100G Ten Bit Interface Proposal

Mark Gustlin and Oded Trainin - Cisco Systems
Med Belhadj – Cortina Systems
Brad Booth - AMCC
Tom Palkert – Xilinx
Subi Krishnamurthy – Force10 Networks
Schelto Van Doorn - Intel

IEEE HSSG, November 2006

Outline

- Considerations for the Interface
- Overview of the Interface
- Configuration Examples
- Alignment Options
- Open Issues
- Summary

Considerations for the interface

- Supports a single rate MAC speed (100G)
- Optical and electrical technologies requires a multi-channel/lane approach.
  - Channels will need to be bonded into one flow
  - Scheme needs to be robust to technology advances allowing future PMDs with reduced numbers of channels
- Minimize complexity of PMDs/Optical Modules
- Low overhead that is independent of packet size
- Enable small buffers
- Allow for differential delay due to wavelengths/fibers
- No auto-negotiation required between end points

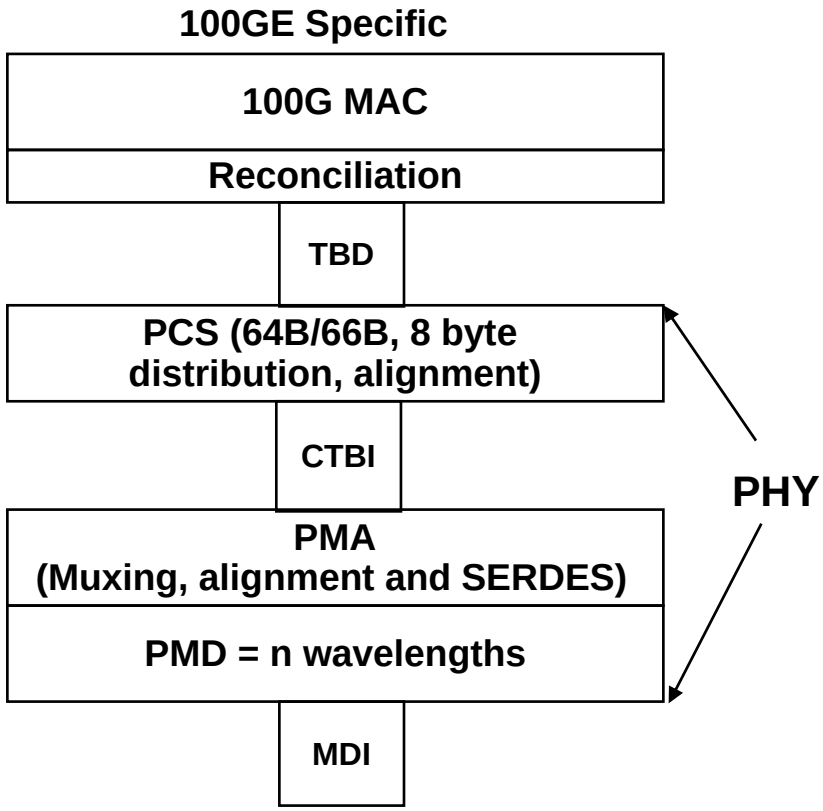
Non-considerations for the interface

Certain features are not considered necessary

- **Support for a scalable MAC**
  - **One MAC rate for the MAC Client is preferred**
- **Resiliency to single lane failures**
  - **Not considered necessary since a system level redundancy would be required anyway (e.g. Cable break, card power failure)**
  - **Working/protection links more than likely to be used**

In IEEE Terminology

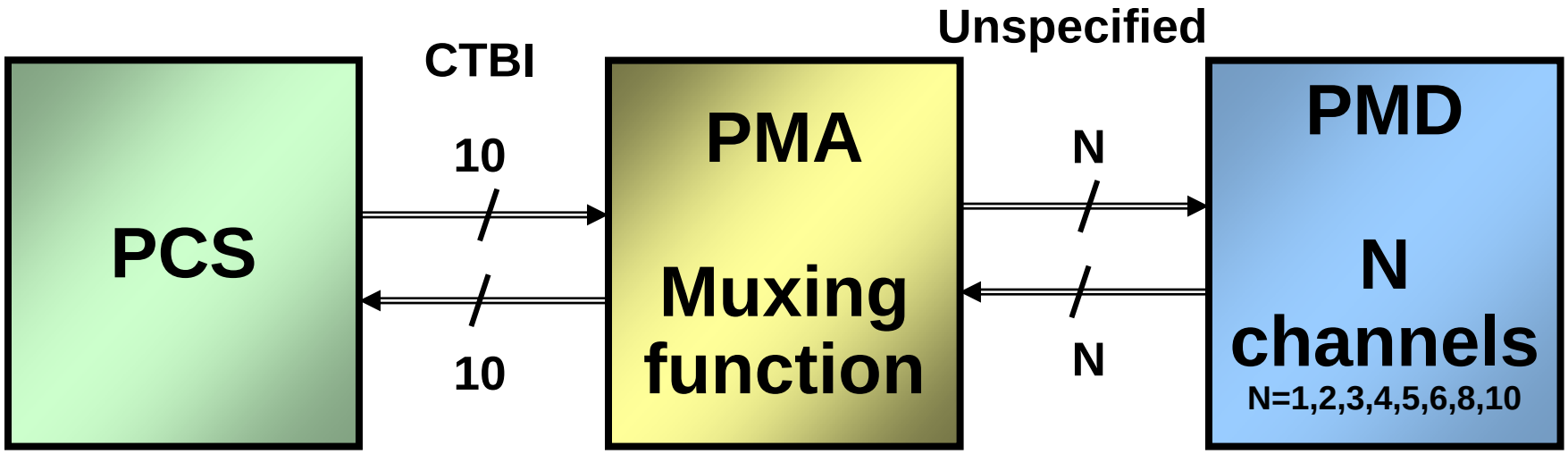
CTBI = 100G Ten Bit Interface
10 lanes at 10.3G (from XSBI)



100G Ten Bit Interface (CTBI) Overview

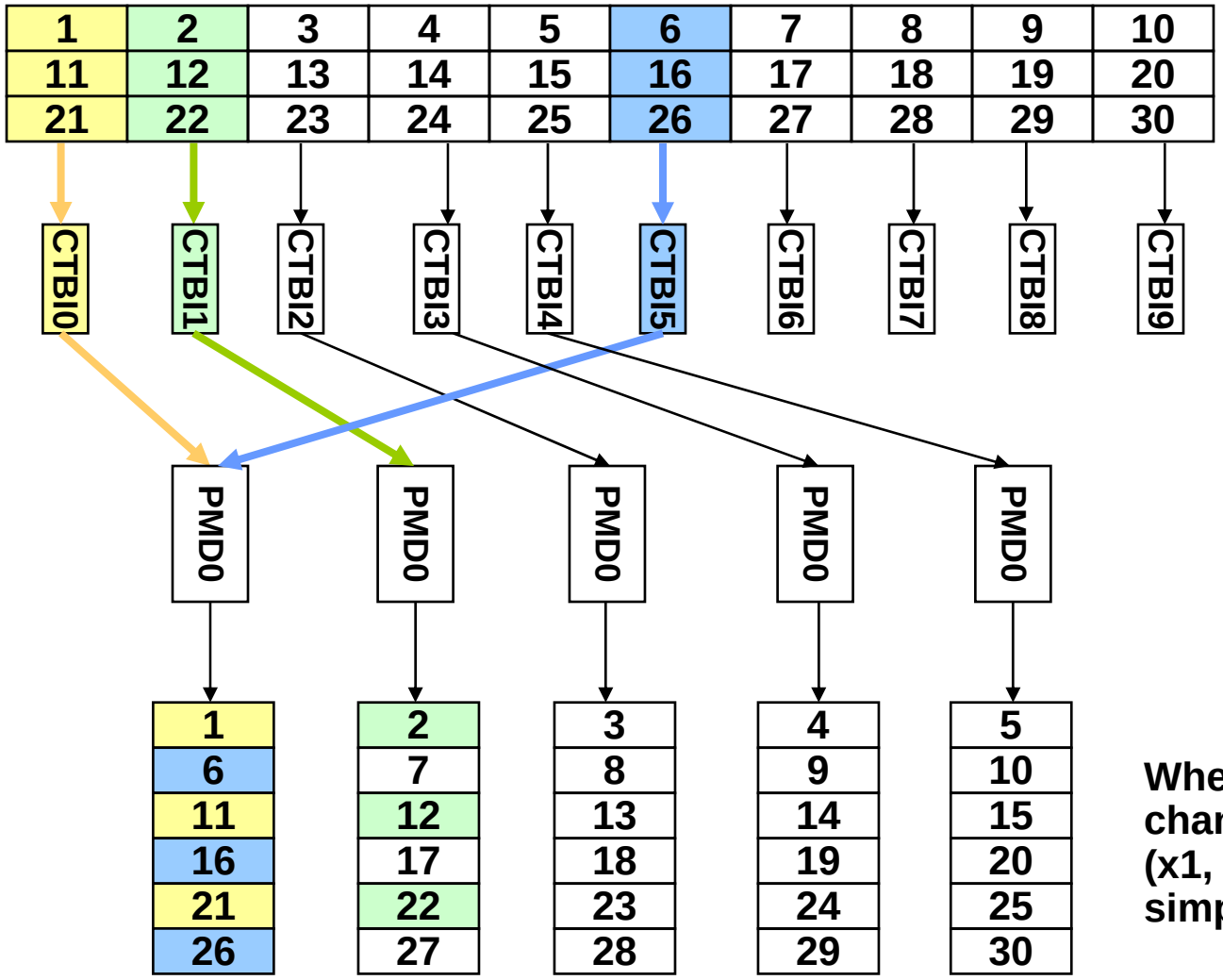
- 10 lanes @ 10.3G SERDES Interface to the PMA/PMD
- Uses 64B/66B encoding
- New 64B/66B encoded alignment mechanism
- Stripe data 8 bytes at a time across the lanes
  - It’s really 66 bits, 8 bytes plus 2 bits of encoding overhead
- Simple alignment and muxing is performed in the PMA
- The same alignment mechanism supports 2 stage alignment across the board electrical interface and PMD link if required.

Mapping from the CTBI to the PMD



Definition of CTBI allows simplified implementation of muxing function to enable support of many variants of PMDs that may be defined.

Example:
64B/66B based muxing (5 lanes of optics)



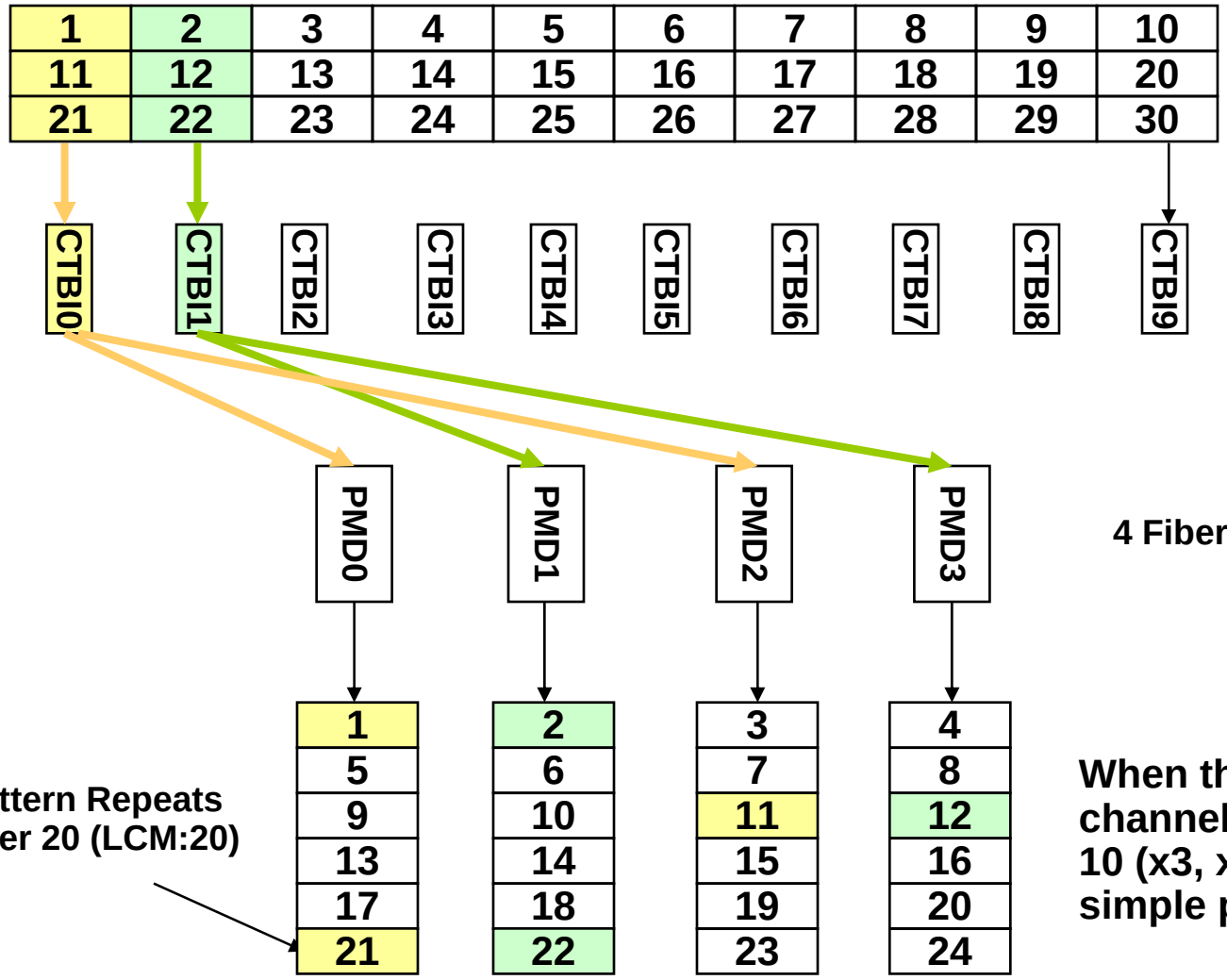
80 bytes at PCS per row

8 bytes per CTBI Lane

5 Fibers/Lambdas

When the number of PMD channels are a factor of 10 (x1, x2, x5, x10), muxing is simple round robin

Example:
64b/66B based muxing (4 lanes of optics)



80 bytes at PCS per row

8 bytes per CTBI Lane

4 Fibers/Lambdas

Pattern Repeats after 20 (LCM:20)

When the number of PMD channels are not a factor of 10 (x3, x4 x6, x8) gearbox is a simple pattern.

Encoded Packet Example – 4 Lane PMD

- **The following is an example of how a frame is encoded, and how it is played out across the interfaces**
- **The Frame can start on any CBTI lane**

CBTI Lane	PMD Lane	Sync	Block Payload							
0	0	10	0x78	Pr	Pr	Pr	Pr	Pr	Pr	SFD
1	1	01	DM	DM	DM	DM	DM	DM	SM	SM
2	2	01	SM	SM	SM	SM	ET	ET	D0	D1
3	3	01	D2	D3	D4	D5	D6	D7	D8	D9
4	0	01	D10	D11	D12	D13	D14	D15	D16	D17
5	1	01	D18	D19	D20	D21	D22	D23	D24	D25
6	2	01	D26	D27	D28	D29	D30	D31	D32	D33
7	3	01	D34	D35	D36	D37	D38	D39	D40	D41
8	0	01	D42	D43	D44	D45	CRC	CRC	CRC	CRC
9	1	10	0x87	00	00	00	00	00	00	00
0	2	10	0x33	00	00	00	00	Pr	Pr	Pr
1	3	01	Pr	Pr	Pr	SFD	DM	DM	DM	DM
2	0	01	DM	DM	SM	SM	SM	SM	SM	SM
3	1	01	ET	ET	D0	D1	D2	D3	D4	D5
4	2	01	D6	D7	D8	D9	D10	D11	D12	D13
5	3	01	D14	D15	D16	D17	D18	D19	D20	D21
6	0	01	D22	D23	D24	D25	D26	D27	D28	D29
7	1	01	D30	D31	D32	D33	D34	D35	D36	D37
8	2	01	D38	D39	D40	D41	D42	D43	D44	D45
9	3	10	0xcc	CRC	CRC	CRC	CRC	00	00	00
0	0	10	0x87	00	00	00	00	00	00	00

Alignment schemes

Alignment of the channels is a key requirement

- **Various schemes can be considered which require different levels of complexity in the PCS and PMA.**
- **One potential scheme is developed further in this presentation**

How to do the alignment?

XAUI uses comma characters sent on all 4 lanes at once:

XGMII													
TXD[7:0]	I	S	Dp	D	D	D	-	D	T	I	I	S	Dp
TXD[15:8]	I	Dp	Dp	D	D	D	-	D	I	I	I	Dp	Dp
TXD[23:16]	I	Dp	Dp	D	D	D	-	D	I	I	I	Dp	Dp
TXD[31:24]	I	Dp	Ds	D	D	D	-	D	I	I	I	Dp	Ds

XAUI													
Lane 0	R	S	Dp	D	D	D	-	D	T	A	R	S	Dp
Lane 1	R	Dp	Dp	D	D	D	-	D	K	A	R	Dp	Dp
Lane 2	R	Dp	Dp	D	D	D	-	D	K	A	R	Dp	Dp
Lane 3	R	Dp	Ds	D	D	D	-	D	K	A	R	Dp	Ds

We can do the same with 8 bytes at a time per lane

Send 8 bytes on each lane every so often.

Frequency would depend on how much overhead is considered acceptable.

How to do the alignment?

CTBI

Lane 0	n-9	A0	1	11	21	31	41	51	61
Lane 1	n-8	A1	2	12	22	32	42	52	62
Lane 2	n-7	A2	3	13	23	33	43	53	63
Lane 3	n-6	A3	4	14	24	34	44	54	64
Lane 4	n-5	A4	5	15	25	35	45	55	65
Lane 5	n-4	A5	6	16	26	36	46	56	66
Lane 6	n-3	A6	7	17	27	37	47	57	67
Lane 7	n-2	A7	8	18	28	38	48	58	68
Lane 8	n-1	A8	9	19	29	39	49	59	69
Lane 9	n	A9	10	20	30	40	50	60	70

Note that each box is 8 bytes

Assume lanes are aligned out of the MAC

Send an 8 byte alignment word on each lane with some frequency (TBD).

Frequency depends on how much overhead is considered acceptable.

Alignment – 10 to 5 Lanes

When you send the alignment on the 10 lane CTBI, what does it look like on an optical interface with less lanes?

Can the same alignment words be re-used? Yes.

Note that each box is 8 bytes

CGMII

Lane 0	n-9	A0	1	11	21	31	41	51	61
Lane 1	n-8	A1	2	12	22	32	42	52	62
Lane 2	n-7	A2	3	13	23	33	43	53	63
Lane 3	n-6	A3	4	14	24	34	44	54	64
Lane 4	n-5	A4	5	15	25	35	45	55	65
Lane 5	n-4	A5	6	16	26	36	46	56	66
Lane 6	n-3	A6	7	17	27	37	47	57	67
Lane 7	n-2	A7	8	18	28	38	48	58	68
Lane 8	n-1	A8	9	19	29	39	49	59	69
Lane 9	n	A9	10	20	30	40	50	60	70

5 Lanes of optics/lambdas It can only align this way

Lane 0	n-4	A0	A5	1	6	11	16	21	26
Lane 1	n-3	A1	A6	2	7	12	17	22	27
Lane 2	n-2	A2	A7	3	8	13	18	23	28
Lane 3	n-1	A3	A8	4	9	14	19	24	29
Lane 4	n	A4	A9	5	10	15	20	25	30

Rx side can do its own alignment based on the above

Alignment - 10 to 4 Lanes

Same concept with 4 optical lanes. At receiver only two potential patterns exist (lanes can also be skewed of course). Align on those.

4 Lanes of optics/lambdas 2 ways the alingment can occur (if no restrictions are place on the mapping)

Lane 0	n-3	A0	A4	A8	3	7	11	15	19
Lane 1	n-2	A1	A5	A9	4	8	12	16	20
Lane 2	n-1	A2	A6	1	5	9	13	17	21
Lane 3	n	A3	A7	2	6	10	14	18	22

Lane 0	n-5	n-1	A2	A6	1	5	9	13	17
Lane 1	n-4	n	A3	A7	2	6	10	14	18
Lane 2	n-3	A0	A4	A8	3	7	11	15	19
Lane 3	n-2	A1	A5	A9	4	8	12	16	20

Rx side can always detect which of the two scenarios it received and do its own alignment based on that (2 vs. 3 word on which lane).

Examples not shown for other lanes counts. In all cases, a detectable set of scenarios exist.

How often to send the Alignment?

How often you send alignment will depend on the following:

- **How much overhead you want to burden the interface with**
  - **Pushes toward slow repetition**
- **How much differential delay you want to be able to support**
  - **Pushes towards slow repetition**
- **How fast you want to be able to do alignment**
  - **Pushes towards fast repetition**

How often to send the Alignment?

- How much bandwidth does the alignment take?
  - 80B at a minimum if it can interrupt a packet
  - Maximum of 159B if it can not interrupt a packet
- How to account for the bandwidth used by alignment?
  - Implement a deficit counter (similar to the XAUI Deficit Counter)
  - Delete IPG to keep average Idles + Alignment = 12 per packet
  - Delete 4 or 8 IPG Bytes per packet?
  - Range of Alignment Deficit Counter (ADC) is 0 to 80
- Potential triggers sending the alignment:
  - Idle plus ADC going to 80 (80 Bytes have been deleted)
  - Fixed interval

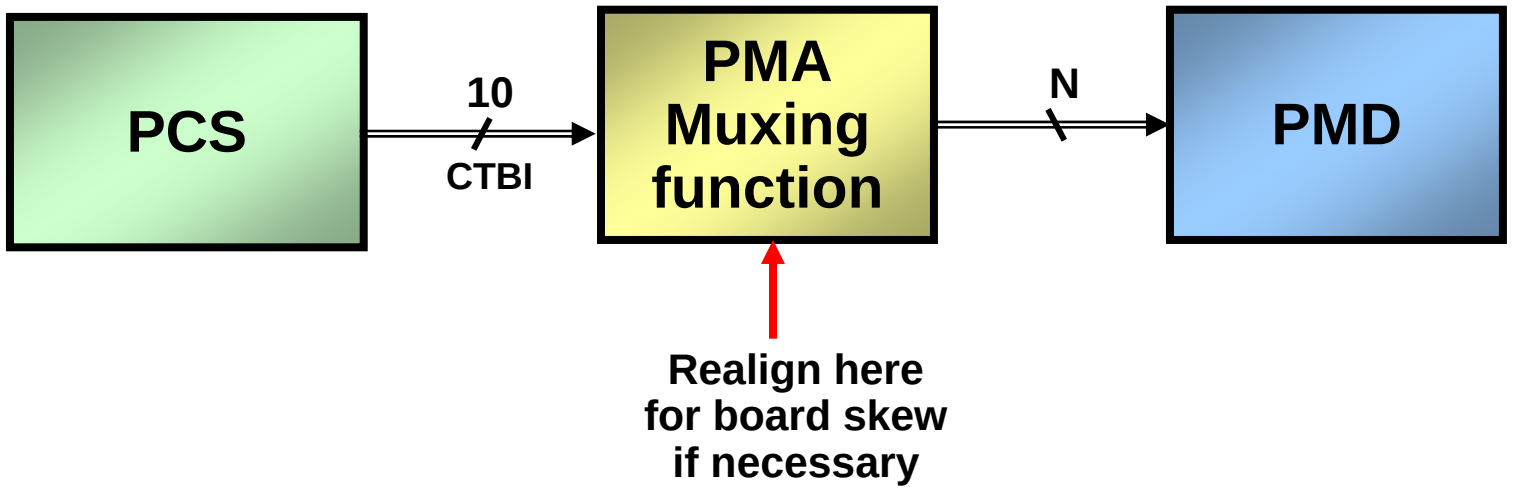
Alternative for Alignment

- **Send alignment on a fixed time basis**
- **For example every 120usec (approximately 1000 x1500B packets)**
- **It interrupts packets on the CTBI**
- **Takes only 0.005% of the Bandwidth**
- **Accept the small loss in BW or speed up the CTBI a little bit**
- **Simpler solution than maintaining the ADC**

A Note About Alignment - Egress

Where is alignment done?

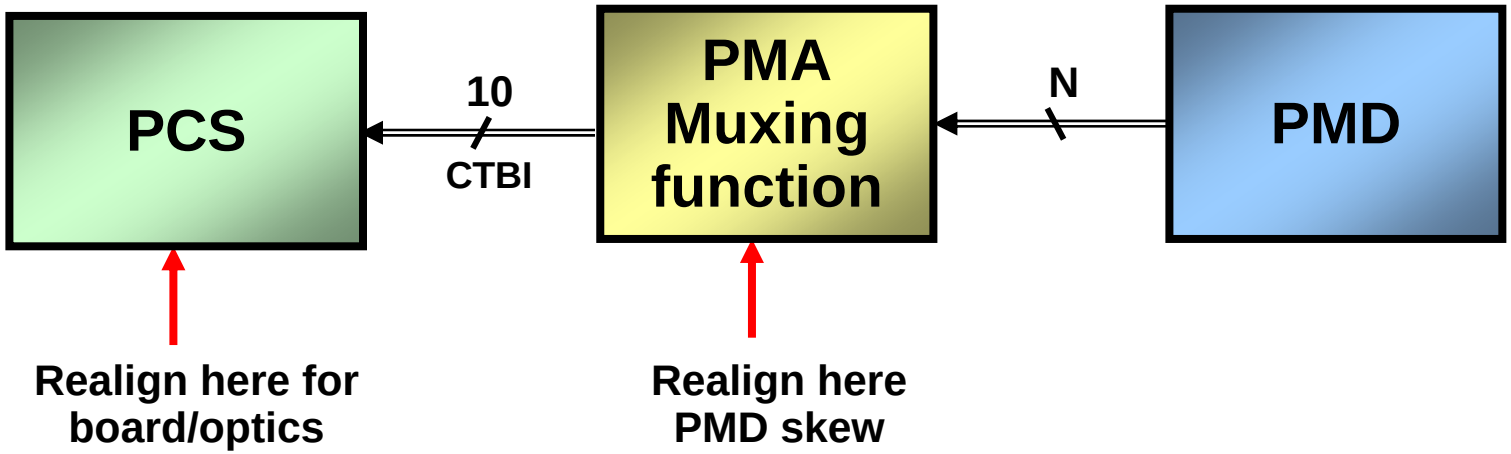
- **Lanes are sent aligned from the PCS**
- **Realigned on the PMA side of the of the CTBI (most of the time)**
- **Then distributed to the optics**
- **In some cases, whenever one lane of the CTBI is always sent on the same lane of the optics, the data does not need to be re-aligned in the PMA (x1,x2,x5,x10)**



A Note About Alignment - Ingress

Where is alignment done?

- **Before the CTBI, the data must be aligned. Sometimes this is trivial (2:10 for instance)**
- **If large skew is expected (say due to wavelength differential delay) then large buffers are in this function. Sometimes this alignment function is trivial such as a single optical interface…first alignment word must be sent on first lane of the CTBI.**
- **After the CTBI, the data must again be aligned to account for board level skew.**



Open Issues

- Scrambling required to maintain sufficient transition density
- Optimal scrambling scheme to be investigated
- Scrambling options include
  - At the PCS aggregate level (before data is striped across lanes)
  - At the CTBI lane level
  - At optical lane level
- We probably can’t scramble the alignment word
- We are investigating ways to make the PMA simpler:
  - Sending multiple alignment words to allow all alignment in the PCS only
  - Performing bit level muxing only at the PMA, requires more complicated muxing in the PCS

Summary

- **CTBI proposed as a potential interface for HSSG**
- **Enables lane bonding/aggregation at electrical and optical levels with a single alignment mechanism**
- **One PCS for many PMDs**
- **Low overhead which is independent of packet size**
- **Minimizes latency, minimizes buffer sizes**