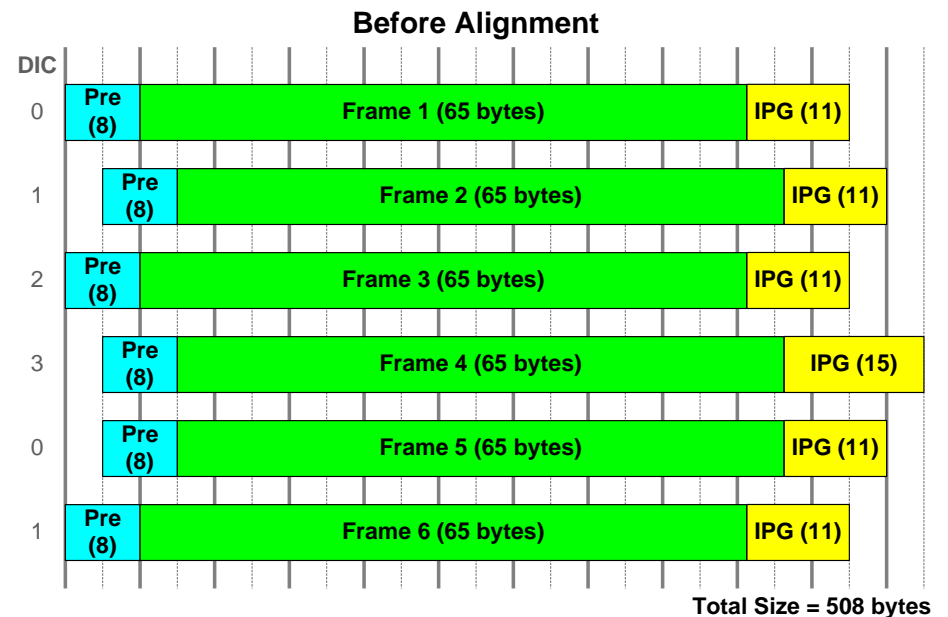
Aligning All Frames – IPG Inflation

- Some frame sequences have a long run of 2-word IPGs and all require alignment. When such sequences are aligned, the net effect is IPG Inflation, where many IPGs in a row **must** have an additional IDLE word inserted.

  – For example, a sequence of 65-byte frames is an unlucky one (or any frames with size MOD 4 = 1)

- In this example, 6 frames, after alignment required 20 bytes more than before the alignment

Before Alignment

DIC

0	Pre (8)	Frame 1 (65 bytes)	IPG (11)
1	Pre (8)	Frame 2 (65 bytes)	IPG (11)
2	Pre (8)	Frame 3 (65 bytes)	IPG (11)
3	Pre (8)	Frame 4 (65 bytes)	IPG (15)
0	Pre (8)	Frame 5 (65 bytes)	IPG (11)
1	Pre (8)	Frame 6 (65 bytes)	IPG (11)

Total Size = 508 bytes

After Alignment

DIC

0	Pre (8)	Frame 1 (65 bytes)	IPG (15)	+4
1	Pre (8)	Frame 2 (65 bytes)	IPG (15)	+4
2	Pre (8)	Frame 3 (65 bytes)	IPG (15)	+4
3	Pre (8)	Frame 4 (65 bytes)	IPG (15)	0
0	Pre (8)	Frame 5 (65 bytes)	IPG (15)	+4
1	Pre (8)	Frame 6 (65 bytes)	IPG (15)	+4

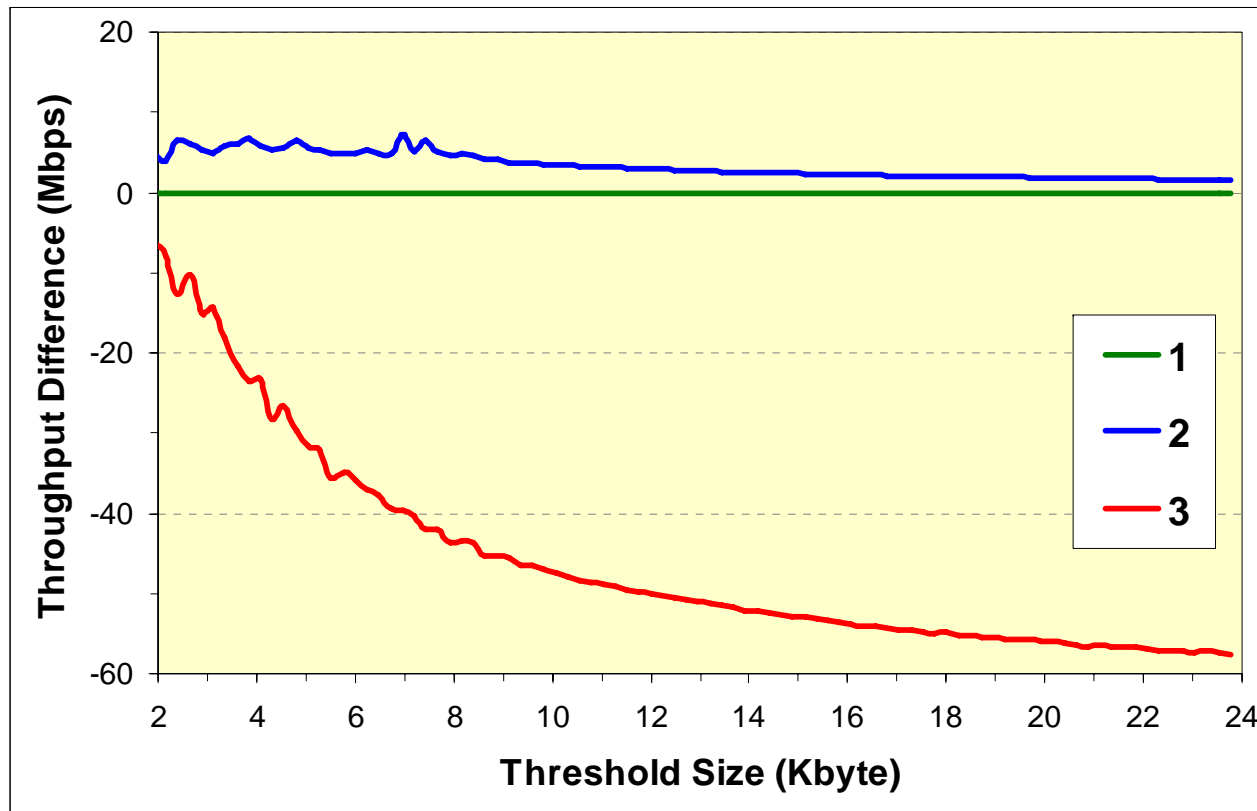**Total Size = 528 bytes**

Accommodating All-Frame Alignment Scheme

- MPCP cannot predict the value of Deficit Idle Counter or the initial positions of /S/ characters, therefore it cannot calculate which frame will have an increased IPG and how many such frames there will be.

- To accommodate alignment of all frames, MPCP should assume the worst case and add 4 bytes for every frame.

- However, this IPG inflation reduces data throughput.

Simulation Results

- The graph shows throughput differences for different schemes as a function of threshold size:

  1. $\text{Throughput}_{\text{No Alignment}}$ - $\text{Throughput}_{\text{No Alignment}}$ (= 0)

  2. $\text{Throughput}_{\text{Align 1}^{\text{st}}\text{ Frame}}$ - $\text{Throughput}_{\text{No Alignment}}$ (< 8 Mbps)

  3. $\text{Throughput}_{\text{Align All Frames}}$ - $\text{Throughput}_{\text{No Alignment}}$ (< 0)

Simulation parameters

- Threshold sizes varied from 7 to 100 FEC codewords (from 1.68 to 24 Kbytes).

- 10 million packets were transmitted for each threshold size.

- System was under heavy load, i.e., there was unlimited supply of packets in each queue.

- Packets followed empirical tri-modal distribution as shown in http://www.ieee802.org/3/efm/public/jul01/presentations/sala_1_0701.pdf.

Summary

- Aligning all /S/ characters to byte 0 decreases performance.

- Aligning only the first /S/ to byte 0 shows a negligible increase of performance (< 8 Mbps)

- What is better:
  - Align the /S/ character of the first frame and save up to 8 Mbit/s of bandwidth

  Or

  - Don't align any characters and have simpler state machines?