

802.3dj D2.2

Comment Resolution

Common Track

Matt Brown (Alphawave Semi), 802.3dj Chief Editor, Matt Brown (Alphawave Semi), Leon Bruckman (Nvidia), Tom Huber (Nokia), Eugene Opsasnick (Broadcom)

Introduction

- This slide package was assembled by the 802.3dj editorial team to provide background and detailed resolutions to aid in comment resolution.
- Specifically, these slides are for the various **common-track** comments.

Topic test blocks to use for test symbol bins

Comment #312

CI 174A SC 174A.9.2 P746 L24 # 312

Mi, Guangcan Huawei Technologies Co., Ltd.

Comment Type T Comment Status X

In a set of $4 \times 544/p$ consecutive test symbols, the description of which $544/p$ test symbols form a test block could be clearer. For example, the test symbols of indices $0, 4, \dots, 4 \times 544/p - 4$ belong to a test block. The test symbols of indices $1, 5, \dots, 4 \times 544/p - 3$ belong to a test block. The test symbols of indices $2, 6, \dots, 4 \times 544/p - 2$ belong to a test block. The test symbols of indices $3, 7, \dots, 4 \times 544/p - 1$ belong to a test block. On the other hand, it is not clear whether all above mentioned test blocks or only one type of them shall be considered by the bin counters.

SuggestedRemedy

Add the suggested description of test blocks in the comment, or any equivalent but concise description. Besides, make it clear which test blocks shall be considered by bin counters.

Proposed Response Response Status O

174A.9.2 PMA block error counters

Test symbols are defined as non-overlapping groups of 5 consecutive PAM4 symbols (10 bits total).

A test block is defined as a set of $544/p$ test symbols composed of every fourth test symbol in a set of $4 \times 544/p$ consecutive test symbols, where p is the number of physical lanes.

17 bin counters are defined for each lane i in the range 0 through $p-1$, as follows (see 176.7.4.7):

- test_block_error_bin_i_k, for k in the range 0 through 15, counts test blocks with k test symbol errors
- test_block_error_bin_i_16p counts test blocks with 16 or more test symbol errors

The concept of the test blocks was introduced and adopted from the following contribution:

https://www.ieee802.org/3/dj/public/24_09/healey_3dj_02a_2409.pdf

In particular, slides 5 and 6 explain the concept of a test block.

174A.9.2 first defines a test symbol as being 5 consecutive PAM4 symbols (10 bits total). This corresponds to an RS(544,514) FEC symbol.

It further defines a set of $4 \times 544 \times p$ consecutive test symbols. This corresponds to 4 interleaved test blocks. Each test block corresponds to 1 RS-FEC codeword (544 FEC symbols per codeword) divided by the number of physical lanes due to the symbol-wise, round-robin distribution to each of the p physical lanes.

Within that set are 4 test blocks of size $544/p$ test symbols as follows:

Test block 1 comprises test symbols 0, 4, 8, ...

Test block 2 comprises test symbols 1, 5, 9, ...

Test block 3 comprises test symbols 2, 6, 10, ...

Test block 4 comprises test symbols 3, 7, 11, ...

It is however not clear which of these test blocks is to be considered.

In fact, that was on purpose as it shouldn't matter if statistics were considered for all test blocks within a set or only a subset of 1, 2, or maybe 3. However, test time could be reduced by considering all test blocks per set. It may be helpful be clear about the expectations.

Suggested changes:

Add text explaining that any subset of test blocks per set may be accumulated for the purposes of this test. Also, add a note pointing out that test time can be minimized by accumulated based on all test blocks.

Topic new term/acronym for path startup

Comment #412

CI 178B	SC 178B.2	P863	L18	# 412
Ran, Adeo Cisco Systems				
Comment Type	E	Comment Status	D	psu Naming (CI)
"Path startup" is a poor term for what is defined by this annex. Paths have been started up before the functionality in this annex was specified. Also, the acronym is in conflict with the well-known Power Supply Unit.				
The functionality can be better described as "Autonomous path startup", or "Auto path startup" (parallel to Auto-Negotiation), which would result in the acronym APS. APS seems to be an available acronym (except maybe EAPS, "Ethernet Automatic Protection Switching").				
The annex name may be changed accordingly but can also stay as it is.				
SuggestedRemedy				
Rename "Path startup" to "Autonomous path startup" and "PSU" to "APS". Implement across the draft with editorial license.				

At the task force ad hoc 2025/10/30, two related straw polls were taken and are shown to the right. The straw polls are recorded in the following contribution:

https://www.ieee802.org/3/dj/public/adhoc/electrical/25_1030/lusted_3dj_adhoc_01a_251030.pdf

Straw poll #3 indicated strong consensus for adopting the term "autonomous path startup" and acronym "APSU".

Straw Poll #2

For the acronym used for "Annex 178B", I prefer to use:

- A. APSU, e.g. "Autonomous Path Start Up"
- B. LTPSU, e.g. "LT Path Start Up"
- C. something else
- D. don't care

(choose one)

Results (all): A: 11, B: 6, C: 9, D: 3

4

Straw Poll #3:

For the acronym used for "Annex 178B", I would support using APSU, e.g. "Autonomous Path Start Up"

Y: 19, N: 6, A: 6

5

Topic new acronym for block error ratio

Comment #18

CI 174A	SC 174A.9	P744	L45	# 18
Brown, Matt		Alphawave Semi		
Comment Type	TR	Comment Status	D	Block error ratio acronym (CK)
The block error ratio parameter is being used a lot in the industry now with various acronyms emerging. Should create a acronym to line everybody up. The letter "B" is taken already for "bit error ratio". The letter "K" has been used for black in color definitions (e.g., CYMK) and would be equally relevant here for "block".				
<i>SuggestedRemedy</i>				
Introduce a new acronym for block error ratio: "KER". Add new acronym to 1.5 "Abbreviations".				

Straw Poll #1

For the acronym of "Block Error Ratio", I prefer to use:

- A. KER
- B. BLER
- C. BKER
- D. don't care

(choose one)

Results (all): A: 6, B: 19, C: 5, D: 3

3

At the task force ad hoc 2025/10/30, a related straw poll was taken is shown to the right. The straw poll is recorded in the following contribution:

https://www.ieee802.org/3/dj/public/adhoc/electrical/25_1030/lusted_3dj_adhoc_01a_251030.pdf

Straw poll #1 indicated strong consensus for adopting the acronym "BLER" to represent the term "block error ratio".

PSU wording

Path startup function description in clauses 116, 169, and 174

Comments 53, 27, 33, 34, 240,

116.2.9 Inter-sublayer link training (ILT)

Inter-sublayer link training (ILT) (see Annex 178B) facilitates the orderly startup of an inter-sublayer link (ISL) and coordinates the startup of a series of ISLs along a path. ILT, ISL, and path are defined in 178B.3.

ILT is used by the following PMD and AUI types:

- 200GBASE-KR1
- 200GBASE-CR1
- 200GBASE-DR1
- 200GBASE-DR1-2
- 400GBASE-KR2
- 400GBASE-CR2
- 400GBASE-DR2
- 400GBASE-DR2-2
- 200GAUI-1 C2C
- 200GAUI-1 C2M
- 400GAUI-2 C2C
- 400GAUI-2 C2M

169.2.10 Inter-sublayer link training (ILT)

Inter-sublayer link training (ILT) (see Annex 178B) facilitates the orderly startup of an inter-sublayer link (ISL) and coordinates the startup of a series of ISLs along a path. ILT, ISL, and path are defined in 178B.3.

ILT is used by the following PMD and AUI types:

- 800GBASE-KR4
- 800GBASE-CR4
- 800GBASE-DR4
- 800GBASE-FR4-500
- 800GBASE-DR4-2
- 800GBASE-FR4
- 800GBASE-LR4
- 800GAUI-4 C2C
- 800GAUI-4 C2M

174.2.12 Inter-sublayer link training (ILT)

Inter-sublayer link training (ILT) (see Annex 178B) facilitates the orderly startup of an inter-sublayer link (ISL) and coordinates the startup of a series of ISLs along a path. ILT, ISL, and path are defined in 178B.3.

ILT is used by the following PMD and AUI types:

- 1.6TBASE-KR8
- 1.6TBASE-CR8
- 1.6TBASE-DR8
- 1.6TBASE-DR8-2
- 1.6TAUI-8 C2C
- 1.6TAUI-8 C2M

Change 116.2.9 to the following:

116.2.9 Path startup (PSU)

Path startup (PSU) functions are defined in Annex 178B. The inter-sublayer link training (ILT) function facilitates the orderly startup of an inter-sublayer link (ISL). The ready to send (RTS) function coordinates the startup of a series of ISLs along a path. Path and ISL are defined in 178B.3.

ILT and RTS functions are specified for use by the following PMD and AUI types:

- 200GBASE-KR1
- 200GBASE-CR1
- 200GBASE-DR1
- 200GBASE-DR1-2
- 400GBASE-KR2
- 400GBASE-CR2
- 400GBASE-DR2
- 400GBASE-DR2-2
- 200GAUI-1 C2C
- 200GAUI-1 C2M
- 400GAUI-2 C2C
- 400GAUI-2 C2M

Update 169.2.10 and 174.2.12 similarly.

Clause 174/178-183

Comments: 53, 154, 153, 35, 38, 45, 242, 62, 67, 49

Table 174-2—PHY type and clause correlation (1.6TBASE-R optical)

PHY type	Clause ^a											
	90	170	171	175	176	177	120F	120G	176C	176D	180	182
	Time Synchronization	RS	1.6TMI	1.6TMI Extender	1.6TBASE-R PCS	1.6TBASE-R SM-PMA	1.6TBASE-R Inner FEC	1.6TAUI-16 C2C	1.6TAUI-16 C2M	1.6TAUI-8 C2C	1.6TAUI-8 C2M	1.6TBASE-DR8-2 PMD
1.6TBASE-DR8	O	M	O	O	M	M	—	O	O	O	O	M
1.6TBASE-DR8-2	O	M	O	O	M	M	M	O	O	O	O	M

^a O = Optional, M = Mandatory.

Table 178-4—Physical Layer clauses associated with the 1.6TBASE-KR8 PMD

Associated clause	1.6TBASE-KR8
73—AN	Required
90—Time Synchronization	Optional
120F—1.6TAUI-16 C2C	Optional ^b
170—1.6 Tb/s RS	Required
170—1.6TMI ^b	Optional
171—1.6TMI extender	Optional
172—1.6TBASE-R PCS	Required
176—1.6TBASE-R SM-PMA	Required ^d
176C—1.6TAUI-8 C2C	Optional ^b
178B—ILT	Required

^a A 1.6TBASE-KR8 PHY may include one instance of 1.6TAUI-n C2C as described in 176B.7.1.

^b The 1.6TMI is an optional interface. However, if the 1.6TMI is not implemented, a conforming implementation behaves functionally as though the RS and 1.6TMI were present.

^c If a 1.6TAUI-n is implemented in a PHY, additional 1.6TBASE-R SM-PMA sublayers are required according to the guidelines in 176B.7.1.

Table 180-4—Physical Layer clauses associated with the 1.6TBASE-DR8 PMD

Associated clause	1.6TBASE-DR8
90—Time Synchronization	Optional
120F—1.6TAUI-16 C2C	Optional ^a
120G—1.6TAUI-16 C2M	Optional ^b
170—1.6Tb/s RS	Required
170—1.6TMI ^b	Optional
171—1.6TMI Extender	Optional
175—1.6TBASE-R PCS	Required
176—1.6TBASE-R SM-PMA	Required ^d
176C—1.6TAUI-8 C2C	Optional ^b
176D—1.6TAUI-8 C2M	Optional ^a
178B—ILT	Required

^a One or two 1.6TAUI-n may be instantiated within a 1.6TBASE-DR8 PHY as described in 176B.7.1.

^b The 1.6TMI is an optional interface. However, if the 1.6TMI is not implemented, a conforming implementation behaves functionally as though the RS and 1.6TMI were present.

^c If a 1.6TAUI-n is implemented in a PHY, additional 1.6TBASE-R SM-PMA sublayers are required according to the guidelines in 176B.7.1.

ILT/RTS functions are mandatory for all 1.6TBASE-R PMDs and for 1.6TAUI-8 C2M and C2C, but not supportable by 1.6TAUI-16 C2M and C2C.

Add a new column in Table 174-2 and Table 174-3 for Annex 178B labelled “ILT/RTS” with “M” with a footnote b on “M” as follows:

^b Refer to PMD clause for details.

In Table 178-4, and similarly for tables for 1.6TBASE Physical Layers in 179, 180, and 182...

Add a footnote “d” on “Required” for the Annex 178B row as follows:

^b Mandatory for the 1.6TBASE-KR8 PMD, and 1.6TAUI-8 C2C and C2M.

Also, change “178B—ILT” to “178B—ILT/RTS”.

Table 174-3—PHY type and clause correlation (1.6TBASE-R electrical)

PHY type	Clause ^a									
	73	90	170	171	175	176	120F	176C	178	179
	Auto-Negotiation	Time Synchronization	RS	1.6TMI	1.6TMI Extender	1.6TBASE-R PCS	1.6TBASE-R SM-PMA	1.6TAUI-16 C2C	1.6TAUI-8 C2C	1.6TBASE-KR8 PMD
1.6TBASE-KR8	M	O	M	O	O	M	M	O	O	M
1.6TBASE-CR8	M	O	M	O	O	M	M	O	O	M

^a O = Optional, M = Mandatory.

Path startup functions in clauses 116/168/174/178-183/185/187

Comments 53, 154, 35, 37, 38, 45, 242, 62, 67,49

Table 169-2—PHY type and clause correlation (800GBASE copper)

PHY type	Clause ^a										
	73	90	116	120F	120G	120H	120I	120J	120K	120L	120M
800GBASE-KR4	M	O	M	O	M	O	M	O	M	O	M
800GBASE-KR8	M	O	M	O	M	O	M	O	M	O	M
800GBASE-CR4	M	O	M	O	M	O	M	O	M	O	M
800GBASE-CR8	M	O	M	O	M	O	M	O	M	O	M

^a O = Optional, M = Mandatory, C = Conditional (refer to PMD clause for details)

Table 169-3—PHY type and clause correlation (800GBASE optical EAM4)

PHY type	Clause ^a										
	73	90	116	120F	120G	120H	120I	120J	120K	120L	120M
800GBASE-SR4	M	O	M	O	M	O	M	O	M	O	M
800GBASE-SR8	M	O	M	O	M	O	M	O	M	O	M
800GBASE-DR4	M	O	M	O	M	O	M	O	M	O	M
800GBASE-FR4-500	M	O	M	O	M	O	M	O	M	O	M
800GBASE-DR8	M	O	M	O	M	O	M	O	M	O	M
800GBASE-DR12	M	O	M	O	M	O	M	O	M	O	M
800GBASE-DR16	M	O	M	O	M	O	M	O	M	O	M
800GBASE-FR4	M	O	M	O	M	O	M	O	M	O	M
800GBASE-LR4	M	O	M	O	M	O	M	O	M	O	M

^a O = Optional, M = Mandatory, C = Conditional (refer to PMD clause for details)

Table 178-3—Physical Layer clauses associated with the 800GBASE-KR4 PMI

Associated clause	800GBASE-KR4
73—AN	Required
90—Time Synchronization	Optional
120F—800GAUI-8 C2C	Optional ¹
170—800 Gb/s RS	Required
170—800GMI ²	Optional
171—800GMI Extender	Optional
172—800GBASE-R PCS	Required
173—800GBASE-R BM-PMA	Conditional ²
176—800GBASE-R SM-PMA	Required ²
176C—800GAUI-4 C2C	Optional ¹
178B—ILT	Required

¹ A 800GBASE-KR4 PHY may include one instance of 800GAUI-8 C2C as described in 170B.6.1.

² The 800GMI is an optional interface. However, if the 800GMI is not implemented, a conforming implementation behaves functionally as though the RS and 800GMI were present.

³ If a 800GAUI-4 is implemented in a PHY, additional 800GBASE-R BM-PMA or SM-PMA sublayers are required according to the guidelines in 170B.6.1.

Table 169-3a—PHY type and clause correlation (800GBASE optical coherent)

PHY type	Clause ^a										
	90	170	171	172	173	120F	176	176C	176D	184	185
800GBASE-LR1	O	M	O	M	C	O	O	C	O	M	—
800GBASE-ER1-20	O	M	O	M	C	O	O	C	O	—	—
800GBASE-ER1	O	M	O	M	C	O	O	C	O	—	—

^a O = Optional, M = Mandatory, C = Conditional (refer to PMD clause for details)

Table 185-1—Physical Layer clauses associated with the 800GBASE-LR1 PMD

Associated clause	800GBASE-LR1
90—Time Synchronization	Optional
120F—800GAUI-8 C2C	Optional ¹
120G—800GAUI-8 C2M	Optional ¹
170—800 Gb/s RS	Required
170—800GMI ²	Optional
171—800GMI Extender	Optional
172—800GBASE-R PCS	Required
173—800GBASE-R BM-PMA	Conditional ²
176—800GBASE-R SM-PMA	Conditional ²
176C—800GAUI-4 C2C	Optional ¹
176D—800GAUI-4 C2M	Optional ¹
184—800GBASE-LR1 Inner FEC	Required

¹ One or two 800GAUI-n may be instantiated within a 800GBASE-LR1 PHY as described in 170B.6.1.

² The 800GMI is an optional interface. However, if the 800GMI is not implemented, a conforming implementation behaves functionally as though the RS and 800GMI were present.

³ If one or two 800GAUI-n are implemented in a PHY, additional 800GBASE-R BM-PMA or SM-PMA sublayers are required according to the guidelines in 170B.6.1.

ILT/RTS functions are mandatory for a small subset of PMDs and AUIs in these tables.

Add a new column in Table 169-2/3a/3 Annex 178B labelled “ILT/RTS”.

Change footnote a to:

^a O = Optional, M = Mandatory, C = Conditional

In Table 169-2/3, for all existing instances of “C” add footnote “x” as follows:

^x Refer to PMD clause for details.

In Table 169-2/3 in the 178B column, for rows with PMDs not defined in this draft put “C” with footnote y as follows:

^y ILT/RTS functions are for mandatory 800GAUI-4 C2C and C2M.

In Table 169-2/3 in the 178B column, for rows with PMDs defined in this draft put “M” with footnote “x” as above.

Similarly update tables 116-3, 116-3aa, 116-3a, 116-3b, 116-4, 116-4a, 116-5, and 116-5a.

For Table 178-3, and similarly for tables for 800GBASE-R Physical Layers in clauses 179 through 183, for the 178B row change “ILT” to “ILT/RTS” and add footnote z on “Required” as follows ...

^z ILT/RTS functions are mandatory for the 800GBASE-KR4 PMD, and 800GAUI-4 C2C and C2M.

Similarly update tables for 200GBASE-R and 400GBASE-R Physical Layers in clauses 178 through 183.

For Table 169-3a in the 178B column, for each row put “C” with footnote x as above.

For Table 185-1 and Table 187-1 add a row for “178B—ILT/RTS” with “Conditional” and footnote “y” as above.

Implement with editorial license.

Clause 116, part 1

Comments: 53, 28,29, 30, 31

116.3.3.3 IS_SIGNAL.indication

Change the text in 116.3.3.3 as follows:

The IS_SIGNAL.indication primitive is generated by the sublayer to the next higher sublayer to indicate the status of the receive process. This primitive is generated by the receive process to propagate the detection of severe error conditions (e.g., no valid signal being received by the sublayer that generates this primitive on IS_UNITDATA.indication in the receive direction) to the next higher sublayer and to indicate the ILT status for Physical Layer implementations that use the ILT function defined in Annex 178B.

116.3.3.3.1 Semantics of the service primitive

Change the text 116.3.3.3.1 as follows:

IS_SIGNAL.indication(SIGNAL_OK)

The SIGNAL_OK parameter can take on one of two values: OK or FAIL. A value of FAIL denotes that invalid data is being presented (rx_symbol parameters undefined) by the sublayer to the next higher sublayer. A value of OK does not guarantee valid data is being presented by the sublayer to the next higher sublayer.

If ILT is not used then the SIGNAL_OK parameter takes one of two values as follows:

- A value of OK indicates that communication with the next lower sublayer is established (but does not guarantee that valid data is being presented to the next higher sublayer).
- A value of FAIL indicates that the sublayer has not established communication to the next lower sublayer, and valid data is not being presented to the next higher sublayer (the rx_symbol parameters are undefined).

If ILT is used then the SIGNAL_OK parameter takes one of four values as follows:

November 2025



Change:

“and to indicate the ILT status for Physical Layer implementations that use the ILT function defined in Annex 178B.”

To:

“and to convey the RTS status between interfaces that use the RTS function (see 178B.6).”

Similarly apply this change to 116.3.3.4

Change:

"If ILT is not used then the SIGNAL_OK parameter takes one of two values as follows:"

To:

"If the RTS function (178B.6) is not used then the SIGNAL_OK parameter takes one of two values as follows:"

Change:

"If ILT is used then the SIGNAL_OK parameter takes one of four values as follows:"

To:

"If the RTS function (see 178B.6) is used then the SIGNAL_OK parameter takes one of four values as follows:"

Apply similar changes to 116.3.3.4.1

Clause 116

Comments: 53, 30

116.3.3.4.1 Semantics of the service primitive

IS_SIGNAL.request(SIGNAL_OK)

The SIGNAL_OK parameter takes on one of four values: OK, FAIL, IN_PROGRESS, or READY. The values IN_PROGRESS and READY are defined only for Physical Layer implementations that use the ILT function defined in Annex 178B.

Change:

"for Physical Layer implementations that use the ILT function"

To:

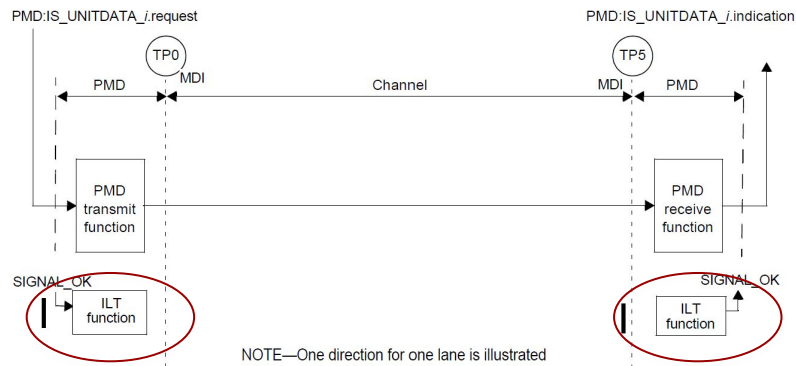
“for service interfaces between interfaces that use the RTS function (see 178B.x)”

Clause 178/179/180/181/182/183/176C

Comment 53, 36, 37, 39, 47, 64, 69, 51, 56

178.8.1 Reference test points

Reference test points are illustrated in Figure 178–2, which shows one direction of a 200GBASE-KR1, 400GBASE-KR2, 800GBASE-KR4, or 1.6TBASE-KR8 link. Additional test points for compliance



Change "ILT function" to "RTS/ILT functions" twice.

(alternately split into two blocks: RTS function (connected to SIGNAL_OK) and ILT function

Same for figures 179-2, 180-2, 181.2, 182-2, 183-2, 176C-2

178.8.9 Inter-sublayer link training (ILT) function

The PMD inter-sublayer link training function specification is identical to that of 179.8.9.

Change the title of 178.8.9 to:
"Path startup (PSU) functions"

Change:
"The PMD inter-sublayer link training function specification is identical to that of 179.8.9."

To:
"The PMD path startup specification is identical to that of 179.8.9."

Clause 179/180/181/182/183

Comments: 53, 40, 42, 61, 66, 71, 53

179.8.4 PMD global signal detect function

The PMD global signal detect function is used by the PMD to indicate the successful completion of the start-up protocol by the inter-sublayer training (ILT) function (see 179.8.9). The variable Global_PMD_signal_detect is set to the value of remote_rts in the ILT function (see 178B.8.2.1).

Change:

"the successful completion of the startup protocol by the inter-sublayer training (ILT) function (see 179.8.9)."

To:

"the successful completion of the inter-sublayer link training (ILT) startup protocol (see 179.8.9)."

179.14 Management variables

PMD control and status variables intended to be accessible via a management system are listed in Table 179–23 and Table 179–24. Additional variables associated with the ILT function are listed in Table 178B–6 and Table 178B–7.

Change:

"Additional variables associated with the ILT function"

To:

"Additional variables associated with the PSU functions"

Apply similar changes to 180.11, 181.11, 182.11, 183.11

Clause 179/176C

Comments: 53, 43, 44

PMDILT	Inter-sublayer link training in PMD	179.8.9	ILT function is implemented in the PMD	PMD:M	Yes [] N/A []
--------	-------------------------------------	---------	--	-------	--------------------

In 179.15.3

Change: "PMDILT" To: "PMDPSU"

Change:

"Inter-sublayer link training in PMD"

To:

"PSU functions in PMD"

Change:

"ILT function is implemented in the PMD"

To:

"PSU functions are implemented in the PMD"

AUIILT	Inter-sublayer link training in AUI-C2C	176C.3	ILT function is implemented in the AUI-C2C	AUI200G: M	Yes [] N/A []
--------	---	--------	--	---------------	--------------------

Apply similar changes in 176C.

Clause 180/181/182/183

Comments: 53, 46, 63, 68, 50

The ILT function indicated in Figure 180–2 is defined in Annex 178B.

In 180.5.1, 181.5.1, 182.5.1, 183.5.1

Change:

"The ILT function indicated in Figure 180–2 is defined in Annex 178B."

To:

"The ILT and RTS functions indicated in Figure 180–2 are defined in Annex 178B."

Other updates related to PSU, ILT, and RTS

Comments: 53

Update any other instances where PSU, ILT, and RTS are discussed and are not consistent with the previous set of related slides.

Other PSU

PSU “support” description

Comments: 414, 11

Cl 178B	SC 178B.7	P868	L1	# 11
Brown, Matt		Alphawave Semi		
Comment Type	TR	Comment Status	D	Scope (Cl)
The ILT is defined assuming that all ISLs in a path support RTS/ILT. There is no guidance on behavior when one or more ISLs in a path do not support do not support those functions. For instance, how does ILT work on an ISL (200 Gb/s per lane) if the other ISLs are 100 Gb/s per lane or lower.				
SuggestedRemedy				
Add guidance for the case where the path does not support path startup.				
Proposed Response	Response Status W			
PROPOSED REJECT.				
As stated in the scope, this is beyond the scope of this Annex.				
The suggested remedy does not provide sufficient detail to implement.				

Cl 178B	SC 178B.4	P 865	L3	# 414
Ran, Adeel		Cisco Systems		
Comment Type	T	Comment Status	D	Definitions (Cl)
The first paragraph and dashed list define "support for PSU" in a very confusing way. The word "support" is overloaded and is used here recursively (support is defined by support). The order of the dashed list is top-down, and the reader needs to read the last item to get a chance to understand what "supported" means, and even then, the last item is defines "An ISL supports" (PSU) using "the interface supports" (functions), which is not well defined, so it's an incomplete definition. Functions are not "supported", they are specified, and should be implemented; these are not optional features.				
Also it is not explained what happens when PSU is not "supported".				
The suggested remedy rewrites this part of 178B.4 without "support", and from the bottom up.				
SuggestedRemedy				
Replace the first paragraph and list with the follows:				
Support for PSU is defined as follows:				
— An ISL between two interfaces can be activated using PSU if these interfaces and the associated sublayers (e.g., PMA, Inner FEC), implement the RTS function (see 178B.6) and the ILT function (see 178B.7), or have equivalent functions.				
— A PHY can be activated using PSU if every ISL within the PHY can be activated using PSU.				
— An xMII Extender can be activated using PSU if every ISL within it can be activated using PSU.				
— A Physical Layer can be activated using PSU if the PHY and xMII Extender (if implemented) can be activated using PSU.				
— A path can be activated using PSU if the Physical Layer at each end can be activated using PSU.				
An ISL, PHY, Physical Layer or path that cannot be activated using PSU may be activated using management or other means beyond the scope of this annex.				
Implement with editorial license.				

Alternate text...

- Support for PSU is defined as follows:
- An ISL can be activated using PSU if the two interfaces and the associated sublayers (e.g., PMA, Inner FEC), implement the RTS function (see 178B.6) and the ILT function (see 178B.7), or have equivalent functions.
 - A PHY can be activated using PSU if every ISL within the PHY can be activated using PSU.
 - An xMII Extender can be activated using PSU if every ISL within it can be activated using PSU.
 - A Physical Layer can be activated using PSU if the PHY and xMII Extender (if implemented) can be activated using PSU.
 - A path can be activated using PSU if the Physical Layer at each end can be activated using PSU.
- An ISL, PHY, Physical Layer, or path that cannot be activated using PSU may be activated using management or other means beyond the scope of this annex.

The last paragraph addresses comment #11.

ILT polarity

Comments: 180

CI 178B	SC 178B.7.7	P 878	L 42	# 180
Dudek, Mike		Marvell		
Comment Type	T	Comment Status	D	Polarity (CI)

Polarity detection and correction is described in 178B.7.7 and required in 179.8.3 and clause 178 and annexes 176C and 176D by reference to 179.8.3. Nothing is however mentioned for the optical clauses leaving it somewhat ambiguous whether it is required or not.

Suggested Remedy

Change the NOTE from "NOTE—Polarity detection and correction is not available when training is disabled." to "NOTE—Polarity detection and correction is not available when training is disabled, or for interfaces using the O1 format."

Proposed Response Response Status W

178B.7.7 Polarity detection and correction

When training starts for each lane, the variable `polarity_correction` is set to false. If inverted frame markers are detected during the frame lock process, the `polarity_correction` variable is set to true.

If `polarity_correction` is true and `local_tf_lock` is true, the lane input shall be corrected by mapping the received PAM4 symbols 0, 1, 2, and 3 to PAM4 symbols 3, 2, 1, and 0, respectively.

NOTE—Polarity detection and correction is not available when training is disabled.

The state of the `polarity_correction` variable persists after training completes, correcting the polarity of the data received when `tx_mode` = data.

From 802.3dj Draft 2.2, 179.8.2 PMD transmit function

The polarity of the PMD output on each of the lanes is either normal, where the highest differential output voltage corresponds to the PAM4 symbol 3 and the lowest differential output voltage corresponds to the PAM4 symbol 0, or inverted, where the highest differential output voltage corresponds to the PAM4 symbol 0 and the lowest differential output voltage corresponds to the PAM4 symbol 3. If training is enabled, either normal or inverted output polarity may be used, since the PMD receiver in the peer interface detects the polarity and corrects it if necessary (see 179.8.3). If training is disabled, the output polarity of each lane shall be normal. Output polarity may be controlled by management in an implementation dependent manner.

From 802.3dj Draft 2.2, 179.8.3 PMD receive function

The polarity of the PMD input on each of the lanes is either normal, where the highest differential input voltage at the MDI input corresponds to `rx_symbol` = three and the lowest differential input voltage corresponds to `rx_symbol` = zero, or inverted, where the highest differential input voltage at the MDI corresponds to `rx_symbol` = zero and the lowest differential input voltage corresponds to `rx_symbol` = three. If training is enabled, the PMD shall detect the polarity during training and correct it if necessary (see 178B.7.7). If training is disabled, the input polarity of each lane shall be normal. Input polarity may be controlled by management in an implementation dependent manner.