MAC/PHY Delay Variability

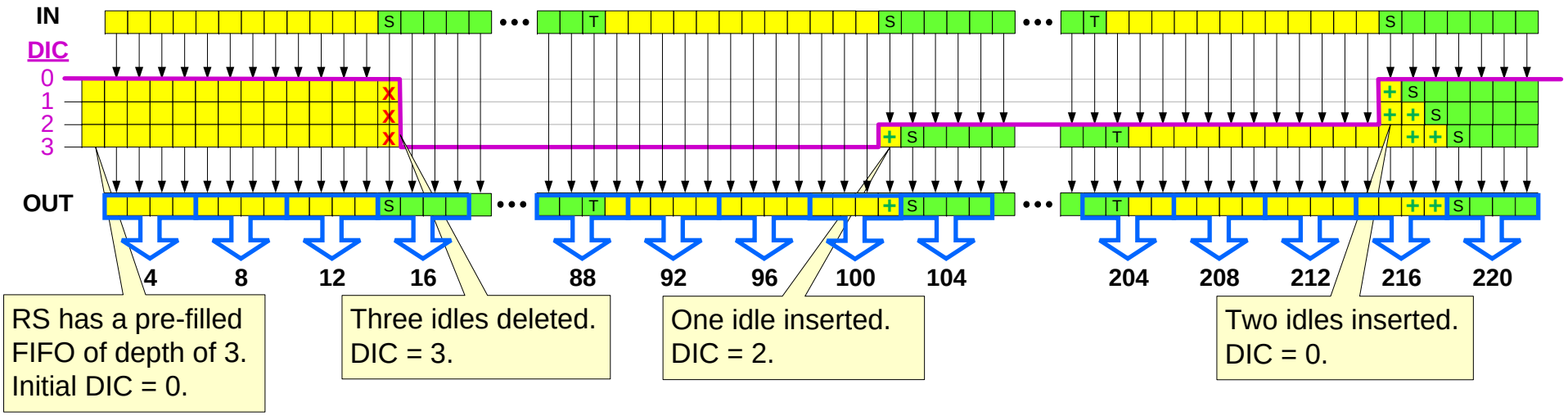
glen.kramer@teknovus.com

Deficit Idle Counter

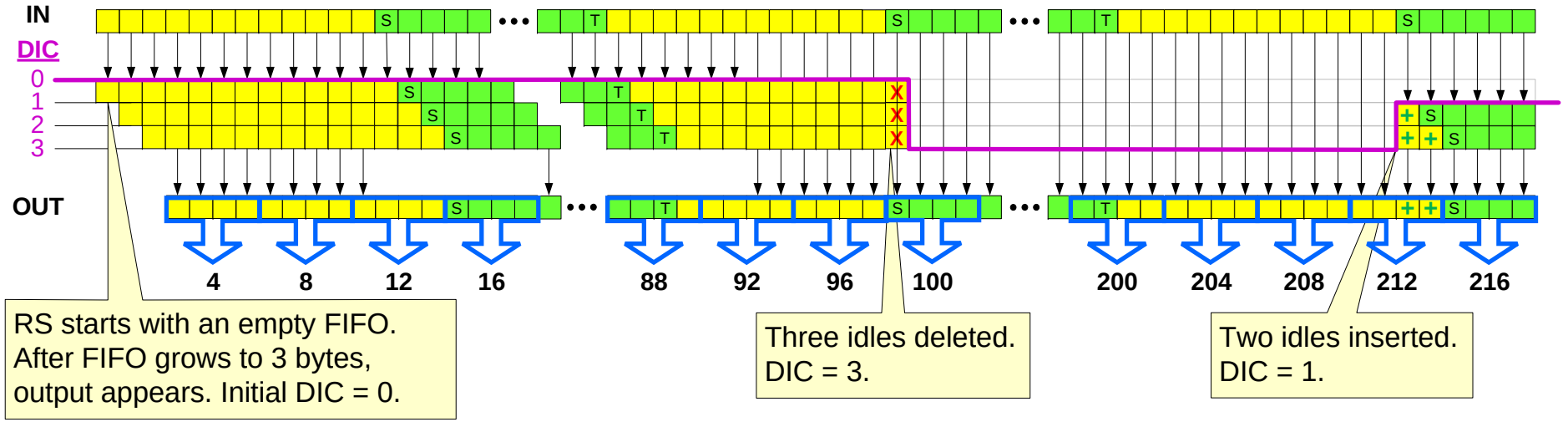
- There are several ways to implement Deficit Idle Counter.
  - Option 1: RS has a pre-filled FIFO
  - Option 2: RS has an empty FIFO
  - Option 3: DIC integrated with MAC
  - Other?
- All of the above options achieve the goal of balancing the average number of transmitted idles. But their interactions with MPCP are different.

Implementations of Deficit Idle Counter

Option 1: RS with a pre-filled FIFO



Option 2: RS with an empty FIFO



FEC_Overhead may be simplified

- If Deficit Idle Counter is implemented as option 2, then the frames will appear to either pass through RS or shift left by n = [0:3] bytes, but never shift right (i.e., IPG will increase only if it has decreased earlier).

- With Option 2, FEC_Overhead() may be simplified:

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

- Also, <u>in the downstream direction</u> alignmentCorrect may be simplified to

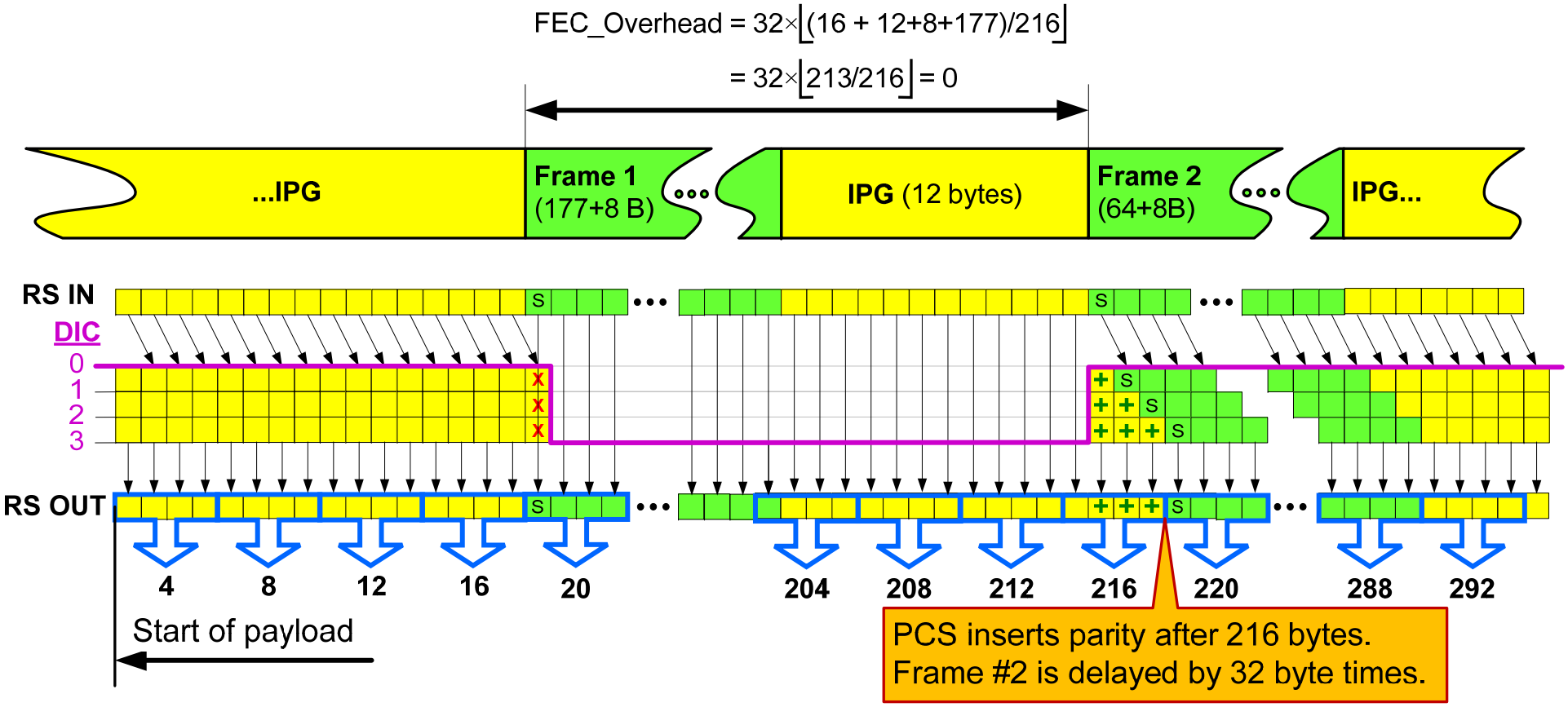
$$alignmentCorrect = (fecOffset < FEC\_PAYLOAD\_SIZE)$$

Upstream Direction is Different

- DIC value is carried over between bursts.
- It is possible that in the previous burst, the IPG between frames has decreased and that caused (relative) IPG increase in the next burst.
- But at the beginning of every burst the FEC alignment is reset, triggered by /S/ of the first frame
  - Previous IPG decrease has no effect on the current burst.
  - If the first frame shifted left, so is the entire FEC alignment.
- IPG increase in the current burst may introduce unacceptable delay variability.

Example of IPG increase

- First frame (177 bytes) is shifted left (IPG decreased), and so is the FEC alignment for the entire burst.
- When IPG before the second frame is increased, previous IPG decrease has no effect. The parity is inserted in front of frame 2 causing 32 byte time delay variability.



Solution for Upstream Problem

- FEC_Overhead() can be simplified as shown on slide 4.

- Definition of alignmentCorrect must ensure that no frames are started when fecOffset is 213, 214, or 215, because it is possible that due to Deficit Idle Counter, such frames will move to byte 216 and parity will be inserted in front of it.
  - **Option 1 (as in D2.2):**

$$alignmentCorrect = (fecOffset < FEC\_PAYLOAD\_SIZE) \text{ and } (fecOffset[1:0] = 0)$$

  - **Option 2:**

$$alignmentCorrect = (fecOffset < FEC\_PAYLOAD\_SIZE - 3)$$