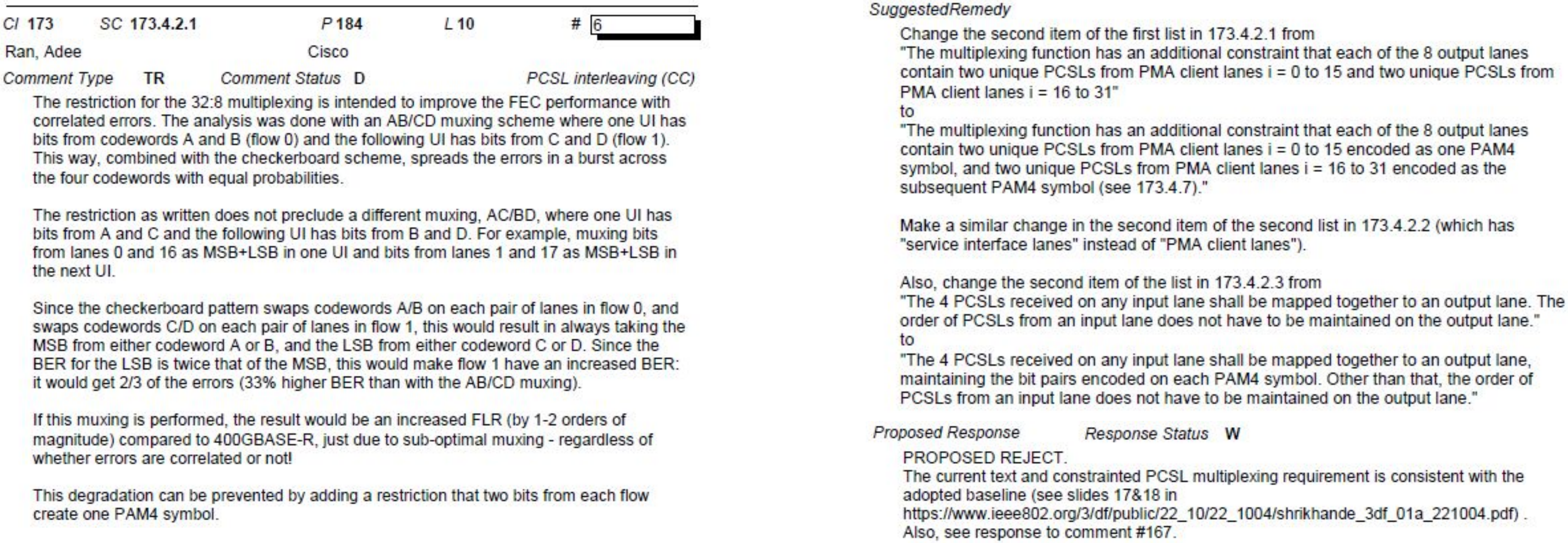
802.3df D1.0
Comment Resolution

P802.3df editorial team

Cross-Clause

CC: PCSL interleaving, FEC performance
Comment 6



Cl **173** *SC* **173.4.2.1** *P* **184** *L* **10** **#** 6

Ran, Adee Cisco

Comment Type **TR** *Comment Status* **D** *PCSL interleaving (CC)*

The restriction for the 32:8 multiplexing is intended to improve the FEC performance with correlated errors. The analysis was done with an AB/CD muxing scheme where one UI has bits from codewords A and B (flow 0) and the following UI has bits from C and D (flow 1). This way, combined with the checkerboard scheme, spreads the errors in a burst across the four codewords with equal probabilities.

The restriction as written does not preclude a different muxing, AC/BD, where one UI has bits from A and C and the following UI has bits from B and D. For example, muxing bits from lanes 0 and 16 as MSB+LSB in one UI and bits from lanes 1 and 17 as MSB+LSB in the next UI.

Since the checkerboard pattern swaps codewords A/B on each pair of lanes in flow 0, and swaps codewords C/D on each pair of lanes in flow 1, this would result in always taking the MSB from either codeword A or B, and the LSB from either codeword C or D. Since the BER for the LSB is twice that of the MSB, this would make flow 1 have an increased BER: it would get 2/3 of the errors (33% higher BER than with the AB/CD muxing).

If this muxing is performed, the result would be an increased FLR (by 1-2 orders of magnitude) compared to 400GBASE-R, just due to sub-optimal muxing - regardless of whether errors are correlated or not!

This degradation can be prevented by adding a restriction that two bits from each flow create one PAM4 symbol.

SuggestedRemedy

Change the second item of the first list in 173.4.2.1 from
"The multiplexing function has an additional constraint that each of the 8 output lanes contain two unique PCSLs from PMA client lanes i = 0 to 15 and two unique PCSLs from PMA client lanes i = 16 to 31"
to
"The multiplexing function has an additional constraint that each of the 8 output lanes contain two unique PCSLs from PMA client lanes i = 0 to 15 encoded as one PAM4 symbol, and two unique PCSLs from PMA client lanes i = 16 to 31 encoded as the subsequent PAM4 symbol (see 173.4.7)."

Make a similar change in the second item of the second list in 173.4.2.2 (which has "service interface lanes" instead of "PMA client lanes").

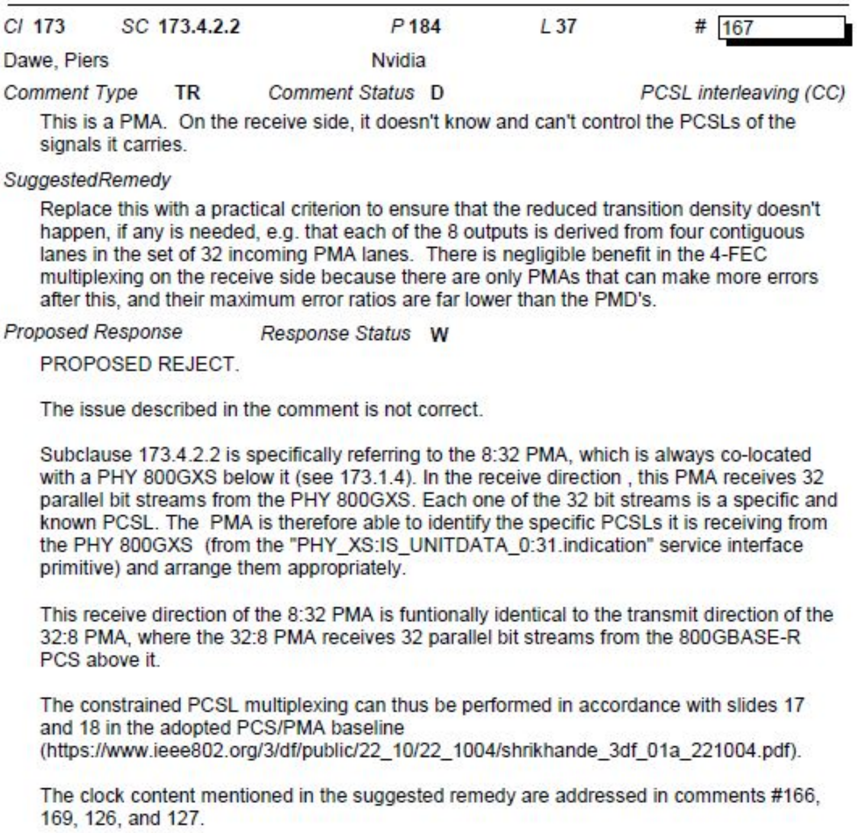
Also, change the second item of the list in 173.4.2.3 from
"The 4 PCSLs received on any input lane shall be mapped together to an output lane. The order of PCSLs from an input lane does not have to be maintained on the output lane."
to
"The 4 PCSLs received on any input lane shall be mapped together to an output lane, maintaining the bit pairs encoded on each PAM4 symbol. Other than that, the order of PCSLs from an input lane does not have to be maintained on the output lane."

Proposed Response *Response Status* **W**

PROPOSED REJECT.
The current text and constrainted PCSL multiplexing requirement is consistent with the adopted baseline (see slides 17&18 in https://www.ieee802.org/3/df/public/22_10/22_1004/shrikhande_3df_01a_221004.pdf) .
Also, see response to comment #167.

There is a related presentation:
https://www.ieee802.org/3/df/public/22_12/ran_3df_01_2212.pdf

CC: PCSL interleaving, known lanes
Comment 167



Cl **173** *SC* **173.4.2.2** *P* **184** *L* **37** # 167

Dawe, Piers Nvidia

Comment Type **TR** *Comment Status* **D** *PCSL interleaving (CC)*

This is a PMA. On the receive side, it doesn't know and can't control the PCSLs of the signals it carries.

SuggestedRemedy

Replace this with a practical criterion to ensure that the reduced transition density doesn't happen, if any is needed, e.g. that each of the 8 outputs is derived from four contiguous lanes in the set of 32 incoming PMA lanes. There is negligible benefit in the 4-FEC multiplexing on the receive side because there are only PMAs that can make more errors after this, and their maximum error ratios are far lower than the PMD's.

Proposed Response *Response Status* **W**

PROPOSED REJECT.

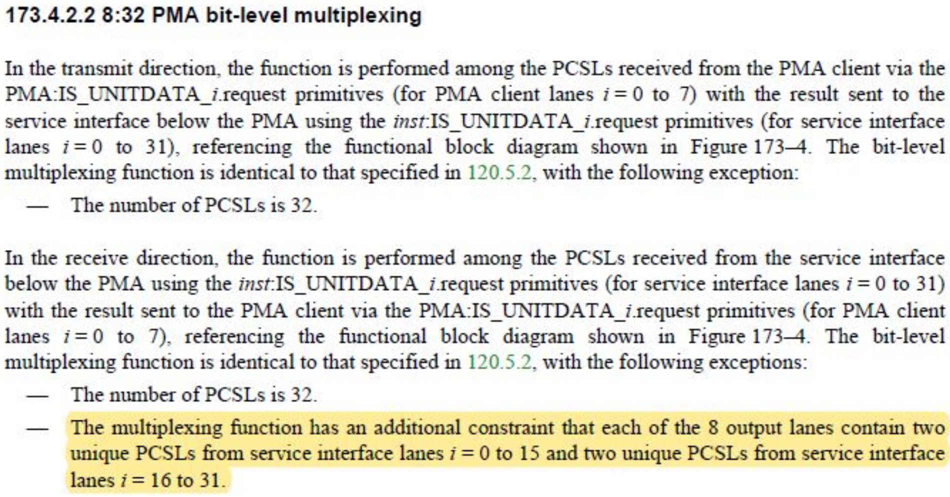
The issue described in the comment is not correct.

Subclause 173.4.2.2 is specifically referring to the 8:32 PMA, which is always co-located with a PHY 800GXS below it (see 173.1.4). In the receive direction , this PMA receives 32 parallel bit streams from the PHY 800GXS. Each one of the 32 bit streams is a specific and known PCSL. The PMA is therefore able to identify the specific PCSLs it is receiving from the PHY 800GXS (from the "PHY_XS:IS_UNITDATA_0:31.indication" service interface primitive) and arrange them appropriately.

This receive direction of the 8:32 PMA is funtionally identical to the transmit direction of the 32:8 PMA, where the 32:8 PMA receives 32 parallel bit streams from the 800GBASE-R PCS above it.

The constrained PCSL multiplexing can thus be performed in accordance with slides 17 and 18 in the adopted PCS/PMA baseline (https://www.ieee802.org/3/df/public/22_10/22_1004/shrikhande_3df_01a_221004.pdf).

The clock content mentioned in the suggested remedy are addressed in comments #166, 169, 126, and 127.



173.4.2.2 8:32 PMA bit-level multiplexing

In the transmit direction, the function is performed among the PCSLs received from the PMA client via the PMA:IS_UNITDATA_*i*.request primitives (for PMA client lanes *i* = 0 to 7) with the result sent to the service interface below the PMA using the *inst*:IS_UNITDATA_*i*.request primitives (for service interface lanes *i* = 0 to 31), referencing the functional block diagram shown in Figure 173–4. The bit-level multiplexing function is identical to that specified in 120.5.2, with the following exception:

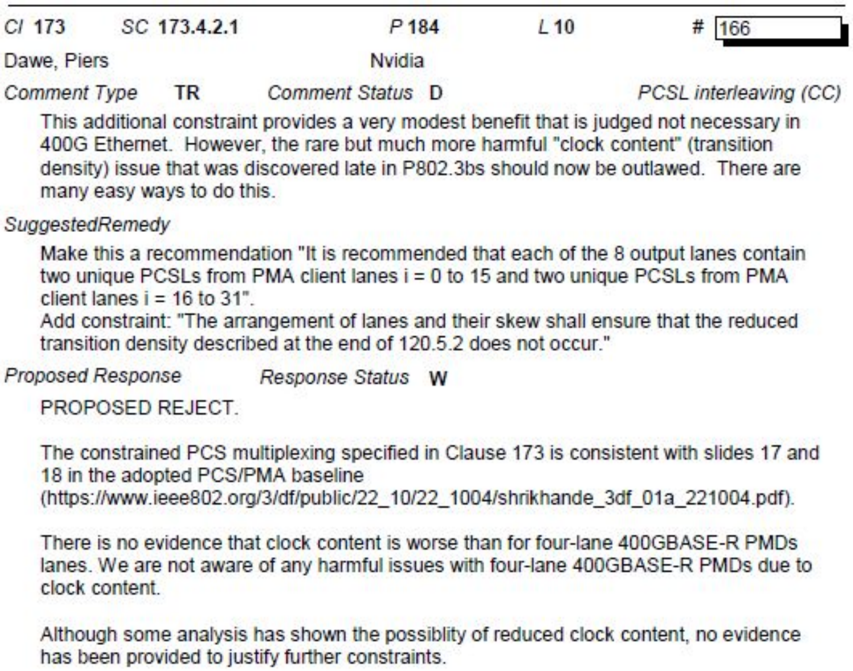
— The number of PCSLs is 32.

In the receive direction, the function is performed among the PCSLs received from the service interface below the PMA using the *inst*:IS_UNITDATA_*i*.request primitives (for service interface lanes *i* = 0 to 31) with the result sent to the PMA client via the PMA:IS_UNITDATA_*i*.request primitives (for PMA client lanes *i* = 0 to 7), referencing the functional block diagram shown in Figure 173–4. The bit-level multiplexing function is identical to that specified in 120.5.2, with the following exceptions:

— The number of PCSLs is 32.

— The multiplexing function has an additional constraint that each of the 8 output lanes contain two unique PCSLs from service interface lanes *i* = 0 to 15 and two unique PCSLs from service interface lanes *i* = 16 to 31.

CC: PCSL interleaving, clock content (part 1)
Comments 166, 169, 126, 127



Cl **173** *SC* **173.4.2.1** *P* **184** *L* **10** # 166

Dawe, Piers Nvidia

Comment Type **TR** *Comment Status* **D** *PCSL interleaving (CC)*

This additional constraint provides a very modest benefit that is judged not necessary in 400G Ethernet. However, the rare but much more harmful "clock content" (transition density) issue that was discovered late in P802.3bs should now be outlawed. There are many easy ways to do this.

SuggestedRemedy

Make this a recommendation "It is recommended that each of the 8 output lanes contain two unique PCSLs from PMA client lanes i = 0 to 15 and two unique PCSLs from PMA client lanes i = 16 to 31".
Add constraint: "The arrangement of lanes and their skew shall ensure that the reduced transition density described at the end of 120.5.2 does not occur."

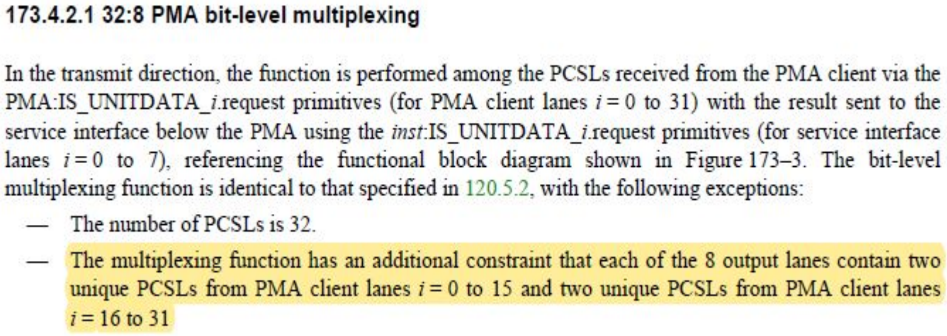
Proposed Response *Response Status* **W**

PROPOSED REJECT.

The constrained PCS multiplexing specified in Clause 173 is consistent with slides 17 and 18 in the adopted PCS/PMA baseline (https://www.ieee802.org/3/df/public/22_10/22_1004/shrikhande_3df_01a_221004.pdf).

There is no evidence that clock content is worse than for four-lane 400GBASE-R PMDs lanes. We are not aware of any harmful issues with four-lane 400GBASE-R PMDs due to clock content.

Although some analysis has shown the possiblity of reduced clock content, no evidence has been provided to justify further constraints.



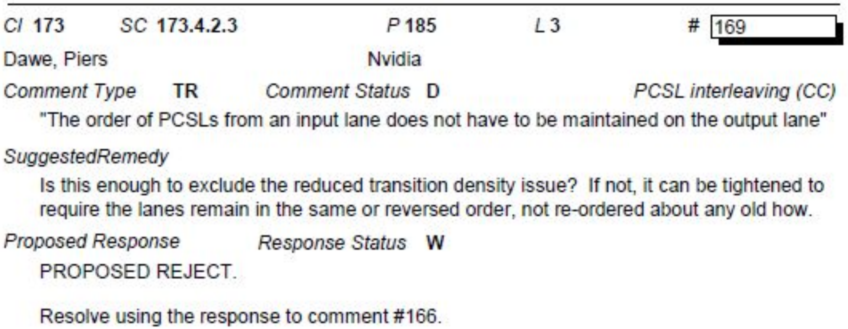
173.4.2.1 32:8 PMA bit-level multiplexing

In the transmit direction, the function is performed among the PCSLs received from the PMA client via the PMA:IS_UNITDATA_*i*.request primitives (for PMA client lanes *i* = 0 to 31) with the result sent to the service interface below the PMA using the *inst*:IS_UNITDATA_*i*.request primitives (for service interface lanes *i* = 0 to 7), referencing the functional block diagram shown in Figure 173–3. The bit-level multiplexing function is identical to that specified in 120.5.2, with the following exceptions:

— The number of PCSLs is 32.

— The multiplexing function has an additional constraint that each of the 8 output lanes contain two unique PCSLs from PMA client lanes *i* = 0 to 15 and two unique PCSLs from PMA client lanes *i* = 16 to 31

CC: PCSL interleaving, clock content (part 2)
Comments 166, 169, 126, 127



Cl **173** *SC* **173.4.2.3** *P* **185** *L* **3** # 169

Dawe, Piers Nvidia

Comment Type **TR** *Comment Status* **D** *PCSL interleaving (CC)*

"The order of PCSLs from an input lane does not have to be maintained on the output lane"

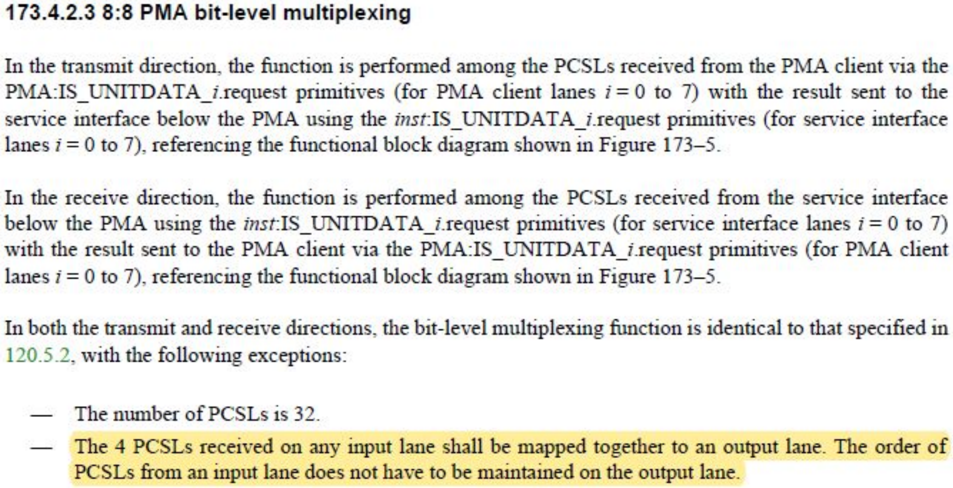
SuggestedRemedy

Is this enough to exclude the reduced transition density issue? If not, it can be tightened to require the lanes remain in the same or reversed order, not re-ordered about any old how.

Proposed Response *Response Status* **W**

PROPOSED REJECT.

Resolve using the response to comment #166.



173.4.2.3 8:8 PMA bit-level multiplexing

In the transmit direction, the function is performed among the PCSLs received from the PMA client via the PMA:IS_UNITDATA_*i*.request primitives (for PMA client lanes *i* = 0 to 7) with the result sent to the service interface below the PMA using the *inst*:IS_UNITDATA_*i*.request primitives (for service interface lanes *i* = 0 to 7), referencing the functional block diagram shown in Figure 173–5.

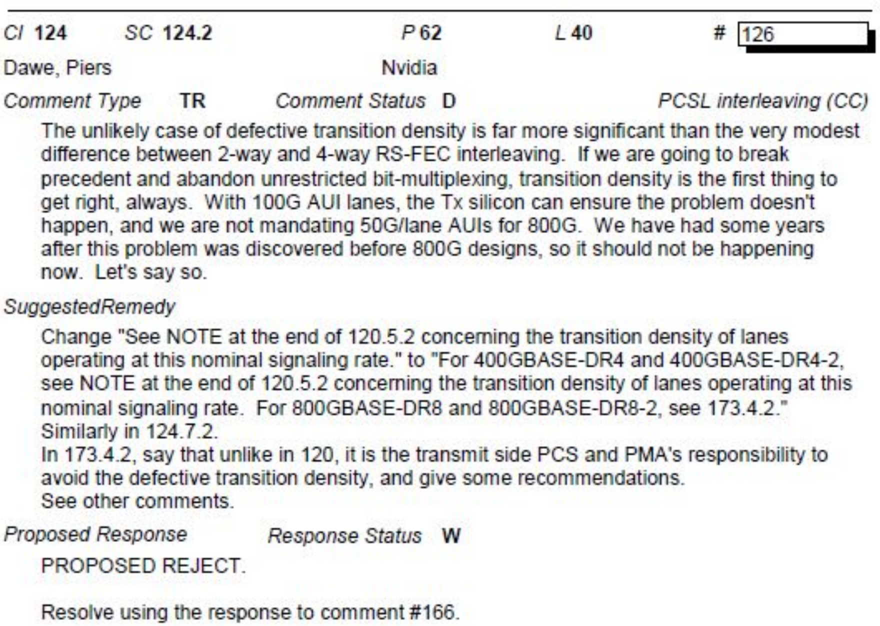
In the receive direction, the function is performed among the PCSLs received from the service interface below the PMA using the *inst*:IS_UNITDATA_*i*.request primitives (for service interface lanes *i* = 0 to 7) with the result sent to the PMA client via the PMA:IS_UNITDATA_*i*.request primitives (for PMA client lanes *i* = 0 to 7), referencing the functional block diagram shown in Figure 173–5.

In both the transmit and receive directions, the bit-level multiplexing function is identical to that specified in 120.5.2, with the following exceptions:

— The number of PCSLs is 32.

— The 4 PCSLs received on any input lane shall be mapped together to an output lane. The order of PCSLs from an input lane does not have to be maintained on the output lane.

CC: PCSL interleaving, clock content (part 3)
Comments 166, 169, 126, 127



Cl **124** *SC* **124.2** *P* **62** *L* **40** **#** 126

Dawe, Piers Nvidia

Comment Type **TR** *Comment Status* **D** *PCSL interleaving (CC)*

The unlikely case of defective transition density is far more significant than the very modest difference between 2-way and 4-way RS-FEC interleaving. If we are going to break precedent and abandon unrestricted bit-multiplexing, transition density is the first thing to get right, always. With 100G AUI lanes, the Tx silicon can ensure the problem doesn't happen, and we are not mandating 50G/lane AUIs for 800G. We have had some years after this problem was discovered before 800G designs, so it should not be happening now. Let's say so.

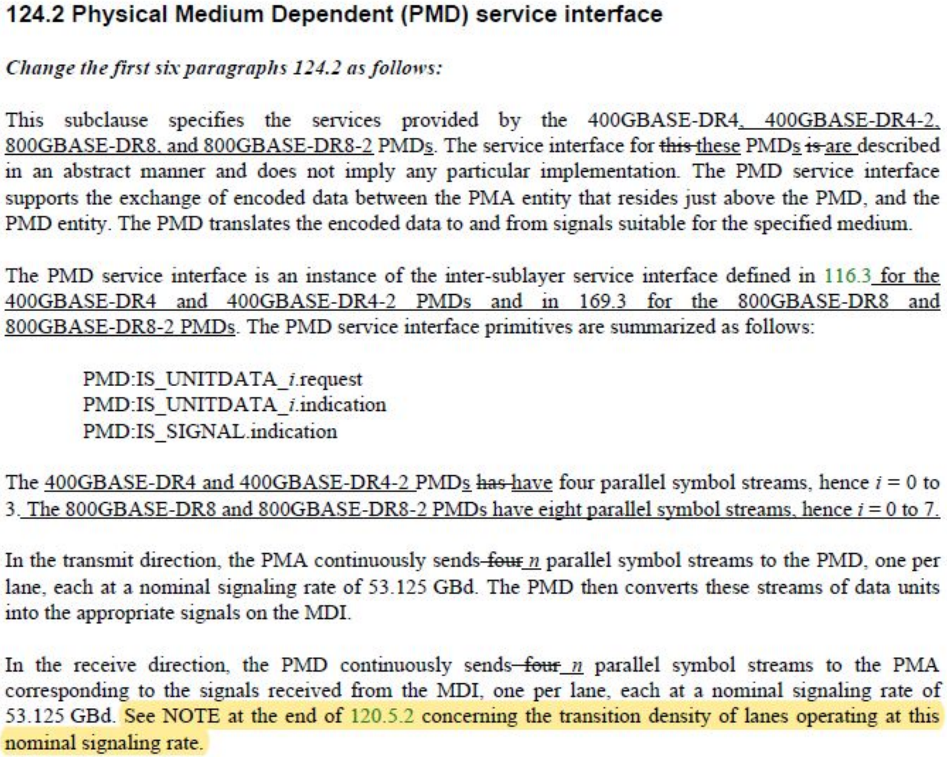
SuggestedRemedy

Change "See NOTE at the end of 120.5.2 concerning the transition density of lanes operating at this nominal signaling rate." to "For 400GBASE-DR4 and 400GBASE-DR4-2, see NOTE at the end of 120.5.2 concerning the transition density of lanes operating at this nominal signaling rate. For 800GBASE-DR8 and 800GBASE-DR8-2, see 173.4.2."
Similarly in 124.7.2.
In 173.4.2, say that unlike in 120, it is the transmit side PCS and PMA's responsibility to avoid the defective transition density, and give some recommendations.
See other comments.

Proposed Response *Response Status* **W**

PROPOSED REJECT.

Resolve using the response to comment #166.



124.2 Physical Medium Dependent (PMD) service interface

Change the first six paragraphs 124.2 as follows:

This subclause specifies the services provided by the 400GBASE-DR4, 400GBASE-DR4-2, 800GBASE-DR8, and 800GBASE-DR8-2 PMDs. The service interface for ~~this~~ these PMDs ~~is~~ are described in an abstract manner and does not imply any particular implementation. The PMD service interface supports the exchange of encoded data between the PMA entity that resides just above the PMD, and the PMD entity. The PMD translates the encoded data to and from signals suitable for the specified medium.

The PMD service interface is an instance of the inter-sublayer service interface defined in 116.3 for the 400GBASE-DR4 and 400GBASE-DR4-2 PMDs and in 169.3 for the 800GBASE-DR8 and 800GBASE-DR8-2 PMDs. The PMD service interface primitives are summarized as follows:

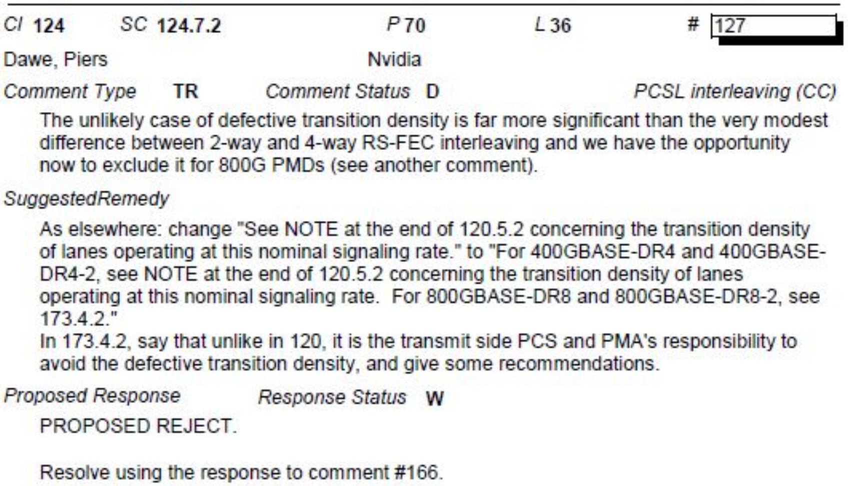
PMD:IS_UNITDATA_*i*.request
PMD:IS_UNITDATA_*i*.indication
PMD:IS_SIGNAL.indication

The 400GBASE-DR4 and 400GBASE-DR4-2 PMDs ~~has~~ have four parallel symbol streams, hence $i = 0$ to 3. The 800GBASE-DR8 and 800GBASE-DR8-2 PMDs have eight parallel symbol streams, hence $i = 0$ to 7.

In the transmit direction, the PMA continuously sends ~~four~~ *n* parallel symbol streams to the PMD, one per lane, each at a nominal signaling rate of 53.125 GBd. The PMD then converts these streams of data units into the appropriate signals on the MDI.

In the receive direction, the PMD continuously sends ~~four~~ *n* parallel symbol streams to the PMA corresponding to the signals received from the MDI, one per lane, each at a nominal signaling rate of 53.125 GBd. See NOTE at the end of 120.5.2 concerning the transition density of lanes operating at this nominal signaling rate.

CC: PCSL interleaving, clock content (part 4) Comments 166, 169, 126, 127



Cl 124 *SC* 124.7.2 *P* 70 *L* 36 # 127

Dawe, Piers Nvidia

Comment Type TR *Comment Status* D *PCSL interleaving (CC)*

The unlikely case of defective transition density is far more significant than the very modest difference between 2-way and 4-way RS-FEC interleaving and we have the opportunity now to exclude it for 800G PMDs (see another comment).

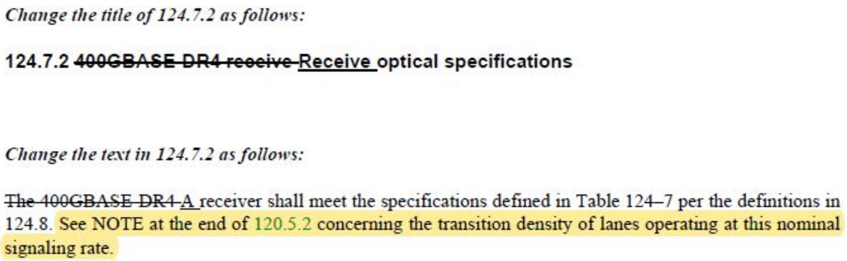
SuggestedRemedy

As elsewhere: change "See NOTE at the end of 120.5.2 concerning the transition density of lanes operating at this nominal signaling rate." to "For 400GBASE-DR4 and 400GBASE-DR4-2, see NOTE at the end of 120.5.2 concerning the transition density of lanes operating at this nominal signaling rate. For 800GBASE-DR8 and 800GBASE-DR8-2, see 173.4.2."
In 173.4.2, say that unlike in 120, it is the transmit side PCS and PMA's responsibility to avoid the defective transition density, and give some recommendations.

Proposed Response *Response Status* W

PROPOSED REJECT.

Resolve using the response to comment #166.

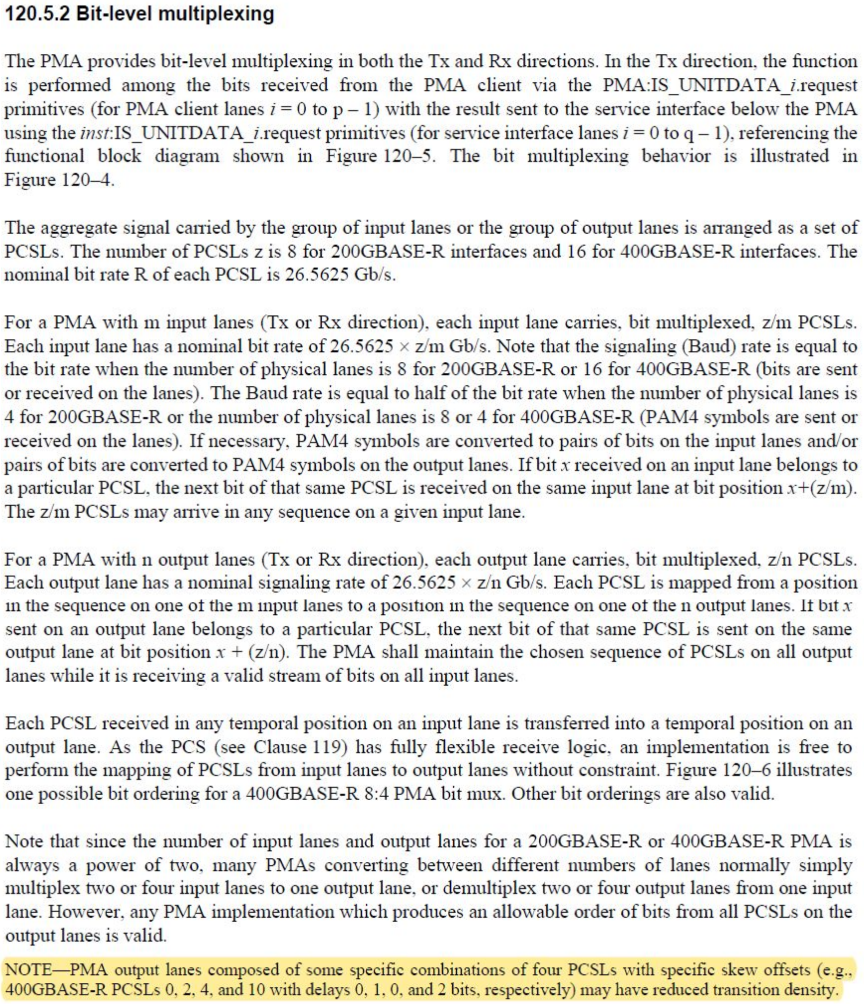


Change the title of 124.7.2 as follows:

124.7.2 ~~400GBASE-DR4 receive~~ Receive optical specifications

Change the text in 124.7.2 as follows:

~~The 400GBASE-DR4~~ A receiver shall meet the specifications defined in Table 124–7 per the definitions in 124.8. See NOTE at the end of 120.5.2 concerning the transition density of lanes operating at this nominal signaling rate.



120.5.2 Bit-level multiplexing

The PMA provides bit-level multiplexing in both the Tx and Rx directions. In the Tx direction, the function is performed among the bits received from the PMA client via the PMA:IS_UNITDATA_*i*.request primitives (for PMA client lanes $i = 0$ to $p - 1$) with the result sent to the service interface below the PMA using the *inst*:IS_UNITDATA_*i*.request primitives (for service interface lanes $i = 0$ to $q - 1$), referencing the functional block diagram shown in Figure 120–5. The bit multiplexing behavior is illustrated in Figure 120–4.

The aggregate signal carried by the group of input lanes or the group of output lanes is arranged as a set of PCSLs. The number of PCSLs z is 8 for 200GBASE-R interfaces and 16 for 400GBASE-R interfaces. The nominal bit rate R of each PCSL is 26.5625 Gb/s.

For a PMA with m input lanes (Tx or Rx direction), each input lane carries, bit multiplexed, z/m PCSLs. Each input lane has a nominal bit rate of 26.5625 × z/m Gb/s. Note that the signaling (Baud) rate is equal to the bit rate when the number of physical lanes is 8 for 200GBASE-R or 16 for 400GBASE-R (bits are sent or received on the lanes). The Baud rate is equal to half of the bit rate when the number of physical lanes is 4 for 200GBASE-R or the number of physical lanes is 8 or 4 for 400GBASE-R (PAM4 symbols are sent or received on the lanes). If necessary, PAM4 symbols are converted to pairs of bits on the input lanes and/or pairs of bits are converted to PAM4 symbols on the output lanes. If bit x received on an input lane belongs to a particular PCSL, the next bit of that same PCSL is received on the same input lane at bit position x+(z/m). The z/m PCSLs may arrive in any sequence on a given input lane.

For a PMA with n output lanes (Tx or Rx direction), each output lane carries, bit multiplexed, z/n PCSLs. Each output lane has a nominal signaling rate of 26.5625 × z/n Gb/s. Each PCSL is mapped from a position in the sequence on one of the m input lanes to a position in the sequence on one of the n output lanes. If bit x sent on an output lane belongs to a particular PCSL, the next bit of that same PCSL is sent on the same output lane at bit position x + (z/n). The PMA shall maintain the chosen sequence of PCSLs on all output lanes while it is receiving a valid stream of bits on all input lanes.

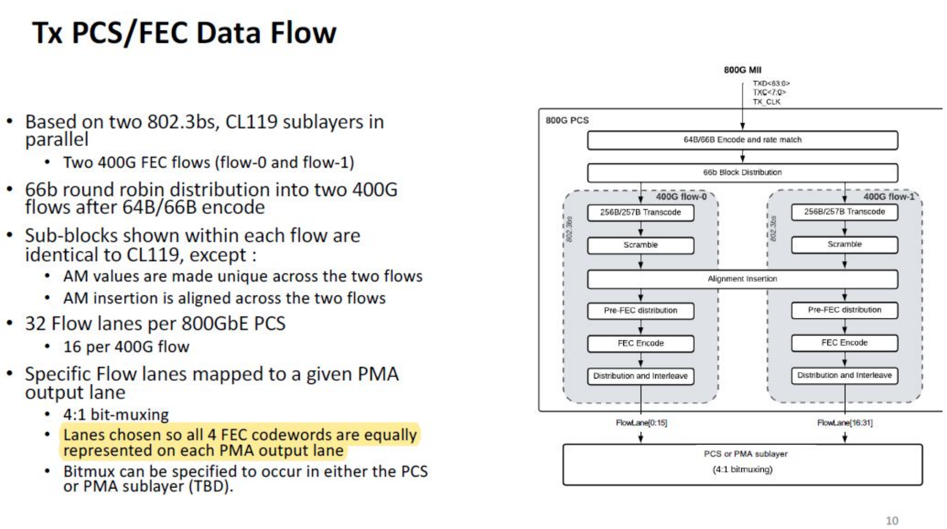
Each PCSL received in any temporal position on an input lane is transferred into a temporal position on an output lane. As the PCS (see Clause 119) has fully flexible receive logic, an implementation is free to perform the mapping of PCSLs from input lanes to output lanes without constraint. Figure 120–6 illustrates one possible bit ordering for a 400GBASE-R 8:4 PMA bit mux. Other bit orderings are also valid.

Note that since the number of input lanes and output lanes for a 200GBASE-R or 400GBASE-R PMA is always a power of two, many PMAs converting between different numbers of lanes normally simply multiplex two or four input lanes to one output lane, or demultiplex two or four output lanes from one input lane. However, any PMA implementation which produces an allowable order of bits from all PCSLs on the output lanes is valid.

NOTE—PMA output lanes composed of some specific combinations of four PCSLs with specific skew offsets (e.g., 400GBASE-R PCSLs 0, 2, 4, and 10 with delays 0, 1, 0, and 2 bits, respectively) may have reduced transition density.

CC: PCSL interleaving, clock content (part 5)
Comments 166, 169, 126, 127

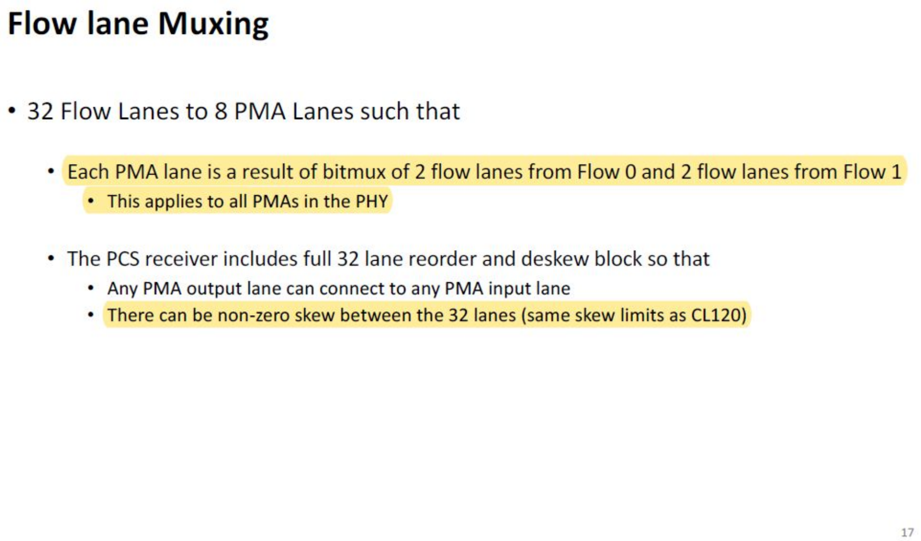
Slides 10 and 17 from adopted baseline:
https://www.ieee802.org/3/df/public/22_10/22_1004/shrikhande_3df_01a_221004.pdf



Tx PCS/FEC Data Flow

- Based on two 802.3bs, CL119 sublayers in parallel
  - Two 400G FEC flows (flow-0 and flow-1)
- 66b round robin distribution into two 400G flows after 64B/66B encode
- Sub-blocks shown within each flow are identical to CL119, except :
  - AM values are made unique across the two flows
  - AM insertion is aligned across the two flows
- 32 Flow lanes per 800GbE PCS
  - 16 per 400G flow
- Specific Flow lanes mapped to a given PMA output lane
  - 4:1 bit-muxing
  - Lanes chosen so all 4 FEC codewords are equally represented on each PMA output lane
  - Bitmux can be specified to occur in either the PCS or PMA sublayer (TBD).

10

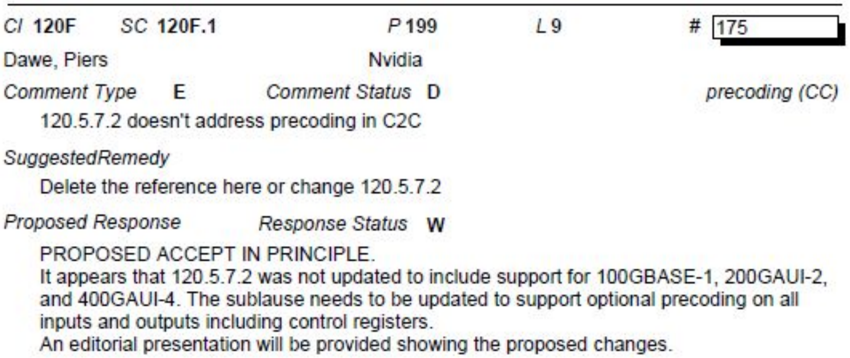


Flow lane Muxing

- 32 Flow Lanes to 8 PMA Lanes such that
  - Each PMA lane is a result of bitmux of 2 flow lanes from Flow 0 and 2 flow lanes from Flow 1
    - This applies to all PMAs in the PHY
  - The PCS receiver includes full 32 lane reorder and deskew block so that
    - Any PMA output lane can connect to any PMA input lane
    - There can be non-zero skew between the 32 lanes (same skew limits as CL120)

17

CC: Precoding (part 1)
Comment #175 - explained



Cl 120F *SC* 120F.1 *P* 199 *L* 9 # 175

Dawe, Piers Nvidia

Comment Type E *Comment Status* D *precoding (CC)*

120.5.7.2 doesn't address precoding in C2C

SuggestedRemedy

Delete the reference here or change 120.5.7.2

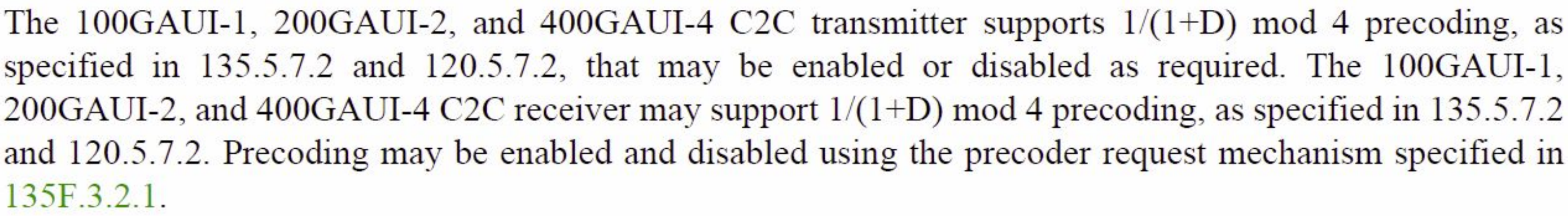
Proposed Response *Response Status* W

PROPOSED ACCEPT IN PRINCIPLE.
It appears that 120.5.7.2 was not updated to include support for 100GBASE-1, 200GAUI-2, and 400GAUI-4. The sublause needs to be updated to support optional precoding on all inputs and outputs including control registers.
An editorial presentation will be provided showing the proposed changes.

In 802.3ck, precoding capability was specified for 100GAUI-1, 200GAUI-2, and 400GAUI-4 (as shown by the last paragraph of 120F.1, below). 135.5.7.2 was updated to include 100GAUI-1, but 120.5.7.2 was not updated to specify this option for 200GAUI-2 and 400GAUI-4.

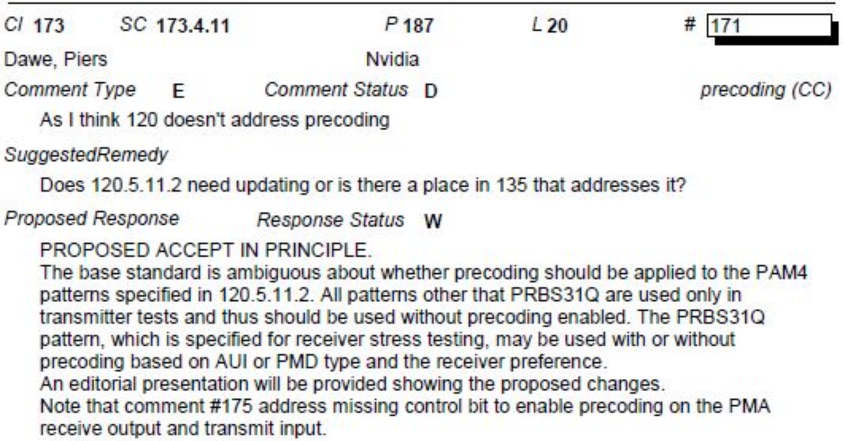
120F.1 Overview

…



The 100GAUI-1, 200GAUI-2, and 400GAUI-4 C2C transmitter supports 1/(1+D) mod 4 precoding, as specified in 135.5.7.2 and 120.5.7.2, that may be enabled or disabled as required. The 100GAUI-1, 200GAUI-2, and 400GAUI-4 C2C receiver may support 1/(1+D) mod 4 precoding, as specified in 135.5.7.2 and 120.5.7.2. Precoding may be enabled and disabled using the precoder request mechanism specified in 135F.3.2.1.

CC: Precoding (part 2)
Comment #171 - explained



Cl 173 *SC* 173.4.11 *P* 187 *L* 20 # 171

Dawe, Piers Nvidia

Comment Type E *Comment Status* D *precoding (CC)*

As I think 120 doesn't address precoding

SuggestedRemedy

Does 120.5.11.2 need updating or is there a place in 135 that addresses it?

Proposed Response *Response Status* W

PROPOSED ACCEPT IN PRINCIPLE.
The base standard is ambiguous about whether precoding should be applied to the PAM4 patterns specified in 120.5.11.2. All patterns other that PRBS31Q are used only in transmitter tests and thus should be used without precoding enabled. The PRBS31Q pattern, which is specified for receiver stress testing, may be used with or without precoding based on AUI or PMD type and the receiver preference.
An editorial presentation will be provided showing the proposed changes.
Note that comment #175 address missing control bit to enable precoding on the PMA receive output and transmit input.

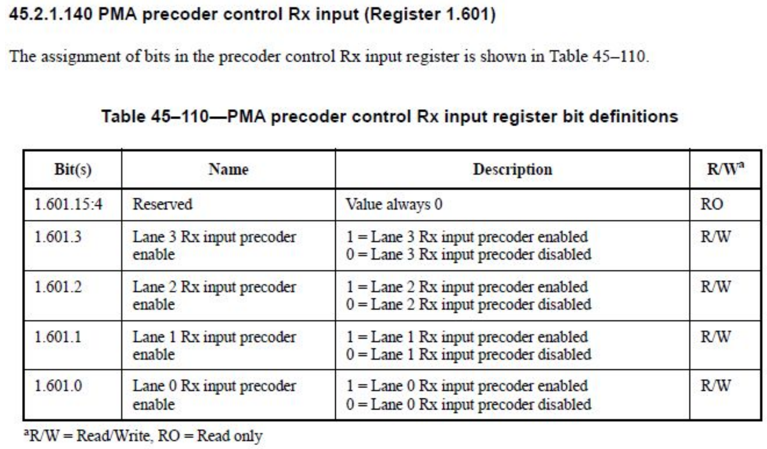
The comment addresses the fact that test patterns are defined without mention of precoding.

Patterns for transmitter testing should be defined without precoding.

However, a receiver may require precoding for meeting its requirements, and therefore precoding should be allowed for the PRBS31Q pattern.

This should be updated in both clause 120 and clause 135, but Clause 135 is out of scope for 802.3df since it deals only with 100GbE and 50GbE.

CC: Precoding (part 3)



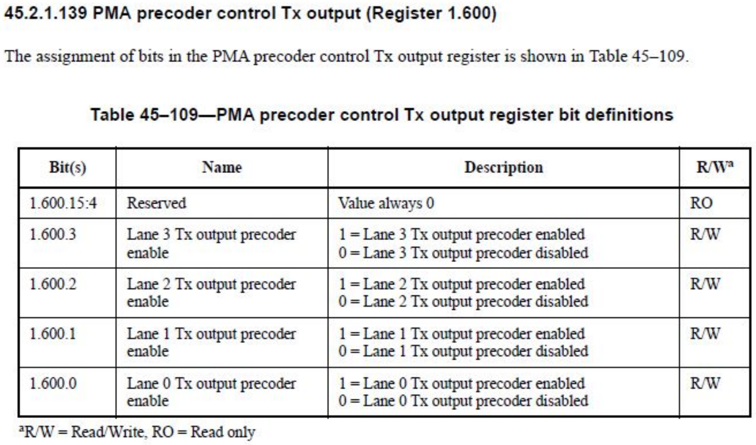
45.2.1.140 PMA precoder control Rx input (Register 1.601)

The assignment of bits in the precoder control Rx input register is shown in Table 45–110.

Table 45–110—PMA precoder control Rx input register bit definitions

Bit(s)	Name	Description	R/W[a]
1.601.15:4	Reserved	Value always 0	RO
1.601.3	Lane 3 Rx input precoder enable	1 = Lane 3 Rx input precoder enabled 0 = Lane 3 Rx input precoder disabled	R/W
1.601.2	Lane 2 Rx input precoder enable	1 = Lane 2 Rx input precoder enabled 0 = Lane 2 Rx input precoder disabled	R/W
1.601.1	Lane 1 Rx input precoder enable	1 = Lane 1 Rx input precoder enabled 0 = Lane 1 Rx input precoder disabled	R/W
1.601.0	Lane 0 Rx input precoder enable	1 = Lane 0 Rx input precoder enabled 0 = Lane 0 Rx input precoder disabled	R/W

[a] R/W = Read/Write, RO = Read only



45.2.1.139 PMA precoder control Tx output (Register 1.600)

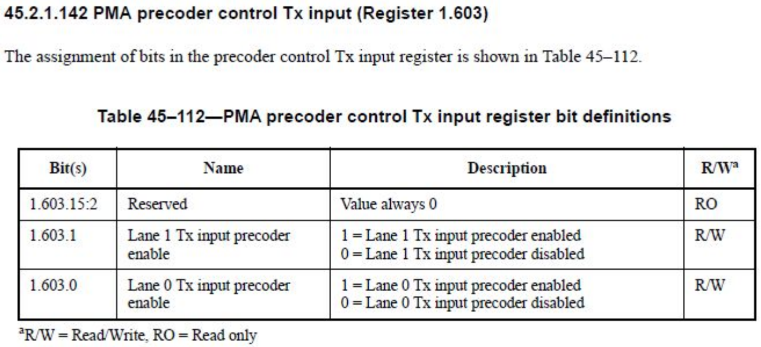
The assignment of bits in the PMA precoder control Tx output register is shown in Table 45–109.

Table 45–109—PMA precoder control Tx output register bit definitions

Bit(s)	Name	Description	R/W[a]
1.600.15:4	Reserved	Value always 0	RO
1.600.3	Lane 3 Tx output precoder enable	1 = Lane 3 Tx output precoder enabled 0 = Lane 3 Tx output precoder disabled	R/W
1.600.2	Lane 2 Tx output precoder enable	1 = Lane 2 Tx output precoder enabled 0 = Lane 2 Tx output precoder disabled	R/W
1.600.1	Lane 1 Tx output precoder enable	1 = Lane 1 Tx output precoder enabled 0 = Lane 1 Tx output precoder disabled	R/W
1.600.0	Lane 0 Tx output precoder enable	1 = Lane 0 Tx output precoder enabled 0 = Lane 0 Tx output precoder disabled	R/W

[a] R/W = Read/Write, RO = Read only

Precoder control bits for RX input and TX output need to be expanded from 4 to 8 lanes.



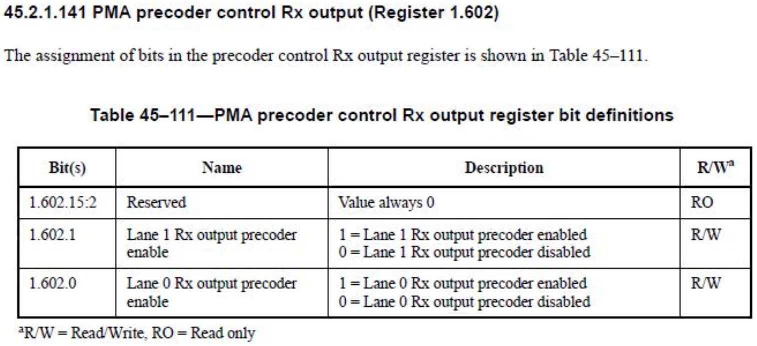
45.2.1.142 PMA precoder control Tx input (Register 1.603)

The assignment of bits in the precoder control Tx input register is shown in Table 45–112.

Table 45–112—PMA precoder control Tx input register bit definitions

Bit(s)	Name	Description	R/W[a]
1.603.15:2	Reserved	Value always 0	RO
1.603.1	Lane 1 Tx input precoder enable	1 = Lane 1 Tx input precoder enabled 0 = Lane 1 Tx input precoder disabled	R/W
1.603.0	Lane 0 Tx input precoder enable	1 = Lane 0 Tx input precoder enabled 0 = Lane 0 Tx input precoder disabled	R/W

[a] R/W = Read/Write, RO = Read only



45.2.1.141 PMA precoder control Rx output (Register 1.602)

The assignment of bits in the precoder control Rx output register is shown in Table 45–111.

Table 45–111—PMA precoder control Rx output register bit definitions

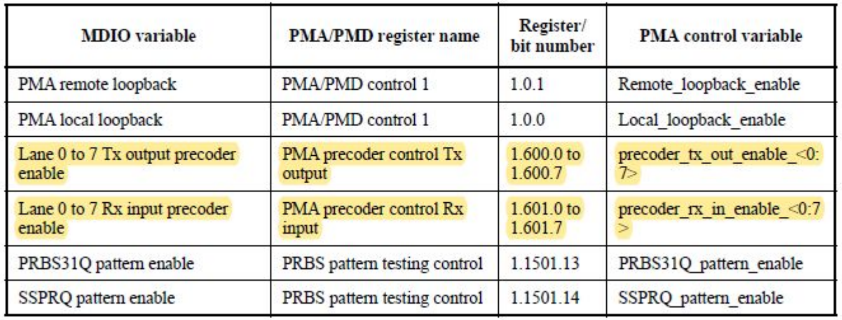
Bit(s)	Name	Description	R/W[a]
1.602.15:2	Reserved	Value always 0	RO
1.602.1	Lane 1 Rx output precoder enable	1 = Lane 1 Rx output precoder enabled 0 = Lane 1 Rx output precoder disabled	R/W
1.602.0	Lane 0 Rx output precoder enable	1 = Lane 0 Rx output precoder enabled 0 = Lane 0 Rx output precoder disabled	R/W

[a] R/W = Read/Write, RO = Read only

Precoder control bits for TX input and RX output need to be expanded from 2 to 8 lanes.

CC: Precoding (part 4)

Table 173–2—MDIO/PMA control variable mapping



MDIO variable	PMA/PMD register name	Register/ bit number	PMA control variable
PMA remote loopback	PMA/PMD control 1	1.0.1	Remote_loopback_enable
PMA local loopback	PMA/PMD control 1	1.0.0	Local_loopback_enable
Lane 0 to 7 Tx output precoder enable	PMA precoder control Tx output	1.600.0 to 1.600.7	precoder_tx_out_enable_<0:7>
Lane 0 to 7 Rx input precoder enable	PMA precoder control Rx input	1.601.0 to 1.601.7	precoder_rx_in_enable_<0:7>
PRBS31Q pattern enable	PRBS pattern testing control	1.1501.13	PRBS31Q_pattern_enable
SSPRQ pattern enable	PRBS pattern testing control	1.1501.14	SSPRQ_pattern_enable

Precoder control bits are missing for TX inputs and RX outputs.

CC: Precoding (part 5)
Comments #175 and #171 - Proposed change for 173.4.7.2

Replace 173.4.7.2 with the following (based on 135.5.7.2 and 120.5.7.2)…

173.4.7.2 Precoding for PAM4 encoded lanes

The precoding specifications in this subclause apply to the input and output lanes of a PMA that are connected to the service interface of an 800GBASE-CR8 or 800GBASE-KR8 PMD, or are part of an 800GAUI-8 C2C link.

The PMA shall provide 1/(1+D) mod 4 precoding capability on each transmit lane and may optionally provide 1/(1+D) mod 4 decoding capability on each receive lane. Precoding is implemented as specified in 135.5.7.2.

The precoder is enabled independently on the Tx output , Rx input, Tx input, and Rx output on each lane. Precoding is enabled and disabled using variables precoder_tx_out_enable_*i*, precoder_rx_in_enable_*i*, precoder_rx_out_enable_*i*, and precoder_tx_in_enable_*i* (where *i* is in the range 0 to 7). If a Clause 45 MDIO is implemented, these variables are accessible through registers as shown in Table 173-2.

If the PMA is connected to the service interface of an 800GBASE-CR8 or 800GBASE-KR8 PMD and training is enabled by the management variable mr_training_enable (see 136.7), then precoder_tx_out_enable_i and precoder_rx_in_enable_i shall be set as determined by the PMD control function in the LINK_READY state on lane *i* (see 136.8.11.7.5 and Figure 136–7). The method by which the PMD control function affects these variables is implementation dependent.

If the PMA is connected to the service interface of an 800GBASE-CR8 or 800GBASE-KR8 PMD and training is disabled by the management variable mr_training_enable, or if the PMA is part of an 800GAUI-8 link, then precoder_tx_out_enable_*i*, precoder_rx_in_enable_*i*, precoder_tx_in_enable_*i*, and precoder_rx_out_enable_*i* are set as required by the implementation. The method described in 135F.3.2.1 may be used for 800GAUI-8 C2C.

CC: Precoding (part 6)
Comments #175 and #171 - Proposed change for 173.4.7.2

In 173.5, Table 173-2 add the following control variables.

precoder_rx_out_enable_0:7, precoder_tx_in_enable_0:7

In Clause 45, add control bits for the following:

precoder_tx_out_enable_4:7, precoder_rx_in_enable_4:7, precoder_rx_out_enable_2:7, precoder_tx_in_enable_2:7

CC: Precoding (part 6) Comments #175 and #171 – Precoding for test patterns

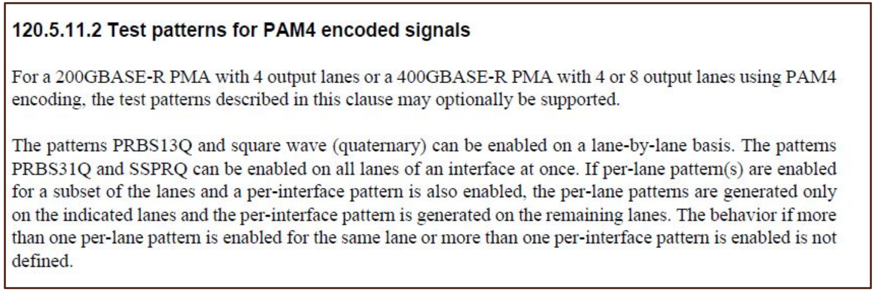
The following changes address ambiguity in the test pattern definitions regarding precoding.

Add the the following text to 120.5.11.2:

All test patterns specified in this subclause are defined without precoding.

Add the following text to 120.5.11.2.2:

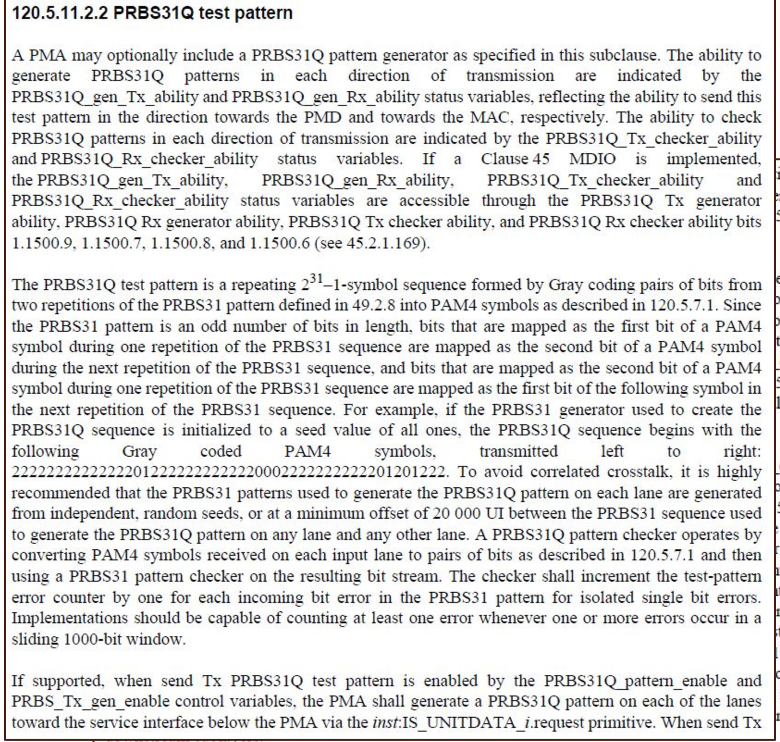
Precoding may be applied to the PRBS31Q pattern by enabling precoding in the PMA output or input as required.



120.5.11.2 Test patterns for PAM4 encoded signals

For a 200GBASE-R PMA with 4 output lanes or a 400GBASE-R PMA with 4 or 8 output lanes using PAM4 encoding, the test patterns described in this clause may optionally be supported.

The patterns PRBS13Q and square wave (quaternary) can be enabled on a lane-by-lane basis. The patterns PRBS31Q and SSPRQ can be enabled on all lanes of an interface at once. If per-lane pattern(s) are enabled for a subset of the lanes and a per-interface pattern is also enabled, the per-lane patterns are generated only on the indicated lanes and the per-interface pattern is generated on the remaining lanes. The behavior if more than one per-lane pattern is enabled for the same lane or more than one per-interface pattern is enabled is not defined.



120.5.11.2.2 PRBS31Q test pattern

A PMA may optionally include a PRBS31Q pattern generator as specified in this subclause. The ability to generate PRBS31Q patterns in each direction of transmission are indicated by the PRBS31Q_gen_Tx_ability and PRBS31Q_gen_Rx_ability status variables, reflecting the ability to send this test pattern in the direction towards the PMD and towards the MAC, respectively. The ability to check PRBS31Q patterns in each direction of transmission are indicated by the PRBS31Q_Tx_checker_ability and PRBS31Q_Rx_checker_ability status variables. If a Clause 45 MDIO is implemented, the PRBS31Q_gen_Tx_ability, PRBS31Q_gen_Rx_ability, PRBS31Q_Tx_checker_ability and PRBS31Q_Rx_checker_ability status variables are accessible through the PRBS31Q Tx generator ability, PRBS31Q Rx generator ability, PRBS31Q Tx checker ability, and PRBS31Q Rx checker ability bits 1.1500.9, 1.1500.7, 1.1500.8, and 1.1500.6 (see 45.2.1.169).

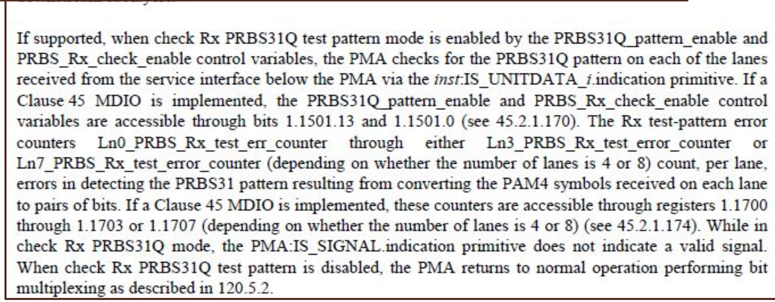
The PRBS31Q test pattern is a repeating 2^{31}–1-symbol sequence formed by Gray coding pairs of bits from two repetitions of the PRBS31 pattern defined in 49.2.8 into PAM4 symbols as described in 120.5.7.1. Since the PRBS31 pattern is an odd number of bits in length, bits that are mapped as the first bit of a PAM4 symbol during one repetition of the PRBS31 sequence are mapped as the second bit of a PAM4 symbol during the next repetition of the PRBS31 sequence, and bits that are mapped as the second bit of a PAM4 symbol during one repetition of the PRBS31 sequence are mapped as the first bit of the following symbol in the next repetition of the PRBS31 sequence. For example, if the PRBS31 generator used to create the PRBS31Q sequence is initialized to a seed value of all ones, the PRBS31Q sequence begins with the following Gray coded PAM4 symbols, transmitted left to right: 22222222222222012222222222220002222222222201201222. To avoid correlated crosstalk, it is highly recommended that the PRBS31 patterns used to generate the PRBS31Q pattern on each lane are generated from independent, random seeds, or at a minimum offset of 20 000 UI between the PRBS31 sequence used to generate the PRBS31Q pattern on any lane and any other lane. A PRBS31Q pattern checker operates by converting PAM4 symbols received on each input lane to pairs of bits as described in 120.5.7.1 and then using a PRBS31 pattern checker on the resulting bit stream. The checker shall increment the test-pattern error counter by one for each incoming bit error in the PRBS31 pattern for isolated single bit errors. Implementations should be capable of counting at least one error whenever one or more errors occur in a sliding 1000-bit window.

If supported, when send Tx PRBS31Q test pattern is enabled by the PRBS31Q_pattern_enable and PRBS_Tx_gen_enable control variables, the PMA shall generate a PRBS31Q pattern on each of the lanes toward the service interface below the PMA via the *inst*:IS_UNITDATA_*i*.request primitive. When send Tx

iplexing as
nable and
501.3 (see

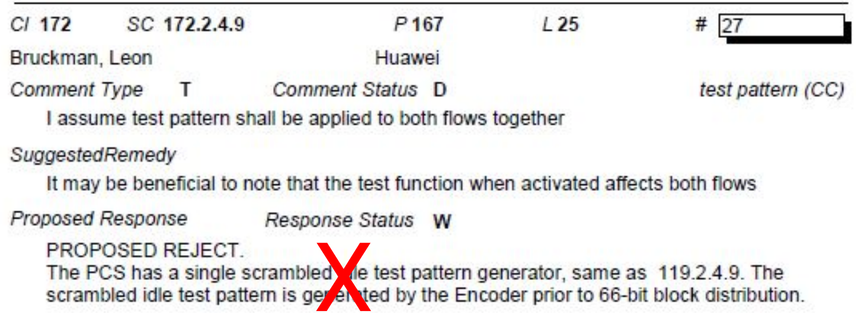
enable and
of the lanes
generates
tatus at the
_enable and
501.1 (see
l operation

_enable and
of the lanes
5 MDIO is
accessible
counters
ter or
t, per lane,
n each lane
ters 1.1600
73). When
or counting
in 120.5.2.
ningful for



If supported, when check Rx PRBS31Q test pattern mode is enabled by the PRBS31Q_pattern_enable and PRBS_Rx_check_enable control variables, the PMA checks for the PRBS31Q pattern on each of the lanes received from the service interface below the PMA via the *inst*:IS_UNITDATA_*i*.indication primitive. If a Clause 45 MDIO is implemented, the PRBS31Q_pattern_enable and PRBS_Rx_check_enable control variables are accessible through bits 1.1501.13 and 1.1501.0 (see 45.2.1.170). The Rx test-pattern error counters Ln0_PRBS_Rx_test_err_counter through either Ln3_PRBS_Rx_test_error_counter or Ln7_PRBS_Rx_test_error_counter (depending on whether the number of lanes is 4 or 8) count, per lane, errors in detecting the PRBS31 pattern resulting from converting the PAM4 symbols received on each lane to pairs of bits. If a Clause 45 MDIO is implemented, these counters are accessible through registers 1.1700 through 1.1703 or 1.1707 (depending on whether the number of lanes is 4 or 8) (see 45.2.1.174). While in check Rx PRBS31Q mode, the PMA:IS_SIGNAL.indication primitive does not indicate a valid signal. When check Rx PRBS31Q test pattern is disabled, the PMA returns to normal operation performing bit multiplexing as described in 120.5.2.

Test Patterns (part 1)
Comments 27, 186



CI 172 *SC* 172.2.4.9 *P* 167 *L* 25 # 27

Bruckman, Leon Huawei

Comment Type T *Comment Status* D *test pattern (CC)*

I assume test pattern shall be applied to both flows together

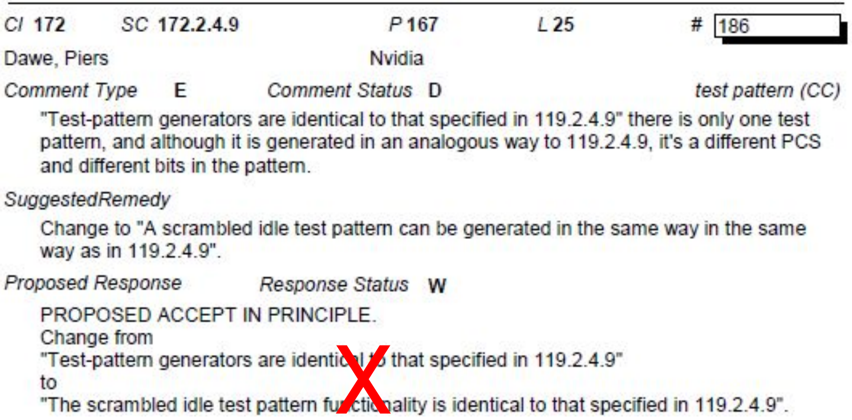
SuggestedRemedy

It may be beneficial to note that the test function when activated affects both flows

Proposed Response *Response Status* W

PROPOSED REJECT.
The PCS has a single scrambled idle test pattern generator, same as 119.2.4.9. The scrambled idle test pattern is generated by the Encoder prior to 66-bit block distribution.

This response needs to be updated.



CI 172 *SC* 172.2.4.9 *P* 167 *L* 25 # 186

Dawe, Piers Nvidia

Comment Type E *Comment Status* D *test pattern (CC)*

"Test-pattern generators are identical to that specified in 119.2.4.9" there is only one test pattern, and although it is generated in an analogous way to 119.2.4.9, it's a different PCS and different bits in the pattern.

SuggestedRemedy

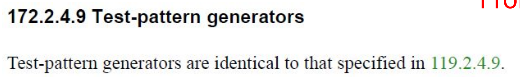
Change to "A scrambled idle test pattern can be generated in the same way in the same way as in 119.2.4.9".

Proposed Response *Response Status* W

PROPOSED ACCEPT IN PRINCIPLE.
Change from
"Test-pattern generators are identical to that specified in 119.2.4.9"
to
"The scrambled idle test pattern functionality is identical to that specified in 119.2.4.9".

This response needs to be updated.

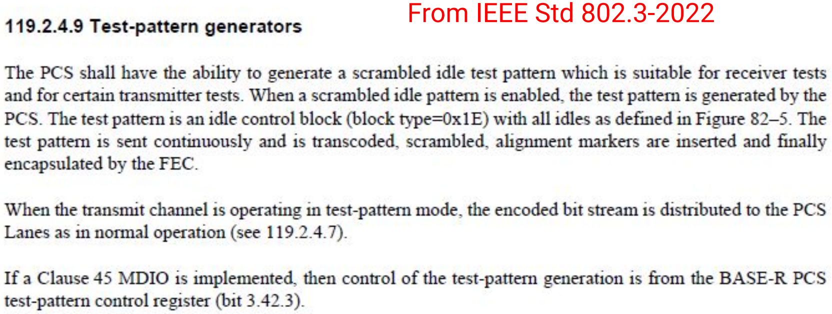
From P802.3df D1.0



172.2.4.9 Test-pattern generators

Test-pattern generators are identical to that specified in 119.2.4.9.

From IEEE Std 802.3-2022



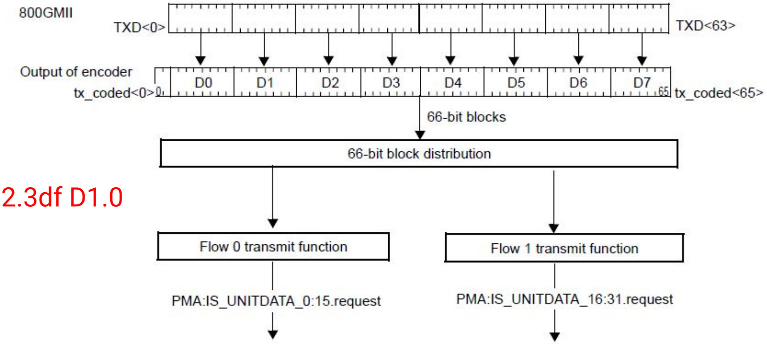
119.2.4.9 Test-pattern generators

The PCS shall have the ability to generate a scrambled idle test pattern which is suitable for receiver tests and for certain transmitter tests. When a scrambled idle pattern is enabled, the test pattern is generated by the PCS. The test pattern is an idle control block (block type=0x1E) with all idles as defined in Figure 82–5. The test pattern is sent continuously and is transcoded, scrambled, alignment markers are inserted and finally encapsulated by the FEC.

When the transmit channel is operating in test-pattern mode, the encoded bit stream is distributed to the PCS Lanes as in normal operation (see 119.2.4.7).

If a Clause 45 MDIO is implemented, then control of the test-pattern generation is from the BASE-R PCS test-pattern control register (bit 3.42.3).

For scrambled idle, an idle block is continuous inserted here.



From P802.3df D1.0

Figure 172–3—800GBASE-R PCS transmit bit ordering and distribution

The scrambled idle is distributed over all 32 PCS lanes.

<u>This is not accurately/clearly described in 119.2.4.9 for 800GbE.</u>

Test Patterns (part 1)
Comments 129, 143

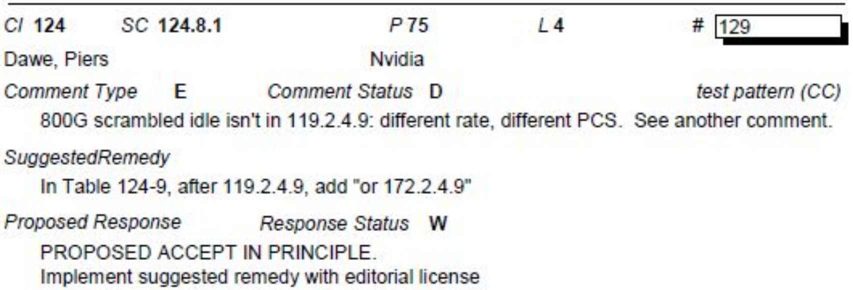
Replace the text in 172.2.4.9 as follows, making it more relevant to 800GbE…

172.2.4.9 Test-pattern generator

The PCS shall have the ability to generate a scrambled idle test pattern which is suitable for receiver tests and for certain transmitter tests. When a scrambled idle pattern is enabled, the test pattern is generated by the PCS. The scrambled idle test pattern is the output of the PCS when the input to the PCS at the 800GMII is a control block with all idle characters.

If a Clause 45 MDIO is implemented, then control of the test-pattern generation is from the BASE-R PCS test-pattern control register (bit 3.42.3).

Test Patterns (part 2)
Comments 129, 143



Cl **124** *SC* **124.8.1** *P* **75** *L* **4** # 129

Dawe, Piers Nvidia

Comment Type **E** *Comment Status* **D** *test pattern (CC)*

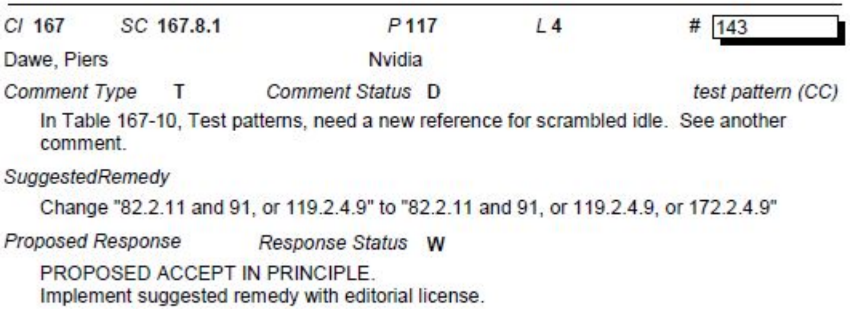
800G scrambled idle isn't in 119.2.4.9: different rate, different PCS. See another comment.

SuggestedRemedy

In Table 124-9, after 119.2.4.9, add "or 172.2.4.9"

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Implement suggested remedy with editorial license



Cl **167** *SC* **167.8.1** *P* **117** *L* **4** # 143

Dawe, Piers Nvidia

Comment Type **T** *Comment Status* **D** *test pattern (CC)*

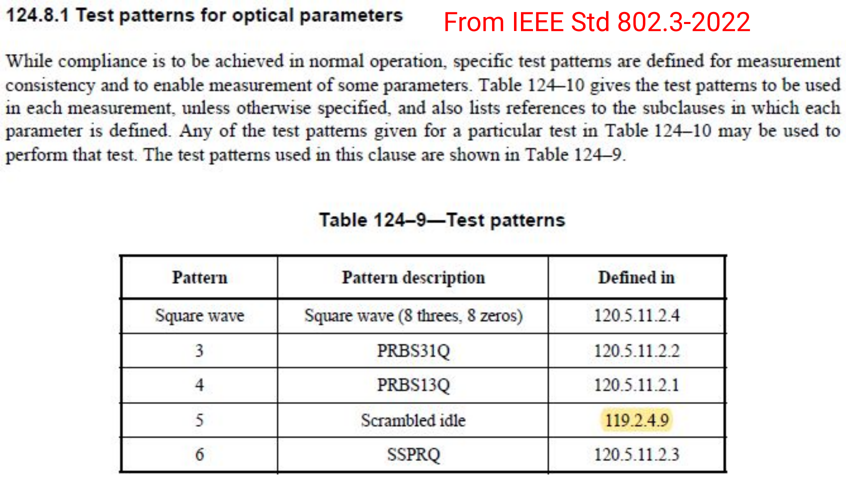
In Table 167-10, Test patterns, need a new reference for scrambled idle. See another comment.

SuggestedRemedy

Change "82.2.11 and 91, or 119.2.4.9" to "82.2.11 and 91, or 119.2.4.9, or 172.2.4.9"

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Implement suggested remedy with editorial license.



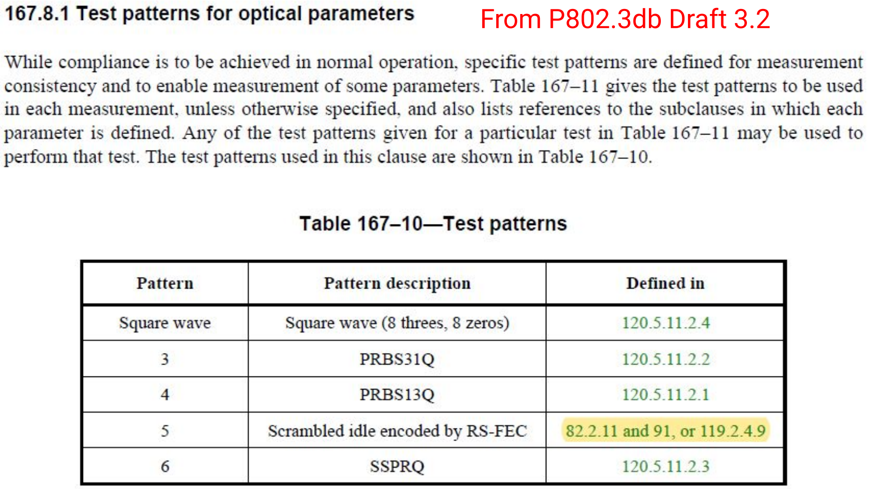
124.8.1 Test patterns for optical parameters

From IEEE Std 802.3-2022

While compliance is to be achieved in normal operation, specific test patterns are defined for measurement consistency and to enable measurement of some parameters. Table 124–10 gives the test patterns to be used in each measurement, unless otherwise specified, and also lists references to the subclauses in which each parameter is defined. Any of the test patterns given for a particular test in Table 124–10 may be used to perform that test. The test patterns used in this clause are shown in Table 124–9.

Table 124–9—Test patterns

Pattern	Pattern description	Defined in
Square wave	Square wave (8 threes, 8 zeros)	120.5.11.2.4
3	PRBS31Q	120.5.11.2.2
4	PRBS13Q	120.5.11.2.1
5	Scrambled idle	119.2.4.9
6	SSPRQ	120.5.11.2.3



167.8.1 Test patterns for optical parameters

From P802.3db Draft 3.2

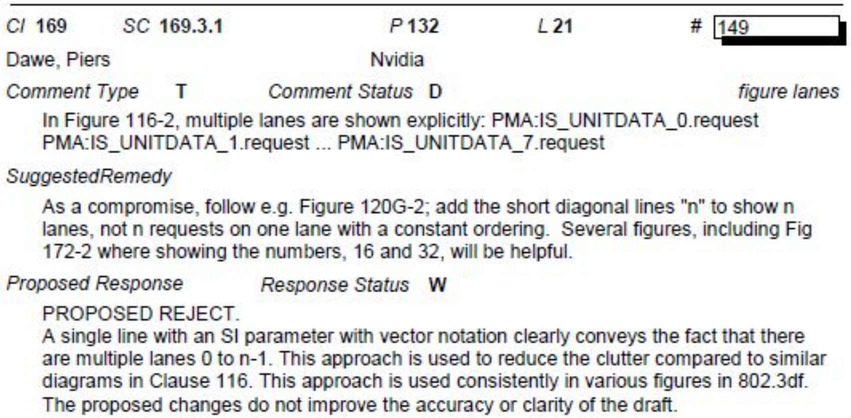
While compliance is to be achieved in normal operation, specific test patterns are defined for measurement consistency and to enable measurement of some parameters. Table 167–11 gives the test patterns to be used in each measurement, unless otherwise specified, and also lists references to the subclauses in which each parameter is defined. Any of the test patterns given for a particular test in Table 167–11 may be used to perform that test. The test patterns used in this clause are shown in Table 167–10.

Table 167–10—Test patterns

Pattern	Pattern description	Defined in
Square wave	Square wave (8 threes, 8 zeros)	120.5.11.2.4
3	PRBS31Q	120.5.11.2.2
4	PRBS13Q	120.5.11.2.1
5	Scrambled idle encoded by RS-FEC	82.2.11 and 91, or 119.2.4.9
6	SSPRQ	120.5.11.2.3

Clause 169 (Matt)

Clause 169: Figure Lanes
Comment 149



Cl 169 *SC* 169.3.1 *P* 132 *L* 21 # 149

Dawe, Piers Nvidia

Comment Type T *Comment Status* D *figure lanes*

In Figure 116-2, multiple lanes are shown explicitly: PMA:IS_UNITDATA_0.request PMA:IS_UNITDATA_1.request ... PMA:IS_UNITDATA_7.request

SuggestedRemedy

As a compromise, follow e.g. Figure 120G-2; add the short diagonal lines "n" to show n lanes, not n requests on one lane with a constant ordering. Several figures, including Fig 172-2 where showing the numbers, 16 and 32, will be helpful.

Proposed Response *Response Status* W

PROPOSED REJECT.

A single line with an SI parameter with vector notation clearly conveys the fact that there are multiple lanes 0 to n-1. This approach is used to reduce the clutter compared to similar diagrams in Clause 116. This approach is used consistently in various figures in 802.3df. The proposed changes do not improve the accuracy or clarity of the draft.

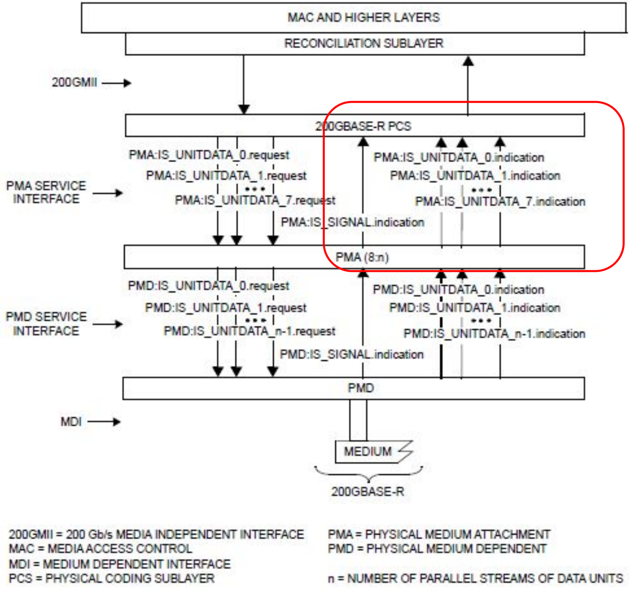

MAC AND HIGHER LAYERS
RECONCILIATION SUBLAYER
200GMII
200GBASE-R PCS
PMA:IS_UNITDATA_0.request
PMA:IS_UNITDATA_1.request
PMA:IS_UNITDATA_7.request
PMA:IS_UNITDATA_0.indication
PMA:IS_UNITDATA_1.indication
PMA:IS_UNITDATA_7.indication
PMA:IS_SIGNAL.indication
PMA SERVICE INTERFACE
PMA (8:n)
PMD:IS_UNITDATA_0.request
PMD:IS_UNITDATA_1.request
PMD:IS_UNITDATA_n-1.request
PMD:IS_UNITDATA_0.indication
PMD:IS_UNITDATA_1.indication
PMD:IS_UNITDATA_n-1.indication
PMD:IS_SIGNAL.indication
PMD SERVICE INTERFACE
PMD
MDI
MEDIUM
200GBASE-R
200GMII = 200 Gb/s MEDIA INDEPENDENT INTERFACE
MAC = MEDIA ACCESS CONTROL
MDI = MEDIUM DEPENDENT INTERFACE
PCS = PHYSICAL CODING SUBLAYER
PMA = PHYSICAL MEDIUM ATTACHMENT
PMD = PHYSICAL MEDIUM DEPENDENT
n = NUMBER OF PARALLEL STREAMS OF DATA UNITS

Figure 116–2—200GBASE-R inter-sublayer service interfaces

The interfaces as shown in Clause 116 were overly cluttered and difficult to read.

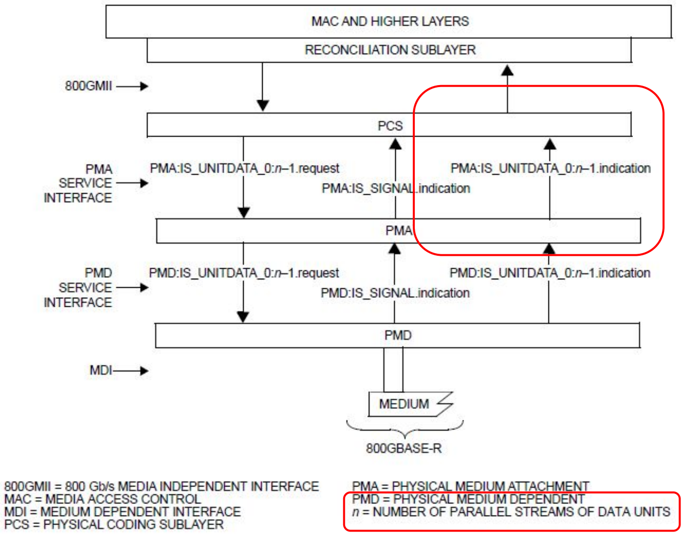
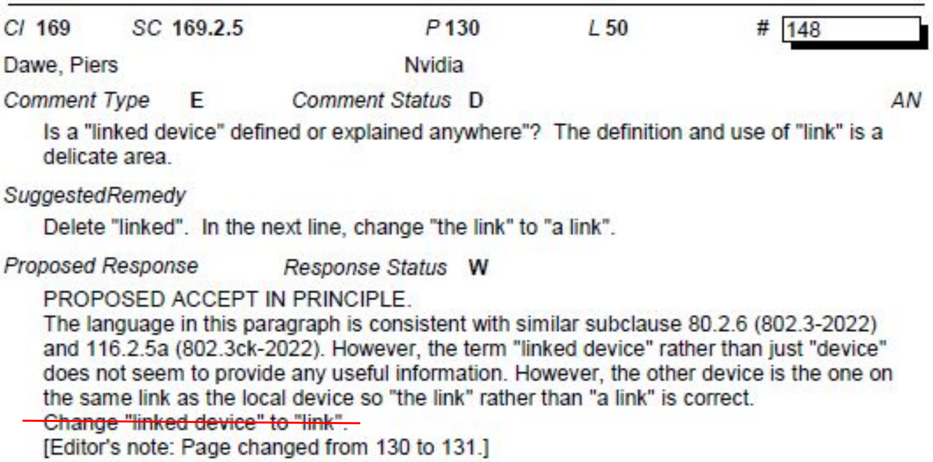

MAC AND HIGHER LAYERS
RECONCILIATION SUBLAYER
800GMII
PCS
PMA SERVICE INTERFACE
PMA:IS_UNITDATA_0:n–1.request
PMA:IS_SIGNAL.indication
PMA:IS_UNITDATA_0:n–1.indication
PMA
PMD SERVICE INTERFACE
PMD:IS_UNITDATA_0:n–1.request
PMD:IS_SIGNAL.indication
PMD:IS_UNITDATA_0:n–1.indication
PMD
MDI
MEDIUM
800GBASE-R
800GMII = 800 Gb/s MEDIA INDEPENDENT INTERFACE
MAC = MEDIA ACCESS CONTROL
MDI = MEDIUM DEPENDENT INTERFACE
PCS = PHYSICAL CODING SUBLAYER
PMA = PHYSICAL MEDIUM ATTACHMENT
PMD = PHYSICAL MEDIUM DEPENDENT
n = NUMBER OF PARALLEL STREAMS OF DATA UNITS

Figure 169–2—800GBASE-R inter-sublayer service interfaces not including 800GMII Extender

In new clauses 169, 173, etc., a parameter vector is used to convey a multitude of lanes, greatly reducing the clutter.
Number of lanes is evident both due to the vector format and the related note in the legend.

Clause 169: AN linked device
Comment 148



Cl 169 *SC* 169.2.5 *P* 130 *L* 50 # 148

Dawe, Piers Nvidia

Comment Type E *Comment Status* D AN

Is a "linked device" defined or explained anywhere"? The definition and use of "link" is a delicate area.

SuggestedRemedy

Delete "linked". In the next line, change "the link" to "a link".

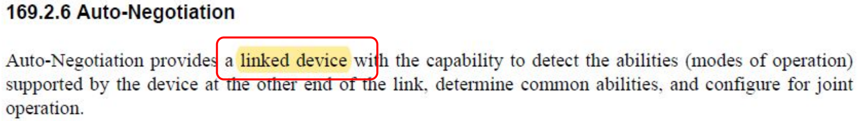
Proposed Response *Response Status* W

PROPOSED ACCEPT IN PRINCIPLE.
The language in this paragraph is consistent with similar subclause 80.2.6 (802.3-2022) and 116.2.5a (802.3ck-2022). However, the term "linked device" rather than just "device" does not seem to provide any useful information. However, the other device is the one on the same link as the local device so "the link" rather than "a link" is correct.
~~Change "linked device" to "link".~~
[Editor's note: Page changed from 130 to 131.]

From P802.3df D1.0…

The final remedy should be:
Change “linked device” to “device”.

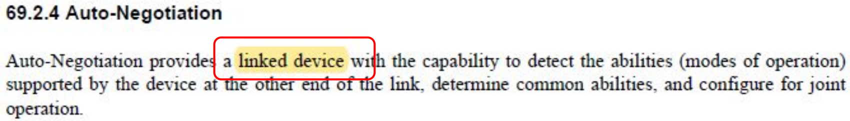
From P802.3df D1.0…



169.2.6 Auto-Negotiation

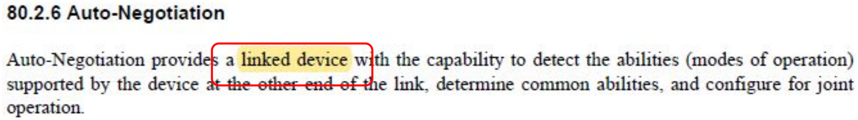
Auto-Negotiation provides a linked device with the capability to detect the abilities (modes of operation) supported by the device at the other end of the link, determine common abilities, and configure for joint operation.

From IEEE Std 802.3-2022



69.2.4 Auto-Negotiation

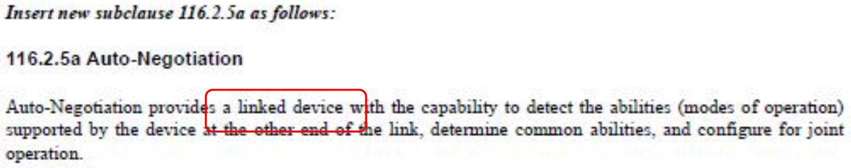
Auto-Negotiation provides a linked device with the capability to detect the abilities (modes of operation) supported by the device at the other end of the link, determine common abilities, and configure for joint operation.



80.2.6 Auto-Negotiation

Auto-Negotiation provides a linked device with the capability to detect the abilities (modes of operation) supported by the device ~~at the other end of the~~ link, determine common abilities, and configure for joint operation.

From P802.3ck D3.3…

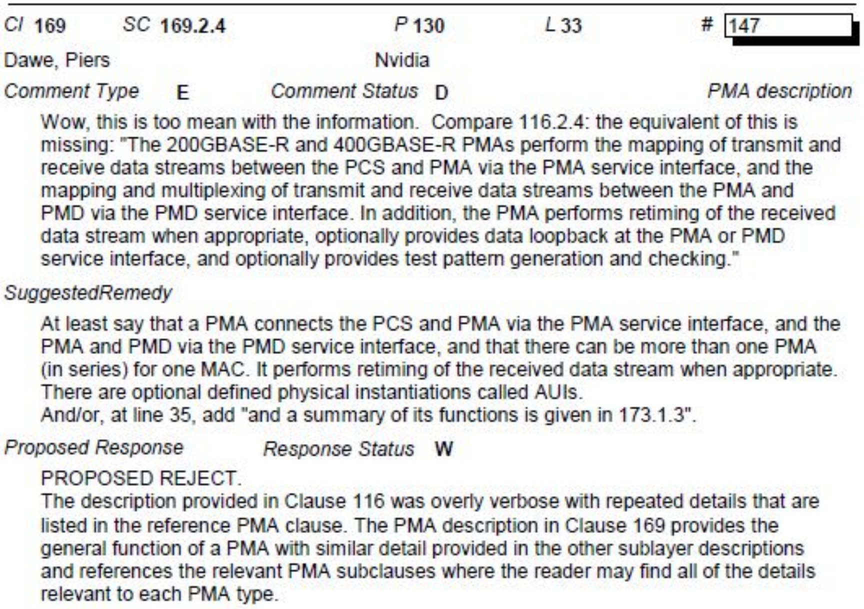


Insert new subclause 116.2.5a as follows:

116.2.5a Auto-Negotiation

Auto-Negotiation provides a linked device with the capability to detect the abilities (modes of operation) supported by the device ~~at the other end of the~~ link, determine common abilities, and configure for joint operation.

Clause 169: PMA description Comment 147



Cl 169 *SC* 169.2.4 *P* 130 *L* 33 # 147

Dawe, Piers Nvidia

Comment Type E *Comment Status* D *PMA description*

Wow, this is too mean with the information. Compare 116.2.4: the equivalent of this is missing: "The 200GBASE-R and 400GBASE-R PMAs perform the mapping of transmit and receive data streams between the PCS and PMA via the PMA service interface, and the mapping and multiplexing of transmit and receive data streams between the PMA and PMD via the PMD service interface. In addition, the PMA performs retiming of the received data stream when appropriate, optionally provides data loopback at the PMA or PMD service interface, and optionally provides test pattern generation and checking."

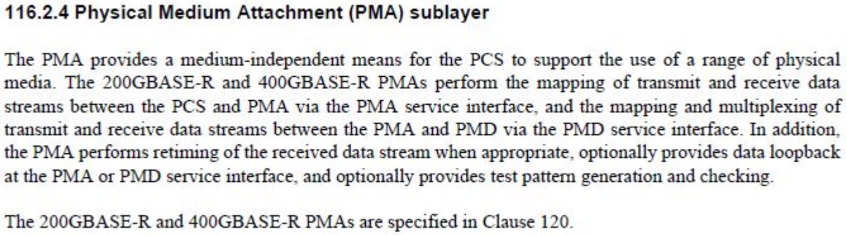
SuggestedRemedy

At least say that a PMA connects the PCS and PMA via the PMA service interface, and the PMA and PMD via the PMD service interface, and that there can be more than one PMA (in series) for one MAC. It performs retiming of the received data stream when appropriate. There are optional defined physical instantiations called AUIs.
And/or, at line 35, add "and a summary of its functions is given in 173.1.3".

Proposed Response *Response Status* W

PROPOSED REJECT.
The description provided in Clause 116 was overly verbose with repeated details that are listed in the reference PMA clause. The PMA description in Clause 169 provides the general function of a PMA with similar detail provided in the other sublayer descriptions and references the relevant PMA subclauses where the reader may find all of the details relevant to each PMA type.

From IEEE Std 802.3-2022, regarding 200G/400G PMAs



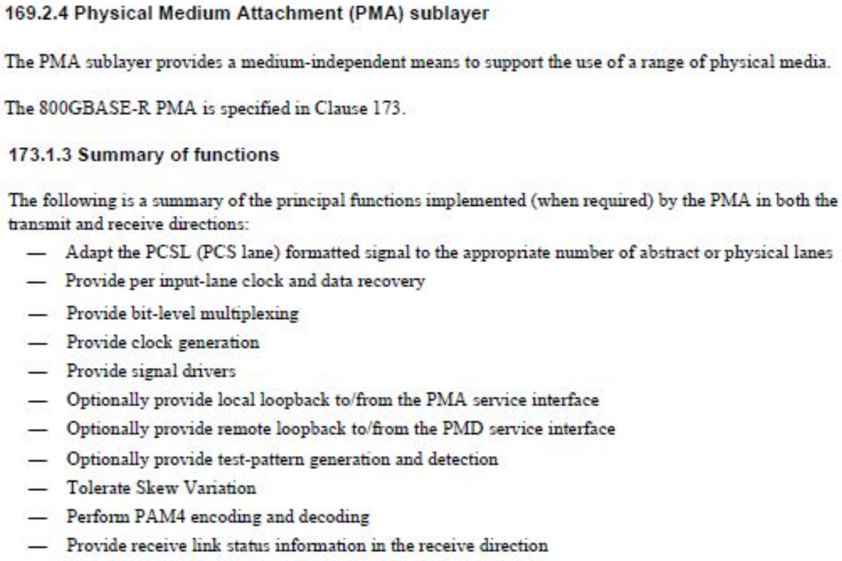
116.2.4 Physical Medium Attachment (PMA) sublayer

The PMA provides a medium-independent means for the PCS to support the use of a range of physical media. The 200GBASE-R and 400GBASE-R PMAs perform the mapping of transmit and receive data streams between the PCS and PMA via the PMA service interface, and the mapping and multiplexing of transmit and receive data streams between the PMA and PMD via the PMD service interface. In addition, the PMA performs retiming of the received data stream when appropriate, optionally provides data loopback at the PMA or PMD service interface, and optionally provides test pattern generation and checking.

The 200GBASE-R and 400GBASE-R PMAs are specified in Clause 120.

Note that 802.3cw defines a new 400GBASE-ZR PMA with different functionality than the 400GBASE-R PMA. Likely for 800 Gb/s 40 km SMF and maybe for 10 km SMF, we'll see a similarly unique PMA defined.

From P802.3df D1.0 regarding 800G PMA



169.2.4 Physical Medium Attachment (PMA) sublayer

The PMA sublayer provides a medium-independent means to support the use of a range of physical media.

The 800GBASE-R PMA is specified in Clause 173.

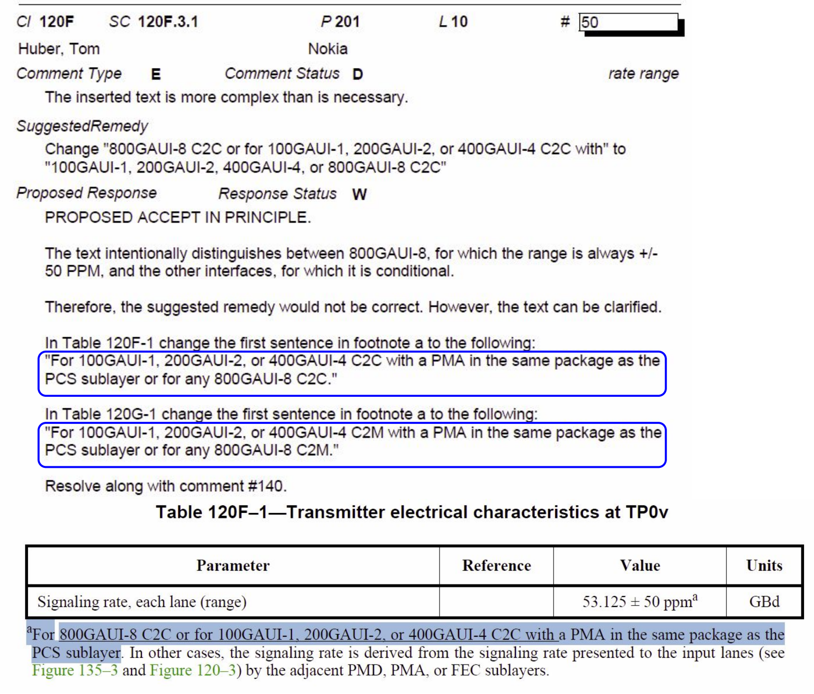
173.1.3 Summary of functions

The following is a summary of the principal functions implemented (when required) by the PMA in both the transmit and receive directions:

- Adapt the PCSL (PCS lane) formatted signal to the appropriate number of abstract or physical lanes
- Provide per input-lane clock and data recovery
- Provide bit-level multiplexing
- Provide clock generation
- Provide signal drivers
- Optionally provide local loopback to/from the PMA service interface
- Optionally provide remote loopback to/from the PMD service interface
- Optionally provide test-pattern generation and detection
- Tolerate Skew Variation
- Perform PAM4 encoding and decoding
- Provide receive link status information in the receive direction

Annex 120F+120G, Clause 162+163 (Adee)

Tx Signaling rate range
Comments 50, 140



Cl **120F** *SC* **120F.3.1** *P* **201** *L* **10** # 50

Huber, Tom Nokia

Comment Type **E** *Comment Status* **D** *rate range*

The inserted text is more complex than is necessary.

SuggestedRemedy

Change "800GAUI-8 C2C or for 100GAUI-1, 200GAUI-2, or 400GAUI-4 C2C with" to "100GAUI-1, 200GAUI-2, 400GAUI-4, or 800GAUI-8 C2C"

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.

The text intentionally distinguishes between 800GAUI-8, for which the range is always +/- 50 PPM, and the other interfaces, for which it is conditional.

Therefore, the suggested remedy would not be correct. However, the text can be clarified.

In Table 120F-1 change the first sentence in footnote a to the following:
"For 100GAUI-1, 200GAUI-2, or 400GAUI-4 C2C with a PMA in the same package as the PCS sublayer or for any 800GAUI-8 C2C."

In Table 120G-1 change the first sentence in footnote a to the following:
"For 100GAUI-1, 200GAUI-2, or 400GAUI-4 C2M with a PMA in the same package as the PCS sublayer or for any 800GAUI-8 C2M."

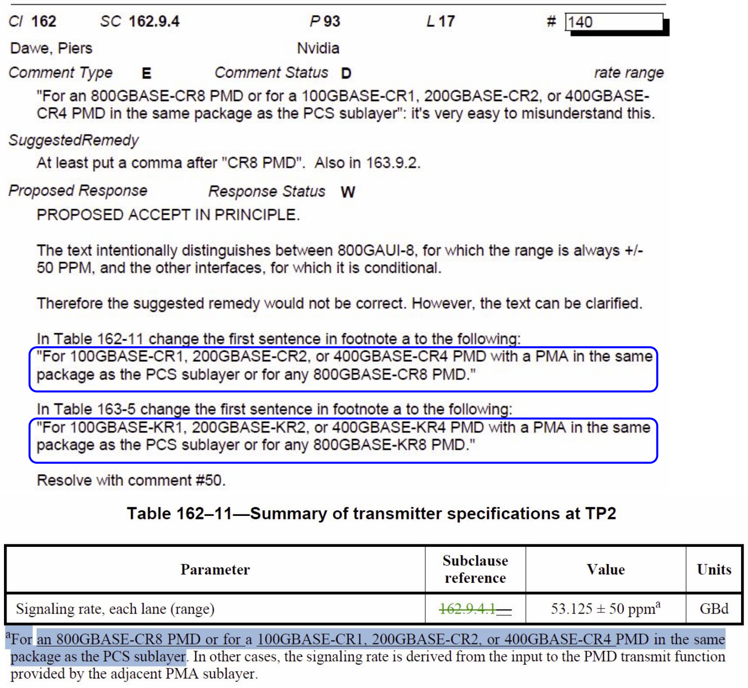
Resolve along with comment #140.

Table 120F–1—Transmitter electrical characteristics at TP0v

Parameter	Reference	Value	Units
Signaling rate, each lane (range)		53.125 ± 50 ppm[a]	GBd

[a]For 800GAUI-8 C2C or for 100GAUI-1, 200GAUI-2, or 400GAUI-4 C2C with a PMA in the same package as the PCS sublayer. In other cases, the signaling rate is derived from the signaling rate presented to the input lanes (see Figure 135–3 and Figure 120–3) by the adjacent PMD, PMA, or FEC sublayers.

Table 120G-1 footnote a has corresponding text for C2M



Cl **162** *SC* **162.9.4** *P* **93** *L* **17** # 140

Dawe, Piers Nvidia

Comment Type **E** *Comment Status* **D** *rate range*

"For an 800GBASE-CR8 PMD or for a 100GBASE-CR1, 200GBASE-CR2, or 400GBASE-CR4 PMD in the same package as the PCS sublayer": it's very easy to misunderstand this.

SuggestedRemedy

At least put a comma after "CR8 PMD". Also in 163.9.2.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.

The text intentionally distinguishes between 800GAUI-8, for which the range is always +/- 50 PPM, and the other interfaces, for which it is conditional.

Therefore the suggested remedy would not be correct. However, the text can be clarified.

In Table 162-11 change the first sentence in footnote a to the following:
"For 100GBASE-CR1, 200GBASE-CR2, or 400GBASE-CR4 PMD with a PMA in the same package as the PCS sublayer or for any 800GBASE-CR8 PMD."

In Table 163-5 change the first sentence in footnote a to the following:
"For 100GBASE-KR1, 200GBASE-KR2, or 400GBASE-KR4 PMD with a PMA in the same package as the PCS sublayer or for any 800GBASE-KR8 PMD."

Resolve with comment #50.

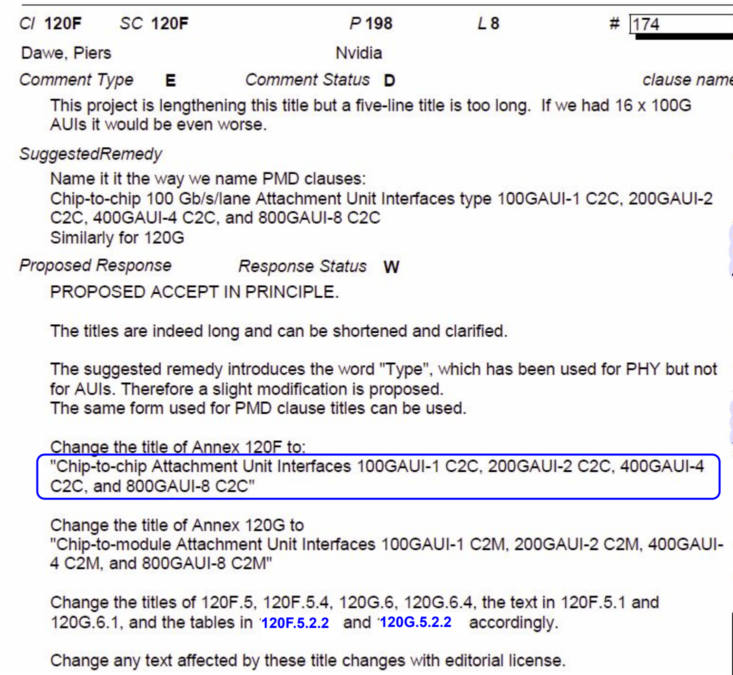
Table 162–11—Summary of transmitter specifications at TP2

Parameter	Subclause reference	Value	Units
Signaling rate, each lane (range)	~~162.9.4.1~~—	53.125 ± 50 ppm[a]	GBd

[a]For an 800GBASE-CR8 PMD or for a 100GBASE-CR1, 200GBASE-CR2, or 400GBASE-CR4 PMD in the same package as the PCS sublayer. In other cases, the signaling rate is derived from the input to the PMD transmit function provided by the adjacent PMA sublayer.

Table 163-5 footnote a has corresponding text for KR

Long annex titles - comment 174



Cl **120F** *SC* **120F** *P* **198** *L* **8** **#** 174

Dawe, Piers Nvidia

Comment Type **E** *Comment Status* **D** *clause name*

This project is lengthening this title but a five-line title is too long. If we had 16 x 100G AUIs it would be even worse.

SuggestedRemedy

Name it it the way we name PMD clauses:
Chip-to-chip 100 Gb/s/lane Attachment Unit Interfaces type 100GAUI-1 C2C, 200GAUI-2 C2C, 400GAUI-4 C2C, and 800GAUI-8 C2C
Similarly for 120G

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.

The titles are indeed long and can be shortened and clarified.

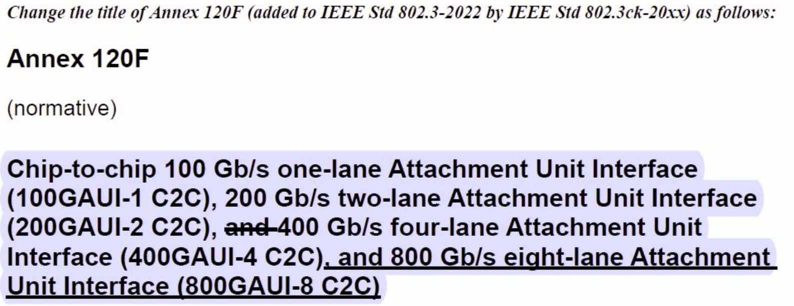
The suggested remedy introduces the word "Type", which has been used for PHY but not for AUIs. Therefore a slight modification is proposed.
The same form used for PMD clause titles can be used.

Change the title of Annex 120F to:
"Chip-to-chip Attachment Unit Interfaces 100GAUI-1 C2C, 200GAUI-2 C2C, 400GAUI-4 C2C, and 800GAUI-8 C2C"

Change the title of Annex 120G to
"Chip-to-module Attachment Unit Interfaces 100GAUI-1 C2M, 200GAUI-2 C2M, 400GAUI-4 C2M, and 800GAUI-8 C2M"

Change the titles of 120F.5, 120F.5.4, 120G.6, 120G.6.4, the text in 120F.5.1 and 120G.6.1, and the tables in 120F.5.2.2 and 120G.5.2.2 accordingly.

Change any text affected by these title changes with editorial license.

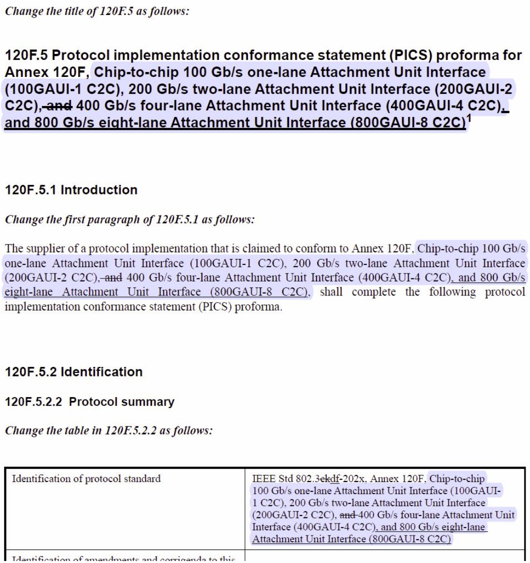


Change the title of Annex 120F (added to IEEE Std 802.3-2022 by IEEE Std 802.3ck-20xx) as follows:

Annex 120F

(normative)

Chip-to-chip 100 Gb/s one-lane Attachment Unit Interface (100GAUI-1 C2C), 200 Gb/s two-lane Attachment Unit Interface (200GAUI-2 C2C), ~~and~~ 400 Gb/s four-lane Attachment Unit Interface (400GAUI-4 C2C), and 800 Gb/s eight-lane Attachment Unit Interface (800GAUI-8 C2C)



Change the title of 120F.5 as follows:

120F.5 Protocol implementation conformance statement (PICS) proforma for Annex 120F, Chip-to-chip 100 Gb/s one-lane Attachment Unit Interface (100GAUI-1 C2C), 200 Gb/s two-lane Attachment Unit Interface (200GAUI-2 C2C), ~~and~~ 400 Gb/s four-lane Attachment Unit Interface (400GAUI-4 C2C), and 800 Gb/s eight-lane Attachment Unit Interface (800GAUI-8 C2C)[1]

120F.5.1 Introduction

Change the first paragraph of 120F.5.1 as follows:

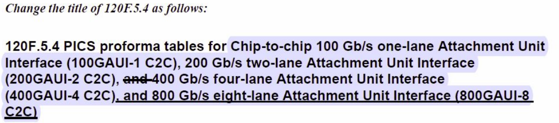
The supplier of a protocol implementation that is claimed to conform to Annex 120F, Chip-to-chip 100 Gb/s one-lane Attachment Unit Interface (100GAUI-1 C2C), 200 Gb/s two-lane Attachment Unit Interface (200GAUI-2 C2C), ~~and~~ 400 Gb/s four-lane Attachment Unit Interface (400GAUI-4 C2C), and 800 Gb/s eight-lane Attachment Unit Interface (800GAUI-8 C2C), shall complete the following protocol implementation conformance statement (PICS) proforma.

120F.5.2 Identification

120F.5.2.2 Protocol summary

Change the table in 120F.5.2.2 as follows:

Identification of protocol standard	IEEE Std 802.3~~ek~~df-202x, Annex 120F, Chip-to-chip 100 Gb/s one-lane Attachment Unit Interface (100GAUI-1 C2C), 200 Gb/s two-lane Attachment Unit Interface (200GAUI-2 C2C), ~~and~~ 400 Gb/s four-lane Attachment Unit Interface (400GAUI-4 C2C), and 800 Gb/s eight-lane Attachment Unit Interface (800GAUI-8 C2C)
Identification of amendments and corrigenda to this	

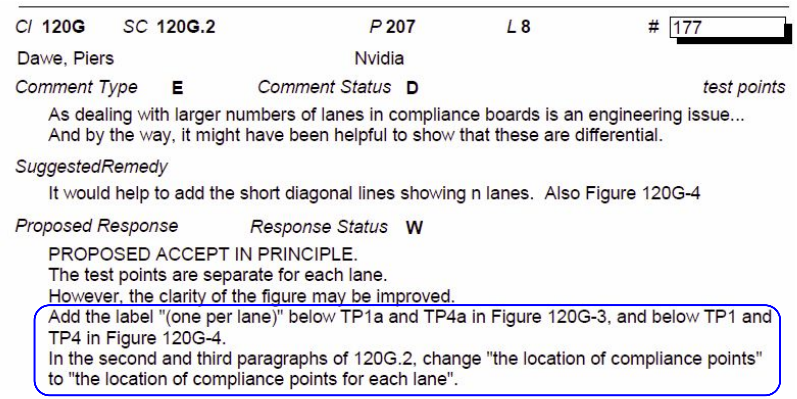


Change the title of 120F.5.4 as follows:

120F.5.4 PICS proforma tables for Chip-to-chip 100 Gb/s one-lane Attachment Unit Interface (100GAUI-1 C2C), 200 Gb/s two-lane Attachment Unit Interface (200GAUI-2 C2C), ~~and~~ 400 Gb/s four-lane Attachment Unit Interface (400GAUI-4 C2C), and 800 Gb/s eight-lane Attachment Unit Interface (800GAUI-8 C2C)

Corresponding changes in Annex 120G-1 for C2M

C2M Test points - comment 177



Cl **120G** *SC* **120G.2** *P* **207** *L* **8** **#** 177

Dawe, Piers Nvidia

Comment Type **E** *Comment Status* **D** *test points*

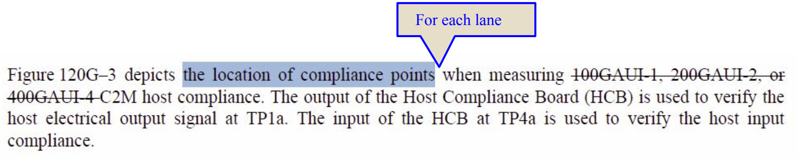
As dealing with larger numbers of lanes in compliance boards is an engineering issue...
And by the way, it might have been helpful to show that these are differential.

SuggestedRemedy

It would help to add the short diagonal lines showing n lanes. Also Figure 120G-4

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
The test points are separate for each lane.
However, the clarity of the figure may be improved.
Add the label "(one per lane)" below TP1a and TP4a in Figure 120G-3, and below TP1 and TP4 in Figure 120G-4.
In the second and third paragraphs of 120G.2, change "the location of compliance points" to "the location of compliance points for each lane".



For each lane

Figure 120G–3 depicts the location of compliance points when measuring ~~100GAUI-1, 200GAUI-2, or 400GAUI-4~~ C2M host compliance. The output of the Host Compliance Board (HCB) is used to verify the host electrical output signal at TP1a. The input of the HCB at TP4a is used to verify the host input compliance.

Original figure

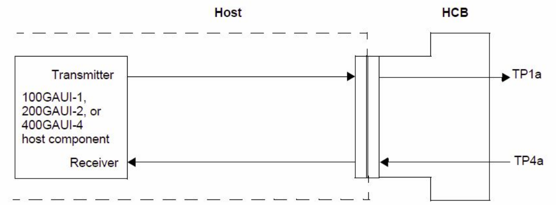


Figure 120G–3—Host compliance points

Replace Figure 120G–3 with the following:

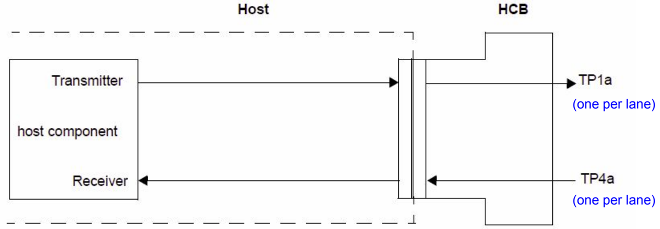
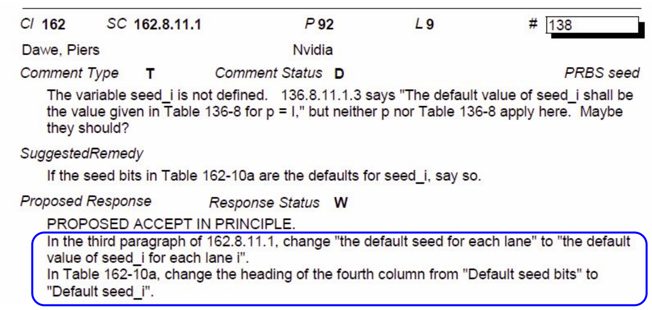


Figure 120G–3—Host compliance points

Corresponding changes in Figure 120G-4 for TP1, TP4

Training pattern PRBS seed - comments 137, 138



Cl **162** *SC* **162.8.11.1** *P* **92** *L* **9** # 138

Dawe, Piers Nvidia

Comment Type **T** *Comment Status* **D** *PRBS seed*

The variable seed_i is not defined. 136.8.11.1.3 says "The default value of seed_i shall be the value given in Table 136-8 for p = I," but neither p nor Table 136-8 apply here. Maybe they should?

SuggestedRemedy

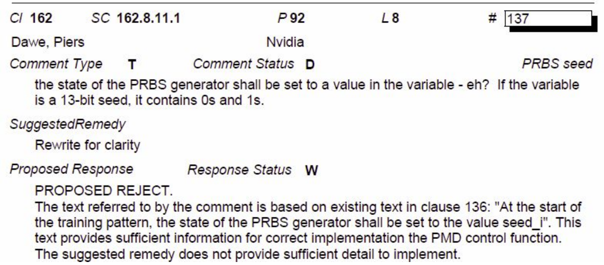
If the seed bits in Table 162-10a are the defaults for seed_i, say so.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.

In the third paragraph of 162.8.11.1, change "the default seed for each lane" to "the default value of seed_i for each lane i".

In Table 162-10a, change the heading of the fourth column from "Default seed bits" to "Default seed_i".



Cl **162** *SC* **162.8.11.1** *P* **92** *L* **8** # 137

Dawe, Piers Nvidia

Comment Type **T** *Comment Status* **D** *PRBS seed*

the state of the PRBS generator shall be set to a value in the variable - eh? If the variable is a 13-bit seed, it contains 0s and 1s.

SuggestedRemedy

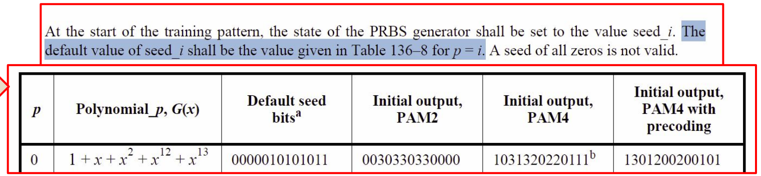
Rewrite for clarity

Proposed Response *Response Status* **W**

PROPOSED REJECT.

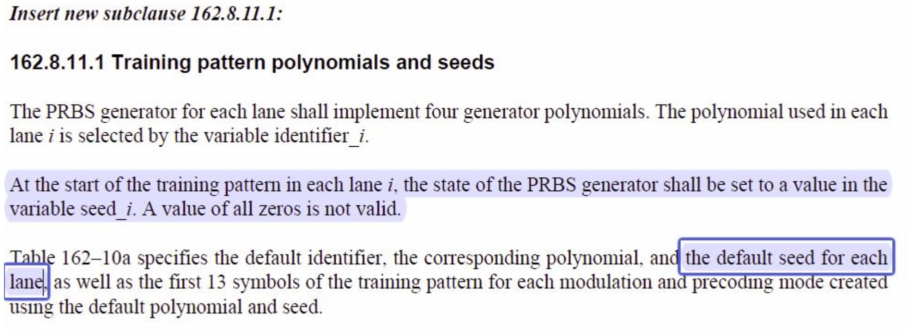
The text referred to by the comment is based on existing text in clause 136: "At the start of the training pattern, the state of the PRBS generator shall be set to the value seed_i". This text provides sufficient information for correct implementation the PMD control function. The suggested remedy does not provide sufficient detail to implement.

Content in clause 136 (reference)



At the start of the training pattern, the state of the PRBS generator shall be set to the value seed_*i*. The default value of seed_*i* shall be the value given in Table 136–8 for *p* = *i*. A seed of all zeros is not valid.

p	Polynomial_*p*, *G*(*x*)	Default seed bits[a]	Initial output, PAM2	Initial output, PAM4	Initial output, PAM4 with precoding
0	$1 + x + x^2 + x^{12} + x^{13}$	0000010101011	0030330330000	1031320220111[b]	1301200200101



Insert new subclause 162.8.11.1:

162.8.11.1 Training pattern polynomials and seeds

The PRBS generator for each lane shall implement four generator polynomials. The polynomial used in each lane *i* is selected by the variable identifier_*i*.

At the start of the training pattern in each lane *i*, the state of the PRBS generator shall be set to a value in the variable seed_*i*. A value of all zeros is not valid.

Table 162–10a specifies the default identifier, the corresponding polynomial, and the default seed for each lane, as well as the first 13 symbols of the training pattern for each modulation and precoding mode created using the default polynomial and seed.

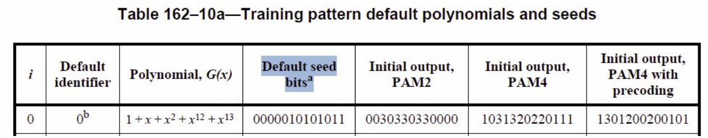
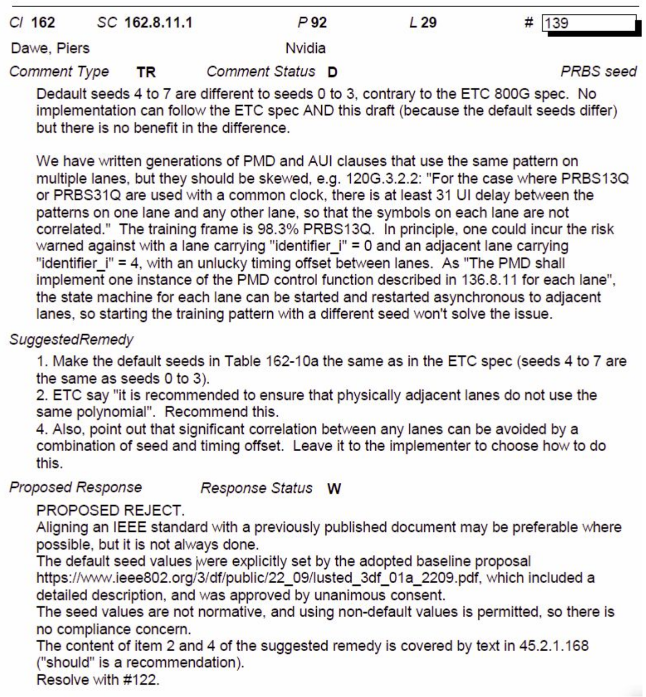


Table 162–10a—Training pattern default polynomials and seeds

i	Default identifier	Polynomial, *G*(*x*)	Default seed bits[a]	Initial output, PAM2	Initial output, PAM4	Initial output, PAM4 with precoding
0	0[b]	$1 + x + x^2 + x^{12} + x^{13}$	0000010101011	0030330330000	1031320220111	1301200200101

Training pattern PRBS seed - comment 139



Cl 162 *SC* 162.8.11.1 *P* 92 *L* 29 # 139

Dawe, Piers Nvidia

Comment Type **TR** *Comment Status* **D** *PRBS seed*

Dedault seeds 4 to 7 are different to seeds 0 to 3, contrary to the ETC 800G spec. No implementation can follow the ETC spec AND this draft (because the default seeds differ) but there is no benefit in the difference.

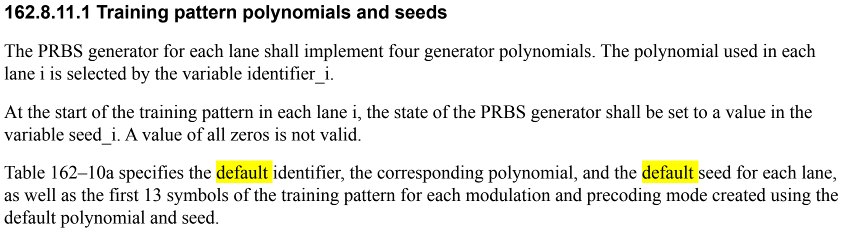
We have written generations of PMD and AUI clauses that use the same pattern on multiple lanes, but they should be skewed, e.g. 120G.3.2.2: "For the case where PRBS13Q or PRBS31Q are used with a common clock, there is at least 31 UI delay between the patterns on one lane and any other lane, so that the symbols on each lane are not correlated." The training frame is 98.3% PRBS13Q. In principle, one could incur the risk warned against with a lane carrying "identifier_i" = 0 and an adjacent lane carrying "identifier_i" = 4, with an unlucky timing offset between lanes. As "The PMD shall implement one instance of the PMD control function described in 136.8.11 for each lane", the state machine for each lane can be started and restarted asynchronous to adjacent lanes, so starting the training pattern with a different seed won't solve the issue.

SuggestedRemedy

1. Make the default seeds in Table 162-10a the same as in the ETC spec (seeds 4 to 7 are the same as seeds 0 to 3).
2. ETC say "it is recommended to ensure that physically adjacent lanes do not use the same polynomial". Recommend this.
4. Also, point out that significant correlation between any lanes can be avoided by a combination of seed and timing offset. Leave it to the implementer to choose how to do this.

Proposed Response *Response Status* **W**

PROPOSED REJECT.
Aligning an IEEE standard with a previously published document may be preferable where possible, but it is not always done.
The default seed values were explicitly set by the adopted baseline proposal https://www.ieee802.org/3/df/public/22_09/lusted_3df_01a_2209.pdf, which included a detailed description, and was approved by unanimous consent.
The seed values are not normative, and using non-default values is permitted, so there is no compliance concern.
The content of item 2 and 4 of the suggested remedy is covered by text in 45.2.1.168 ("should" is a recommendation).
Resolve with #122.



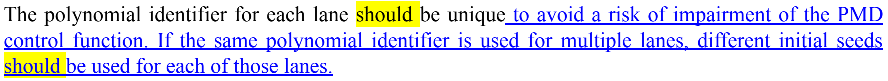
162.8.11.1 Training pattern polynomials and seeds

The PRBS generator for each lane shall implement four generator polynomials. The polynomial used in each lane i is selected by the variable identifier_i.

At the start of the training pattern in each lane i, the state of the PRBS generator shall be set to a value in the variable seed_i. A value of all zeros is not valid.

Table 162–10a specifies the default identifier, the corresponding polynomial, and the default seed for each lane, as well as the first 13 symbols of the training pattern for each modulation and precoding mode created using the default polynomial and seed.

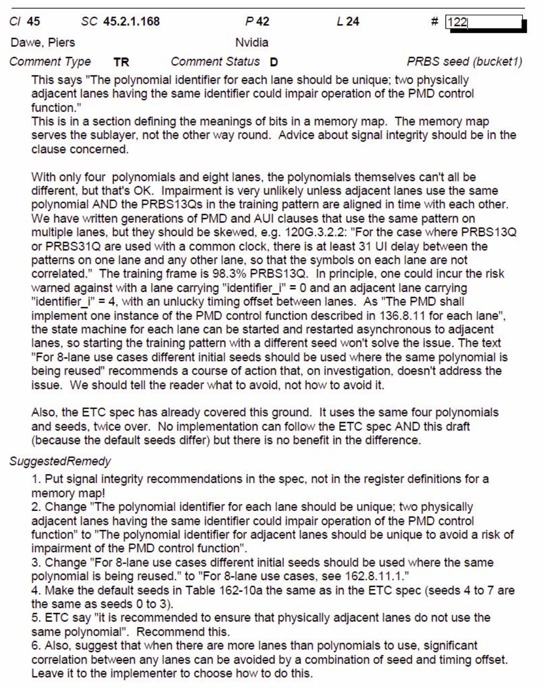
The corresponding text in 45.2.1.168 (per proposed response to bucket comment 122) is:



The polynomial identifier for each lane should be unique to avoid a risk of impairment of the PMD control function. If the same polynomial identifier is used for multiple lanes, different initial seeds should be used for each of those lanes.

For reference: adopted baseline proposal
https://www.ieee802.org/3/df/public/22_09/lusted_3df_01a_2209.pdf

Training pattern PRBS seed - comment 122 (bucket; for reference only)



Cl **45** *SC* **45.2.1.168** *P* **42** *L* **24** # 122

Dawe, Piers Nvidia

Comment Type **TR** *Comment Status* **D** *PRBS seed (bucket1)*

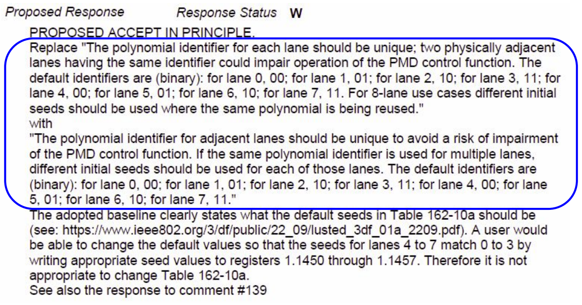
This says "The polynomial identifier for each lane should be unique; two physically adjacent lanes having the same identifier could impair operation of the PMD control function."
This is in a section defining the meanings of bits in a memory map. The memory map serves the sublayer, not the other way round. Advice about signal integrity should be in the clause concerned.

With only four polynomials and eight lanes, the polynomials themselves can't all be different, but that's OK. Impairment is very unlikely unless adjacent lanes use the same polynomial AND the PRBS13Qs in the training pattern are aligned in time with each other. We have written generations of PMD and AUI clauses that use the same pattern on multiple lanes, but they should be skewed, e.g. 120G.3.2.2: "For the case where PRBS13Q or PRBS31Q are used with a common clock, there is at least 31 UI delay between the patterns on one lane and any other lane, so that the symbols on each lane are not correlated." The training frame is 98.3% PRBS13Q. In principle, one could incur the risk warned against with a lane carrying "identifier_i" = 0 and an adjacent lane carrying "identifier_i" = 4, with an unlucky timing offset between lanes. As "The PMD shall implement one instance of the PMD control function described in 136.8.11 for each lane", the state machine for each lane can be started and restarted asynchronous to adjacent lanes, so starting the training pattern with a different seed won't solve the issue. The text "For 8-lane use cases different initial seeds should be used where the same polynomial is being reused" recommends a course of action that, on investigation, doesn't address the issue. We should tell the reader what to avoid, not how to avoid it.

Also, the ETC spec has already covered this ground. It uses the same four polynomials and seeds, twice over. No implementation can follow the ETC spec AND this draft (because the default seeds differ) but there is no benefit in the difference.

SuggestedRemedy

1. Put signal integrity recommendations in the spec, not in the register definitions for a memory map!
2. Change "The polynomial identifier for each lane should be unique; two physically adjacent lanes having the same identifier could impair operation of the PMD control function" to "The polynomial identifier for adjacent lanes should be unique to avoid a risk of impairment of the PMD control function".
3. Change "For 8-lane use cases different initial seeds should be used where the same polynomial is being reused." to "For 8-lane use cases, see 162.8.11.1."
4. Make the default seeds in Table 162-10a the same as in the ETC spec (seeds 4 to 7 are the same as seeds 0 to 3).
5. ETC say "it is recommended to ensure that physically adjacent lanes do not use the same polynomial". Recommend this.
6. Also, suggest that when there are more lanes than polynomials to use, significant correlation between any lanes can be avoided by a combination of seed and timing offset. Leave it to the implementer to choose how to do this.



Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Replace "The polynomial identifier for each lane should be unique; two physically adjacent lanes having the same identifier could impair operation of the PMD control function. The default identifiers are (binary): for lane 0, 00; for lane 1, 01; for lane 2, 10; for lane 3, 11; for lane 4, 00; for lane 5, 01; for lane 6, 10; for lane 7, 11. For 8-lane use cases different initial seeds should be used where the same polynomial is being reused."
with
"The polynomial identifier for adjacent lanes should be unique to avoid a risk of impairment of the PMD control function. If the same polynomial identifier is used for multiple lanes, different initial seeds should be used for each of those lanes. The default identifiers are (binary): for lane 0, 00; for lane 1, 01; for lane 2, 10; for lane 3, 11; for lane 4, 00; for lane 5, 01; for lane 6, 10; for lane 7, 11."
The adopted baseline clearly states what the default seeds in Table 162-10a should be (see: https://www.ieee802.org/3/df/public/22_09/lusted_3df_01a_2209.pdf). A user would be able to change the default values so that the seeds for lanes 4 to 7 match 0 to 3 by writing appropriate seed values to registers 1.1450 through 1.1457. Therefore it is not appropriate to change Table 162-10a.
See also the response to comment #139

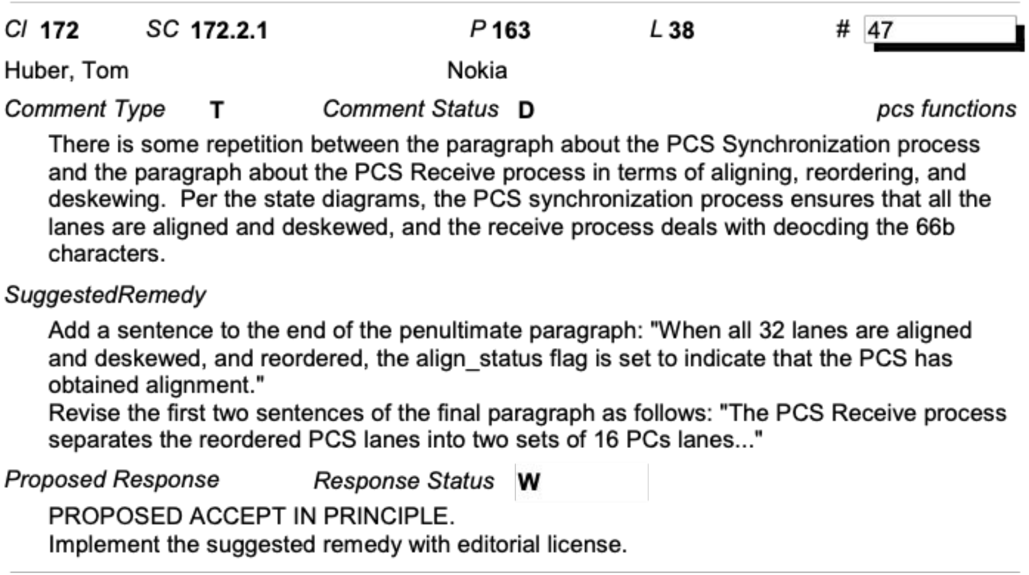
45.2.1.168 PMD training pattern lanes 0 through 73 (Register 1.1450 through 1.1457~~3~~)

The assignment of bits in the PMD training pattern lane 0 register is shown in Table 45–133. The assignment of bits in the PMD training pattern lanes 1 through 73 registers are defined similarly to lane 0. Register 1.1450 controls the PMD training pattern for PMD lane 0; register 1.1451 controls the PMD training pattern for PMD lane 1; etc.

Register bits 12:11 contain a 2-bit identifier that selects the polynomial used for training in the particular PMD lane according to the definition in 92.7.12 and 136.8.11.1.3. The polynomial identifier for each lane should be unique to avoid a risk of impairment of the PMD control function. If the same polynomial identifier is used for multiple lanes, different initial seeds should be used for each of those lanes~~; two physically adjacent lanes having the same identifier could impair operation of the PMD control function~~. The default identifiers are (binary): for lane 0, 00; for lane 1, 01; for lane 2, 10; for lane 3, 11; for lane 4, 00; for lane 5, 01; for lane 6, 10; for lane 7, 11.~~ For 8-lane use cases different initial seeds should be used where the same polynomial is being reused.~~

Clause 172 (Kapil)

PCS functions (part 1)
Comment #47



Cl **172** *SC* **172.2.1** *P* **163** *L* **38** # 47

Huber, Tom Nokia

Comment Type **T** *Comment Status* **D** *pcs functions*

There is some repetition between the paragraph about the PCS Synchronization process and the paragraph about the PCS Receive process in terms of aligning, reordering, and deskewing. Per the state diagrams, the PCS synchronization process ensures that all the lanes are aligned and deskewed, and the receive process deals with deocding the 66b characters.

SuggestedRemedy

Add a sentence to the end of the penultimate paragraph: "When all 32 lanes are aligned and deskewed, and reordered, the align_status flag is set to indicate that the PCS has obtained alignment."
Revise the first two sentences of the final paragraph as follows: "The PCS Receive process separates the reordered PCS lanes into two sets of 16 PCs lanes..."

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Implement the suggested remedy with editorial license.

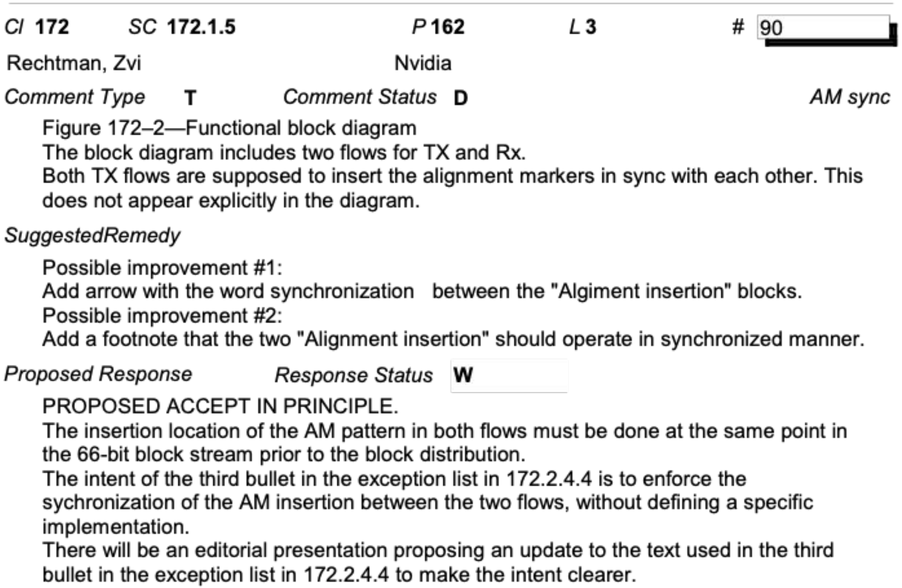
PCS functions (part 2)
Comment #47 - proposed changes

In 172.2.1, the last two paragraphs are edited per the comment

The PCS Synchronization process continuously monitors PMA:IS_SIGNAL.indication(SIGNAL_OK). When SIGNAL_OK indicates OK, then the PCS synchronization process accepts data units via the PMA:IS_UNITDATA_0:31.indication primitive. It attains alignment marker lock based on the common marker (CM) portion that is periodically transmitted on every PCS lane. After alignment markers are found on all PCS lanes, the individual PCS lanes are identified using the unique marker portion (UM) and then reordered,~~ and ~~deskewed, and the align_status flag is set. Note that a particular transmit PCS lane can be received on any receive lane of the service interface due to the skew and multiplexing that occurs in the path.

~~The PCS Receive process aligns, deskews, reorders the 32 PCS lanes, and sets the align_status flag to indicate whether the PCS has obtained alignment. The reordered PCS lanes are separated into two sets of 16 PCS lanes belonging to each flow. ~~The PCS Receive process separates the reordered PCS lanes into two sets of 16 PCS lanes belonging to each flow. Within a flow, the data from the 16 PCS lanes is de-interleaved, processed by the FEC decoder, and re-interleaved on a 10-bit basis to form a single data stream. The alignment markers are removed, the data is descrambled and reverse transcoded back to 66-bit blocks. A 66-bit block collection function merges the 66-bit blocks from the two flows in a round-robin fashion into a single stream of blocks that are then 64B/66B decoded.

AM Sync (part 1)
Comment #90



Cl **172** *SC* **172.1.5** *P* **162** *L* **3** # 90

Rechtman, Zvi Nvidia

Comment Type **T** *Comment Status* **D** *AM sync*

Figure 172–2—Functional block diagram
The block diagram includes two flows for TX and Rx.
Both TX flows are supposed to insert the alignment markers in sync with each other. This does not appear explicitly in the diagram.

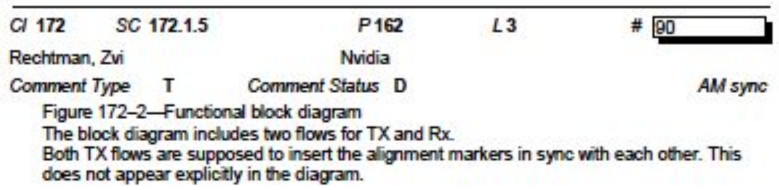
SuggestedRemedy

Possible improvement #1:
Add arrow with the word synchronization between the "Algiment insertion" blocks.
Possible improvement #2:
Add a footnote that the two "Alignment insertion" should operate in synchronized manner.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
The insertion location of the AM pattern in both flows must be done at the same point in the 66-bit block stream prior to the block distribution.
The intent of the third bullet in the exception list in 172.2.4.4 is to enforce the sychronization of the AM insertion between the two flows, without defining a specific implementation.
There will be an editorial presentation proposing an update to the text used in the third bullet in the exception list in 172.2.4.4 to make the intent clearer.

AM Sync (part 2)
Comments related to #90: [91, 159, 108, 180, 9, 60]

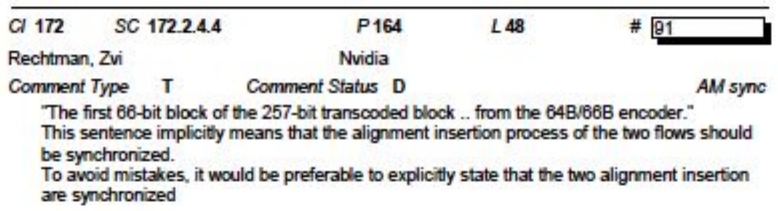


Cl **172** *SC* **172.1.5** *P* **162** *L* **3** **#** 90

Rechtman, Zvi — Nvidia

Comment Type **T** *Comment Status* **D** *AM sync*

Figure 172–2—Functional block diagram
The block diagram includes two flows for TX and Rx.
Both TX flows are supposed to insert the alignment markers in sync with each other. This does not appear explicitly in the diagram.

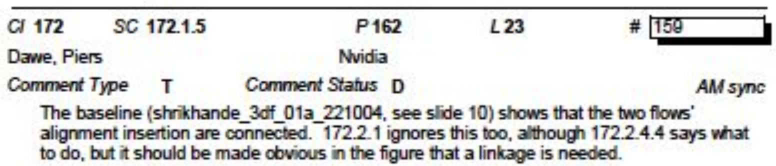


Cl **172** *SC* **172.2.4.4** *P* **164** *L* **48** **#** 91

Rechtman, Zvi — Nvidia

Comment Type **T** *Comment Status* **D** *AM sync*

"The first 66-bit block of the 257-bit transcoded block .. from the 64B/66B encoder."
This sentence implicitly means that the alignment insertion process of the two flows should be synchronized.
To avoid mistakes, it would be preferable to explicitly state that the two alignment insertion are synchronized

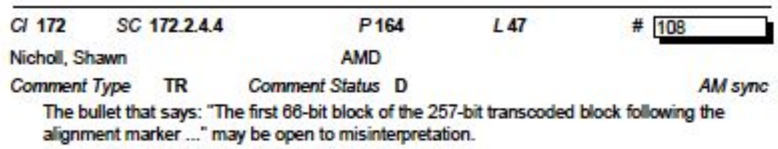


Cl **172** *SC* **172.1.5** *P* **162** *L* **23** **#** 159

Dawe, Piers — Nvidia

Comment Type **T** *Comment Status* **D** *AM sync*

The baseline (shrikhande_3df_01a_221004, see slide 10) shows that the two flows' alignment insertion are connected. 172.2.1 ignores this too, although 172.2.4.4 says what to do, but it should be made obvious in the figure that a linkage is needed.

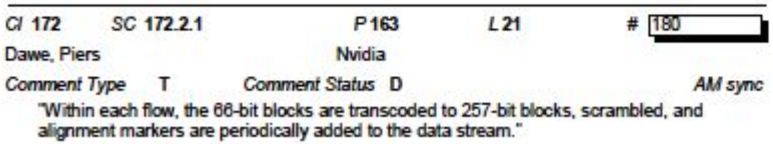


Cl **172** *SC* **172.2.4.4** *P* **164** *L* **47** **#** 108

Nicholl, Shawn — AMD

Comment Type **TR** *Comment Status* **D** *AM sync*

The bullet that says: "The first 66-bit block of the 257-bit transcoded block following the alignment marker ..." may be open to misinterpretation.

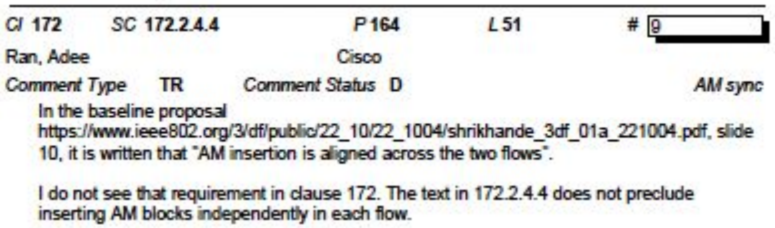


Cl **172** *SC* **172.2.1** *P* **163** *L* **21** **#** 180

Dawe, Piers — Nvidia

Comment Type **T** *Comment Status* **D** *AM sync*

"Within each flow, the 66-bit blocks are transcoded to 257-bit blocks, scrambled, and alignment markers are periodically added to the data stream."



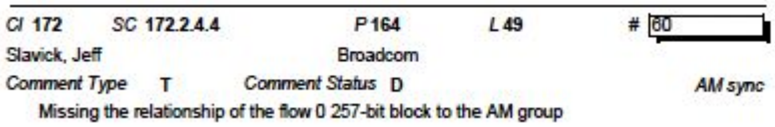
Cl **172** *SC* **172.2.4.4** *P* **164** *L* **51** **#** 9

Ran, Adee — Cisco

Comment Type **TR** *Comment Status* **D** *AM sync*

In the baseline proposal https://www.ieee802.org/3/df/public/22_10/22_1004/shrikhande_3df_01a_221004.pdf, slide 10, it is written that "AM insertion is aligned across the two flows".

I do not see that requirement in clause 172. The text in 172.2.4.4 does not preclude inserting AM blocks independently in each flow.



Cl **172** *SC* **172.2.4.4** *P* **164** *L* **49** **#** 60

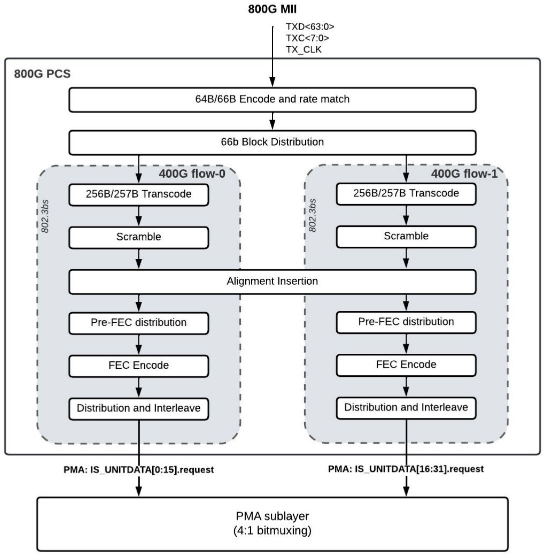
Slavick, Jeff — Broadcom

Comment Type **T** *Comment Status* **D** *AM sync*

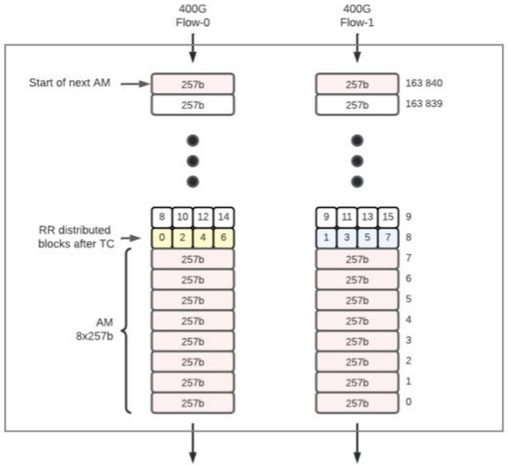
Missing the relationship of the flow 0 257-bit block to the AM group

AM Sync (part 3)
AM insertion in the Baseline

Baseline Proposal Diagram Slide 12 (shrikhande_3df_01a_221004.pdf)



Baseline Proposal Diagram Slide 13 (shrikhande_3df_01a_221004.pdf)



The transmit diagram in the baseline (slide 12) shows Alignment insertion as a single block across 2 flows.

The key information showing how the AM insertion is "synchronized" across the two flows is present in the figure on slide 13, which shows the two Alignment marker groups of flow 0 and flow 1 are followed by 66-bit blocks numbered (0,2,4,6) in flow 0 and (1,3,5,7) in flow 1.

Only stating that the AM insertion should be "synchronized" or showing a single AM insertion block is not sufficient.

AM Sync (part 4)
AM insertion in D1.0 172.2.4.4

172.2.4.4 Alignment marker mapping and insertion

The alignment marker mapping and insertion in each flow is identical to the 400GBASE-R alignment marker and insertion function specified in 119.2.4.4 with the following exceptions:

-Alignment marker encoding values for flow 0 are specified in Table 172–1.
-Alignment marker encoding values for flow 1 are specified in Table 172–2 and the variable x in 119.2.4.4.2 takes the values of PCS lane number minus 16.
-The first 66-bit block of the 257-bit transcoded block following the alignment marker group in flow 1 shall be the 66-bit block that followed the first 66-bit block of the 257-bit transcoded block in flow 0 from the original 66-bit block stream from the 64B/66B encoder.

D1.0 uses the highlighted text in 172.2.4.4 to specify the AM insertion requirement shown in slide 13 of the baseline, without defining a specific implementation.

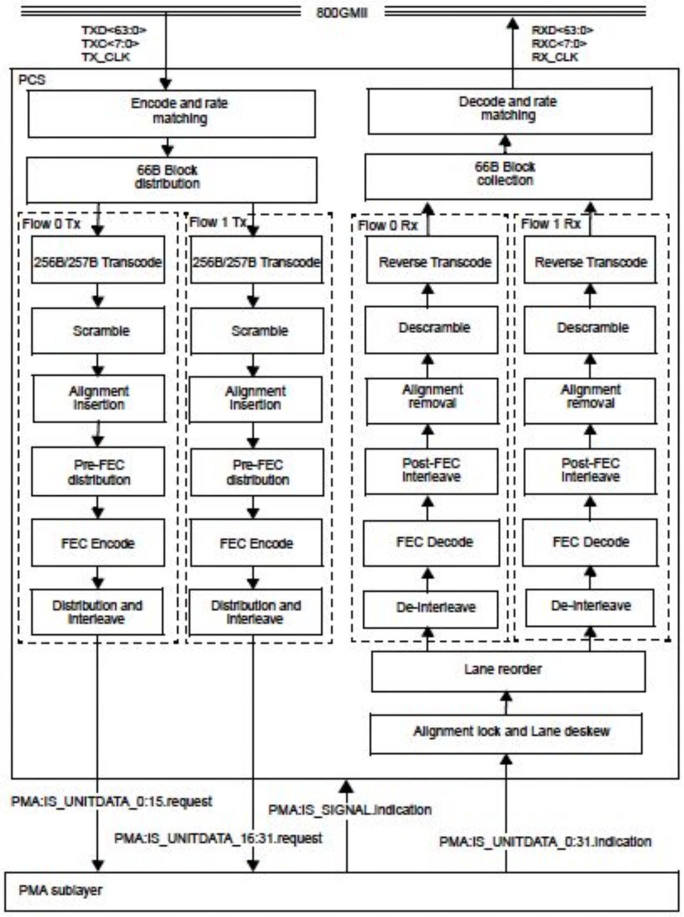


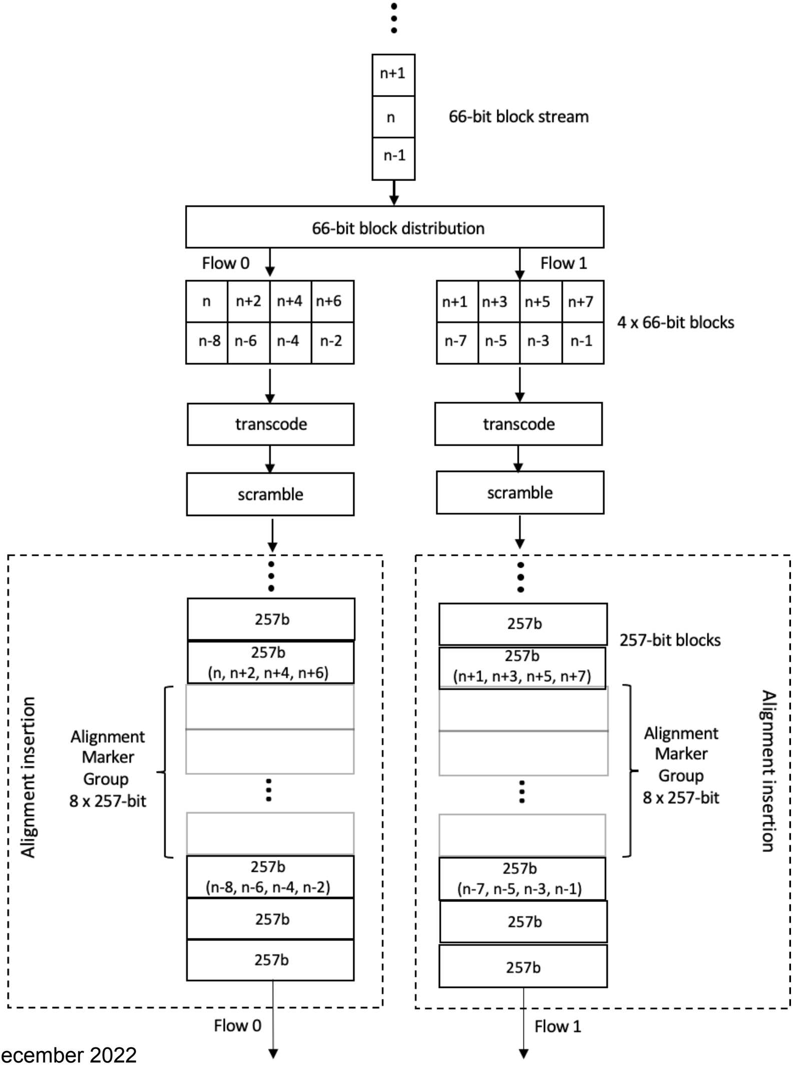
Figure 172–2—Functional block diagram

AM Sync (part 5)
Proposed changes in 172.2.4.4

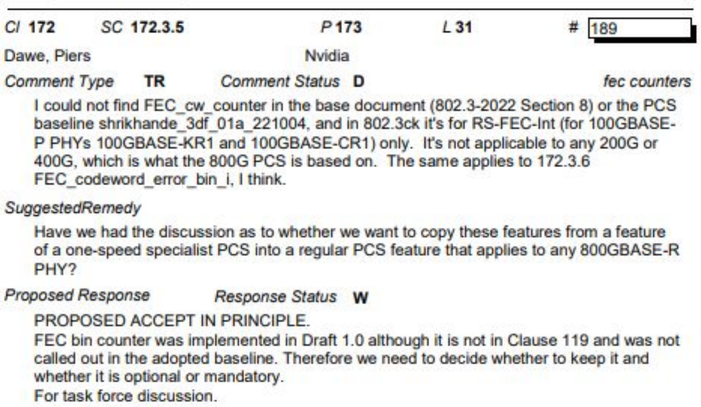
Include the following diagram in 172.2.4.4, with editorial license.

Change the 3rd exception bullet in 172.2.4.4 to the following, with editorial license.

“The alignment marker insertion within each flow shall occur at the same point relative to the original stream of 66-bit blocks before block distribution. This requirement is illustrated in Figure 172-x where the alignment marker group is inserted prior to the 257-bit block containing the 66-bit block “n” for flow 0 and prior to the 257-bit block containing the 66-bit block “n+1” for flow 1. ”



Clause 172: FEC bin counters (part 1)
Comments 189, 4, 63, 64



Cl **172** *SC* **172.3.5** *P* **173** *L* **31** # 189

Dawe, Piers Nvidia

Comment Type **TR** *Comment Status* **D** *fec counters*

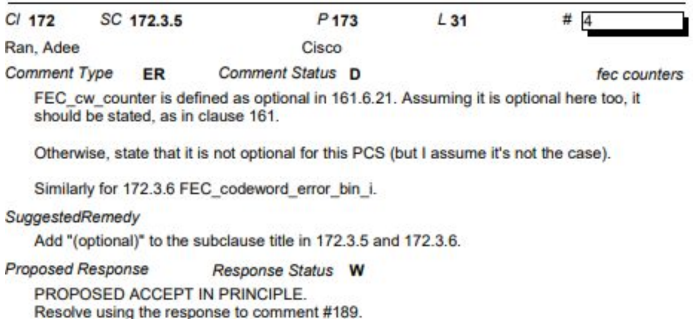
I could not find FEC_cw_counter in the base document (802.3-2022 Section 8) or the PCS baseline shrikhande_3df_01a_221004, and in 802.3ck it's for RS-FEC-Int (for 100GBASE-P PHYs 100GBASE-KR1 and 100GBASE-CR1) only. It's not applicable to any 200G or 400G, which is what the 800G PCS is based on. The same applies to 172.3.6 FEC_codeword_error_bin_i, I think.

SuggestedRemedy

Have we had the discussion as to whether we want to copy these features from a feature of a one-speed specialist PCS into a regular PCS feature that applies to any 800GBASE-R PHY?

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
FEC bin counter was implemented in Draft 1.0 although it is not in Clause 119 and was not called out in the adopted baseline. Therefore we need to decide whether to keep it and whether it is optional or mandatory.
For task force discussion.



Cl **172** *SC* **172.3.5** *P* **173** *L* **31** # 4

Ran, Adee Cisco

Comment Type **ER** *Comment Status* **D** *fec counters*

FEC_cw_counter is defined as optional in 161.6.21. Assuming it is optional here too, it should be stated, as in clause 161.

Otherwise, state that it is not optional for this PCS (but I assume it's not the case).

Similarly for 172.3.6 FEC_codeword_error_bin_i.

SuggestedRemedy

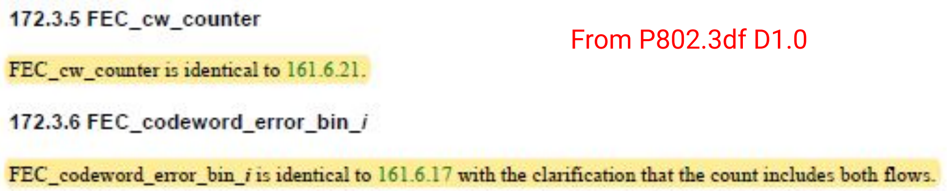
Add "(optional)" to the subclause title in 172.3.5 and 172.3.6.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Resolve using the response to comment #189.

Comment #189 indicates that although these counters were added in Clause 161 in 3ck, they were not explicitly called out in the adopted 3df PCS baseline.

Comment #4 states that it is not clear whether the counters should be optional (as in clause 161) or mandatory.



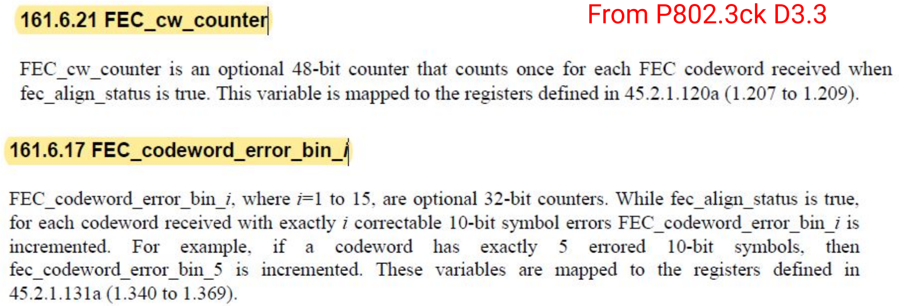
From P802.3df D1.0

172.3.5 FEC_cw_counter

FEC_cw_counter is identical to 161.6.21.

172.3.6 FEC_codeword_error_bin_*i*

FEC_codeword_error_bin_*i* is identical to 161.6.17 with the clarification that the count includes both flows.



From P802.3ck D3.3

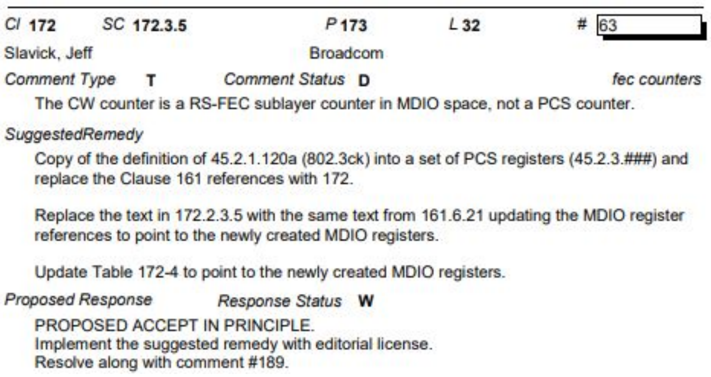
161.6.21 FEC_cw_counter

FEC_cw_counter is an optional 48-bit counter that counts once for each FEC codeword received when fec_align_status is true. This variable is mapped to the registers defined in 45.2.1.120a (1.207 to 1.209).

161.6.17 FEC_codeword_error_bin_*i*

FEC_codeword_error_bin_*i*, where *i*=1 to 15, are optional 32-bit counters. While fec_align_status is true, for each codeword received with exactly *i* correctable 10-bit symbol errors FEC_codeword_error_bin_*i* is incremented. For example, if a codeword has exactly 5 errored 10-bit symbols, then fec_codeword_error_bin_5 is incremented. These variables are mapped to the registers defined in 45.2.1.131a (1.340 to 1.369).

Clause 172: FEC bin counters (part 2)
Comments 189, 4, 63, 64



Cl **172** *SC* **172.3.5** *P* **173** *L* **32** # 63

Slavick, Jeff Broadcom

Comment Type **T** *Comment Status* **D** *fec counters*

The CW counter is a RS-FEC sublayer counter in MDIO space, not a PCS counter.

SuggestedRemedy

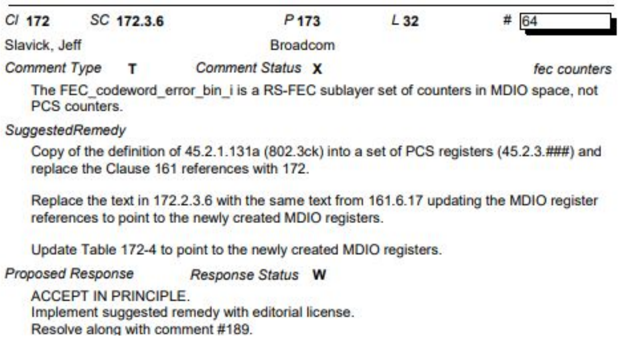
Copy of the definition of 45.2.1.120a (802.3ck) into a set of PCS registers (45.2.3.###) and replace the Clause 161 references with 172.

Replace the text in 172.2.3.5 with the same text from 161.6.21 updating the MDIO register references to point to the newly created MDIO registers.

Update Table 172-4 to point to the newly created MDIO registers.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Implement the suggested remedy with editorial license.
Resolve along with comment #189.



Cl **172** *SC* **172.3.6** *P* **173** *L* **32** # 64

Slavick, Jeff Broadcom

Comment Type **T** *Comment Status* **X** *fec counters*

The FEC_codeword_error_bin_i is a RS-FEC sublayer set of counters in MDIO space, not PCS counters.

SuggestedRemedy

Copy of the definition of 45.2.1.131a (802.3ck) into a set of PCS registers (45.2.3.###) and replace the Clause 161 references with 172.

Replace the text in 172.2.3.6 with the same text from 161.6.17 updating the MDIO register references to point to the newly created MDIO registers.

Update Table 172-4 to point to the newly created MDIO registers.

Proposed Response *Response Status* **W**

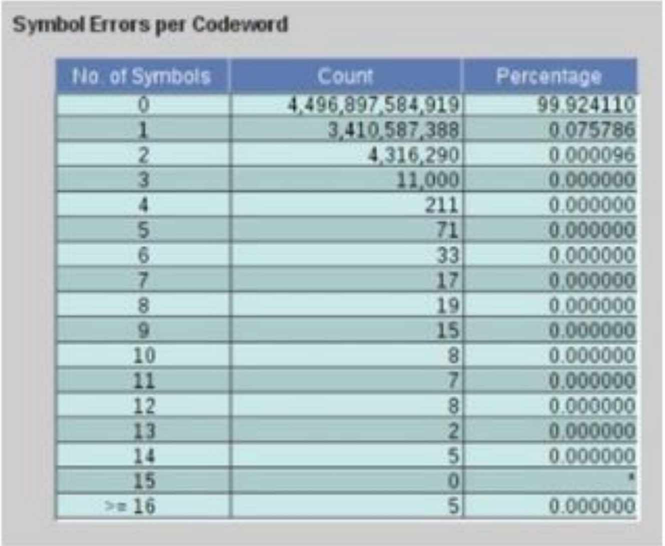
ACCEPT IN PRINCIPLE.
Implement suggested remedy with editorial license.
Resolve along with comment #189.

Comments #63 and #64 indicate that if these counters are to be included in the draft, they need to be mapped to a set of PCS MDIO registers and not a set of RS-FEC MDIO registers (as they are in the current draft, and likely a hold over from the reference to Clause 161 from P802.3ck) .

Clause 172: FEC bin counters (part 3)
Comments 189, 4, 63, 64

FEC bin counter background

- Added in 3ck for Clause 161 (RS-FEC-Int sublayer)
  - https://www.ieee802.org/3/ck/public/20_03/gustlin_3ck_01_0320.pdf
- Comprises 15 x 32 bit counters
  - # of codewords with 1 symbol error
  - # of code words with 2 symbol errors, etc ….
- These counters are valuable because:
  - Indicate margin to FEC cliff
  - Provide insight into error statistics (random or burst)
- Optional in Clause 161 as feature added late in the day
- Although not specifically included in Clause 119, many implementations include these counters (for reasons cited above)



Symbol Errors per Codeword

No. of Symbols	Count	Percentage
0	4,496,897,584,919	99.924110
1	3,410,587,388	0.075786
2	4,316,290	0.000096
3	11,000	0.000000
4	211	0.000000
5	71	0.000000
6	33	0.000000
7	17	0.000000
8	19	0.000000
9	15	0.000000
10	8	0.000000
11	7	0.000000
12	8	0.000000
13	2	0.000000
14	5	0.000000
15	0	*
>= 16	5	0.000000

Figure 4: Classic errored symbol per codeword view of a 400GE link

Source:

https://www.viavisolutions.com/en-us/literature/test-and-validate-fec-implementations-white-papers-books-en.pdf

Clause 172: FEC bin counters (part 4)
Comments 189, 4, 63, 64

Proposed straw polls

Straw poll #xxx

I support keeping the FEC bin counters (FEC_codeword_error_bin_i) and FEC cw counter (FEC_cw_counter) currently defined in 172.3.6 and 172.3.5 respectively.

Y:
N:

Straw poll #xxx

I support the FEC bin counters (FEC_codeword_error_bin_i) and FEC cw counter (FEC_cw_counter) currently defined in 172.3.6 and 172.3.5 respectively, being:

A: Mandatory
B: Optional
C: Need more information

Clause 173 (Gary)

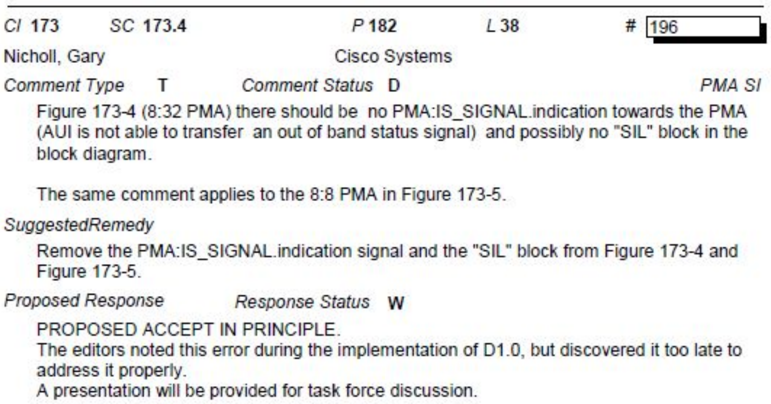
PMA service interface clarifications
(in support of comments 29, 162, 196 and 197)

Gary Nicholl - Cisco
Matt Brown, Xiang He - Huawei
Jeff Slavick - Broadcom

Introduction

- Several comments were received related to the PMA service interface
- These comments were primarily related to the IS_SIGNAL.indication primitive and the fact that:
  - this signal is not supported over an 800GAUI-8 interface
  - this signal is not generated by a PHY 800GXS
- The editorial team also noted that the 800GXS service interfaces are not explicitly defined in Clause 171
  - simple reference to the PCS clause (Clause 172) is not really sufficient, especially for the “PHY 800GXS”
  - need to define the “DTE 800GXS” and “PHY 800GXS” service interfaces in Clause 171 (i.e. PHY 800GXS service interface receives IS_SIGNAL.request and does not generate IS_SIGNAL.indication)
- The editorial team also found that Figure 169-3 needs to up updated related to the comments identified above.

Related Comments



Cl **173** *SC* **173.4** *P* **182** *L* **38** # 196

Nicholl, Gary Cisco Systems

Comment Type **T** *Comment Status* **D** *PMA SI*

Figure 173-4 (8:32 PMA) there should be no PMA:IS_SIGNAL.indication towards the PMA (AUI is not able to transfer an out of band status signal) and possibly no "SIL" block in the block diagram.

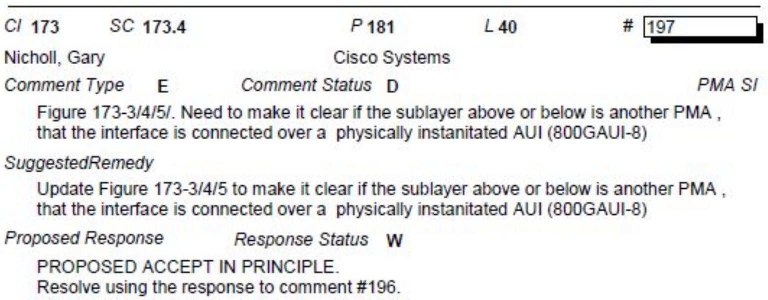
The same comment applies to the 8:8 PMA in Figure 173-5.

SuggestedRemedy

Remove the PMA:IS_SIGNAL.indication signal and the "SIL" block from Figure 173-4 and Figure 173-5.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
The editors noted this error during the implementation of D1.0, but discovered it too late to address it properly.
A presentation will be provided for task force discussion.



Cl **173** *SC* **173.4** *P* **181** *L* **40** # 197

Nicholl, Gary Cisco Systems

Comment Type **E** *Comment Status* **D** *PMA SI*

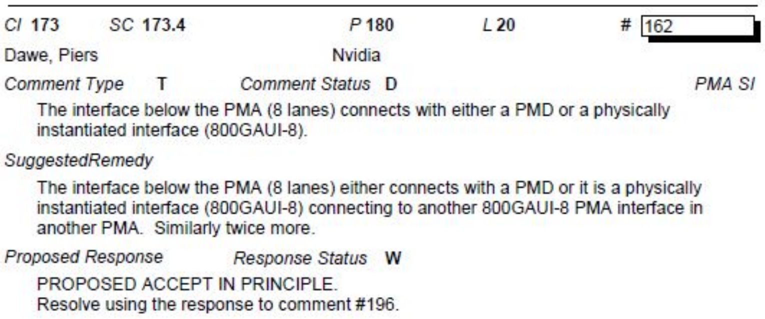
Figure 173-3/4/5/. Need to make it clear if the sublayer above or below is another PMA , that the interface is connected over a physically instanitated AUI (800GAUI-8)

SuggestedRemedy

Update Figure 173-3/4/5 to make it clear if the sublayer above or below is another PMA , that the interface is connected over a physically instanitated AUI (800GAUI-8)

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Resolve using the response to comment #196.



Cl **173** *SC* **173.4** *P* **180** *L* **20** # 162

Dawe, Piers Nvidia

Comment Type **T** *Comment Status* **D** *PMA SI*

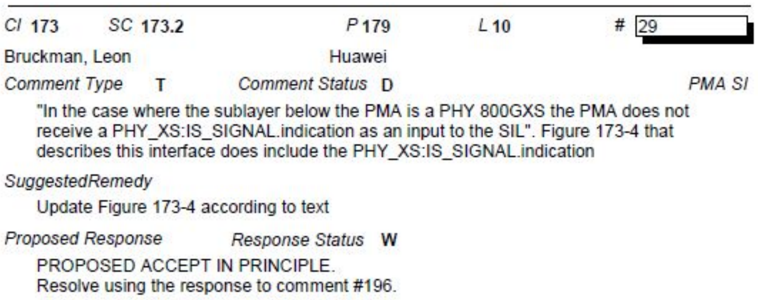
The interface below the PMA (8 lanes) connects with either a PMD or a physically instantiated interface (800GAUI-8).

SuggestedRemedy

The interface below the PMA (8 lanes) either connects with a PMD or it is a physically instantiated interface (800GAUI-8) connecting to another 800GAUI-8 PMA interface in another PMA. Similarly twice more.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Resolve using the response to comment #196.



Cl **173** *SC* **173.2** *P* **179** *L* **10** # 29

Bruckman, Leon Huawei

Comment Type **T** *Comment Status* **D** *PMA SI*

"In the case where the sublayer below the PMA is a PHY 800GXS the PMA does not receive a PHY_XS:IS_SIGNAL.indication as an input to the SIL". Figure 173-4 that describes this interface does include the PHY_XS:IS_SIGNAL.indication

SuggestedRemedy

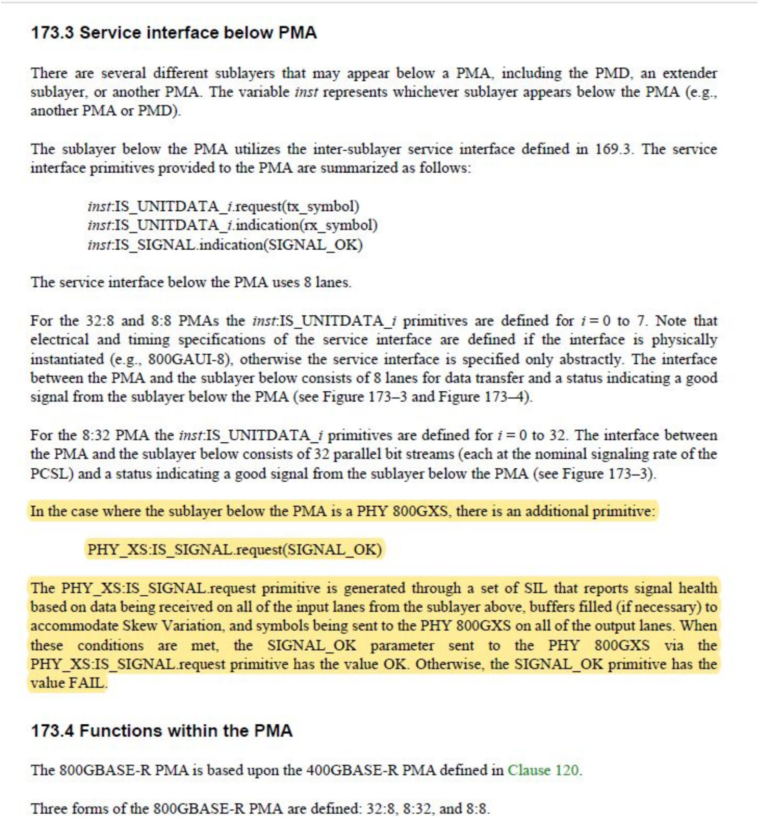
Update Figure 173-4 according to text

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Resolve using the response to comment #196.

PHY 800GXS service interface (background)

- ❖ The editorial team noted that the highlighted text in 173.3 defines an additional service interface primitive for the PHY XS.
- ❖ This primitive is not mentioned anywhere else in the draft and is not included in any of the service interface diagrams.
- ❖ This should be defined in Clause 171, including a full description of the DTE 800GXS and PHY 800GXS service interfaces (rather than just being mentioned in 173.3)



173.3 Service interface below PMA

There are several different sublayers that may appear below a PMA, including the PMD, an extender sublayer, or another PMA. The variable *inst* represents whichever sublayer appears below the PMA (e.g., another PMA or PMD).

The sublayer below the PMA utilizes the inter-sublayer service interface defined in 169.3. The service interface primitives provided to the PMA are summarized as follows:

inst:IS_UNITDATA_*i*.request(tx_symbol)
inst:IS_UNITDATA_*i*.indication(rx_symbol)
inst:IS_SIGNAL.indication(SIGNAL_OK)

The service interface below the PMA uses 8 lanes.

For the 32:8 and 8:8 PMAs the *inst*:IS_UNITDATA_*i* primitives are defined for *i* = 0 to 7. Note that electrical and timing specifications of the service interface are defined if the interface is physically instantiated (e.g., 800GAUI-8), otherwise the service interface is specified only abstractly. The interface between the PMA and the sublayer below consists of 8 lanes for data transfer and a status indicating a good signal from the sublayer below the PMA (see Figure 173–3 and Figure 173–4).

For the 8:32 PMA the *inst*:IS_UNITDATA_*i* primitives are defined for *i* = 0 to 32. The interface between the PMA and the sublayer below consists of 32 parallel bit streams (each at the nominal signaling rate of the PCSL) and a status indicating a good signal from the sublayer below the PMA (see Figure 173–3).

In the case where the sublayer below the PMA is a PHY 800GXS, there is an additional primitive:

PHY_XS:IS_SIGNAL.request(SIGNAL_OK)

The PHY_XS:IS_SIGNAL.request primitive is generated through a set of SIL that reports signal health based on data being received on all of the input lanes from the sublayer above, buffers filled (if necessary) to accommodate Skew Variation, and symbols being sent to the PHY 800GXS on all of the output lanes. When these conditions are met, the SIGNAL_OK parameter sent to the PHY 800GXS via the PHY_XS:IS_SIGNAL.request primitive has the value OK. Otherwise, the SIGNAL_OK primitive has the value FAIL.

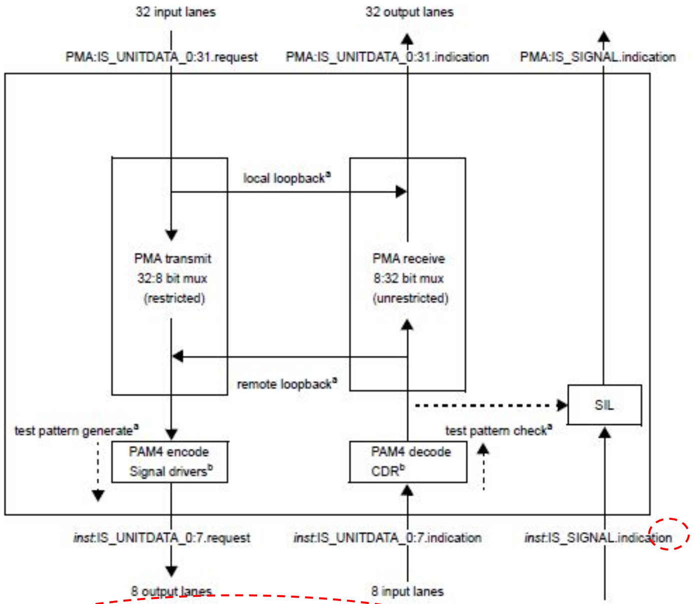
173.4 Functions within the PMA

The 800GBASE-R PMA is based upon the 400GBASE-R PMA defined in Clause 120.

Three forms of the 800GBASE-R PMA are defined: 32:8, 8:32, and 8:8.

Figure 173-3 (32:8 PMA)

Figure in 802.3df D1.0:

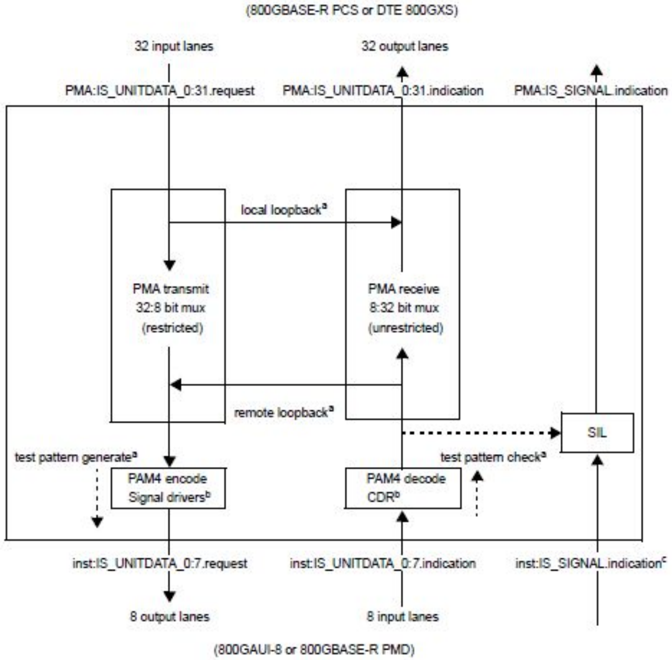


inst: PMD or PMA, depending on which sublayer is below this PMA
SIL: Signal Indication Logic

[a] Optional.
[b] If physically instantiated interface (800GAUI-8) immediately below this PMA.

Figure 173–3—32:8 PMA Functional Block Diagram

Replace with the following:



inst: PMA[b] or PMD[c]
SIL: Signal Indication Logic

[a] Optional.
[b] If an 800GAUI-8 is below the PMA.
[c] If the sublayer below the PMA is a PMD.

Figure 173–3—32:8 PMA Functional Block Diagram

Figure 173-4 (8:32 PMA)

Figure in 802.3df D1.0:

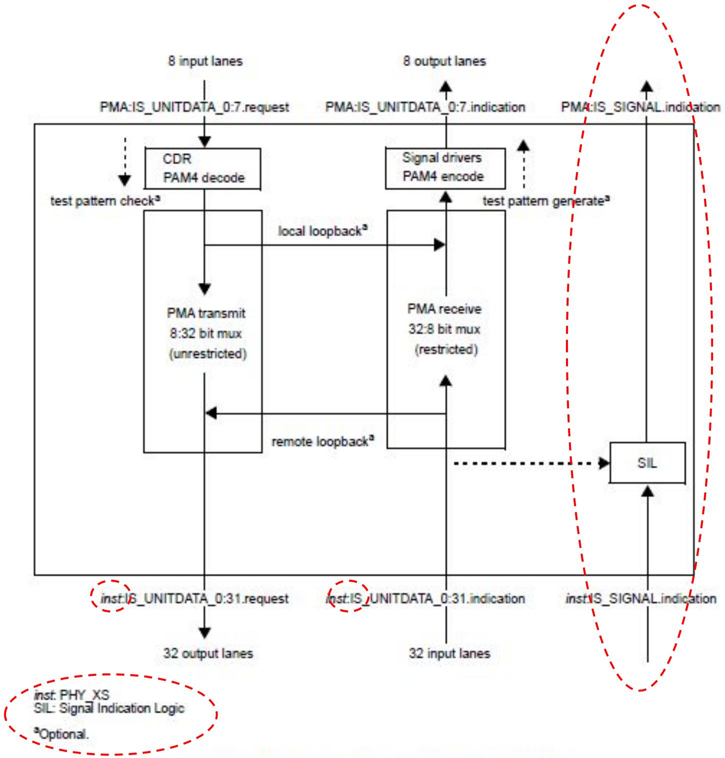
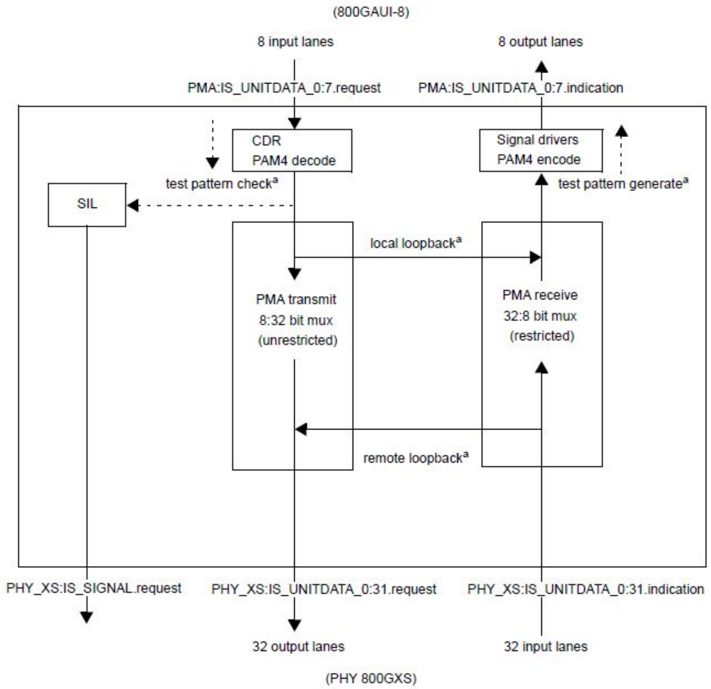


Figure 173–4—8:32 PMA Functional Block Diagram

Replace with the following:



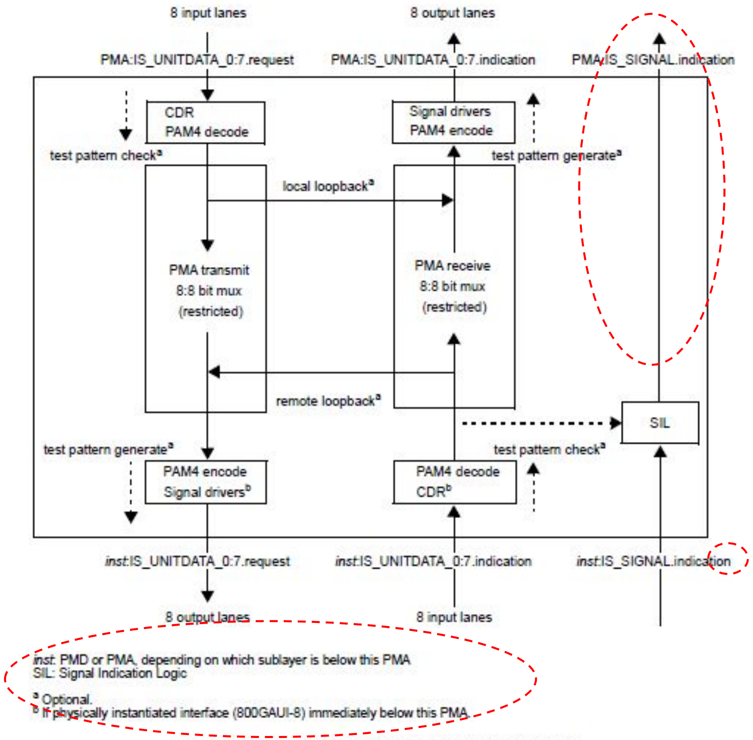
SIL: Signal Indication Logic

^{a}Optional.

Figure 173–4—8:32 PMA Functional Block Diagram

Figure 173-5 (8:8 PMA)

Figure in 802.3df D1.0:

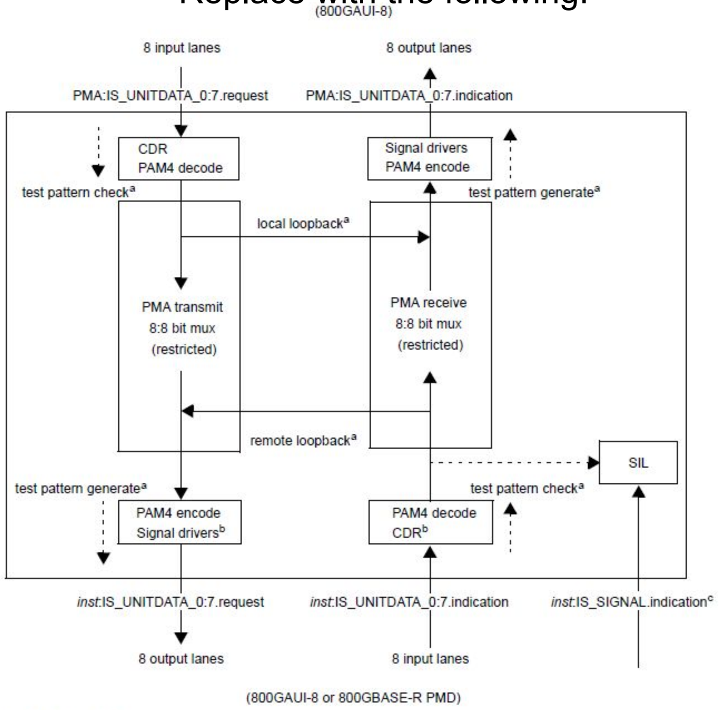


inst: PMD or PMA, depending on which sublayer is below this PMA
SIL: Signal Indication Logic

[a] Optional.
[b] If physically instantiated interface (800GAUI-8) immediately below this PMA

Figure 173–5—8:8 PMA Functional Block Diagram

Replace with the following:



inst: PMA[b] or PMD[c]
SIL: Signal Indication Logic

[a]Optional.
[b]If an 800GAUI-8 is below the PMA.
[c]If the sublayer below the PMA is a PMD.

Figure 173–5—8:8 PMA Functional Block Diagram

DTE 800GXS (Clause 171)

- The DTE 800GXS functionality and service interface is not explicitly defined in Clause 171
- The functional diagram for the DTE XS would be identical to the PCS as shown in Figure 172-2 (with the exception that “PCS” should be labelled “DTE XS”).
- The DTE 800GXS service interface is identical to the 800GBASE-R PCS service interface that is defined in Clause 172.

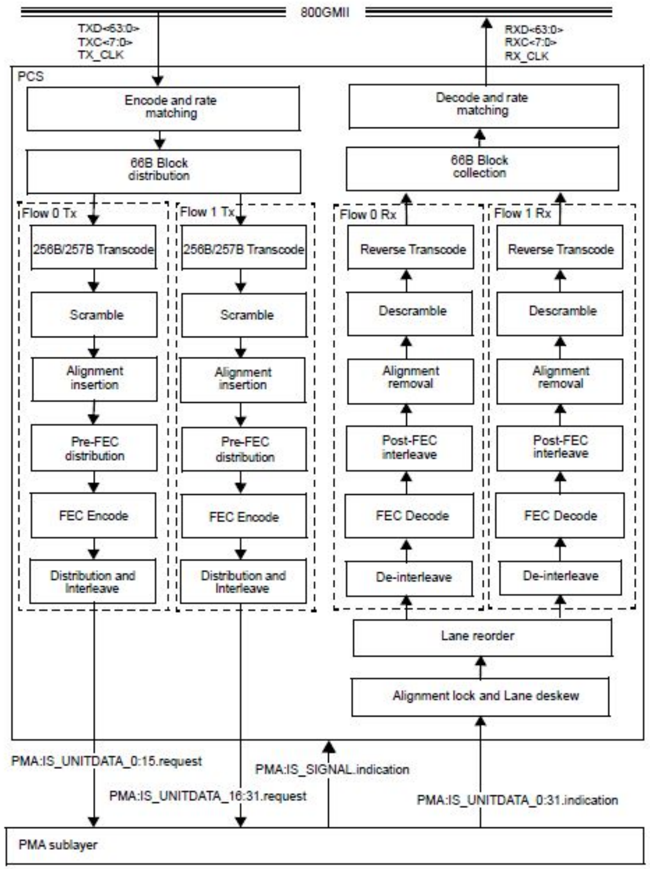


Figure 172–2—Functional block diagram

PHY 800GXS (Clause 171)

❖ The PHY 800GXS functionality and service interface is not explicitly defined in Clause 171.

❖ The functional block diagram for the PHY 800GXS is shown on this slide.

❖ The PHY 800GXS is essentially an upside down 800GBASE-R PCS, and as a result the service interface is somewhat different to the 800GBASE-R PCS service interface that is defined in Clause 172 (for example the service interface receives an IS_SIGNAL.request signal and rather than an IS_SIGNAL.indication signal)

❖ Clause 171 should be updated to show the PHY 800GXS functional block diagram and to define the PHY 800GXS service interface.

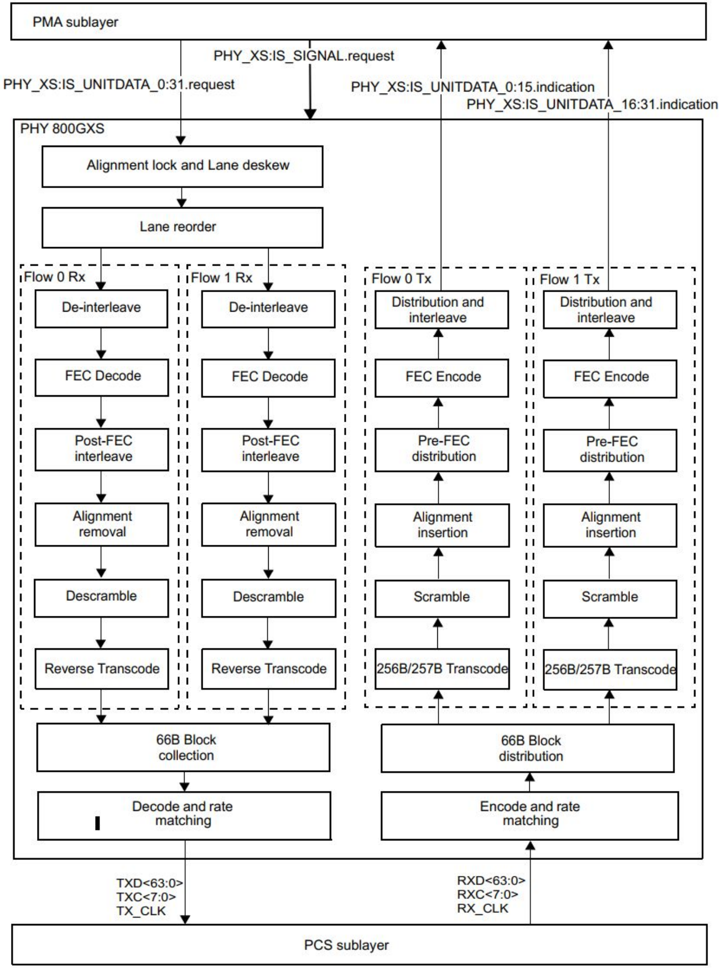
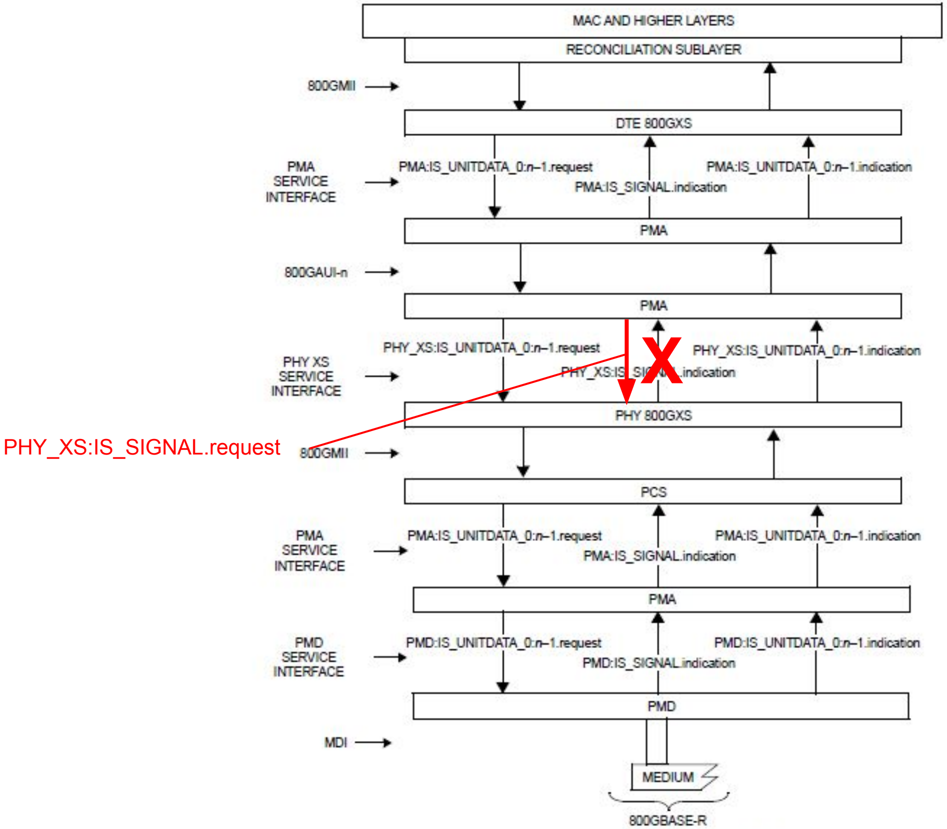


Figure 169-3

- ❖ Update Figure 169-3 as shown.



800GMII = 800 Gb/s MEDIA INDEPENDENT INTERFACE
800GXS = 800 Gb/s PHY EXTENDER SUBLAYER
DTE = DATA TERMINAL EQUIPMENT
MAC = MEDIA ACCESS CONTROL
MDI = MEDIUM DEPENDENT INTERFACE
PCS = PHYSICAL CODING SUBLAYER

PHY = PHYSICAL LAYER DEVICE
PMA = PHYSICAL MEDIUM ATTACHMENT
PMD = PHYSICAL MEDIUM DEPENDENT
XS = EXTENDER SUBLAYER
n = NUMBER OF PARALLEL STREAMS OF DATA UNITS

Figure 169–3—800GBASE-R inter-sublayer service interfaces including 800GMII Extender

Recommendations

- Update Figures 173-3/4/5 with the corresponding figures shown on previous slides, and update any associated text accordingly.
- Update Clause 171 according to previous slides, add DTE 800GXS and PHY 800GXS service interface definitions, and update any associated text accordingly.
- Update Figure 169-3 based on the figure shown in previous slide and update any associated text accordingly.

Clause 173
CDR (bucket1), Comment # 165

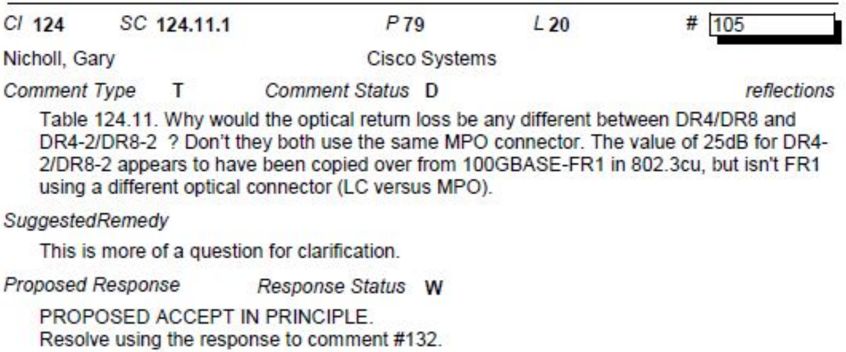
173.4.1 Per input-lane clock and data recovery (CDR)

If the interface between the PMA client and the PMA is physically instantiated as 800GAUI-8, the PMA shall meet the electrical and timing specifications as specified in Annex 120F or Annex 120G as appropriate at the PMA service interface.

If the interface between the sublayer below the PMA and the PMA is physically instantiated as 800GAUI-8, the PMA shall meet the electrical and timing specifications ~~at the service interface below the PMA~~ as specified in Annex 120F or Annex 120G as appropriate at the service interface below the PMA.

Clause 124 (Peter)

Reflections and return loss
Comments 105, 131



Cl **124** *SC* **124.11.1** *P* **79** *L* **20** **#** 105

Nicholl, Gary Cisco Systems

Comment Type **T** *Comment Status* **D** *reflections*

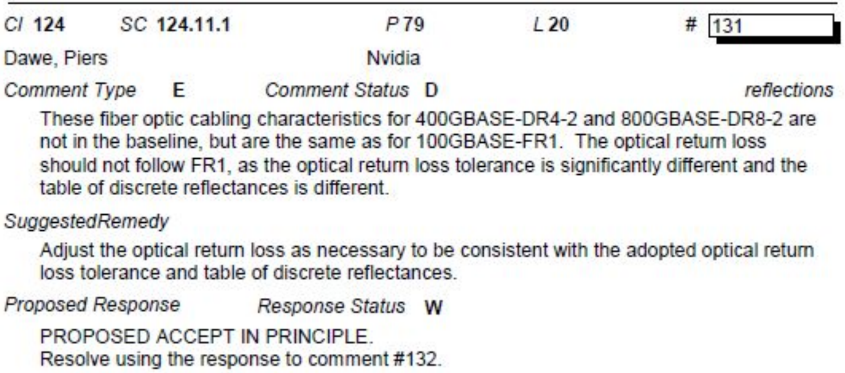
Table 124.11. Why would the optical return loss be any different between DR4/DR8 and DR4-2/DR8-2 ? Don't they both use the same MPO connector. The value of 25dB for DR4-2/DR8-2 appears to have been copied over from 100GBASE-FR1 in 802.3cu, but isn't FR1 using a different optical connector (LC versus MPO).

SuggestedRemedy

This is more of a question for clarification.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Resolve using the response to comment #132.



Cl **124** *SC* **124.11.1** *P* **79** *L* **20** **#** 131

Dawe, Piers Nvidia

Comment Type **E** *Comment Status* **D** *reflections*

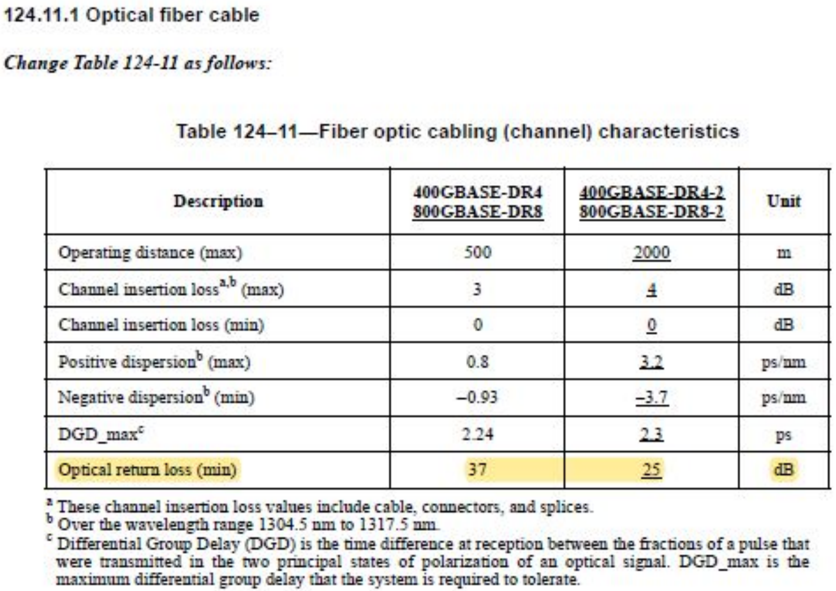
These fiber optic cabling characteristics for 400GBASE-DR4-2 and 800GBASE-DR8-2 are not in the baseline, but are the same as for 100GBASE-FR1. The optical return loss should not follow FR1, as the optical return loss tolerance is significantly different and the table of discrete reflectances is different.

SuggestedRemedy

Adjust the optical return loss as necessary to be consistent with the adopted optical return loss tolerance and table of discrete reflectances.

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Resolve using the response to comment #132.



124.11.1 Optical fiber cable

Change Table 124-11 as follows:

Table 124–11—Fiber optic cabling (channel) characteristics

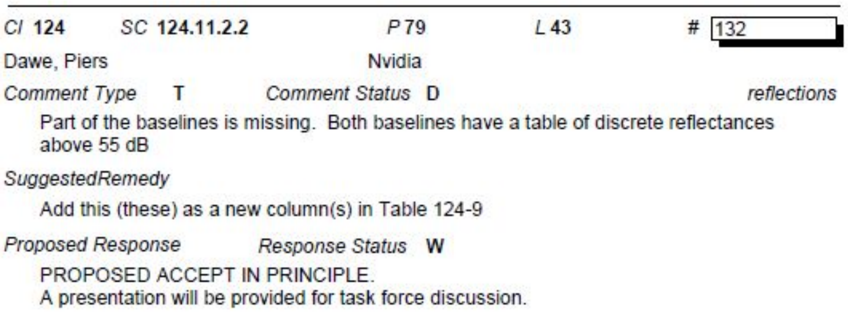
Description	400GBASE-DR4 800GBASE-DR8	400GBASE-DR4-2 800GBASE-DR8-2	Unit
Operating distance (max)	500	2000	m
Channel insertion loss[a,b] (max)	3	4	dB
Channel insertion loss (min)	0	0	dB
Positive dispersion[b] (max)	0.8	3.2	ps/nm
Negative dispersion[b] (min)	–0.93	–3.7	ps/nm
DGD_max[c]	2.24	2.3	ps
Optical return loss (min)	37	25	dB

[a] These channel insertion loss values include cable, connectors, and splices.
[b] Over the wavelength range 1304.5 nm to 1317.5 nm.
[c] Differential Group Delay (DGD) is the time difference at reception between the fractions of a pulse that were transmitted in the two principal states of polarization of an optical signal. DGD_max is the maximum differential group delay that the system is required to tolerate.

The comments are totally right. There should be no difference between DR4/DR8 on one hand and DR4-2/DR8-2 on the other hand. The usage of 100GBASE-FR1 as a reference point was incorrect because of presence of a single fiber instead of a ribbon fiber with 4/8 fibers in one direction.

Thus, in Table-124-11 change 25 dB for Optical return loss (min) to 37 dB.

Reflections and return loss
Comments 132, 133



Cl **124** *SC* **124.11.2.2** *P* **79** *L* **43** # 132

Dawe, Piers Nvidia

Comment Type **T** *Comment Status* **D** *reflections*

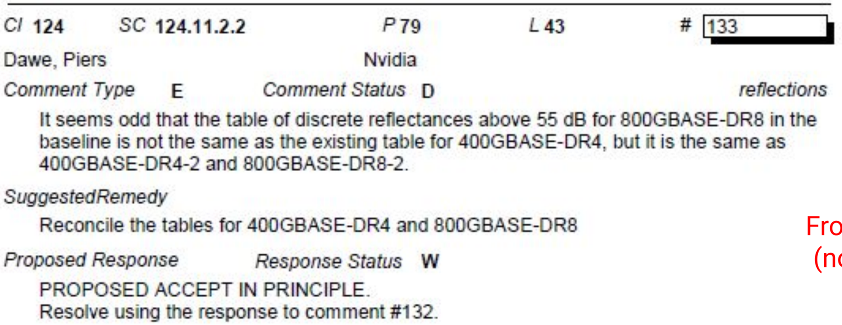
Part of the baselines is missing. Both baselines have a table of discrete reflectances above 55 dB

SuggestedRemedy

Add this (these) as a new column(s) in Table 124-9

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
A presentation will be provided for task force discussion.



Cl **124** *SC* **124.11.2.2** *P* **79** *L* **43** # 133

Dawe, Piers Nvidia

Comment Type **E** *Comment Status* **D** *reflections*

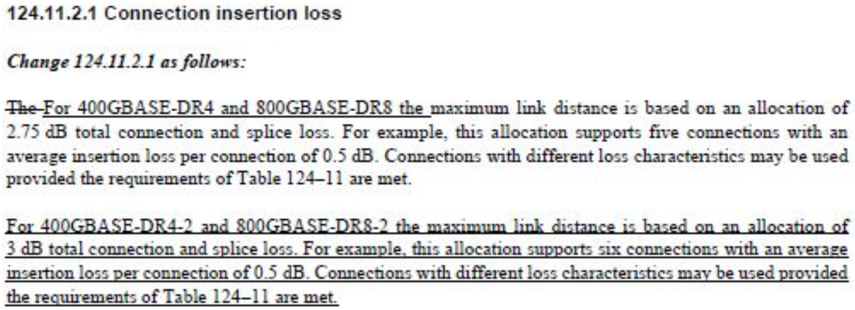
It seems odd that the table of discrete reflectances above 55 dB for 800GBASE-DR8 in the baseline is not the same as the existing table for 400GBASE-DR4, but it is the same as 400GBASE-DR4-2 and 800GBASE-DR8-2.

SuggestedRemedy

Reconcile the tables for 400GBASE-DR4 and 800GBASE-DR8

Proposed Response *Response Status* **W**

PROPOSED ACCEPT IN PRINCIPLE.
Resolve using the response to comment #132.



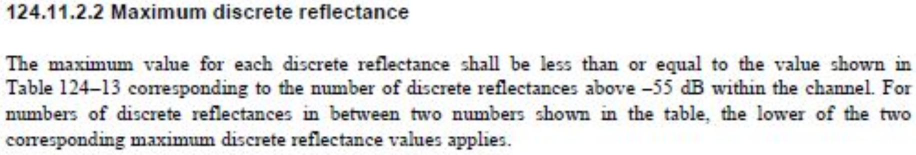
124.11.2.1 Connection insertion loss

Change 124.11.2.1 as follows:

~~The~~ For 400GBASE-DR4 and 800GBASE-DR8 the maximum link distance is based on an allocation of 2.75 dB total connection and splice loss. For example, this allocation supports five connections with an average insertion loss per connection of 0.5 dB. Connections with different loss characteristics may be used provided the requirements of Table 124–11 are met.

For 400GBASE-DR4-2 and 800GBASE-DR8-2 the maximum link distance is based on an allocation of 3 dB total connection and splice loss. For example, this allocation supports six connections with an average insertion loss per connection of 0.5 dB. Connections with different loss characteristics may be used provided the requirements of Table 124–11 are met.

The comments are totally right. There should be no difference between DR4/DR8 on one hand and DR4-2/DR8-2 on the other hand. The usage of 100GBASE-FR1 as a reference point was incorrect. However that has already been taken into account.
Table 124-13 in the in-force Clause 124 has not been modified (as such not shown in P802.3-df D1.0) and is therefore valid for all 4 DR PMD types.
Therefore the proposed should be "reject" because no changes to the draft are require.



124.11.2.2 Maximum discrete reflectance

The maximum value for each discrete reflectance shall be less than or equal to the value shown in Table 124–13 corresponding to the number of discrete reflectances above –55 dB within the channel. For numbers of discrete reflectances in between two numbers shown in the table, the lower of the two corresponding maximum discrete reflectance values applies.

From IEEE Std 802.3-2022
(not amended in 802.3df)

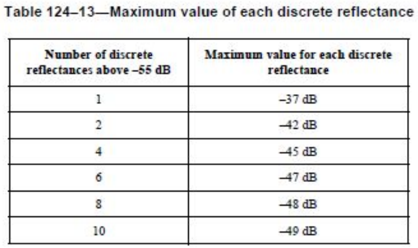


Table 124–13—Maximum value of each discrete reflectance

Number of discrete reflectances above –55 dB	Maximum value for each discrete reflectance
1	–37 dB
2	–42 dB
4	–45 dB
6	–47 dB
8	–48 dB
10	–49 dB