

1 **Comment #55, #56, #57, #63, #70, #104, #105, #106, #107, #117, #120, #121, #122, #126, #128, #326, #327, #399,**  
2 **(TDL #268, #269, #358, #143 from D2.2)**  
3

4 Single-signature DLL state machines:

- 5 1. From comment #155 D2.2, the assignment 'pd\_dll\_power\_type <= parameter\_type' in the INITIALIZE state is not required  
6 (Figure 145-44, PD DLL state machine) and was already implemented in D2.3. Removing it also from Table 145-39.
- 7 2. From comment #167 D2.2, the assignment 'pse\_dll\_power\_type <= pse\_power\_type' in the INITIALIZE state is not required  
8 (Figure 145-43, PSE DLL state machine) and was already implemented in D2.3. Removing it also from Table 145-39.
- 9 3. Removing pse\_dll\_power\_type from the variable list. It is not used by the state machines Figures 145-43 and Figure 145-44.
- 10 4. Removing pse\_power\_type from the variable list. It is not used by the state machines Figures 145-43 and Figure 145-44.
- 11 5. pd\_dll\_single\_or\_dual was removed the entire draft due to the following reasons: (a) PD knows if it is a single-signature or  
12 dual-signature. (b) This information is in the TLVs fields anyway. As a result PD will use the correct DLL state machine.
- 13 6. pse\_dll\_single\_or\_dual was replaced with the variable sig\_type which already exists and generated but the connection check  
14 function. It is needed for PSE DLL state machine only.

15  
16 Dual-signature DLL state machines:

- 17 7. The suffix "(M)" was replace with "(X)" to prevent confusion with the Mode word.
  - 18 8. The use of "Alternative" in a variable name was replaced with "Alt" for having shorter variable names.
  - 19 9. PD DLL state machine variables must use variable names with the suffix XXX\_mode~~(M)~~(X) (e.g.  
20 PDRequestedPowerValue\_mode(X)) and their attributes need to use XXXA and XXXB (e.g.  
21 aLldpXdot3LocPDRequestedPowerValueA and aLldpXdot3LocPDRequestedPowerValueB without the suffix "Mode" or "Alt"  
22 due to the fact that the same attribute may be used in PSE or PD and using PSE attribute with the suffix "Mode" is confusing.
  - 23 10. PSE DLL state machine variables must use variable names with the suffix XXX\_Alt(X) (e.g. PSEAllocatedPowerValue\_Alt(X)  
24 and their attributes need to use XXXA and XXXB (e.g. aLldpXdot3LocPSEAllocatedPowerValueA and  
25 aLldpXdot3LocPSEAllocatedPowerValueA without the suffix "Mode" or "Alt" due to the fact that the same attribute may be  
26 used in PSE or PD and using PSE attribute with the suffix "Mode" is confusing.
  - 27 11. From comment #167 D2.2, the assignment 'pse\_dll\_power\_type <= pse\_power\_type' in the INITIALIZE state is not required  
28 (Figure 145-47, PSE DLL state machine). Removing pse\_dll\_power\_type from the variable list. It is not used by the state  
29 machines Figures 145-47 and Figure 145-48.
  - 30 12. From comment #155 D2.2, the assignment 'pd\_dll\_power\_type <= parameter\_type' in the INITIALIZE state is not required  
31 (Figure 145-48, PD DLL state machine). Removing pse\_power\_type from the variable list. It is not used by the state machines  
32 Figures 145-47 and Figure 145-48.
  - 33 13. Removing the variables pd\_dll\_power\_type and pse\_dll\_power\_types from Table 145-40.
  - 34 14. In PSE\_POWER\_REVIEW state, the local\_system\_change\_Alt~~(M)~~(X) was reset to prevent it from being stuck in endless loop  
35 (set to local\_system\_change\_Alt~~(M)~~(X) <= FALSE). This is done according to Clause 21.5.2, "Once set, variables retain their  
36 values as long as succeeding blocks contain no references to them." As a result, we need to explicitly change it in the state  
37 when needed. Other option would be to treat the exit from RUNNING state to PSE\_POWER\_REVIEW as a triggering pulse but in this case we need to  
38 define this behavior in the local\_system\_change\_Alt~~(M)~~(X) variable definition. Same issues with the MIRROR\_UPDATE state for  
39 pse\_power\_update\_Alt~~(M)~~(X) <= True. Both methods were simulated and work. I choose the first one which is simple and covered by Clause 21.5.2.
  - 40 15. Missing "\_mode~~(M)~~(X)" or "\_Alt~~(M)~~(X)" in some of the parameters
  - 41 16. Adding missing dual-signature variables to clause 30 and updating clause 79 Tables.
- 42 See Annex A for naming variables and attributes concept.

43  
44 **Proposed Remedy:**

45 **Implement the following baseline with the proposed additions, deletions and changes.**  
46

47

48

Proposed Baseline starts here

50

51

*Make the following changes (insertion, deletions etc.) to 145.5*

52

53

**145.5 Data Link Layer classification**

54

55

56

**145.5.3 Power control state diagrams**

57

The power control state diagrams for PSEs and PDs specify the externally observable behavior of a PSE and PD Data Link Layer classification respectively.

58

59

Data Link Layer classification of PSEs connected to a single-signature PD, shall provide the behavior in the state diagram defined in Figure 145-43 and Figure 145-45. Data Link Layer classification of PSEs connected to a dual-signature PD, shall provide the behavior in the state diagram defined in Figure 145-47 .

60

61

62

Single-signature PD Data Link Layer classification shall provide the behavior of the state diagram defined in Figure 145-46 44 and Figure 145-44. Dual-signature PD Data Link Layer classification shall provide the behavior of the state diagram defined in Figure 145-48.

63

64

**145.5.3.1 Conventions**

65

66

67

*Editor to splits in 145.5.3.2, 145.5.3.3, 145.5.3.4 and 145.5.3.5 the constants, variable and function list into one for PSE and one for PD (See example in the dual-signature DLL state machine).*

68

69

**145.5.3.2 Single-signature system Constants**

70

71

72

**145.5.3.3 Single-signature system Variables**

73

~~pd\_dll\_single\_or\_dual~~

74

~~A variable in the PD power control state diagram, defined in Figure 145-44, that indicates if the PD is a single signature PD or a dual signature PD. Values: single: A single signature PD configuration is connected to the PI. dual: A dual signature PD configuration is connected to the PI.~~

75

76

~~pse\_dll\_power\_type~~

77

~~A control variable output by the PD power control state diagram, defined in Figure 33-49, that indicates the PSE Type as 1 or 2, see 79.3.2.4.1.~~

78

~~Values:~~

79

~~1: The PSE is a Type 1 PSE, for a Type 1 PSE~~

80

~~2: The PSE is a Type 2 PSE, for Type 2, Type 3, or Type 4PSEs~~

81

82

~~sig\_type~~

83

~~pse\_dll\_single\_or\_dual~~

84

~~A variable in the PSE power control state diagram defined in Figure 145-43 (generated from the do\_exn\_check do\_cxn\_chk function of the Type 3 and Type 4 PSE state diagram in Figure 145-13) which indicates if the PSE is connected to a single-signature PD or dual-signature PD.~~

85

~~Values:~~

86

~~Invalid: Neither single-signature PD nor dual-signature PD connection check signature has been found. This include an open circuit condition.~~

87

~~single: A single-signature PD configuration is connected to the PI.~~

88

~~dual: A dual-signature PD configuration is connected to the PI.~~

89

90

91

92

93

94

95

96

97

98

99

100

101

102 ~~pse\_power\_type~~  
 103 ~~A control variable that indicates to the PD the type of PSE by which it is being powered.~~  
 104 ~~Values:-~~  
 105 ~~1: The PSE is a Type 1 PSE.~~  
 106 ~~2: The PSE is a Type 2, Type 3, or Type 4 PSE.~~

107 **145.5.3.4 Timers**

108 .....

109  
 110 **145.5.3.5 Functions**

111 .....

112  
 113  
 114 Remove from Table 145-39 the variables `pd_dll_power_type` and `pse_dll_power_type`.

115 **Table 145–39—Attribute to state diagram variable cross-reference**

Entity	Attribute	Mapping	State diagram variable
oLldpXdot3LocSystemsGroup Object Class			
PSE	aLldpXdot3LocPDRRequestedPowerValue	←	PDRequestedPowerValueEcho
	aLldpXdot3LocPSEAllocatedPowerValue	←	PSEAllocatedPowerValue
	<del>aLldpXdot3LocReady</del>	←	<del>pse_dll_ready</del>
	aLldpXdot3LocPSEAutoclassSupport	←	PSEAutoclassSupport
	aLldpXdot3LocAutoclassCompleted	←	PSEAutoclassCompleted
PD	aLldpXdot3LocPDRRequestedPowerValue	←	PDRequestedPowerValue
	aLldpXdot3LocPSEAllocatedPowerValue	←	PSEAllocatedPowerValueEcho
	<del>aLldpXdot3LocReady</del>	←	<del>pd_dll_ready</del>
	aLldpXdot3LocAutoclassRequest	←	PDAutoclassRequest
oLldpXdot3RemSystemsGroup Object Class			
PSE	aLldpXdot3RemPDRRequestedPowerValue	→	MirroredPDRRequestedPowerValue
	aLldpXdot3RemPSEAllocatedPowerValue	→	MirroredPSEAllocatedPowerValueEcho
	<del>aLldpXdot3RemPowerType</del>		<del>pd_dll_power_type</del>
	<del>Values:-</del>		<del>Values:-</del>
	<del>-11</del>	→	<del>-01</del>
	<del>-01</del>	→	<del>-10</del>
	aLldpXdot3RemPSEAutoclassSupport	→	MirroredPSEAutoclassSupport
	aLldpXdot3RemAutoclassCompleted	→	MirroredPSEAutoclassCompleted
PD	aLldpXdot3RemPSEAllocatedPowerValue	→	MirroredPSEAllocatedPowerValue
	aLldpXdot3RemPDRRequestedPowerValue	→	MirroredPDRRequestedPowerValueEcho
	<del>aLldpXdot3RemPowerType</del>		<del>pse_dll_power_type</del>
	<del>Value<sup>†</sup></del>		<del>Value<sup>†</sup></del>
	<del>-10</del>	→	<del>-01</del>
	<del>-00</del>	→	<del>-10</del>
	aLldpXdot3RemAutoclassRequest	→	MirroredPDAutoclassRequest

116 <sup>†</sup>Other value combinations mapping from aLldpXdot3RemPowerType\_mode(M) to pd\_dll\_power\_type or pse\_dll\_power\_type are not possible.

117 **145.5.3.6 State diagrams**  
 118 The general state change procedure for PSEs is shown in Figure 145–43.  
 119 **Make the following changes in Figure 145-43:**  
 120

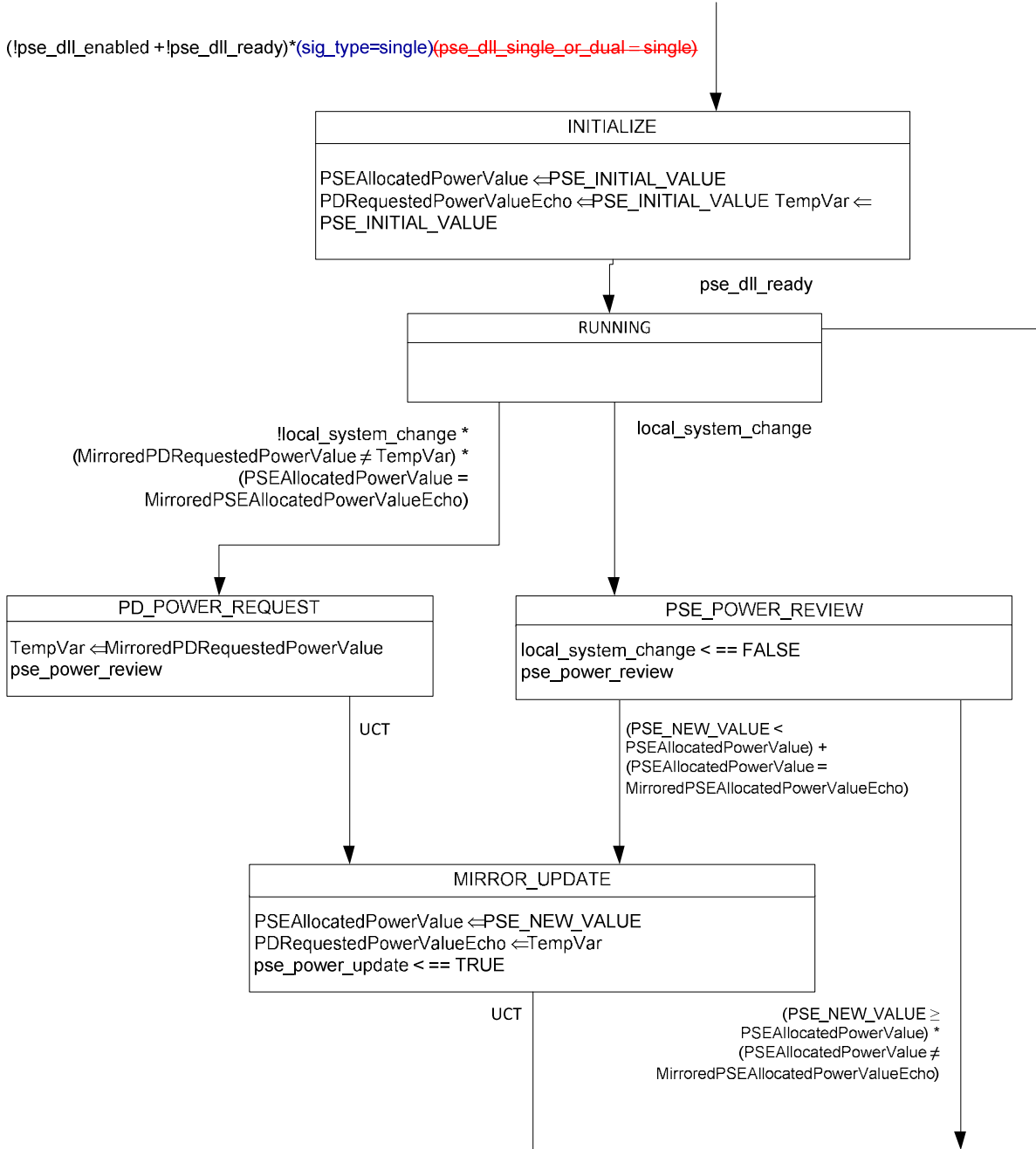


Figure 145–43—PSE power control state diagram

121  
 122  
 123  
 124

125 The general state change procedure for PDs is shown in Figure 145-44.

126 Make the following changes in Figure 145-44:

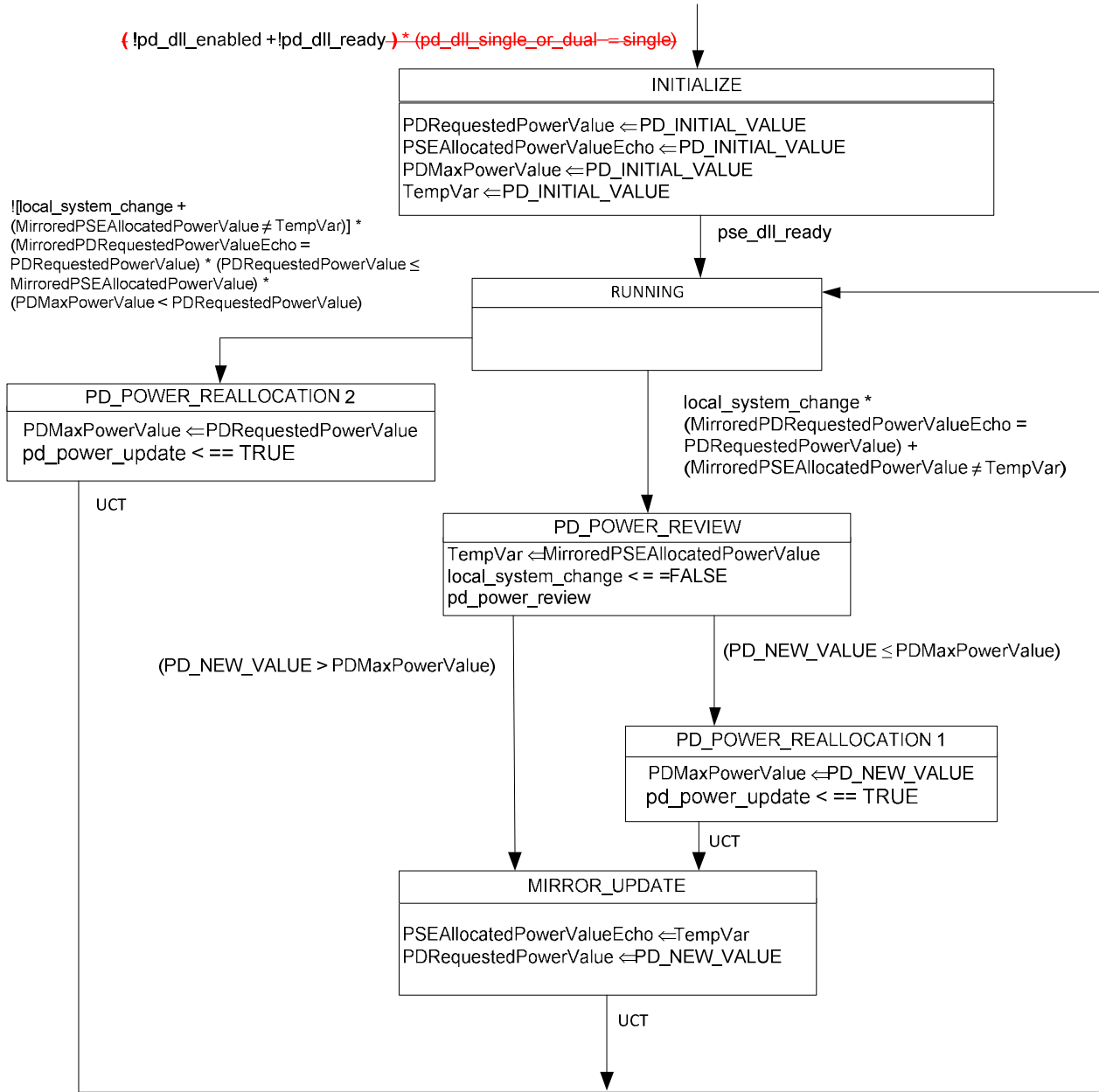


Figure 145-44—PD power control state diagram

127  
 128  
 129  
 130  
 131  
 132  
 133  
 134  
 135  
 136  
 137  
 138  
 139

**Not part of the baseline**

The DLL state machines for dual-signature PSE and PD were edited to have separate constants, variables and functions for each state machine.

**Editing Instructions**

1. PSE DLL state machine constants and variables have "Alt(X)" suffix and PD DLL state machine constants and variables have "mode(X)" suffix. Editor to verify implementation of this rule.
2. Dual-signature attributes names (e.g. aLldpXdot3RemPDRequestedPowerValueA) have been updated and are not contain "Alternative" or "Mode" they are ended only with "A" or "B" to indicate ModeA or ModeB or AltA or ALTB. Editor to verify implementation for clause 145.3, 30, and 79.
3. Editor to update subclause numbers.
4. Editor to implement yseboodt\_04\_0317.pdf if accepted (regarding the constants) and make the necessary changes for dual-signature DLL SM.

**145.5.3.7 Dual-signature system constants – PSE state diagram**

Variables PD\_DLLMAX\_VALUE\_M), PD\_INITIAL\_VALUE, and PSE\_INITIAL\_VALUE\_Alt(X), are is quantized to fit the available resolution. Additional information on power levels for Class 5 Classes 6 and 8 may be found in 145.3.8.2.1.

This is not part of the base line

The following items: PD\_DLLMAX\_VALUE\_mode(X), PD\_INITIAL\_VALUE\_mode(X) MirroredPSEAllocatedPowerValue\_mode(X), PDMaxPowerValue\_mode(X), PDRequestedPowerValue\_mode(X), PSEAllocatedPowerValueEcho\_mode(X) pd\_dll\_enabled, pd\_dll\_ready, pd\_power\_review\_mode(X) were updated with "mode(X)" and moved to dual-signature DLL PD state machine section.

~~PD\_DLLMAX\_VALUE\_mode(M)~~

~~This value is derived from pd\_max\_power\_(M) variable (145.3.3.9) described as follows:~~

~~pd\_max\_power PD\_DLLMAX\_VALUE\_mode(M)~~

~~1 39~~

~~2 65~~

~~3 130~~

~~4 255~~

~~5 355~~

~~PD\_INITIAL\_VALUE\_mode(M)~~

~~This value is derived as follows from the pd\_max\_power\_mode(M) variable (145.3.3.9) used in the PD state diagram (Figure 145-29):~~

~~pd\_max\_power\_mode(M) PD\_INITIAL\_VALUE\_mode(M)~~

~~1 ≤ 39~~

~~2 ≤ 65~~

~~3 ≤ 130~~

~~4 ≤ 255~~

~~5 ≤ 355~~

PSE\_INITIAL\_VALUE\_mode(M)\_Alt(X)

This value is derived as follows from pd\_allocated\_pwr\_pri or pd\_allocated\_pwr\_sec, as defined in 145.2.5.4, which is used in the state diagrams in 145.2.5.7:

pd\_allocated\_pwr\_pri PSE\_INITIAL\_VALUE\_mode(M)\_Alt(X)

pd\_allocated\_pwr\_sec

1 39

2 65

3 130

4 255

5 355

[single\_or\_dual is not used by the state machine]

~~single\_or\_dual~~

187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247

~~This variable indicates if the connected PD is a single signature PD or a dual signature PD.  
Values: single: A single signature PD configuration is connected to the PI. dual: A dual signature PD configuration is connected to the PI.~~

### 145.5.3.8 Dual-signature ~~system~~ Variables – PSE state diagram

The PSE power control state diagram (Figure 145-4347) and PD power control state diagram (Figure 145-44) use the following variables:

#### MirroredPDRequestedPowerValue\_Alt(X)mode(M)

The copy of the PD Requested Power Value field for Alternative mode(M) (X) in the Power Via MDI TLV that the PSE receives from the remote system in units of 0.1 W. This variable is mapped from the aLldpXdot3RemPDRequestedPowerValueA and aLldpXdot3RemPDRequestedPowerValueB attributes (30.12.3.1.18a and 30.12.3.1.18b).

When a PD mode is not active, the value is set to zero.

#### MirroredPDRequestedPowerValueEcho\_Alt(X)mode(M)

The copy of the PD Requested Power Value field for Alternative mode(M) (X) in the Power Via MDI TLV that the PD receives from the remote system. This variable is mapped from the aLldpXdot3RemPDRequestedPowerValueA and aLldpXdot3RemPDRequestedPowerValueB attributes (30.12.3.1.18a and 30.12.3.1.18b).

Values: 0 through 499.

When a PD mode is not active, the value is set to zero.

#### ~~MirroredPSEAllocatedPowerValue\_mode(M) \_\_\_\_\_~~

~~The copy of the PSE Allocated Power Value field for mode(M) in the Power Via MDI TLV that the PD receives from the remote system in units of 0.1 W. This variable is mapped from the aLldpXdot3RemPSEAllocatedPowerValue attribute ( ).~~

~~Values: 0 through 499.~~

~~When a PD mode is not active, the value is set to zero.~~

#### MirroredPSEAllocatedPowerValueEcho\_mode(M)\_Alt(X)

The copy of the PSE Allocated Power Value field for mode Alternative (X)(M) in the Power Via MDI TLV that the PSE receives from the remote system. This variable is mapped from the aLldpXdot3RemPSEAllocatedPowerValueA and aLldpXdot3RemPSEAllocatedPowerValueB attributes (30.12.3.1.18c and 30.12.3.1.18d).

Values: 0 through 499

When a PD mode is not active, the value is set to zero.

#### PDRequestedPowerValueEcho\_Alt(X)mode(M)

This variable is updated by the PSE state diagram. This variable maps into the aLldpXdot3LocPDRequestedPowerValueA and aLldpXdot3LocPDRequestedPowerValueB attribute (30.12.2.1.18a and 30.12.2.1.18b).

Values: 0 through 499

When a PD mode is not active, the value shall be set to zero.

#### ~~PDMaxPowerValue\_mode(M) \_\_\_\_\_~~

~~Integer that indicates the actual PD power value of the local system in units of 0.1 W. The actual PD power value for a PD is the maximum input average power (see 145.3.8.2) the PD ever draws under the current power allocation.~~

~~Values: 0 through 499.~~

~~When a PD mode is not active, the value shall be set to zero.~~

#### ~~PDRequestedPowerValue\_mode(M) \_\_\_\_\_~~

~~Integer that indicates the actual PD power value of the local system in units of 0.1 W. The actual PD power value for a PD is the maximum input average power (see 145.3.8.2) the PD ever draws under the current power allocation.~~

~~Values: 0 through 499. When a PD mode is not active, the value shall be set to zero.~~

#### PSEAllocatedPowerValue\_mode(M)\_Alt(X)

Integer that indicates the PSE allocated power value in the PSE in units of 0.1 W. The value is the maximum input average power (see 145.3.8.2) the PD ever draws. This variable maps to the aLldpXdot3LocPSEAllocatedPowerValueA and aLldpXdot3LocPSEAllocatedPowerValueB attribute (30.12.2.1.18c and 30.12.2.1.18d).

Values: 0 through 499

When a PD mode is not active, the value shall be set to zero.

248 ~~PSEAllocatedPowerValueEcho\_mode(M)~~

249 ~~This variable is updated by the PD state diagram. This variable maps into the~~

250 ~~aLldpXdot3LocPSEAllocatedPowerValue attribute~~

251 ~~()~~

252 ~~Values: 0 through 499.~~

253 ~~When a PD mode is not active, the value shall be set to zero.~~

254

255 TempVar\_(M) Alt(X)

256 A temporary variable used to store a Power Value in units of 0.1 W.

257 Values: 0 through 499.

258 ~~When a PD mode is not active, the value shall be set to zero.~~

259

260 local\_system\_change Alt(X)

261 An implementation-specific control variable that indicates that the local system wants to change

262 the allocated power value. In a PSE, this indicates it is going to change the power allocated to the

263 PD over Alternative X. ~~In a PD, this indicates it is going to request a new power allocation from the PSE.~~

264 Values:

265 FALSE: The local system does not wants to change the power allocation.

266 TRUE: The local system wants to change the power allocation.

267

268 ~~pd\_dll\_enabled~~

269 ~~A variable output by the PD state diagram (Figure 145-26) to indicate if the PD Data Link Layer~~

270 ~~classification mechanism is enabled.~~

271 ~~Values:~~

272 ~~FALSE: PD Data Link Layer classification is not enabled.~~

273 ~~TRUE: PD Data Link Layer classification is enabled.~~

274

275 ~~pd\_dll\_power\_type~~

276 ~~A Type 1 and Type 2 PSE state diagram control variable that indicates the Type of PD that is connected to the PSE as advertised~~

277 ~~through Data Link Layer classification. Type 3 and Type 4 PSE state diagrams do not use this variable.~~

278 ~~Values:-~~

279 ~~1: PD is a Type 1 PD (default).~~

280 ~~2: PD is a Type 2 PD.~~

281

282 ~~pd\_dll\_ready~~

283 ~~An implementation specific control variable that indicates that the PD has initialized Data Link~~

284 ~~Layer classification. This variable maps into the aLldpXdot3LocReady attribute (30.12.2.1.20).~~

285 ~~Values:~~

286 ~~FALSE: Data Link Layer classification has not completed initialization.~~

287 ~~TRUE: Data Link Layer classification has completed initialization.~~

288

289 pse\_dll\_enabled

290 A variable output by the PSE state diagram (Figure 145-13) to indicate if the PSE Data Link Layer classification

291 mechanism is enabled.

292 Values:

293 FALSE: PSE Data Link Layer classification is not enabled.

294 TRUE: PSE Data Link Layer classification is enabled.

295

296 ~~pse\_dll\_power\_type~~

297 ~~A control variable output by the PD power control state diagram, defined in Figure 145-44, that indicates the PSE Type as~~

298 ~~1 or 2, see 79.3.2.4.1.~~

299 ~~Values:-~~

300 ~~1: The PSE is a Type 1 PSE, for a Type 1 PSE.~~

301 ~~2: The PSE is a Type 2 PSE, for a Type 2, 3 and, 4 PSEs~~

302

303



304 pse\_dll\_ready Alt(X)  
305 An implementation-specific control variable that indicates that the PSE has initialized Data Link Layer classification over  
306 Alternative (X). This variable maps into the aLldpXdot3LocReady attribute (30.12.2.1.20).  
307 Values:  
308 FALSE: Data Link Layer classification has not completed initialization.  
309 TRUE: Data Link Layer classification has completed initialization.

311 ~~pse\_power\_type~~  
312 ~~———— A control variable that indicates to the PD the type of PSE by which it is being powered.~~  
313 ~~———— Values:~~  
314 ~~———— 1: The PSE is a Type 1 PSE.~~  
315 ~~———— 2: The PSE is a Type 2, Type 3, or Type 4 PSE.~~

317 ~~pd\_dll\_single\_or\_dual~~  
318 ~~A control variable output by PD power control state diagram, defined in Figure 145-44, that indicates if the PD is a~~  
319 ~~single-signature PD or a dual-signature PD. Type 3 and Type 4 PD state diagrams do not use this variable.~~  
320 ~~Values:~~  
321 ~~single: A single-signature PD configuration is connected to the PI.~~  
322 ~~dual: A dual-signature PD configuration is connected to the PI.~~

324 sig\_type  
325 A variable generated from the do\_cxn\_chk function in Figure 145-13 which indicates if the PSE is connected to a single-signature  
326 PD or dual-signature PD.  
327 Values:  
328 Invalid: Neither single-signature PD nor dual-signature PD connection check signature has been found. This include an  
329 open circuit condition.  
330 single: A single-signature PD configuration is connected to the PI.  
331 dual: A dual-signature PD configuration is connected to the PI.

334 ~~pse\_dll\_single\_or\_dual~~  
335 ~~A control variable output by PSE power control state diagram defined in Figure 145-43 (generated from the~~  
336 ~~do\_cxn\_check function of the Type 3 and Type 4 PSE state diagram in Figure 145-13 which indicates if the PSE is~~  
337 ~~connected to a single-signature PD or dual-signature PD.~~  
338 ~~Values:~~  
339 ~~invalid: ————— Neither a single-signature PD nor a dual-signature PD connection check signature has been~~  
340 ~~found. This includes an open circuit condition.~~  
341 ~~Single: ————— A single-signature PD configuration is connected to the PI.~~  
342 ~~dual: ————— A dual-signature PD configuration is connected to the PI.~~

### 344 ~~145.5.3.9 Dual-signature system Functions~~

### 345 145.5.3.9 Dual-signature Functions – PSE state diagram

347 pse\_power\_review Alt(X)  
348 This function evaluates the power allocation or budget of the PSE based on local system changes. The function returns the  
349 following variables:  
350 PSE\_NEW\_VALUE Alt(X):  
351 The new maximum power value that the PSE expects the PD to draw in units of 0.1 W.

352 ~~pd\_power\_review~~  
353 ~~———— This function evaluates the power requirements of the PD based on local system changes and/or~~  
354 ~~changes in the PSE allocated power value. The function returns the following variables:~~  
355 ~~———— PD\_NEW\_VALUE:~~  
356 ~~———— The new maximum power value that the PD wants to draw in units of 0.1 W.~~

360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415

**145.5.3.X1 Dual-signature constants – PD state diagram**

Variables PD\_DLLMAX\_VALUE\_mode(X) and PD\_INITIAL\_VALUE\_mode(X), are quantized to fit the available resolution. Additional information on power level for Class 5 may be found in 145.3.8.2.1.

PD\_DLLMAX\_VALUE\_mode(X)

This value is derived from pd\_max\_power\_mode(X) variable (145.3.3.9) described as follows:

<u>pd_max_power_mode(X)</u>	<u>PD_DLLMAX_VALUE_mode(X)</u>
<u>1</u>	<u>39</u>
<u>2</u>	<u>65</u>
<u>3</u>	<u>130</u>
<u>4</u>	<u>255</u>
<u>5</u>	<u>355</u>

PD\_INITIAL\_VALUE\_mode(X)

This value is derived as follows from the pd\_max\_power\_mode(X) variable (145.3.3.9) used in the PD state diagram (Figure 145-29):

<u>pd_max_power_mode(X)</u>	<u>PD_INITIAL_VALUE_mode(X)</u>
<u>1</u>	<u>≤39</u>
<u>2</u>	<u>≤ 65</u>
<u>3</u>	<u>≤ 130</u>
<u>4</u>	<u>≤255</u>
<u>5</u>	<u>≤355</u>

**145.5.3.X2 Dual-signature Variables – PD state diagram**

The PD power control state diagram (Figure 145-48) use the following variables:

MirroredPDRequestedPowerValueEcho\_mode(X)

The copy of the PD Requested Power Value filed for mode(X) in the Power Via MDI TLV that the PD receives from the remote system. This variable is mapped from the aLldpXdot3RemPDRequestedPowerValueModeA and aLldpXdot3RemPDRequestedPowerValueModeB attributes (30.12.3.1.18a and 30.12.3.1.18b).

Values: 0 through 499.

When a PD mode is not active, the value is set to zero.

MirroredPSEAllocatedPowerValue\_mode(X)

The copy of the PSE Allocated Power Value field for mode(X) in the Power Via MDI TLV that the PD receives from the remote system in units of 0.1 W. This variable is mapped from the aLldpXdot3RemPSEAllocatedPowerValueA and aLldpXdot3RemPSEAllocatedPowerValueB attributes (30.12.3.1.18c and 30.12.3.1.18d).

Values: 0 through 499

When a PD mode is not active, the value is set to zero.

PDMaxPowerValue\_mode(X)

Integer that indicates the actual PD power value of the local system in units of 0.1 W. The actual PD power value for a PD is the maximum input average power (see 145.3.8.2) the PD ever draws under the current power allocation.

Values: 0 through 499

When a PD mode is not active, the value shall be set to zero.

PDRequestedPowerValue\_mode(X)

Integer that indicates the actual PD power value of the local system in units of 0.1 W. The actual PD power value for a PD is the maximum input average power (see 145.3.8.2) the PD ever draws under the current power allocation.

Values: 0 through 499

When a PD mode is not active, the value shall be set to zero.

416  
417 PSEAllocatedPowerValue\_mode(X)  
418 Integer that indicates the PSE allocated power value in the PSE in units of 0.1 W. The value is the maximum input  
419 average power (see 145.3.8.2) the PD ever draws. This variable maps to the aLldpXdot3LocPSEAllocatedPowerValueA  
420 and aLldpXdot3LocPSEAllocatedPowerValueB attribute (30.12.2.1.18c and 30.12.2.1.18d).  
421 Values: 0 through 499  
422 When a PD mode is not active, the value shall be set to zero.  
423  
424 PSEAllocatedPowerValueEcho\_mode(X)  
425 This variable is updated by the PD state diagram. This variable maps into the  
426 aLldpXdot3LocPSEAllocatedPowerValueA and aLldpXdot3LocPSEAllocatedPowerValueB attribute  
427 (30.12.2.1.18c and 30.12.2.1.18d).  
428 Values: 0 through 499.  
429 When a PD mode is not active, the value shall be set to zero.  
430  
431 TempVar\_mode(X)  
432 A temporary variable used to store a Power Value in units of 0.1 W.  
433 Values: 0 through 499.  
434 ~~When a PD mode is not active, the value shall be set to zero.~~  
435  
436 local\_system\_change\_mode(X)  
437 An implementation-specific control variable that indicates that the local system wants to change  
438 the requested power value. In a PD, this indicates it is going to change the power requested by the PD over  
439 mode X.  
440 Values:  
441 FALSE: The local system does not want to change the requested power.  
442 TRUE: The local system wants to change the requested power.  
443  
444 pd\_dll\_enabled\_mode(X)  
445 A variable output by the PD state diagram (Figure 145–26) to indicate if the PD Data Link Layer  
446 classification mechanism is enabled over mode (X).  
447 Values:  
448 FALSE: PD Data Link Layer classification is not enabled.  
449 TRUE: PD Data Link Layer classification is enabled.  
450  
451  
452 pd\_dll\_ready\_mode(X)  
453 An implementation-specific control variable that indicates that the PD has initialized Data Link  
454 Layer classification over mode (X). This variable maps into the aLldpXdot3LocReady attribute (30.12.2.1.20).  
455 Values:  
456 FALSE: Data Link Layer classification has not completed initialization.  
457 TRUE: Data Link Layer classification has completed initialization.  
458  
459 **145.5.3.X3 Dual-signature Functions – PD state diagram**  
460  
461 pd\_power\_review\_mode(X)  
462 This function evaluates the power requirements of the PD based on local system changes and/or  
463 changes in the PSE allocated power value. The function returns the following variables:  
464 PD\_NEW\_VALUE\_mode(X):  
465 The new maximum power value that the PD wants to draw in units of 0.1 W.  
466  
467  
468

469  
470

**Make the following changes to Table 145-40:**

This is not part of the base line

- New variables were added to clause 145.5, clause 30 and clause 79 to support dual-signature DLL state machine and related TLVs per the following concept based on single-signature DLL state machine with the relevant changes for dual signature. The concept used for single signature is based on the following example:  
**State diagram variable e.g. PDRequestedPowerValueEcho** from Figure 145-43 PSE DLL state diagram is shown in Table 145-39. Table 145-39 shows that **PDRequestedPowerValueEcho** is mapped to **aLldpXdot3LocPDRequestedPowerValue**. **PDRequestedPowerValueEcho** is defined in the variable list of the state diagram.  
**aLldpXdot3LocPDRequestedPowerValue** is defined in clause 30 and in clause 79.
- In Table 145-40 the column **Attribute** doesn't have "mode" or "Alt" designation due to shared use in PSE and PD. See Lennart comment marked as mode\_Alt\_shared.

471

472 **Table 145-40—Attribute to state diagram variable cross-reference**

Entity	Attribute	Mapping	State diagram variable
<b>oLldpXdot3LocSystemsGroup Object Class</b>			
PSE	<a href="#">aLldpXdot3LocPDRequestedPowerValueA</a>	<==	PDRequestedPowerValueEcho_modeALT(MX=A)
	<a href="#">aLldpXdot3LocPDRequestedPowerValueB</a>	<==	PDRequestedPowerValueEcho_ALT(X=B)
	<a href="#">aLldpXdot3LocPSEAllocatedPowerValueA</a>	<==	PSEAllocatedPowerValue_mode(M) Alt(X=A)
	<a href="#">aLldpXdot3LocPSEAllocatedPowerValueB</a>	<==	PSEAllocatedPowerValue_Alt(X=B)
	<a href="#">aLldpXdot3LocReadyA</a> <a href="#">aLldpXdot3LocReadyB</a>	<==	pse_dll_ready_Alt(X=A) pse_dll_ready_Alt(X=A)
PD	<a href="#">aLldpXdot3LocPDRequestedPowerValueA</a>	<==	PDRequestedPowerValue_mode(MX=A)
	<a href="#">aLldpXdot3LocPDRequestedPowerValueB</a>	<==	PDRequestedPowerValue_mode(X=B)
	<a href="#">aLldpXdot3LocPSEAllocatedPowerValueA</a>	<==	PSEAllocatedPowerValueEcho_mode(MX=A)
	<a href="#">aLldpXdot3LocPSEAllocatedPowerValueB</a>	<==	PSEAllocatedPowerValueEcho_mode(X=B)
	<a href="#">aLldpXdot3LocReadyA</a> <a href="#">aLldpXdot3LocReadyB</a>	<==	pd_dll_ready_mode(X) pd_dll_ready_mode(X)
<b>oLldpXdot3RemSystemsGroup Object Class</b>			
PSE	<a href="#">aLldpXdot3RemPDRequestedPowerValueA</a>	→	MirroredPDRequestedPowerValue_modeALT(MX=A)
	<a href="#">aLldpXdot3RemPDRequestedPowerValueB</a>	→	MirroredPDRequestedPowerValue_ALT(X=A)
	<a href="#">aLldpXdot3RemPSEAllocatedPowerValueA</a>	→	MirroredPSEAllocatedPowerValueEcho_mode(M) Alt(X=A)
	<a href="#">aLldpXdot3RemPSEAllocatedPowerValueB</a>	→	MirroredPSEAllocatedPowerValueEcho_Alt(X=B)
	<del>aLldpXdot3RemPowerType- Value<sup>†</sup> -11 -01</del>	<del>→ →</del>	<del>pd_dll_power_type- Value<sup>†</sup> -01 -10</del>
PD	<a href="#">aLldpXdot3RemPSEAllocatedPowerValueA</a>	→	MirroredPSEAllocatedPowerValue_mode(X=A)
	<a href="#">aLldpXdot3RemPSEAllocatedPowerValueB</a>	→	MirroredPSEAllocatedPowerValue_mode(X=B)
	<a href="#">aLldpXdot3RemPDRequestedPowerValueA</a>	→	MirroredPDRequestedPowerValueEcho_mode(X=A)
	<a href="#">aLldpXdot3RemPDRequestedPowerValueB</a>	→	MirroredPDRequestedPowerValueEcho_mode(X=B)
	<del>aLldpXdot3RemPowerType Value<sup>†</sup> -10 -00</del>	<del>→ →</del>	<del>pse_dll_power_type Value<sup>†</sup> -01 -10</del>

473  
474  
475  
476  
477

<sup>†</sup>Other value combinations mapping from aLldpXdot3RemPowerType\_mode(M) to pd\_dll\_power\_type or pse\_dll\_power\_type are not possible.

478 *Update the following PSE state diagram Figure 145-47 per the following changes. In addition:*  
 479 *-Verify that all variable extensions are with “\_Alt(X)” only.*

Not part of the baseline  
 Before the PD is going to IDLE and its pd\_dll\_enabled\_Alt(X) became FALSE, The TLV field need to be filled with value=0. This was implemented by adding new state IDLE.

480 **145.5.3.10 Dual-signature State diagrams**

481 The general state change procedure for PSEs is shown in Figure 145-47.

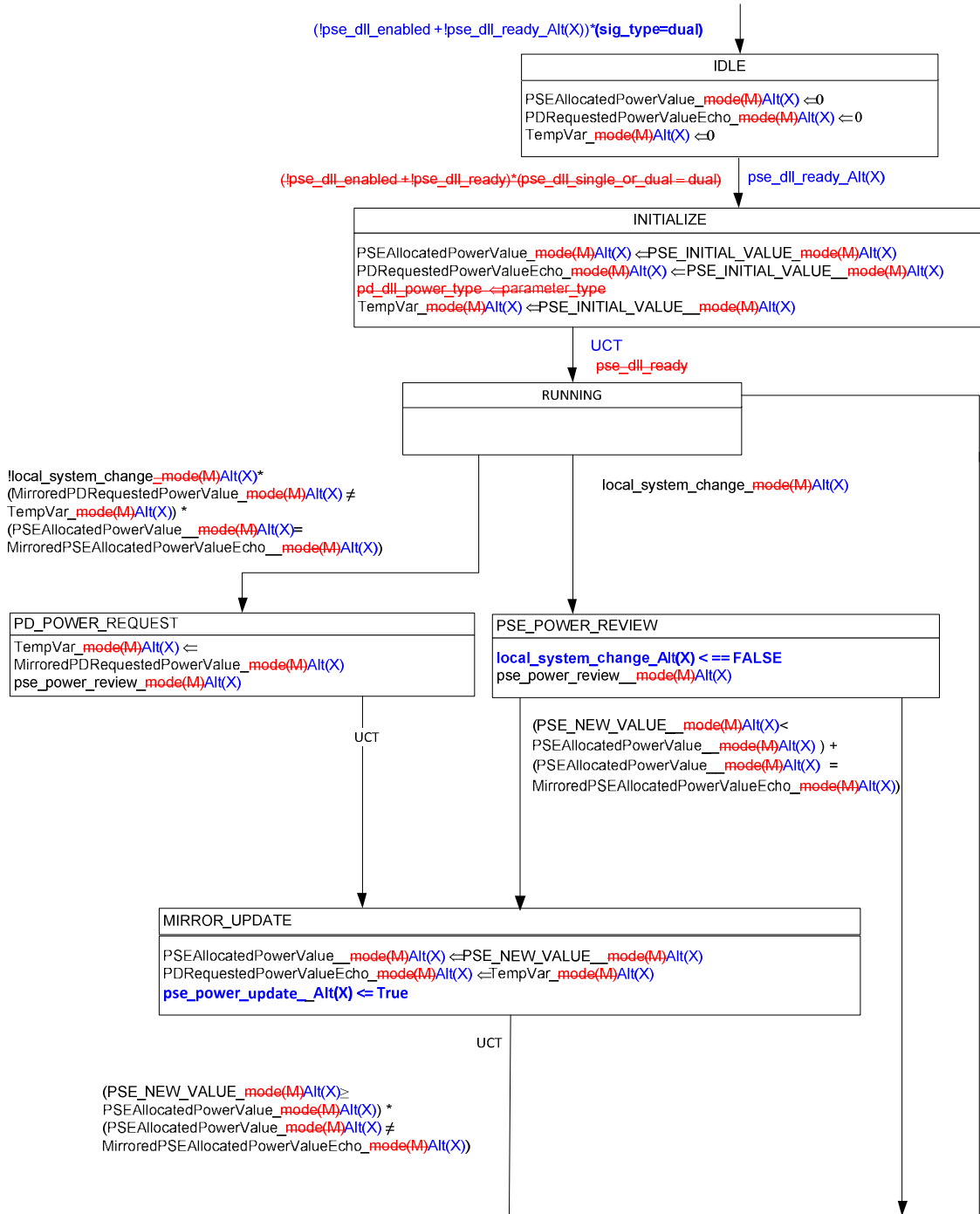


Figure 145-47—PSE power control state diagram Alternative (X) when connected to a dual-signature PD mode (X)

482  
 483  
 484

485 *Update the following PSE state diagram Figure 145-48. In addition:*  
 486 *-Verify that all variable extensions are with “\_mode(X)” only.*

487 The general state change procedure for PDs is shown in Figure 145-48.

Not part of the baseline

Before the PD is going to IDLE and its pd\_dll\_enabled\_mode(X) became FALSE, The TLV field need to be filled with value=0. This was implemented by adding new state IDLE.

488

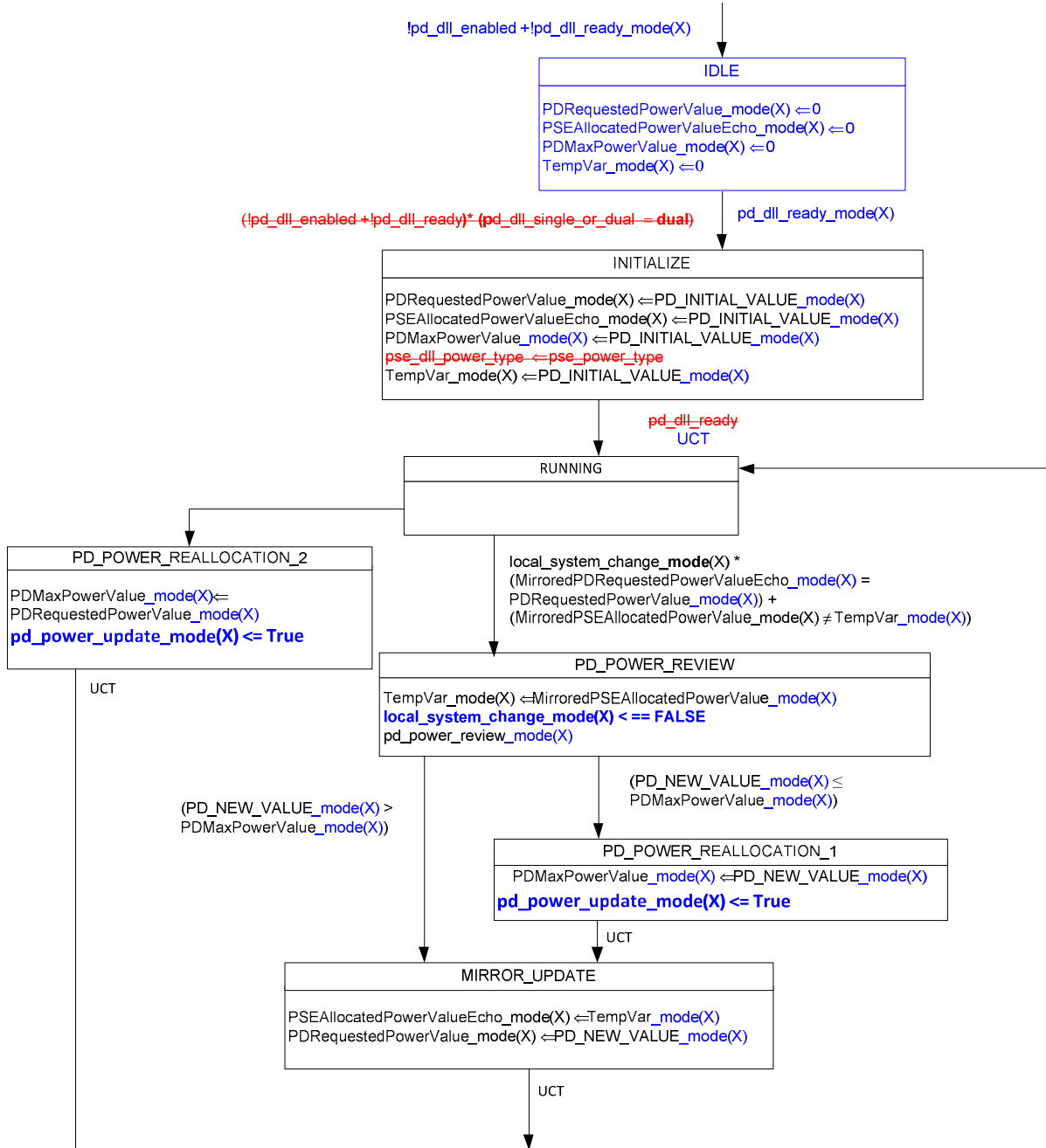


Figure 145-48—Dual-signature PD power control state diagram mode(X)

489  
 490  
 491

492 **145.5.4 State change procedure across a link ([single-signature](#))**  
493 .....  
494 .....  
495 **145.5.4.1 PSE state change procedure across a link (single-signature)**

496 .....  
497 .....  
498 **145.5.4.2 PD state change procedure across a link (single-signature)**  
499 .....

500  
501 *-Make the following additions/changes:*  
502 *-In addition verify the following if I missed some of it ☺.:*  
503 *1. Whenever applicable, replace from index “M” to index “X”.*  
504 *2. Please verify that PSE State machine variable e.g. PSEAllocatedPowerValue\_Alt(X) will have only \_Alt(X)*  
505 *extension and not \_mode(X) extension.*  
506 *3. Please verify that PD State machine variable e.g. PDMaxPowerValue\_mode(X), will have only \_mode(X) extension*  
507 *and not \_Alt(X) extension or anything else.*  
508 *4. Attributes e.g. aLldpXdot3LocPDRequestedPowerValueX will appear without the “Mode” or “mode” or “Alt” or*  
509 *“Alternative” extensions e.g. aLldpXdot3LocPDRequestedPowerValueA or aLldpXdot3LocPDRequestedPowerValueB*  
510 *etc.*

511 **[145.5.5 State change procedure across a link \(dual-signature\)](#)**

512 The PSE and PD utilize the LLDPDUs to advertise their various attributes to the other entity.

513  
514 The PD may request a new power value through the aLldpXdot3LocPDRequestedPowerValueA or  
515 aLldpXdot3LocPDRequestedPowerValueB (30.12.2.1.18a and 30.12.2.1.18b) attribute in the oLldpXdot3LocSystemsGroup  
516 object class. The request appears to the PSE as a change to the aLldpXdot3RemPDRequestedPowerValueA and  
517 aLldpXdot3RemPDRequestedPowerValueB (30.12.3.1.18a and 30.12.3.1.18b) attribute in the oLldpXdot3RemSystemsGroup  
518 object class.

519  
520 The PSE responds to the PD’s request through the aLldpXdot3LocPSEAllocatedPowerValueA and  
521 aLldpXdot3LocPSEAllocatedPowerValueB (30.12.2.1.18c and 30.12.2.1.18d) attribute in the oLldpXdot3LocSystemsGroup  
522 object class. The PSE also copies the value of the aLldpXdot3RemPDRequestedPowerValueA and  
523 aLldpXdot3RemPDRequestedPowerValueB (30.12.3.1.18a and 30.12.3.1.18b) in the oLldpXdot3RemSystemsGroup object class  
524 to the aLldpXdot3LocPDRequestedPowerValueA and aLldpXdot3LocPDRequestedPowerValueB (30.12.2.1.Z1 and  
525 30.12.2.1.Z2) in the oLldpXdot3LocSystemsGroup object class. This appears to the PD as a change to the  
526 aLldpXdot3RemPSEAllocatedPowerValueA and aLldpXdot3RemPSEAllocatedPowerValueB (30.12.3.1.18c and 30.12.3.1.18d)  
527 attribute in the oLldpXdot3RemSystemsGroup object class.

528  
529 The PSE may allocate a new power value through the aLldpXdot3LocPSEAllocatedPowerValueA and  
530 aLldpXdot3LocPSEAllocatedPowerValueB (30.12.2.1.18c and 30.12.2.1.18d) attribute in the oLldpXdot3LocSystemsGroup  
531 object class. The request appears to the PD as a change to the aLldpXdot3RemPSEAllocatedPowerValueA and  
532 aLldpXdot3RemPSEAllocatedPowerValueB (30.12.3.1.18c and 30.12.3.1.18d) attribute in the oLldpXdot3RemSystemsGroup  
533 object class. The PD responds to a PSE’s request through the aLldpXdot3LocPDRequestedPowerValueA and  
534 aLldpXdot3LocPDRequestedPowerValueB (30.12.2.1.18a and 30.12.2.1.18b) attribute in the oLldpXdot3LocSystemsGroup  
535 object class. The PD also copies the value of the aLldpXdot3RemPSEAllocatedPowerValueA and  
536 aLldpXdot3RemPSEAllocatedPowerValueB (30.12.3.1.18c and 30.12.3.1.18d) attribute in the oLldpXdot3RemSystemsGroup  
537 object class to the aLldpXdot3LocPSEAllocatedPowerValueA and aLldpXdot3LocPSEAllocatedPowerValueB (30.12.2.1.18c  
538 and 30.12.2.1.18d) attribute in the oLldpXdot3LocSystemsGroup object class. This appears to the PSE as a change to the  
539 aLldpXdot3RemPDRequestedPowerValueA and aLldpXdot3RemPDRequestedPowerValueB (30.12.3.1.18a and 30.12.3.1.18b)  
540 attribute in the oLldpXdot3RemSystemsGroup object class.

541 The state diagrams describe the behavior above.

542



543 **145.5.5.1 PSE state change procedure across a link (dual-signature)**

544 A PSE is considered to be in sync with the PD when the value of PSEAllocatedPowerValue~~\_mode(M)~~ Alt(X) matches the value  
 545 of MirroredPSEAllocatedPowerValueEcho~~\_mode(M)~~ Alt(X). When the PSE is not in sync with the PD, the PSE is allowed to  
 546 change its power allocation.

547 During normal operation, the PSE is in the RUNNING state. If the PSE wants to initiate a change in the PD allocation, the  
 548 local\_system\_change Alt(X) ~~\_mode(M)~~ is asserted and the PSE enters the PSE\_POWER\_REVIEW state, where a new power  
 549 allocation value, PSE\_NEW\_VALUE Alt(X) ~~\_mode(M)~~, is computed. If the PSE is in sync with the PD or if  
 550 PSE\_NEW\_VALUE~~\_mode(M)~~ Alt(X) is smaller than PSEAllocatedPowerValue~~\_mode(M)~~ Alt(X), it enters the  
 551 MIRROR\_UPDATE state where PSE\_NEW\_VALUE~~\_mode(M)~~ Alt(X) is assigned to  
 552 PSEAllocatedPowerValue~~\_mode(M)~~ Alt(X). It also updates PDRRequestedPowerValueEcho Alt(X) ~~\_mode(M)~~ and returns to the  
 553 RUNNING state.

554 If the PSE’s previously stored MirroredPDRRequestedPowerValue Alt(X) ~~\_mode(M)~~ changes, a request by the PD to change its  
 555 power allocation is recognized. It entertains this request only when it is in sync with the PD. The PSE examines the request by  
 556 entering the PD\_POWER\_REQUEST state. A new power allocation value, PSE\_NEW\_VALUE~~\_mode(M)~~ Alt(X), is computed.  
 557 It then enters the MIRROR\_UPDATE state where PSE\_NEW\_VALUE~~\_mode(M)~~ Alt(X) is assigned to  
 558 PSEAllocatedPowerValue~~\_mode(M)~~ Alt(X). It also updates PDRRequestedPowerValueEcho Alt(X) ~~\_mode(M)~~ and returns to the  
 559 RUNNING state.

560 **145.5.4.4 145.5.2 PD state change procedure across a link (dual-signature)**

561 A PD is considered to be in sync with the PSE when the value of PDRRequestedPowerValue~~\_mode(M)~~(X) matches the value of  
 562 MirroredPDRRequestedPowerValueEcho~~\_mode(M)~~(X). The PD is not allowed to change its maximum power draw or the requested  
 563 power value when it is not in sync with the PSE.

564 During normal operation, the PD is in the RUNNING state. If the PD’s previously stored  
 565 MirroredPSEAllocatedPowerValue~~\_mode(M)~~ is changed or local\_system\_change~~\_mode(M)~~(X) is asserted by the PD so as to  
 566 change its power allocation, the PD enters the PD\_POWER\_REVIEW state. In this state, the PD evaluates the change and  
 567 generates an updated power value called PD\_NEW\_VALUE~~\_mode(M)~~(X). If PD\_NEW\_VALUE~~\_mode(M)~~(X) is less than  
 568 PDMaxPowerValue~~\_mode(M)~~(X), it updates PDMaxPowerValue~~\_mode(M)~~(X) in the PD\_POWER\_REALLOCATION1 state.  
 569 The PD finally enters the MIRROR\_UPDATE state where PD\_NEW\_VALUE~~\_mode(M)~~(X) is assigned to  
 570 PDRRequestedPowerValue~~\_mode(M)~~(X). It also updates PSEAllocatedPowerValueEcho~~\_mode(M)~~(X) and returns to the RUNNING  
 571 state.

572 In the above flow, if PD\_NEW\_VALUE~~\_mode(M)~~(X) is greater than PDMaxPowerValue~~\_mode(M)~~(X), the PD waits until it is in  
 573 sync with the PSE and the PSE grants the higher power value. When this condition arises, the PD enters the PD\_POWER  
 574 REALLOCATION\_2 state. In this state, the PD assigns PDMaxPowerValue~~\_mode(M)~~(X) to  
 575 PDRRequestedPowerValue~~\_mode(M)~~(X) and returns to the RUNNING state.

576 **145.5.5 145.5.6 Autoclass**

577 A PSE can indicate it supports DLL Autoclass by means of the.....

578 **Make the following changes to clause 79:**

579  
 580 **Table 79–9—IEEE 802.3 Organizationally Specific TLV/LLDP Local System Group managed object class cross references**

TLV name	TLV variable	LLDP Local System Group managed object class attribute
Power via MDI	PD requested power value mode A(A)	aLldpXdot3LocPDRRequestedPowerValueA <del>ModeA</del>
	PD requested power value mode B(B)	aLldpXdot3LocPDRRequestedPowerValueB <del>ModeB</del>
	PSE allocated power value Alternative A	aLldpXdot3LocPSEAllocatedPowerValueA <del>AlternativeA</del>
	PSE allocated power value <del>Alternative A</del> Alternative B	aLldpXdot3LocPSEAllocatedPowerValueB <del>AlternativeB</del>

581  
 582 **Table 79–10—IEEE 802.3 Organizationally Specific TLV/LLDP Remote System Group managed object class cross references**



TLV name	TLV variable	LLDP Remote System Group managed object class attribute
Power via MDI	PD requested power value mode A	aLldpXdot3RemPDRRequestedPowerValue <del>A</del> <u>ModeA</u>
	PD requested power value mode B	aLldpXdot3RemPDRRequestedPowerValue <del>B</del> <u>ModeB</u>
	PSE allocated power value Alternative A	aLldpXdot3RemPSEAllocatedPowerValue <del>A</del> <u>AlternativeA</u>
	PSE allocated power value <del>Alternative A</del> <u>Alternative B</u>	aLldpXdot3RemPSEAllocatedPowerValue <del>B</del> <u>AlternativeB</u>

584

**Add the following definitions to clause 30:**

585

**30.12.2.1.18a aLldpXdot3LocPDRRequestedPowerValue~~A~~ModeA**

586

ATTRIBUTE

587

APPROPRIATE SYNTAX:

588

INTEGER

589

BEHAVIOUR DEFINED AS:

590

A GET attribute that returns the PD requested power value for the Mode A pairset in units of 0.1 W, ~~as defined in~~

591

~~Equation (79-1), where aLldpXdot3LocPDRRequestedPowerValueModeA is X~~. For a PD, it is the power value that the PD has currently requested from the remote system for the Mode A pairset. For a PSE, it is the power value for the ~~Mode~~Alternative A

592

pairset that the PSE mirrors back to the remote system.;

593

594

595

**30.12.2.1.18b aLldpXdot3LocPDRRequestedPowerValue~~B~~ModeB**

596

ATTRIBUTE

597

APPROPRIATE SYNTAX:

598

INTEGER

599

BEHAVIOUR DEFINED AS:

600

A GET attribute that returns the PD requested power value for the Mode B pairset in units of 0.1 W, ~~as defined in~~

601

~~Equation (79-1), where aLldpXdot3LocPDRRequestedPowerValueModeB is X~~. For a PD, it is the power value that the PD has currently requested from the remote system for the Mode B pairset. For a PSE, it is the power value for the ~~Alternative~~Mode B

602

pairset that the PSE mirrors back to the remote system.;

603

604

605

**30.12.2.1.18c aLldpXdot3LocPSEAllocatedPowerValueA**

606

ATTRIBUTE

607

APPROPRIATE SYNTAX:

608

INTEGER

609

BEHAVIOUR DEFINED AS:

610

A GET attribute that returns the PSE allocated power value for the Alternative A pairset in units of 0.1 W, ~~as defined in~~

611

~~Equation (79-2), where aLldpXdot3LocPSEAllocatedPowerValueAlternativeA is X~~. For a PSE, it is the power value for the Alternative A pairset that the PSE has currently allocated to the remote system. For a PD, it is the power value for the ~~Alternative~~mode A

612

pairset that the PD mirrors back to the remote system.;

613

614

615

616

**30.12.2.1.18d aLldpXdot3LocPSEAllocatedPowerValue~~B~~AlternativeB**

617

ATTRIBUTE

618

APPROPRIATE SYNTAX:

619

INTEGER

620

BEHAVIOUR DEFINED AS:

621

A GET attribute that returns the PSE allocated power value for the Alternative B pairset in units of 0.1 W, ~~as defined in~~

622

~~Equation (79-2), where aLldpXdot3LocPSEAllocatedPowerValueAlternativeB is X~~. For a PSE, it is the power value for the Alternative B pairset that the PSE has currently allocated to the remote system. For a PD, it is the power value for the ~~Alternative~~mode B

623

pairset that the PD mirrors back to the remote system.;

624

625

626

**30.12.3.1.18a aLldpXdot3RemPDRRequestedPowerValue~~A~~ModeA**

627

ATTRIBUTE

628

APPROPRIATE SYNTAX:

629

INTEGER

630

BEHAVIOUR DEFINED AS:

631

A GET attribute that returns the PD requested power value for the Mode A pairset that was used by the remote system to compute the power value that is ~~has~~ currently allocated to the PD. For a PSE, it is the PD requested power value for the ~~Mode~~Alternative A

632

pairset received from the remote system. The definition and encoding of PD requested power value for the Mode A pairset is the same as described in aLldpXdot3LocPDRRequestedPowerValue~~A~~ModeA (30.12.2.1.18a).;

633

634

635

636

**30.12.3.1.18b aLldpXdot3RemPDRRequestedPowerValue~~B~~ModeB**

637 ATTRIBUTE  
638 APPROPRIATE SYNTAX:  
639 INTEGER  
640 BEHAVIOUR DEFINED AS:  
641 A GET attribute that returns the PD requested power value for the Mode B pairset that was used by the remote system to  
642 compute the power value that is ~~has~~ currently allocated to the PD. For a PSE, it is the PD requested power value for the ~~Mode~~  
643 [Alternative B](#) pairset received from the remote system. The definition and encoding of PD requested power value for the Mode B  
644 pairset is the same as described in aLldpXdot3LocPDRrequestedPowerValueBModeB (30.12.2.1.18b).;  
645  
646 **30.12.3.1.18c aLldpXdot3RemPSEAllocatedPowerValueAAlternativeA**  
647 ATTRIBUTE  
648 APPROPRIATE SYNTAX:  
649 INTEGER  
650 BEHAVIOUR DEFINED AS:  
651 A GET attribute that returns the PSE allocated power value for the Alternative A pairset received from the remote system.  
652 For a PSE, it is the PSE allocated power value for the Alternative A pairset that was used by the remote system to compute the  
653 power value that it has currently requested from the PSE. For a PD, it is the PSE allocated power value for the ~~Alternative-mode~~ [A](#)  
654 pairset received from the remote system. The definition and encoding of PSE allocated power value for the Alternative A pairset is  
655 the same as described in aLldpXdot3LocPSEAllocatedPowerValueAAlternativeA (30.12.2.1.18c).;  
656  
657 **30.12.3.1.18d aLldpXdot3RemPSEAllocatedPowerValueBAlternativeB**  
658 ATTRIBUTE  
659 APPROPRIATE SYNTAX:  
660 INTEGER  
661 BEHAVIOUR DEFINED AS:  
662 A GET attribute that returns the PSE allocated power value for the Alternative B pairset received from the remote system.  
663 For a PSE, it is the PSE allocated power value for the Alternative B pairset that was used by the remote system to compute the  
664 power value that it has currently requested from the PSE. For a PD, it is the PSE allocated power value for the ~~mode~~ [Alternative B](#)  
665 pairset received from the remote system. The definition and encoding of PSE allocated power value for the Alternative B pairset is  
666 the same as described in aLldpXdot3LocPSEAllocatedPowerValueBAlternativeB (30.12.2.1.18d).;  
667  
668 **30.12.2.1.18g aLldpXdot3LocPowerClassxAModeA**  
669 ATTRIBUTE  
670 APPROPRIATE SYNTAX:  
671 An ENUMERATED VALUE that has one of the following entries:  
672 pClassPSE PSE  
673 pClassPD PD  
674 BEHAVIOUR DEFINED AS:  
675 A read-only value that identifies the port Class of the given port associated with the local system over Alternative A for a  
676 PSE or over mode A for a PD.;  
677  
678 **30.12.2.1.18h aLldpXdot3LocPowerClassxBModeB**  
679 ATTRIBUTE  
680 APPROPRIATE SYNTAX:  
681 An ENUMERATED VALUE that has one of the following entries:  
682 pClassPSE PSE  
683 pClassPD PD  
684 BEHAVIOUR DEFINED AS:  
685 A read-only value that identifies the port Class of the given port associated with the local system over Alternative B for a  
686 PSE or over mode B for a PD.;  
687  
688 **30.12.3.1.18g aLldpXdot3RemPowerClassxAModeA**  
689 ATTRIBUTE  
690 APPROPRIATE SYNTAX:  
691 An ENUMERATED VALUE that has one of the following entries:  
692 pClassPSE PSE  
693 pClassPD PD  
694 BEHAVIOUR DEFINED AS:  
695 A read-only value that identifies the port Class of the given port associated with the local system over Alternative B for a  
696 PSE or over mode B for a PD.;

696 **30.12.3.1.18h aLldpXdot3RemPowerClassxBModeB**

697 ATTRIBUTE

698 APPROPRIATE SYNTAX:

699 An ENUMERATED VALUE that has one of the following entries:

700 pClassPSE PSE

701 pClassPD PD

702 BEHAVIOUR DEFINED AS:

703 A read-only value that identifies the port Class of the given port associated with the local system [over Alternative B for a](#)  
 704 [PSE or over mode B for a PD.](#);

705

706 **Add the following lines to Table 30-7**

707 aLldpXdot3RemPDRRequestedPowerValueA, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

708 aLldpXdot3RemPDRRequestedPowerValueB, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

709 aLldpXdot3RemPSEAllocatedPowerValueA, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

710 aLldpXdot3RemPSEAllocatedPowerValueB, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

711

712 **Change the following in Table 30-7**

713 aLldpXdot3LocPSEAllocatedPowerValueAModeA, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

714 aLldpXdot3LocPSEAllocatedPowerValueBModeB, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

715 aLldpXdot3LocPDRRequestedPowerValueAModeA, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

716 aLldpXdot3LocPDRRequestedPowerValueBModeB, ATTRIBUTE, GET, column= LLDP Power via MDI Remote Package (conditional)

717

**End of Proposed Baseline**

720 **Annex A- Naming variables and attributes concept**

721

#	Subject	Single-signature example	Dual-Signature example
	<b>PSE</b>		
1	Physical layer state machine variable	pse_power_update	pse_power_update_pri pse_power_update_sec
2	DLL state machine variable e.g. 1	PSEAllocatedPowerValue	PSEAllocatedPowerValue_Alt(X)
3	DLL state machine variable e.g. 2	PDRRequestedPowerValueEcho	PDRRequestedPowerValueEcho_Alt(X)
4	Attribute name e.g. 1	aLldpXdot3LocPSEAllocatedPowerValue	aLldpXdot3LocPSEAllocatedPowerValueA aLldpXdot3LocPSEAllocatedPowerValueB
5	Attribute name e.g. 2	aLldpXdot3LocPDRRequestedPowerValue	aLldpXdot3LocPDRRequestedPowerValueA aLldpXdot3LocPDRRequestedPowerValueB
6	Filed name embedded in a text.	e.g. "The copy of the PSE Allocated Power Value field in the power via MDI TLV..."	e.g. "The copy of the PSE Allocated Power Value field for <b>Alternative (X)</b> in the power via MDI TLV..."

722

#	Subject	Single-signature example	Dual-Signature example
	<b>PD</b>		
1	Physical layer state machine variable	pd_undefined	pd_undefined_mode(X)
2	DLL state machine variable e.g. 1	MirroredPDRRequestedPowerValueEcho	MirroredPDRRequestedPowerValueEcho_mode(X)
3	DLL state machine variable e.g. 2	PSEAllocatedPowerValue	PSEAllocatedPowerValue_mode(X)
4	Attribute name e.g. 1	aLldpXdot3RemPSEAllocatedPowerValue	aLldpXdot3RemPSEAllocatedPowerValueA aLldpXdot3RemPSEAllocatedPowerValueB
5	Attribute name e.g. 2	aLldpXdot3LocPSEAllocatedPowerValue	aLldpXdot3LocPSEAllocatedPowerValueA aLldpXdot3LocPSEAllocatedPowerValueB
6	Filed name embedded in a text.	e.g. "The copy of the PD Requested Power Value filed in the Power Via MDI TLV..."	e.g. "The copy of the PD Requested Power Value filed for mode(X) in the Power Via MDI TLV..."

723

724

725