

RS-FEC Codeword Monitoring for 802.3cd

(in support of comment #14 against D2.1)

Adee Ran

Intel Corp.

Contributors / Supporters

- Kent Lusted, Intel
- Upen Reddy Kareti, Cisco

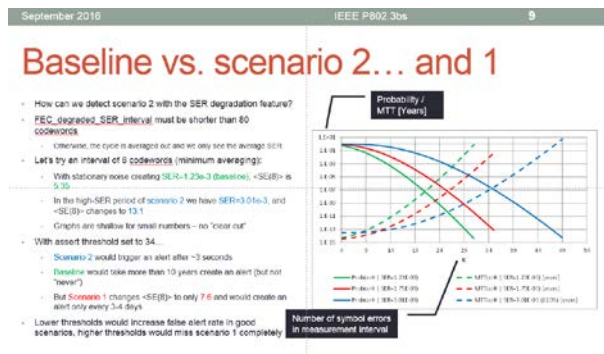
Problem statement

- **How can we estimate link quality / margin with FEC?**
 - i.e, predict how often frames will be lost before it actually occurs
- The Frame Loss Ratio (FLR) can be derived from the Uncorrectable Codeword Ratio (UCR), given the frame size and IPG
 - This was discussed in [ran_020415_25GE_adhoc](#) (which analyzes the RS(528,514)) based on previous work by Messrs. Brown and Anslow, referenced in that presentation
- **Assuming uncorrelated errors** (stationary noise), UCR can be calculated from the Symbol Error Ratio (SER)
 - The information we have in RS-FEC registers enables measurement of SER
 - SER is correlated to pre-FEC BER/DER so this measurement gives some estimate on the PMD/PMA performance
- **Problem: Errors are not necessarily uncorrelated**
 - Measured SER, and PMD performance, may yield optimistic estimates of FLR

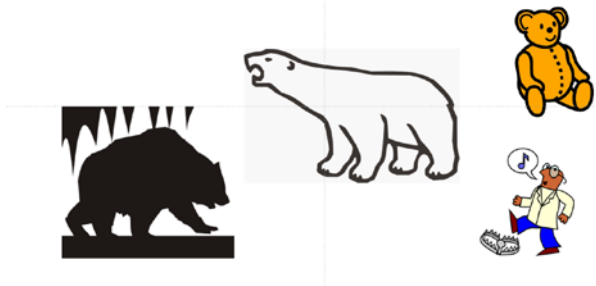
As described earlier...

(See ran 3bs 01a 0916)

- The FEC degradation solution is good for monitoring the PMD/PMA performance, but it is questionable how it can be used to predict failure rate (expressed as FLR, MTBF, etc.)
- Three scenarios were shown
 - Non-stationary noise conditions can cause much shorter MTBF than what would be calculated assuming uncorrelated errors
 - Selection of thresholds for degradation depends on scenario
 - Incorrect thresholds may cause frequent false alerts (especially in a large network) or unanticipated faults
- Do we have something better available?



What kind of BER do we want to catch?



Errored codeword counters

- The FEC decoder inherently knows, for each codeword, how many symbols were corrected (up to a maximum of 15 correctable errors)
- Counting codewords in separate counters, according to the number of errored symbols, would enable better understanding of the error statistics
 - e.g. estimate/extrapolate probabilities of encountering codewords with more errors than already seen, up to non-correctable codewords
 - From that, FLR or MTBF can be calculated
 - This is a soft metric, not an alert

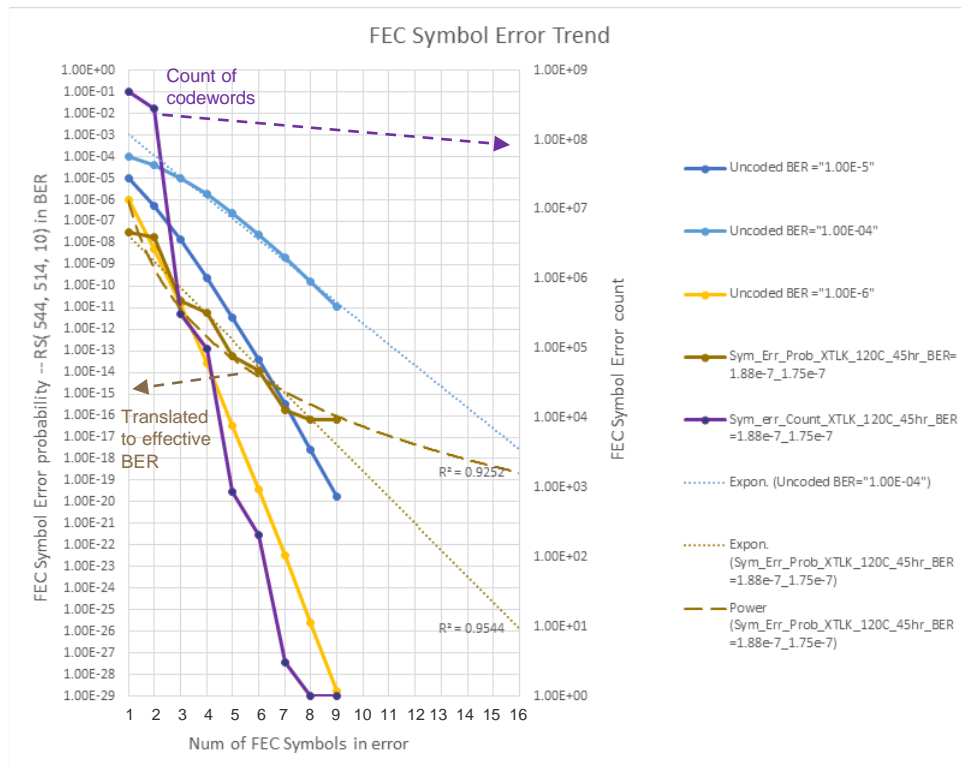
Why now (again)?

- In [ran_3bs_02a_0916](#) it was suggested to use this information instead of simple symbol error count to detect degradation
 - The associated comment was rejected: “There was no support for changing the FEC degrade feature along the lines in ran_3bs_02a_0916”
 - Presumably the errors on the optical links (most of the error budget) are nearly uncorrelated, so the concern for 802.3bs is low
 - Also, 400G/200G interleaved FEC is quite tolerant to bursts
- In this project we have full electrical links, and non-interleaved FEC
 - Expect more correlated errors and non-stationary patterns
- Also, the volumes for electrical links are likely to be higher + high variability of margins
 - Network management could benefit from soft margin assessment
- The proposed feature adds the required registers to observe the statistics, but no mechanism for signaling to the link partner
 - It is not as a replacement to the FEC degrade feature

What do we need?

- Ideally up to 15 registers to hold codeword counters per number of errored symbols
- The low-symbol-error counters are expected to advance very quickly, so have limited value
 - For a 50 Gb/s link a minimally compliant PMD BER ($2.4e-4$), with uncorrelated errors:
 - The 8-error counter would advance ~500/second
 - The 11-error counter would advance ~1/second, ~4000/hour
 - The 15-error counter would advance once every 3 hours
 - Uncorrectable codeword (16 errors) occurs once every 38 hours
 - For non-minimally compliant links the counters may advance less often
 - Reading the registers periodically (even once per hour) can provide the required information for extrapolation
 - This is beyond the scope of the standard. But see example in the next slide
- There is already a counter for uncorrectable codewords (for the RS(544,514), more than 15 errors)
- Suggestion is to allocate 16-bit counters for codewords with 8 through 15 errors.
 - This will not be useful for the RS(528,514) FEC but that is out of scope anyway.

Possible extrapolation of measurements



- Data and plot courtesy of Upen Reddy Kareti
- Shows how measurements of corrected symbols per codeword may be extrapolated to higher numbers and predict uncorrectable codewords
- There is more than one way to do it

DETAILED PROPOSAL

Proposed text for clause 134 (I)

- *Insert new subclause after 134.5.3.3.2 (under 134.5.3.3, Reed-Solomon decoder)*

134.5.3.3.3 FEC codeword monitoring

The Reed-Solomon decoder may optionally provide the ability to count codewords according to the number of corrected symbols. The presence of this option is indicated by the assertion of the FEC_codeword_monitor_ability variable (see 134.6.X). When this option is provided, it is enabled by the assertion of the FEC_codeword_monitor_enable variable (see 134.6.Y).

When FEC codeword monitoring is enabled, the Reed-Solomon decoder counts codewords with eight to fifteen FEC symbol errors in separate counters, fec_codeword_monitor_count_*i* (*i*=8 to 15), such that a codeword with *i* FEC symbols corrected by the decoder causes increment of fec_codeword_monitor_count_*i*.

Proposed text for clause 134 (II)

- *Insert new subclauses 134.6.X, 134.6.Y, and 134.6.Z (under 134.6, RS-FEC MDIO function mapping)*

134.6.X FEC_codeword_monitor_ability

This variable is set to one when the FEC decoder has the codeword monitoring ability (see 134.5.3.3.3), and is set to zero if this ability is not supported. It is mapped to the bit defined in 45.2.1.102 (1.201.5).

134.6.Y FEC_codeword_monitor_enable

This variable controls the FEC decoder codeword monitoring when the ability is supported (see 134.5.3.3.3). When set to one, codeword monitoring is enabled. When set to zero, codeword monitoring is disabled. Writes to this bit are ignored and reads return a zero if the FEC decoder does not have the codeword monitoring ability. This variable is mapped to the bit defined in 45.2.1.101 (1.200.5).

134.6.Z FEC_codeword_monitor_counter_i

FEC_codeword_monitor_counter_i, where $i=8$ to 15, is a 16-bit counter that counts once for each codeword in which exactly i FEC symbol were corrected, if FEC_codeword_monitor_enable is true. These counters are mapped to the registers defined in 45.2.1.115d.

Note: these subclauses may fit either after 134.6.6 (FEC_bypass_indication_ability) or at the end of 134.6

Proposed text for clauses 91 and 119

- New feature description subclauses similar to 134.5.3.3.2 (The following slides assume 91.5.3.3.2 and 119.2.5.3.1)
 - Clause 91 text should be limited to 100GBASE-CR2, 100GBASE-KR2, 100GBASE-SR2, and 100GBASE-DR PHYs
 - Clause 119 text should be limited to 200GBASE-CR4, 200GBASE-KR4, and 200GBASE-SR4 PHYs
- Variables subclauses as in 134.6
- Implement with editorial license

Proposed text for clause 45 (I)

- In **45.2.1.101 RS-FEC control register (Register 1.200)**

Change the first row of **Table 45–79** and insert a new row after it:

Bit(s)	Name	Description	R/W
1.200.15:6	Reserved	Value always 0	RO
1.200.5	FEC codeword monitor enable	1 = FEC codeword monitor counts codewords 0 = FEC codeword monitor does not count codewords	R/W

Insert new **45.2.1.101.aa** (before the current subclause) and renumber subsequent subclauses:

45.2.1.101.aa FEC codeword monitor enable (1.200.5)

This bit controls the RS-FEC codeword monitoring (see 91.5.3.3.2, 119.2.5.3.1, and 134.5.3.3.3). When set to a one, this bit enables codeword monitoring. When set to a zero, codeword monitoring is disabled. Writes to this bit are ignored and reads return a zero if the FEC does not have the codeword monitoring ability.

Proposed text for clause 45 (II)

- In **45.2.1.102 RS-FEC status register (Register 1.201)**

Change the second row of **Table 45–78** and insert a new row after it:

Bit(s)	Name	Description	R/W
1.201.6	Reserved	Value always 0	RO
1.201.5	FEC codeword monitor ability	1 = RS-FEC decoder has the FEC codeword monitor ability 0 = RS-FEC decoder does not have the FEC codeword monitor ability	RO

- Insert new **45.2.1.101.6b** (before the current subclause) and renumber subsequent subclauses:
45.2.1.101.6b FEC codeword monitor ability (1.201.5)
This bit is set to one to indicate that the decoder has the codeword monitoring ability (see 91.5.3.3.2, 119.2.5.3.1, and 134.5.3.3.3). This bit is set to zero if this ability is not supported.

Proposed text for clause 45 (III)

- Insert new subclause after **45.2.1.115c**

45.2.1.115d: RS-FEC codeword monitor counters (Registers 1.658 through 1.665)

The RS-FEC codeword monitor counters are defined in 91.5.3.3.2, 119.2.5.3.1, and 134.5.3.3.3. Register 1.658 contains RS-FEC codeword monitor counter 8, and the assignment of bits in this register is shown in Table 45–90xxx. Registers 1.659 through 1.665 contain RS-FEC codeword monitor counters 9 through 15 respectively, and their bit assignments are equivalent to that of RS-FEC codeword monitor counter 8 register, for the corresponding counters.

For each of these registers, the bits shall be reset to all zeros when the register is read by the management function or upon PHY reset, and shall be held at all ones in the case of overflow.

Table 45–90xxx—RS-FEC codeword monitor counter 8 register bit definitions

Bit(s)	Name	Description	R/W ^a
1.658.15:0	RS-FEC codeword monitor counter 8	FEC_codeword_monitor_counter_8	RO, NR

^aRO = Read only, NR = Non Roll-over

QUESTIONS/COMMENTS?

Thank you