Assessing burst rate and MTTFPA in real systems

Adee Ran, Intel Corporation

October 14th, 2013

Problem statement

- We would like to use a DFE in the reference receiver of CAUI-4
- If CAUI-4 isn't protected by RS-FEC (existing 100GBASE-LR4), having a DFE in the path creates concerns about error propagation, bursts, and MTTFPA
- Nothing in any of the current or proposed CAUI-N specifications prevents using a DFE or addresses error bursts in any way…
- A receiver can include a DFE and suffer from severe error propagation when deployed in a specific system. There is no indication of this situation. It may even be compliant!

Receiver compliance

- Our usual practice is to define receiver compliance tests to ensure the desired behavior. Can we do that for error propagation?
- Two major issues come to mind:
  - How difficult it is to perform the test? (time)
  - How meaningful is it? (would passing the test guarantee the receiver doesn’t create bursts?)

Test Time

- Testing directly just for BER<1e-15 would take 32 hours
  - Assuming 3e15 bits on each lane for statistical significance, and all 4 lanes tested in parallel
- If BER=1e-15 and p(EP)=0.03, then mean time between bursts of 2 errors in any lane is 90 hours; test time 270 hours
- Can we shorten test time by extrapolation (e.g. adding Gaussian noise and drawing a bathtub curve)?
  - May be reasonable for BER testing; but adding noise also increases p(EP), in a way dependent on the original noise PDF; As presented previously
  - At the desired p(EP) of ~1e-2 the noise statistics is far from Gaussian – so extrapolation of p(EP) based on a shortened test time would be incorrect
  - Defining the test requirements for equivalent of the desired p(EP) would be a problem

Coverage

- Besides the noise PDF, error propagation also depends on DFE coefficients, which in turn depend on the actual channel in use
- What channel should be used in an EP compliance test?
  - A benign channel may cause EP not to occur at all; but the receiver might still create EP in a real system
  - Adding “controlled ISI” to a channel to emulate real systems may be hard to define
    - We may end up validating (in a convoluted way) the receiver’s DFE coefficients
    - Should a receiver without a DFE pass such a test?

Another problem statement

- Error propagation is not an attribute of the receiver alone – it also depends on the channel (and on the transmitter part of it)
  - Perhaps mostly on these
- Testing the receiver alone is not very meaningful
  - Relying on receiver compliance tests alone may provide a false feeling of safety…

Should we specify p(EP) or burst rate for a whole link?

- A direct test on a CAUI-4 link is possible… but there's nothing to define: "build your system and see if it works"
- But do we absolutely need a compliance test?
  - We have an analysis method that can predict burst probability for a given channel and reference TX/RX
  - We have shown that, on several channels, achieving a low-enough p(burst) is feasible with limited DFE taps
  - Even in "bad links", bursts are quite infrequent and "mostly harmless"
  - If most links have healthy margins and "bad links" are rare enough, maybe we just need a way to indicate that an operational link is unsafe?

Another way to address MTTFPA concerns

- Even short bursts should be rare, and they are about as harmful as single errors
- It would be nice if they can monitored in the field, during actual operation, to identify “weak links”
  - An identified “bad-MTTFPA link” can be serviced months after it is deployed, without causing a real risk.
  - Ideally, allowing network management to calculate MTTFPA and identify weak links in the field could eliminate most of the concerns.

Identifying bursts in operational systems

- Proposed below is a simple method of identifying error bursts and measuring their rate during regular system operation, **based on the existing BIP mechanism**.
- This can in principle be done without any new specifications; but may be made more useful with an additional feature proposed later.

How?

- Assumption: CAUI-4 RX connects directly or indirectly to a PMA(20:4) attached to the RX 100GBASE-R PCS.
- A burst of errors on one of the CAUI-4 lanes is thus striped between up to 5 PCS lanes (PCSLs).
  - For burst lengths of up to 5, the error bits will be striped to one PCSL each.

PMA demux from CAUI-4 to PCS

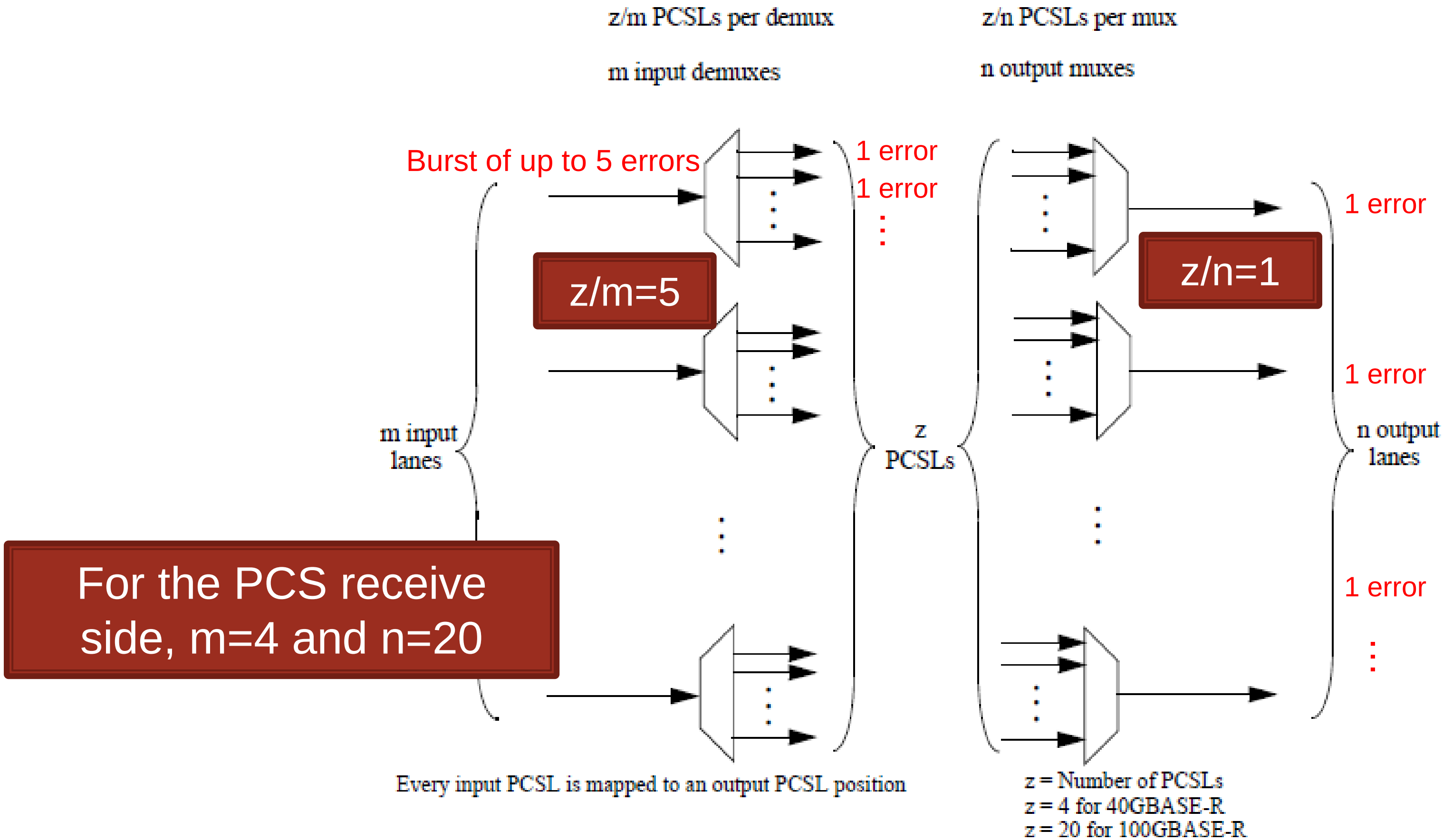

Figure 83–4—PMA bit mux operation used in both Tx and Rx directions

Identifying bursts

- PCS detects errors on each PCSL separately using the BIP field in alignment markers (AMs)
  - BIP will detect any single bit error since the last AM.
  - Assuming CAUI-4 BER<1e15, having more than one burst on any PCSL between adjacent AMs is extremely unlikely.
- After PCS lane alignment, AMs from all 20 lanes are available together.
- When a burst of length L occurs, exactly L out of the 20 AMs will have BIP mismatch.

Identifying bursts

- If the full link operates at BER=1e-12 then errors are expected once per 10 seconds…
  - An isolated error will cause one of 20 the BIP counters to advance
  - If the error is propagated into a burst, more than one counter will advance
  - If one reads all 20 BIP counters once per second (noting that they are clear-on-read) then:
    - Reading all zeros means no error have occurred during this second
    - Reading 1 on one counter means a single error has occurred
    - Reading 1 on L counters (L≤5) means a single burst of L errors has occurred
    - Anything else is probably a major link-wide event (assumed unlikely)
- Under assumed BER levels, bursts are detectable!
- Is this a sufficient solution for network management?

Proposed improvement

- Monitoring can be easier with a new PCS feature:
  - Whenever a set of AMs is received, define L as the number of lanes with mismatched BIP (= the burst length)
  - Alert: if L≥3 is detected, assert hi_ber until the next set of AMs
    - This is expected to be an extremely rare event in good links
    - When it occurs, it will cause PCS_status=false and XGMII set to LOCAL_FAULT for ~42 microseconds, temporary pausing traffic
    - Also exposed through MDIO, latched-high
  - Monitor: define 5 new burst counters, one per value of L (1…5)
    - Whenever L>0, increment counter L
    - Define the burst counters in clause 82 and make them available through optional MDIO (current BIP counters are only defined in MDIO clause 45, so are optional!)
- Optional feature, but recommend/require it for implementations that include a CAUI-4 interface.

Estimating MTTFPA for an operational system

- Assumption: all four lanes have same BER and p(EP), therefore same p(burst≥4)
- Measure the rate of single errors f_1 over time; estimate 4-lane BER as $p_1 = f_1 \cdot UI$
- Measure the rate of 2-error bursts f_2 over time; estimate p(EP) as $p_2 = f_2/f_1 \cdot UI$
- Estimated p(burst≥4) for the whole CAUI-4 link is $p_1 \cdot p_2^3$

Estimating MTTFPA – cont.

- Assume frames are 179*64=11456 bits long
  - Slightly below MTU limit
  - Shorter frames improve MTTFPA; and below 2944 bits, CRC can always detect up to 5 errors [1]
- Adding IPG and sync headers yields 11880 bits at the PCS
- There are 11264 out of 11880 locations where a dangerous 4-error burst can be placed
  - Excluding all sync headers, last 3 blocks and IPG
- Assume a 4-error burst starting on these locations can create a CRC collision with $p=2^{-32}$

[1] Koopman, P. "32-bit cyclic redundancy codes for Internet applications", Proc. DSN 2002. See table 1.

Estimating MTTFPA – cont.

- Estimated MTTFPA is

$$\frac{11880/4\ UI}{p(burst \geq 4) \cdot 2^{-32} \cdot 11264}$$

$$\cong \frac{1.4 \cdot 10^{-9}}{p(burst \geq 4)}\ years$$

- Example: if all four lanes have BER=1e-15 and p(EP)=0.03, we get MTTFPA ≈13 billion years.
- This is a pessimistic estimate
  - Assumes max frame size, no idles, and all lanes are worst case
  - A considerable margin is built-in!

How fast is MTTFPA estimation?

- Let's consider a CAUI-4 which operates at worst-case compliant conditions (stated above)
- Estimate how fast the counters advance for this system, and compare to cases when either its BER or its p(EP) are increased.
- Results, shown in the next slide, show that safe/unsafe decision can be made within a few days of operation.

Results

Scenario	**BER=1e-15; EPP=0.03**	**BER=1e-14; EPP=0.03**	**BER=1e-15; EPP=0.3**
Mean time to a single error (any BIP mismatch)	**2.7 hours**	**16 minutes**	**2.7 hours**
Mean time to burst with L=2	**3.7 days**	**9 hours**	**9 hours**
Mean time to burst with L=3	125 days	12 days	30 hours
Mean time to burst with L=4	380 years	38 years	14 days
MTTFPA estimate	13 billion years	1.3 billion years	13 million years
Mean time to false count of 2 uncorrelated errors	6,000 years	60 years	6,000 years

Proposals

- **Add "burst" detector using multi-lane BIP mismatch as a new optional PCS feature**
  - Required if PCS is attached to a PMA with CAUI-4
  - Burst length above 2 shall cause <u>assertion of hi_ber</u> (and disrupt traffic, so it can't be ignored)
  - Define <u>counters per length</u> in clause 82, and map to addresses in clause 45 – enabling monitoring by management
- **Define a <u>shortened BER compliance test</u> using bathtub extrapolation**
  - Either refer to a BERT scan or explicitly define jitter/noise levels
  - Details can be worked out, if the principle is accepted
- **Avoid defining a compliance test for error propagation.**

Summary

- Burst probability and MTTFPA depends on full link rather than on receiver alone.
- Measurement can be performed in the field.
- MTTFPA assessment is simple (formulas are provided).
- Time constants allow identification and maintenance with negligible risk.

To Do

- Define new feature text in clause 82 (82.2.14 seems the place to do it)
- Allocate new MDIO registers for counters in clause 45
- Define shortened BER test

Backup

Compatibility issues

- Clause 82 does not require PCS to check anything on the AMs as a group… This wasn't required before
- Chips with 100GBASE-R PCS and CAUI-4 interface are new, so if the DFE is on the PCS side there is no backward compatibility problem
- Only uncovered case is using CAUI-4 C2C as a C2M extension for an LR4 module, when the remote host is legacy (CAUI-10) LR4 and does not have burst counters
  - In this case, error propagation caused by the "extension" transmit-side RX DFE can be detected in the remote host by polling BIP counters.