Latency Analysis

William Lo

10 January 2024

Summary of Current Discussions and Findings

- Want single 100Mb/s solution to work at
  - Servos – 100 meters, < 1.5us latency
  - Long reach – 500 meters, latency is less important
- PAM3 vs PAM4 roughly equal on performance at 500 meters
- Turn off RS-FEC to reduce latency for low latency applications
  - Transmit encoding looks the same for long reach and low latency
- Volumes
  - Low latency > 10M ports/year (servo)
  - Short reach ~ 2-3M (windfarm spurs)
  - Long reach ~ 100K ports/year (windfarm trunks)

  https://www.ieee802.org/3/SPEP2P/public/SPE_long_term_cfi.pdf
- Some discussions on RS-FEC, Block encoding, Bounded Disparity

What Has Not Been Discussed

- Focus has been on long reach (1% of the market) with the assumption RS-FEC is turned off at receiver for low latency (99% of the market)
  - Are we considering the correct tradeoffs between latency and bandwidth

- No analysis on block coding latency
- No analysis on RS-FEC buffering latency that is incurred with FEC turned off
- How does intrinsic safety requirements translate to bounded disparity
  - Is there a line model we can use judge different schemes

Latency Discussion

- The following is based on latency discussions in

https://www.ieee802.org/3/bp/public/jul14/Lo_3bp_01a_0714.pdf

- Algorithm latency is the minimum theoretical latency

- Implementation latency assumed to be 0 in current discussion as this is vendor dependent

Latency Definitions

- **Algorithmic Latency**
  - Amount of time waiting to collect data before algorithm can be applied
    - Aggregate data in 8N/(8N+1) encoder
    - RS TX data delay to avoid underflow
    - RS RX frame aggregation
- **Implementation Latency**
  - Circuit latency
    - Pipelining, FIFOing
    - RS parity computation
    - RS Error correction
    - DSP processing
    - Circuit propagation delays
- **Total Latency = Algorithmic + Implementation for round trip**
  - GMII → TX → RX → GMII

IEEE 802.3bp RTPGE – July 2014 Plenary Meeting 3

MARVELL

AXONNE

Algorithm Latency of Low Latency PHY with FEC correction turned off using long reach PCS coding

- Algorithm Delay = A + B + C + D where
  - A = Block encoder latency
  - B = RS encoder underflow prevention
  - C = Symbol conversion at receiver needed for bounded disparity
  - D = Block decoder latency (this cannot be 0 since there is no delay in FEC to take advantage of)
- Encoder latency
  - 64/65 – need to get all 64 bits from MII before we know how to encode it.

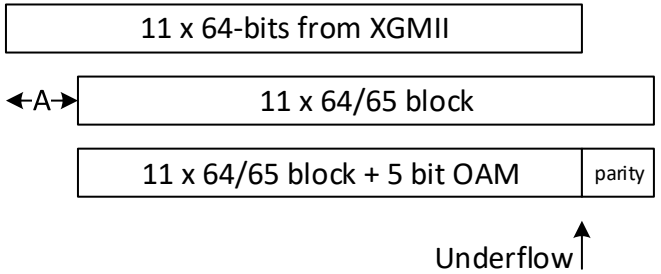
    64 x 10ns = 640ns
  - 80/81 – 80 x 10ns = 800ns

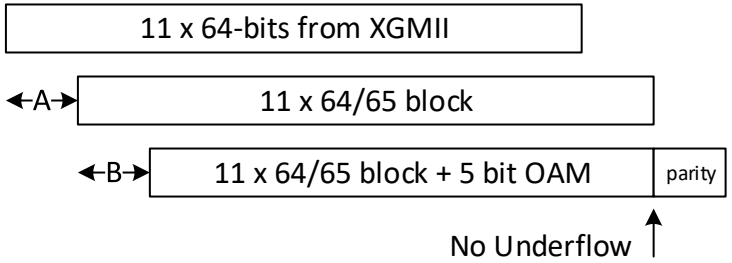
Algorithm Latency of Low Latency PHY with FEC correction turned off using long reach PCS coding

- RS Encoder Underflow prevention
  - Need to delay the duration parity is transmitted
    - Actually slightly less since OAM bits are stuffed in but ignore this to simplify analysis
  - Use example from slide 5 below, first option

    64/65 coding , RS(96, 90) GF(2^8), 8b/10b bounded disparity, PAM4, 68.1818 Mbaud

    https://www.ieee802.org/3/dg/public/May_2022/Tingting_3dg_01_25_10_2023.pdf
  - 6 symbols x 8-bit symbol x (10/8) x (1/2) / 68.1818 Mbaud = 440ns





Algorithm Latency of Low Latency PHY with FEC correction turned off using long reach PCS coding

- 8b/10 symbol conversion
  - 5 symbols = 10-bits (5 /68.1818 Mbaud) = 73.3ns
- Decoder latency
  - 64/65 – Worst case need to wait until 17th bit of 64/65 to start byte decoding
  - Get 8 bits at a time so worst case is 3x8=24 bits to get to the 17th bit
  - 15 symbols = 24 bits (15 / 68.1818 Mbaud) = 220ns

Input Data	data ctrl header	Block Payload							
Bit Position: / Data Block Format:	0	1							64
D_0 D_1 D_2 D_3/D_4 D_5 D_6 D_7	0	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
Control Block Formats:		Block							
C_0 C_1 C_2 C_3/C_4 C_5 C_6 C_7	1	0x1E	C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7
C_0 C_1 C_2 C_3/O_4 D_5 D_6 D_7	1	0x2D	C_0	C_1	C_2	C_3	O_4	D_5	D_6	D_7
C_0 C_1 C_2 C_3/S_4 D_5 D_6 D_7	1	0x33	C_0	C_1	C_2	C_3		D_5	D_6	D_7
O_0 D_1 D_2 D_3/S_4 D_5 D_6 D_7	1	0x66	D_1	D_2	D_3	O_0		D_5	D_6	D_7
O_0 D_1 D_2 D_3/O_4 D_5 D_6 D_7	1	0x55	D_1	D_2	D_3	O_0	O_4	D_5	D_6	D_7
S_0 D_1 D_2 D_3/D_4 D_5 D_6 D_7	1	0x78	D_1	D_2	D_3	D_4		D_5	D_6	D_7
O_0 D_1 D_2 D_3/C_4 C_5 C_6 C_7	1	0x4B	D_1	D_2	D_3	O_0	C_4	C_5	C_6	C_7
T_0 C_1 C_2 C_3/C_4 C_5 C_6 C_7	1	0x87		C_1	C_2	C_3	C_4	C_5	C_6	C_7
D_0 T_1 C_2 C_3/C_4 C_5 C_6 C_7	1	0x99	D_0		C_2	C_3	C_4	C_5	C_6	C_7
D_0 D_1 T_2 C_3/C_4 C_5 C_6 C_7	1	0xAA	D_0	D_1		C_3	C_4	C_5	C_6	C_7
D_0 D_1 D_2 T_3/C_4 C_5 C_6 C_7	1	0xB4	D_0	D_1	D_2		C_4	C_5	C_6	C_7
D_0 D_1 D_2 D_3/T_4 C_5 C_6 C_7	1	0xCC	D_0	D_1	D_2	D_3		C_5	C_6	C_7
D_0 D_1 D_2 D_3/D_4 T_5 C_6 C_7	1	0xD2	D_0	D_1	D_2	D_3	D_4		C_6	C_7
D_0 D_1 D_2 D_3/D_4 D_5 T_6 C_7	1	0xE1	D_0	D_1	D_2	D_3	D_4	D_5		C_7
D_0 D_1 D_2 D_3/D_4 D_5 D_6 T_7	1	0xFF	D_0	D_1	D_2	D_3	D_4	D_5	D_6	

Total Algorithm Latency of Example

- Encoder latency (64/65) = 640ns
- RS encoder underflow prevention = 440ns
- 8b/10 Symbol conversion at receiver = 73.3
- Decoder latency (64/65) = 220ns
  - Realistically the entire block is decoded so will wait 640ns
- Total algorithm latency = 1373.3 ns

- Margin left for implementation = 126.7ns = 1500ns – 1373.3ns
  - Margin most likely not sufficient for implementation
  - And no benefit of FEC correction

- Cannot ignore encoder and decoder latency!
  - 62.6% of total algorithm latency! (640+220)/1373.3

Bounded Disparity

- Do we really need it with long scrambler sequence
  - 100BASE-TX had issues with short scrambler and killer packet
- What is the probability of creating a long unbalanced run?
- How long is too long?
- High bandwidth overhead to implement
  - 8b/10b is 25% overhead
- Is 8b/10b followed by PAM 4 coding really bounded disparity

  ie. D21.5 = 1010101010 every 2-bit converted to PAM 4 results in no transitions

  ie. D10.2 = 0101010101
- If needed, is there a better way to bound PAM4 disparity

Proposal

- Focus on optimizing low latency PHY since it is 99% of the market.
- Put the burden on the long reach PHY cost if vendor wants to implement dual mode.
  - Do we abandon the one solution fits all?
  - It is ok to have 2 solutions if we focus on keeping the expensive components of the PHY as similar as possible.

- It is possible to have some FEC protection and meet low latency targets
  - Do we want FEC protection for low latency?
  - Details to be presented at next interim meeting with and without bounded disparity

THANK YOU