100G backplane PAM4 PHY encoding

IEEE P802.3bj

January 2012, Newport Beach

Matt Brown – AppliedMicro

Sudeep Bhoja – Broadcom

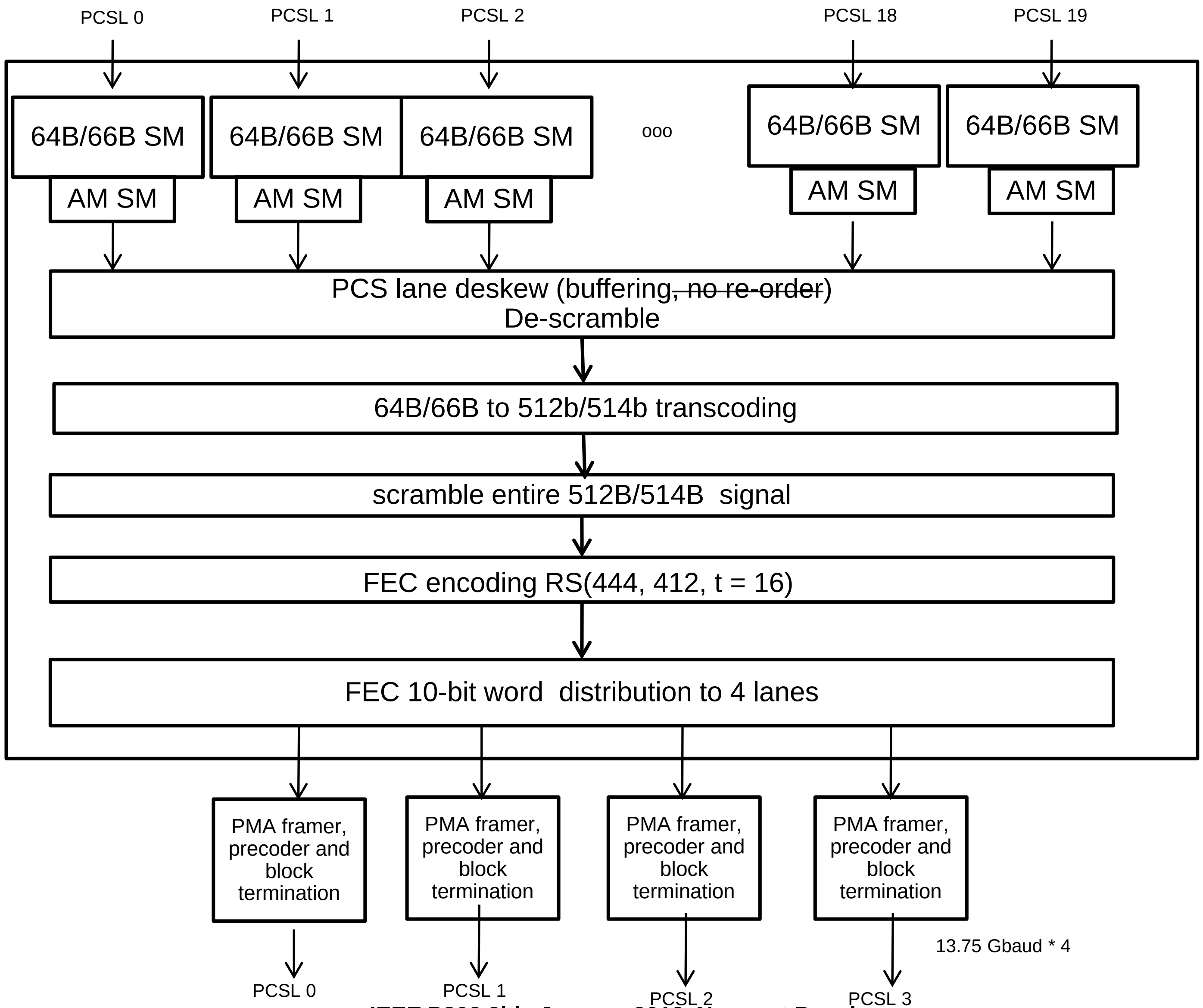
Contributors and Supporters

- Ran Adee, Intel
- Will Bliss, Broadcom
- David Chalupsky, Intel
- Dariush Dabiri, APM
- Dan Dove, APM
- Howard Frazier, Broadcom
- Ali Ghiasi, Broadcom
- Dimitrios Giannakopoulos, APM
- Sanjay Kasturia, Inphi
- Kent Lusted, Intel
- Richard Mellitz, Intel
- Venkatesh Nagapudi, APM
- Vasu Parthasarathy, Broadcom

Transmitter process

- Transcoding: 512B/514B
- FEC: RS(444,412,T=16,M=10)
- PAM4 Symbols: Gray mapping,
  - {+1,+1/3,-1/3,-1} map to {10,11,01,00}
- Precoding: 1/(1+D) MOD 4
- PAM4 block termination: 1 PAM4 termination symbol per 32 PAM4 symbols
  - 63 data bits per 32 PAM4 symbols
- PAM4 symbol rate: 88 * 156.25 MHz = 13.75 Gbaud
- Tx pre-emphasis: 3 taps, one pre, one post
  - same structure as for 10GBASE-KR
- PAM4 test methodology and parameters addressed in bliss_01a_0911.

Tx encoding flow

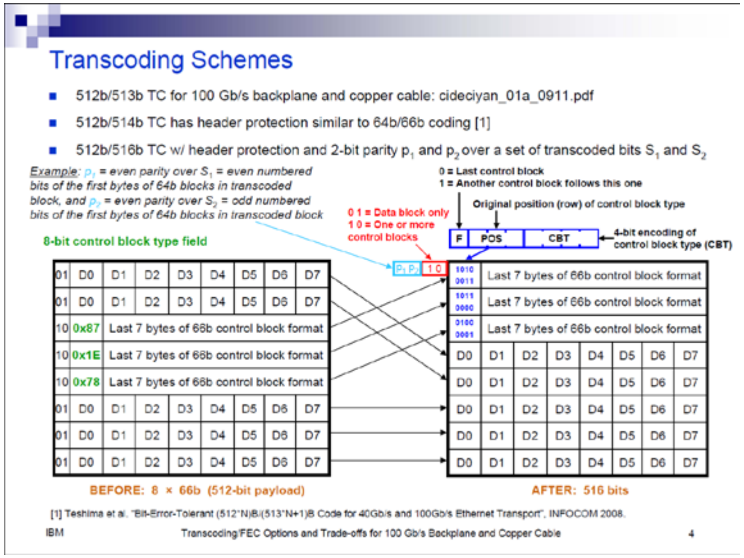


PCS Lane Processing

- Synchronize to 64B/66B blocks on each PCS lane per 802.3ba 82.2.11.
- Synchronize to PCS alignment markers (64B/66B blocks) on each PCS lane per 802.3ba 82.2.12.
- Align (or deskew) PCS lanes based on alignment markers per 802.3ba 82.2.12.
- Descramble 64B/66B blocks per 82.2.15.
  - Required for transcoding.
- Same as for NRZ PHY.

Transcoding

- Use 512B/514B transcoding
  - per cideciyan_01a_0911 and cideciyan_01a_1111.
- Map 8x 64B/66B blocks for each 512B/514B block.
  - Cycle through PCSLs, one 64B/66B block at a time.
  - See following slide.
- Should be the same as for NRZ PHY.



from cideciyan_01a_1111.pdf

Mapping 64B/66B blocks to 512B/514B

64B/66B blocks from PCS lanes arriving in time, bottom arrives first.
~~Note: Lanes may not be in the order shown. Reording is not required or recommended.~~

PCSL0	PCSL1	PCSL1	…	PCSL18	PCSL19
…	…	…	…	…	…
0.3	1.3	2.3	…	18.3	19.3
0.2	1.2	2.2	…	18.2	19.2
0.1	1.1	2.1	…	18.1	19.1
0.0	1.0	2.0	…	18.0	19.0

Block label <PCSL>.<64B/66B block index>

map to groups of 8 64B/66B blocks

8x8 64B/66B blocks

0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
1	8.0	9.0	10.0	11.0	12.0	13.0	14.0	15.0
2	16.0	17.0	18.0	19.0	0.1	1.1	2.1	3.1
…	…	…	…	…	…	…	…	…
7	16.2	17.2	18.2	19.2	0.3	1.3	2.3	3.3

512B/514B header bits (2/block)

transcode 8x 64B/66B blocks to transcode blocks

8x 512B/514B blocks

Note: Showing 8 512B/514B blocks here since 8 of these blocks map to each FEC frame.

0	2	512 bits
1	2	512 bits
2	2	512 bits
…	2	512 bits
7	2	512 bits

Scrambling

- Use self-synchronizing scrambler
  - Same scrambler as for PCS in 802.3ba 82.2.5.
  - All data bits including the 512B/514B header bits are scrambled.
  - Should be the same as for the NRZ PHY.

FEC

- RS(444,412,T=16,M=10) code format
  - single, efficient, dual-purpose (NRZ/PAM4) FEC core is possible if FEC generator math specified similarly for both
- FEC frame content
  - correctable payload = 412*10 = 4120 bits
  - parity = 32*10 = 320 bits
  - data = 64x 64B/66B blocks transcoded to 8x 512B/514B blocks
    - total data = 4112 bits
    - 8 dummy bits (4120-4112) per FEC frame required
    - 8 zeros added (assumed) for parity calculation
    - Payload words 408-411 will contain 8 data bits and 2 dummy bits.
      - one 8-bit word will end up on each of the 4 PMA lanes
    - dummy bits not transmitted
- FEC encoding is mandatory; negotiation is not required.

13.75GBaud Precoding/FEC Summary

RS(444, 412, t = 16)	Delta (dB)	Coding Gain (dB)
Random Error		7.12
DFE Burst Error Penalty	-0.88	6.24
Extended KR channel 6.7% over clocking loss	-1.0	5.24 (<100ns total latency)

- ~6.7% over clocking (88*156.25 MHz)
- 5.34 dB Coding gain for Extended KR channel
- Overhead includes FEC parity & PAM4 block termination

Comparison of RS FEC candidate codes

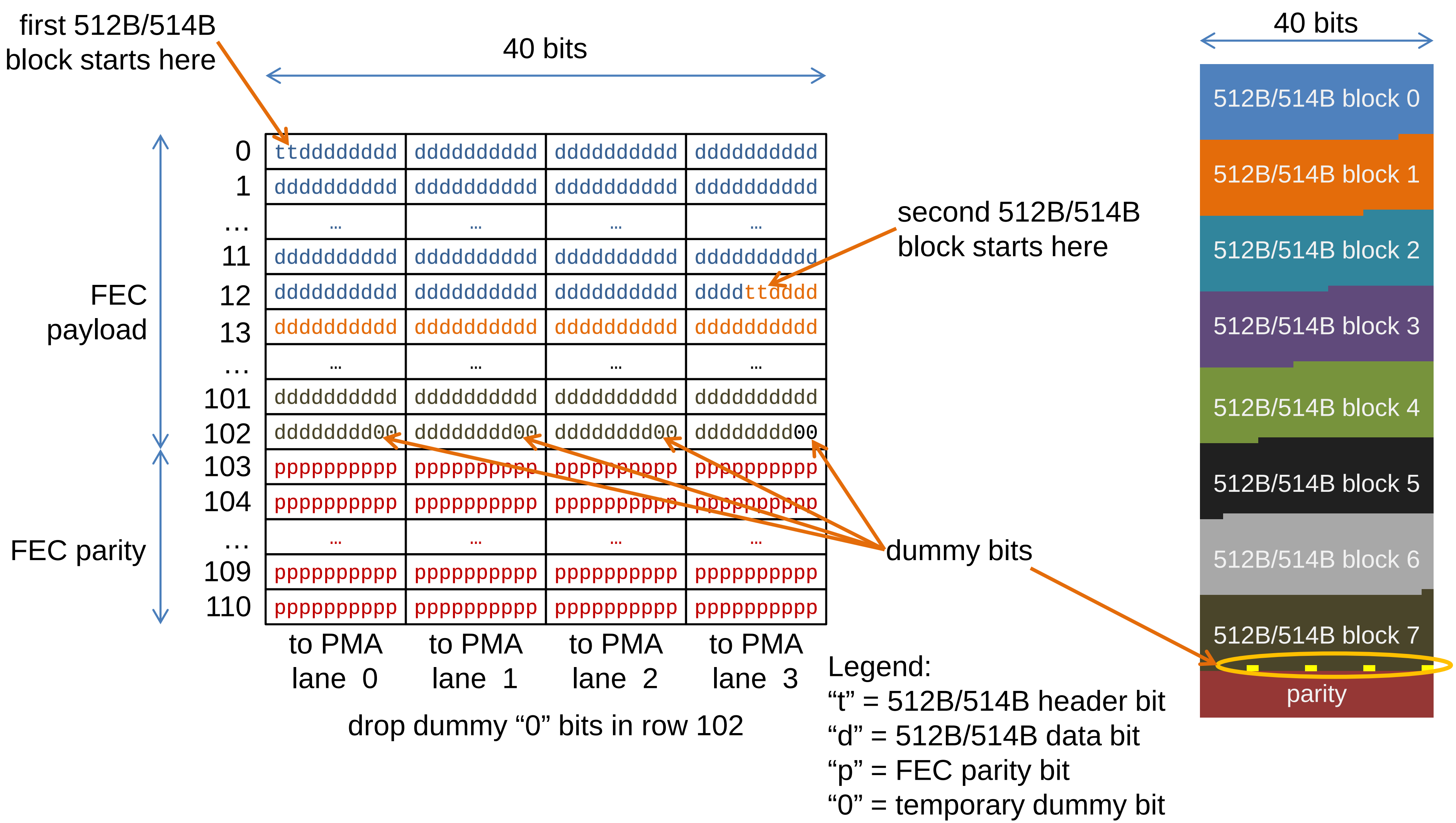
GF(2^10)	Total Coding Gain (dB)	Burst Coding Gain (dB)	Latency (ns)
RS(444, 412, t = 16)	5.24	6.24	82 - 123
RS(550, 520, t = 15)	5.1	5.9	102 - 154
RS(546, 520, t = 13)	4.9	5.6	102 - 154
RS(544, 520, t = 12)	5	5.6	102 - 154
RS(540, 520, t = 10)	4.9	5.2	102 - 154

- Codes in bhoja_01_0911 and cideciyan_01_1111 (found using computer search)
- RS(444, 412, t = 16) has best coding gain within 100ns target latency
  - Example implementation of 460K gates in 40nm CMOS has 99.9ns latency

Mapping 512B/514B blocks to FEC frame

- 512B/514B blocks are concatenated and decimated into 10-bit FEC words.
  - Except for last four FEC words which are 8 data bits with 2 pad bits each (see FEC slide).

FEC frame structure (assuming PMA sync)



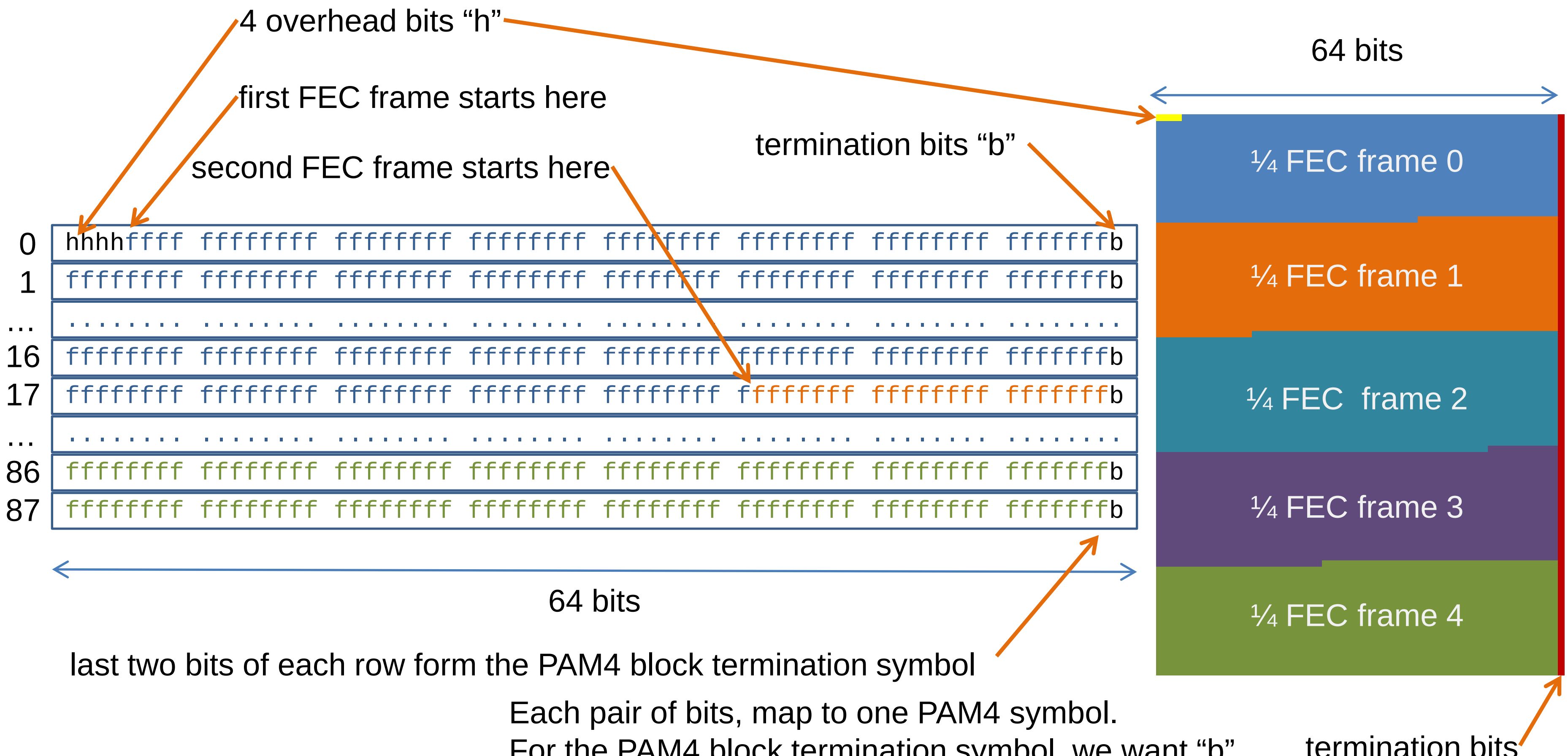
Mapping FEC to PMA lanes

- Cycle through FEC 10-bit words through each of the 4 PMA lanes.
  - The FEC frame contains 444 10-bit words
  - For each FEC frame, 111 10-bit words are destined for each of the four PMA lanes.
  - FEC words (i+j*4) go to lane i
    - i is {0,1,2,3}, where i represents the lane #
    - j is {0,1,2,…,110}, j indexes the FEC words destined for each lane
  - Note that for FEC words 408 to 411, only the 8 data bits are transferred to each lane.

PMA Frame

- PMA frame generated for each PMA lane.
- PMA frame is composed of…
  - 5 quarter FEC frames, 5*(4440-8)/4 = 5540 bits
  - 4 overhead bits
    - essential to give a resultant PAM4 symbol rate of 88 * 156.25 MHz
    - various possible applications discussed on subsequent slide
  - 88 PAM4 block termination bits
    - 1 termination bit per 63 data bits
  - 5632 bits total

PMA frame structure (one per lane)



Each pair of bits, map to one PAM4 symbol.
For the PAM4 block termination symbol, we want "b" and the preceding bit "f" to indicate +1 or -1 so …
For gray mapping, b = 0, always!
if the preceding bit is 1, then 10 maps to +1
if the preceding bit is 0, then 00 maps to -1

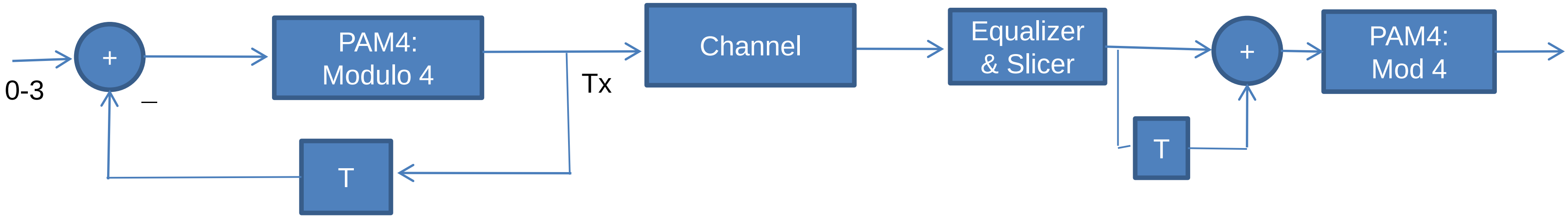
Legend:
"f" = bits from 5 FEC frames
"h" = overhead bits
"b" = block termination bits

PMA Frame Overhead Bits

- Each PMA per-lane frame has 4 overhead bits.
- Must be randomized or at least “friendly”.
- Various applications …
  - PMA frame alignment (see previous slide)
  - lane identification
  - control channel for remote transmitter control
  - vendor specific use

Pre-Coding

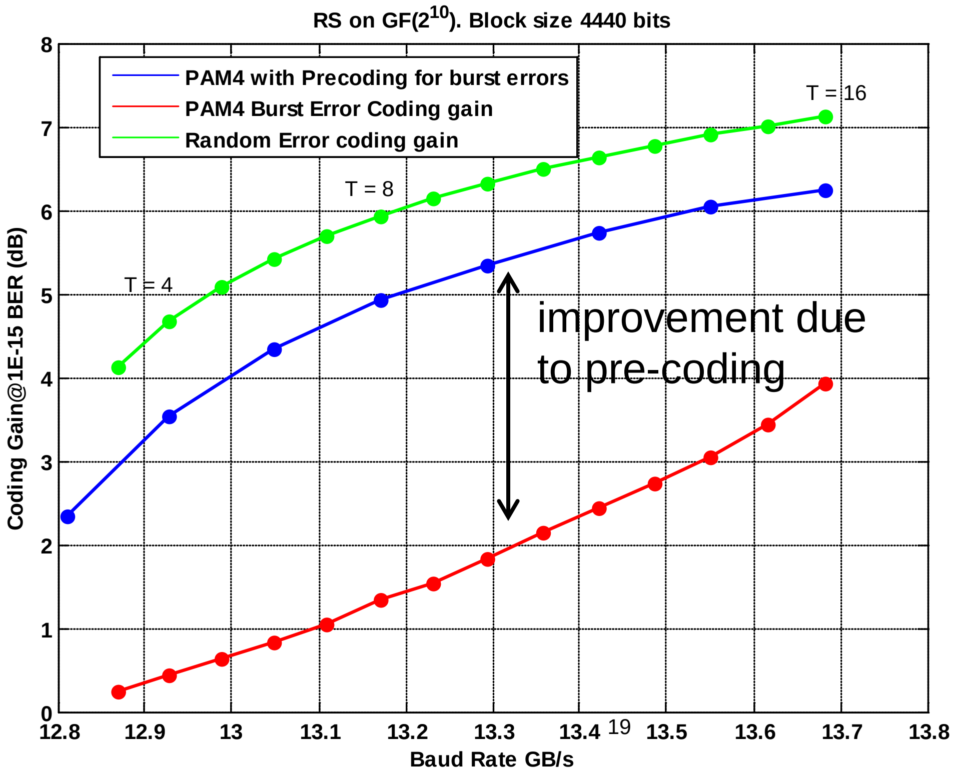
- 1/(1+D) modulus 4 pre-coding
  - See bliss_01_0311, “Signaling Terminology; PAM-M and Partial Response Precoders”
  - Rx uses a (1+D) mod 4 after slicing
- Simple to implement
- Very low Complexity; similar complexity to duo-binary precoder.
- Pre-coding is mandatory; negotiation is not required.



Motivation for pre-coding

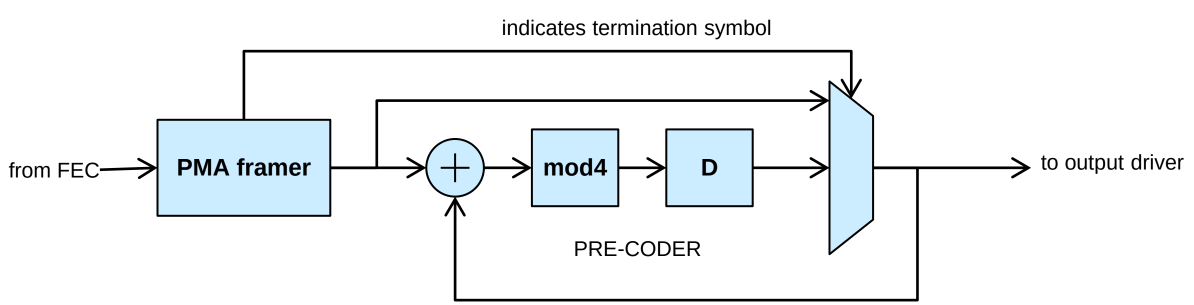
- Mitigates error propagation in DFE and MLSD receivers.
  - Greatly reduces number of errors per burst.
  - For 1-tap DFE, reduces burst to two errors, one at beginning and one at end
  - For MLSD see dabiri_01_0911 "Enabling Improved DSP Based Receivers for 100G Backplane"

- Graph shows improved coding gain (blue) due to precoding.
- The delta between burst error and random error is ~1.0dB with 1/(1+D) mod 4 precoding



PAM4 Block Termination

- PAM4 block termination symbol every 32 PAM4 symbols
  - For efficiency, each PAM4 termination symbol transmits one data bit.
    - 63 data bits sent every 32 PAM4 symbols
    - Increases baud rate by 64/63.
  - Each PAM4 block termination symbol is mapped to either +1 or -1.
  - At the transmitter, termination added within the precoder.
  - At the detector, termination removed after the detector.
- See dabiri_01_0112.
- PAM4 block termination encoding is mandatory; negotiation is not required.



Functional representation of block termination and pre-coding

Motivation for PAM4 Block Termination

- Block termination by transmitting known PAM4 symbols on a regular cycle enables…
  - efficient and effective MLSD, maximum likelihood sequence detection (dabiri_01_0911)
  - parallel DFE implementations
    - Keshab K. Parhi, Pipelining of parallel multiplexor loop and Decision Feedback Equalizers, ICASSP, 2004

PAM4 encoding

- Gray mapping
  - pre-coder output {10, 11, 01, 00} maps to {+1,+1/3,-1/3,-1}
  - based on 2B1Q coding used in HDSL and ISDN

PMA synchonization

- Lock to PAM4 termination blocks by searching for PAM4 termination symbols
  - PAM4 termination symbols (1 in 32) are always either +1 or -1.
  - Similar to framing on 10 or 01 sequence for 64B/66B, can borrow and modify 64B/66B synchronization state machine.
- Lock to PMA frame
  - Use known content of overhead bits.
    - Once locked to the PAM4 termination blocks, look for 4 bits (2 PAM4 symbols) every 88 rows.
    - Again, similar to 64B/66B synchronization.

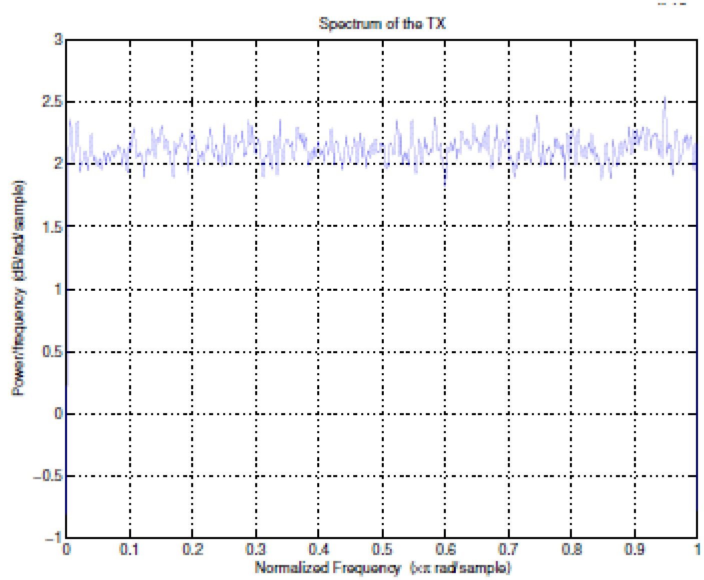
Energy Efficient Ethernet Operation

- Fast synchronization for REFRESH and WAKE.
  - Synchronize on PAM4 termination symbols.
  - Use prescribed sequence to accelerate synchronization.
- For REFRESH, PCS and FEC not required.
  - Replace with scrambled sequence.
    - Similar to EEE/LPI for 10GBASE-T.
- For WAKE, rapid alignment markers not required by the PHY ~~transmitter and~~ receiver.
  - Will still be required at the PCS RX at the PCS end point.
- No significant impact to work being done in EEE consensus group.
  - Compatible and complementary with PCS state machine in Gustlin_02_1111.

Thanks!

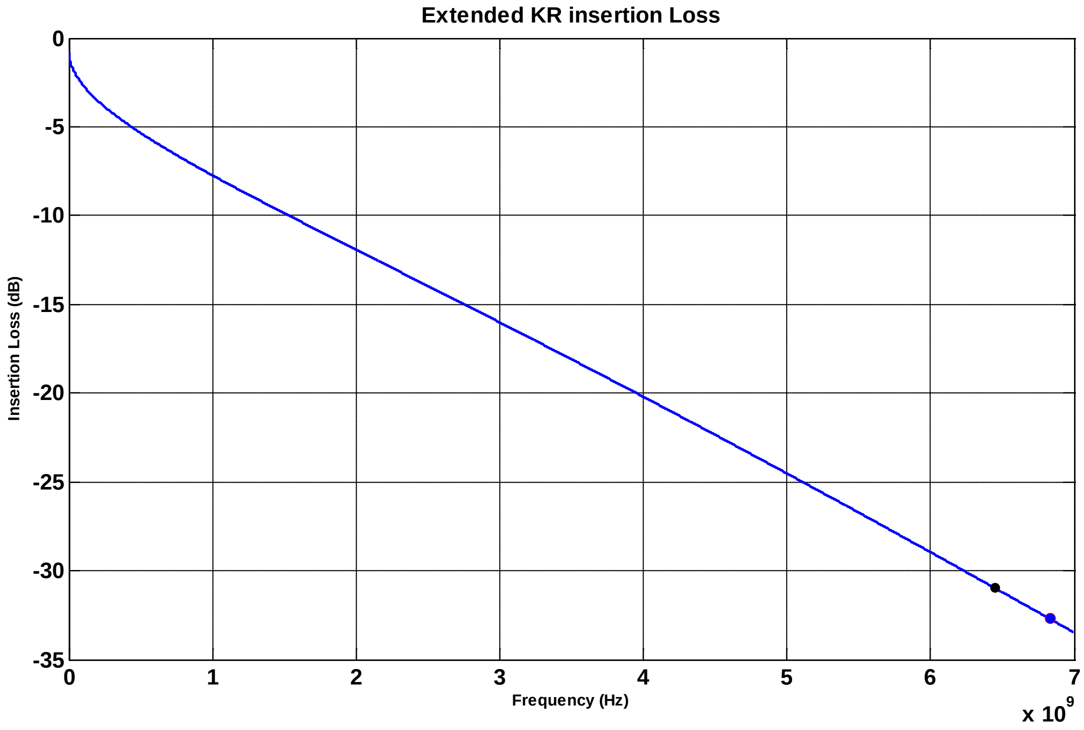
BACKUP SLIDES

Power spectrum with PAM4 block termination symbols



- The simulated spectrum above shows no spectral content due to block termination symbols.
- Pattern is repeating structure (not content) of 32 PAM4 symbols…
  - 31 random PAM4 symbols in {-1,-1/3,+1/3,-1} * 3
  - 1 random PAM4 symbol in {-1, +1} * 3

PAM4 SNR Loss due to Over clocking

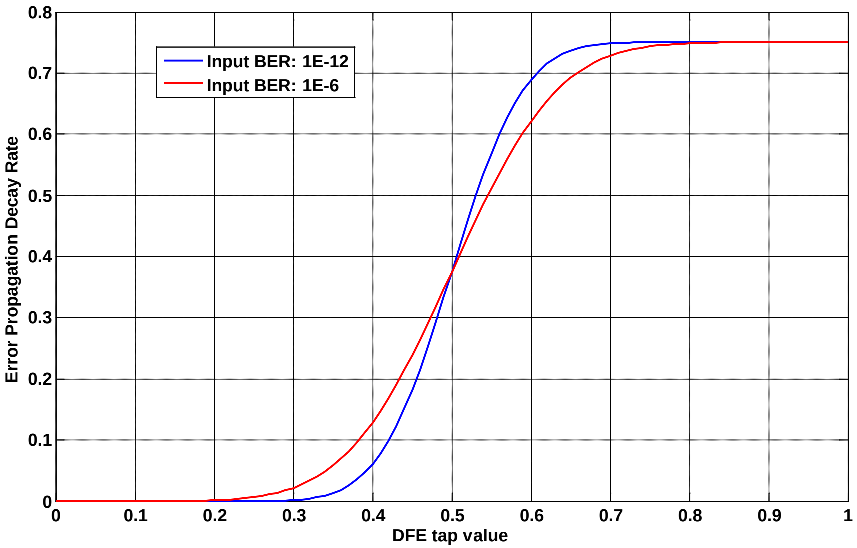


For FEC baud rate of 13.67G, the SNR loss due to over clocking

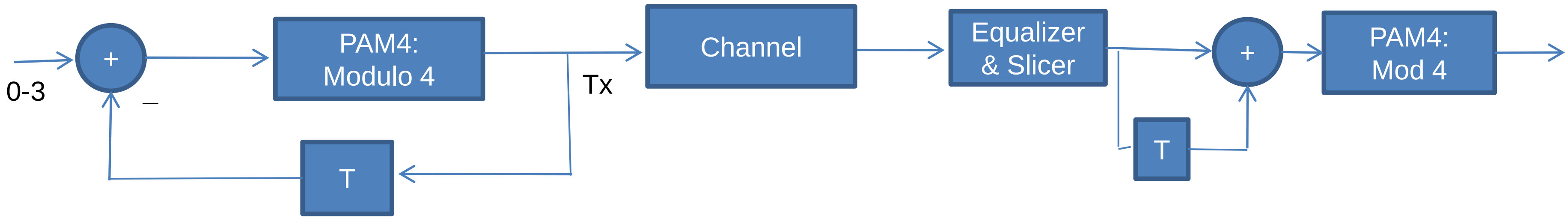
- $SNR_{delta} = (IL_{6.84GHz} - IL_{6.45\ GHz})/2 = 0.9dB$

Precoding Motivation: PAM4 DFE bursts

- DFE's are well known to multiply errors in the feedback loop
  - A single error will become a burst error
- Consider PAM4 1-tap DFE with tap coeff = 1
  - If previous decision is wrong, then there is 3/4 probability of making a successive error
  - i.e. Probability of K consecutive errors = $(3/4)^k$
- Lower 1st DFE tap between 0.6 to 1 have similar burst length as tap coefficient of 1
  - Tap of 1: 0.75^k
  - Tap of 0.7: 0.72^k
  - Tap of 0.6: 0.62^k
- A single random error may consume multiple Reed Solomon words
  - Burst error coding gain is lower than coding gain for random errors

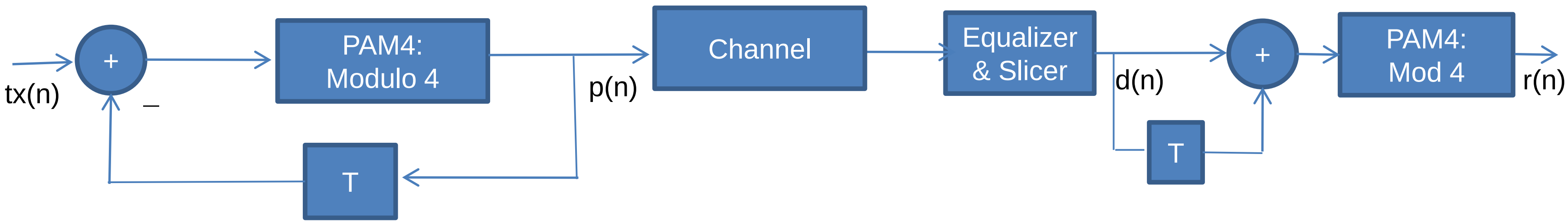


1/(1+D) Precoding for DFE burst errors



- The burst error length of the DFE error events for PAM4 can be reduced by using precoding
- PAM4 Tx precoding uses a 1/(1+D) mod 4
  - –See bliss_01_0311, "**Signaling Terminology; PAM-M and Partial Response Precoders"**
  - –Rx uses a (1+D) mod 4 after slicing
- Simple to implement
- Very low Complexity; similar complexity to duo-binary precoder
- Reduces 1 tap DFE burst error runs into 2 errors per error event
  - –One error at the entry, one error at the exit

1/(1+D) Precoding worked example



- Precoder Input : tx(n)
  - 2 2 2 2 0 3 2 0 1 3 3 0 0 0 0 2 3 0 3
- Precoder Output : p(n)
  - 0 2 0 2 2 1 1 3 2 1 2 2 2 2 2 0 3 1 2
- DFE, Slicer Output : d(n)
  - 0 1 1 1 3 0 2 2 3 0 3 1 3 1 3 0 3 1 2
- Error Event : p(n) – d(n)
  - 0 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 0 0 0 0
- Decoder Output after 1+D at Rx : r(n)
  - 2 1 2 2 0 3 2 0 1 3 3 0 0 0 0 3 3 0 3

Entry Error

Exit Error

This example does not include the PAM4 block termination.