MTTFPA Analysis and Updates for Stateless 64B/66B Coding

Eugene Opsasnick – Broadcom

802.3df December 2022

Supporters

- Shimon Muller, Enfabrica
- Mark Gustlin, Cisco
- Adee Ran, Cisco
- Xinyuan Wang, Huawei
- Xiang He, Huawei
- Gary Nicholl, Cisco
- Shawn Nicholl, AMD
- Jerry Pepper, Keysight
- Eric Maniloff, Ciena
- David Ofelt, Juniper Networks
- Kent Lusted, Intel Corporation

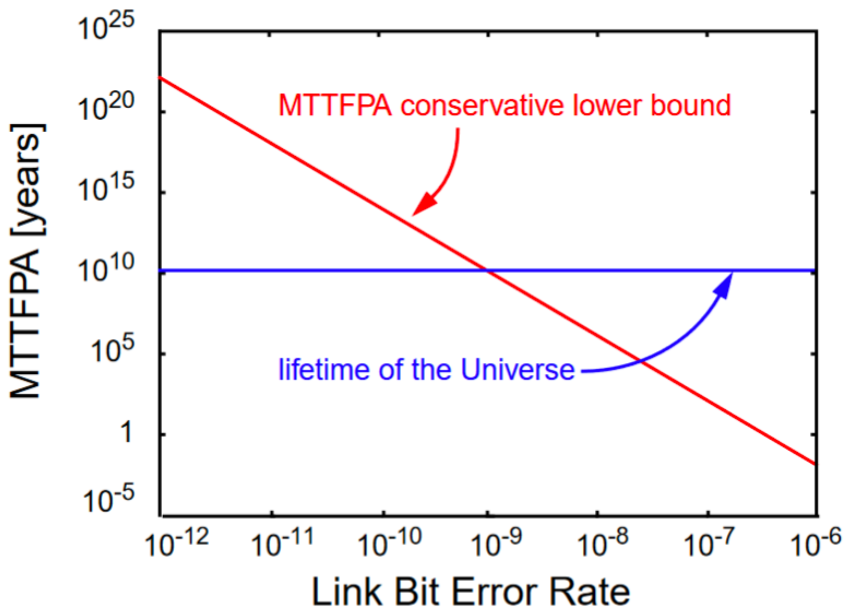
Introduction

- In opsasnick_3df_01a_221005, it was presented that the state machines used to follow 64B/66B coding block transitions and do the substitution of EBLOCKS are getting harder to implement for faster port speeds due to ever-wider datapaths for 800GbE and 1.6TbE.
- A few assertions regarding error detection were made:
  1. The state machines were created before RS-FEC was introduced.
  2. The state machines were designed to provide a hamming distance of 4 (same as CRC32) for catching errors in the 66B block field types and sync headers.
  3. Newer interfaces that require RS-FEC have better protection of the block field type, and no longer require the state machines for error protection.
- This presentation attempts to prove the validity of these assertions and proposes a new, optional, alternative to the 64B/66B coding state machines that remains compatible with them.

Background on MTTFPA Analysis

- The metric generally used in previous 802.3 projects is that the Mean Time to False Packet Acceptance (MTTFPA) should be greater than the lifetime of the universe (13.8 billion years).

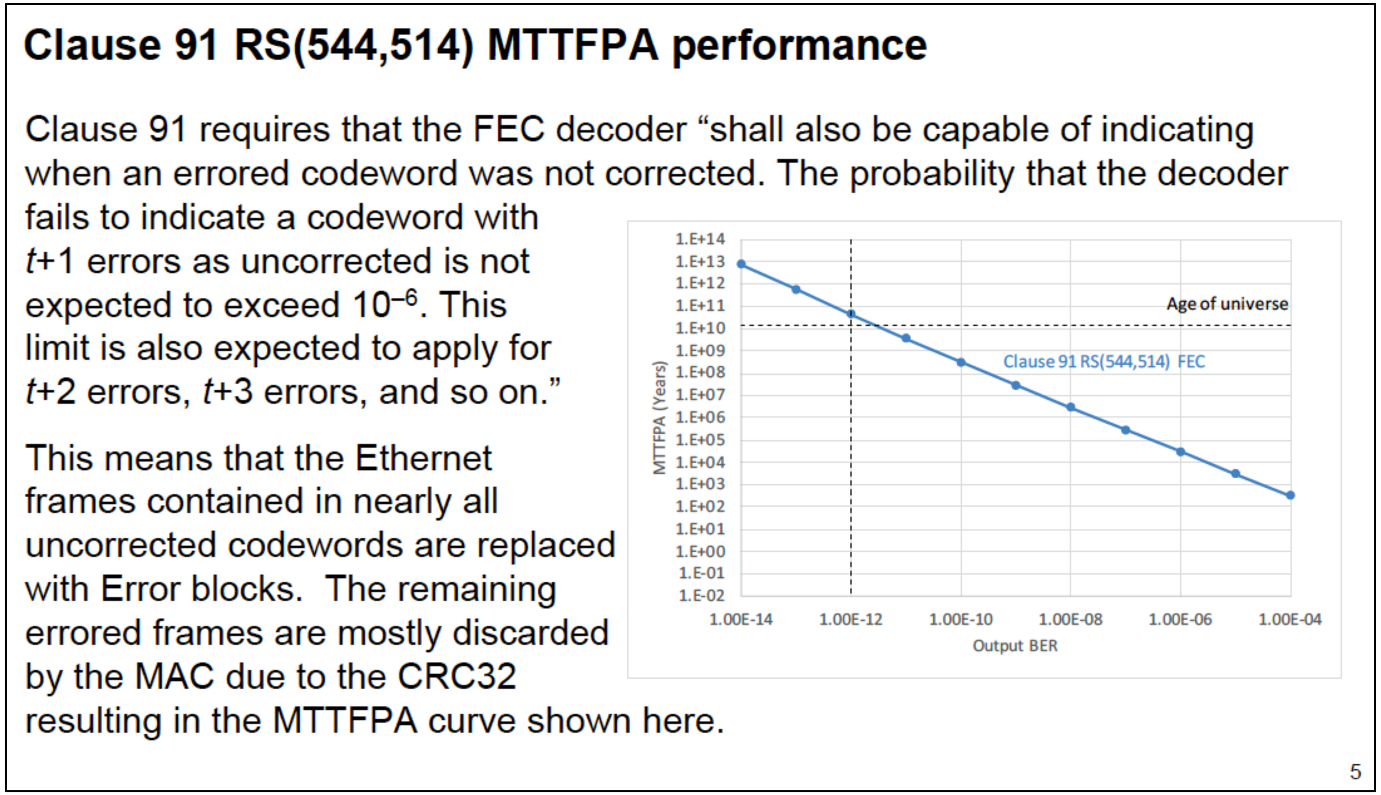
False Packet Acceptance Rate



- For 802.3ae (10 GbE), walker_1_0300 shows this graph.
  - It relies on the 64B/66B state machines to provide detection of up to 3 bit errors in consecutive block codes and sync headers (same as the ethernet CRC32 for frame data).

More Recent MTTFPA Analysis with RS-FEC

- For 802.3ct, anslow_3ct_01_0519 does a MTTFPA analysis for FEC-based interfaces and asserts on slide 5:



Clause 91 RS(544,514) MTTFPA performance

Clause 91 requires that the FEC decoder “shall also be capable of indicating when an errored codeword was not corrected. The probability that the decoder fails to indicate a codeword with *t*+1 errors as uncorrected is not expected to exceed 10^{-6}. This limit is also expected to apply for *t*+2 errors, *t*+3 errors, and so on.”

This means that the Ethernet frames contained in nearly all uncorrected codewords are replaced with Error blocks. The remaining errored frames are mostly discarded by the MAC due to the CRC32 resulting in the MTTFPA curve shown here.

- Based on this graph:
  - The 800GbE objective for output BER of 1E-13 meets the MTTFPA goal using RS(544,514) alone.

Note: 10^{-6} is the FEC escape probability for RS(528, 514). The slide text quotes 10^{-6} probability, but the graph is labeled RS(544,514). The FEC escape probability for RS(544, 514) is 10^{-16} as shown on a later slide.

MTTFPA for 800GbE with RS(544, 514)

- When a Codeword completes RX FEC decode it can be in one of three states:
  1. Data is correct (no errors or all errors are corrected)
  2. Data is incorrect and is known to be incorrect – decoder fail.
     - Data is replaced with EBLOCKs
  3. Data is incorrect and is thought to be correct – decoder error, FEC escape.
     - Incorrect data is passed to upper sub-layers as if it is correct
     - This is the one we are interested in

- Verify MTTFPA with a conservative calculation using RS-FEC and CRC32
  1. Calculate the probability of an uncorrectable RS(544, 514) CW (>15 symbol errors) and convert to FLR for worst-case packet size (min size packet with min IPG).
  2. Multiply by the probability of a FEC escape.
     - And assume any number of random errors in FEC output.
  3. Multiply by the probability of a false packet passing the ethernet CRC check (CRC32 escape).

Conservative RS Codeword Error Rate & FLR

- The RS(544, 514) codeword error rate (CER) is given by:
  - CER = $\sum_{i=t+1}^{N}\binom{N}{i}SER^{i}(1-SER)^{N-i}$
  - Where SER = 1 - (1 - BER_{in})^{10} and N=544, t=15 for RS(544, 514)
- For 100G/lane PMDs, assume BER = 2.4E-4
  - Plus, up to 4 electrical AUIs, add 1E-5 per AUI.
  - Further, to account for burst errors on the electrical links, multiply by 4.
    - For a=0.75, the average burst length is 4. (reference: slide 9 of anslow_3cd_01_0716)
- Total BER_{in} = 2.4E-4 + (4 * (4E-5)) = 4.0E-4
- Conservative CER = 1.30E-9
- Frame Loss Ratio (FLR) for 800GE with 4 interleaved CWs is 4.125*CER = 5.35E-9

RS-FEC Escapes (Uncorrectable & Undetected CW)

- From Clause 91.5.3.3:
  - “The probability that the decoder fails to indicate a codeword with *t*+1 errors as uncorrected is not expected to exceed **10^{-6}**. This limit is also expected to apply for *t*+2 errors, *t*+3 errors, and so on.”
  - This applies to the RS(528, 514) code.
- From Clause 119.2.5.3:
  - “The probability that the decoder fails to indicate a codeword with *t*+1 errors as uncorrected is not expected to exceed **10^{-16}**. This limit is also expected to apply for *t*+2 errors, *t*+3 errors, and so on.”
  - This applies to the RS(544, 514) code.
  - Use **10^{-16}** for probability of RS(544, 514) FEC escape.

Ethernet CRC32 Escapes

- For 32-bit ethernet CRC and random data, there is probability of 2^{-32} that the CRC is correct.
    - 2^{-32} = 2.33E-10

Putting the MTTFPA Probability Together

- Probability of undetected error packet from an undetected FEC error:
  - (FLR for FEC CWs with >15 FEC symbol errors) * Prob(FEC escape) * Prob(CRC32 escape)
  - 5.35E-9 * 1E-16 * 2.33E-10 = 1.24E-34
- At 800GE, an RS-FEC codeword arrives every 6.4ns
- MTTFPA = 6.4E-9 seconds / 1.24E-34 = 1.63E+18 years
  - Age of the universe ≈ 1.38E+10 years

FEC_bypass_indication Considerations

- When FEC_bypass_indication is enabled, the calculation changes
  - Corrected FEC data is allowed to be forwarded while ignoring the “uncorrectable” indication from the RX FEC decoder.
  - This means the 10^{-16} factor for FEC escapes cannot be used for calculations
  - MTTFPA = 6.4E-9 / (5.35E-9 * 2.33E-10) = 163 years
  - This can be made better with better input BER

MTTFPA for FEC_bypass_indication

BER_{in}	MTTFPA
4.0E-4	163 years
2.4E-4	259,049 years
5.0E-5	7.91E+15 years
1.0E-5	9.87E+26 years

- FEC_bypass_indication also requires additional error monitoring (see CL 119.2.5.3, paragraph 5)
- The RS-FEC decoder counts symbols errors and asserts hi_ser if over a threshold and the link can be dropped.
- This raises the MTTFPA when FEC_bypass_indication is enabled
- The threshold can be re-visited separately if needed.

Proposed Optional Stateless 64B/66B Encode

- Stateless encode can be done by looking at the “input” values of a block and one block before it. This guarantees good sequences on transmission.

Reset	Current block T_TYPE (tx_raw)	Previous block T_TYPE (tx_raw)	Current Block result (tx_coded)	Current block output type
1	*	*	LBLOCK_T	Local fault
0	S	C + T	ENCODE(tx_raw)	S
0	D	S + D		D
0	T	S + D		T
0	C	C + T + LI		C
0	LI	C + T + LI		LI
0	E	*	EBLOCK_T	Error block
0	S + D + T + C + LI	Anything other than above	EBLOCK_T	Error block

Proposed Optional Stateless 64B/66B Decode

- Stateless decode can be done without considering any previous blocks, unless the immediately previous block is known to be bad.
  - If a block has a known error, then the descrambler can cause the next block to also have errors. Therefore, only decode a block if the previous block is error-free

Reset	Current block R_TYPE (rx_coded)	Previous block R_TYPE (rx_coded)	Current Block result (rx_raw)	Current block output type
1	*	*	LBLOCK_R	Local fault
0	S + D + T + C + LI	~E	DECODE(rx_coded)	S or D or T or C or LI
0	S + D + T + C + LI	E	EBLOCK_R	Error block
0	E	*	EBLOCK_R	Error block

Summary

- Propose the “Stateless” TX 64B/66B Encode Table on slide #12
  - To be an option to the TX State Diagram (Fig. 119-14) for 8x100 interfaces
  - Only requires looking at only one previous block, in addition to the current block to guarantee correct block sequences are transmitted
  - This maintains compatibility with any implementation using the RX state machine.
- Propose the “Stateless” RX 64B/66B Decode Table on slide #13
  - To be an option to the RX State Diagram (Fig, 119-15) for 8x100 interfaces
  - RS(544, 514) FEC decoder alone is sufficient to meet the MTTFPA goal.
  - Simplifies the RX 64B/66B decode implementations by operating on each block independently.
  - Must still throw out one block after a bad FEC CW or received EBLOCK due to the de-scrambler.
- FEC_Bypass_Indication should not be enabled under normal conditions
  - Based on the BER_{in}, FEC_Bypass_Indication should only be used with low-loss channels.