Cl **33** *SC* **33.1.4** *P* *L* # 1

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of Geoff Thompson (1271)

Move as much of the cabling specification to cabling documents as possible. (This RR was entered as a tracking mechanism for Thompson Comment #59 against P802.3REVbx/D2.0 during initial WG ballot. Resolution of this comment was given over to P802.3bt as they will have Cl 33 open.)

SuggestedRemedy

see maint_1271.pdf for proposed text

Proposed Response *Response Status* **O**

Cl **33** *SC* **33** *P* *L* # 2

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of George Zimmermann (1273)

Text in the existing standard is ambiguous and is inconsistent with the more precise definition in the definitions section. The imprecise language “generic term” does not point to a specific interface point necessary for the specifications attached to the PI, including a pin-out. In contrast the language in the definitions section is more precise.

SuggestedRemedy

Change: The Power Interface (PI) is the generic term that refers to the mechanical and electrical interface between the PSE or PD and the transmission medium. To: The Power Interface (PI) is the mechanical and electrical interface between the Power Sourcing Equipment (PSE) or Powered Device (PD) and the transmission medium as defined in 1.4.324 (1.4.336 in P802.3bx/D2.0). In an Endpoint PSE and in a PD the Power Interface is the MDI as defined in 1.4.256 (1.4.268 in P802.3bx/D2.0)

Proposed Response *Response Status* **O**

Cl **33** *SC* **33** *P* *L* # 3

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of David Law (1276)
CLAUSE NUMBER: 1.4, 1.5, 33.1

The IEEE Std 802.3-2012 keywords include 'Power over Ethernet', however 'Power over Ethernet' and 'PoE' do not appear anywhere within the body of the standard.

SuggestedRemedy

[1] Add the following new definition in alphanumeric order to IEEE Std 802.3 subclause 1.4 'Definitions':

1.4.xxx IEEE 802.3 Power over Ethernet (IEEE 802.3 PoE): A system consisting of one PSE and one PD that provides power across balanced twisted-pair cabling. (See IEEE Std 802.3, Clause 33).

[2] Add the following new definition in alphanumeric order to IEEE Std 802.3 subclause 1.5 'Abbreviation':

PoE Power over Ethernet

[3] Modify the first paragraph of IEEE Std 802.3 subclause 33.1 'Overview' to read as follows:

This clause defines the functional and electrical characteristics for providing a Power over Ethernet (PoE) system for deployment over balanced twisted-pair cabling. The system consists of two optional power (non-data) entities, a Powered Device (PD) and Power Sourcing Equipment (PSE), for use with the MAU defined in Clause 14 and the PHYs defined in Clause 25 and Clause 40. These entities allow devices to draw/supply power using the same generic cabling as is used for data transmission.

Proposed Response *Response Status* **O**

Cl **33** *SC* **33** *P* *L* # 4

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of Koussalya Balasubramanian (1277)

PDs in the field turn on their DC-DC load during inrush. This leads to PD cap not charging up fully (even if PD cap is <180uf PSE is followinginrush rules from Section 33.2.7.5). This may lead to operational problems after inrush. There is a Voff requirement in PD table 33-18 to ensure power supply remains turned off for V<30V, but customers seem to read this as applicable only "after power on" not during "power on"- hence ether turn on their DC-DC during inrush causing problems.

SuggestedRemedy

Request the following text be added as note to section 33.4.1
Add the following to section 33.3.7.3
"PDs shall not draw more than the maximum current allowed by a PSE during inrush as outlined in section 33.2.7.5"
Change 2nd paragraph of Section 33.3.7.1 as follows (change shown in _underline_) "The PD shall _not_ turn on until a voltage _greater_than_Voff_and_less than or equal to Von"

Proposed Response *Response Status* **O**

Cl **33** *SC* **33.1.4** *P* *L* # 5

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of Geoff Thompson (1278)

The "definitions" for:
Iport (1.4.234)
Vpd (1.4.425)
Vpse (1.4.426)
are incorrectly placed in the definitions clause of the overall standard for terms (1.4).
They are not terms, They are parameters, as such they belong within the technical clause in which they are used.

SuggestedRemedy

Text is not to be changed.
Existing text is to be moved to appropriate placement within clause 33. Suggested placement is adjacent to Icable definition in 33.1.4.

Proposed Response *Response Status* **O**

Cl **33** *SC* **33** *P* *L* # 6

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of Lennart Yseboodt (1294)

The entry arc into IDLE reads "[(Vpd < Vreset) + !power_received] * mdi_power_required * !pd_reset"
The effect is that at ANY voltage below Vport_pd min, this condition will apply and reset the state machine to IDLE. The intent is to allow a global override to reset the SM to IDLE when the PI voltage drops below Vreset, but this has been done incorrectly.
See comment #213 against D1.5 for P802.3bt

SuggestedRemedy

This MR pertains to Figure 33-16, page 657, the PD state machine.
Change the entry arc condition into IDLE to read:
(Vpd < Vreset) * mdi_power_required * !pd_reset

Proposed Response *Response Status* **O**

Cl **33** *SC* **33** *P* *L* # 7

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of Lennart Yseboodt (1300)

The existing text in 33.3.7.9 says "When VPort_PD max is applied across the PI...". VPort_PD max is a single voltage point of 57.0V. This has the effect of invalidating the requirement. The intent of this requirement was to not backfeed at any voltage from 0V to 57V. In addition the existing text is not 100% clear on where the 100K load resistor is to be placed.

SuggestedRemedy

This MR pertains to 33.3.7.9 on Backfeed voltage.
Replace the text in 33.3.7.9 as follows:

"When any voltage in the range of 0V to VPort_PD max is applied across the PI at either polarity specified on the conductors for Mode A according to Table 33–13, the voltage measured across the PI for Mode B with a 100 kOhm load resistor connected across Mode B shall not exceed Vbfd max as specified in Table 33–18. When any voltage in the range of 0V to VPort_PD max is applied across the PI at either polarity specified on the conductors for Mode B according to Table 33–13, the voltage measured across the PI for Mode A with a 100 kOhm load resistor connected across Mode A shall not exceed Vbfd max."

Proposed Response *Response Status* **O**

Cl **33** *SC* **33** *P* *L* # 8

Jones, Chad
Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of David Law (1306)

There is an assignment to the pd_dll_power_type variable in the NITIALIZE state of Figure 33–46 'PSE power control state diagram' as well as a mapping to it in Table 33–41 'Attribute to state diagram variable cross-reference' so effectively there are two sources to this variable. There is a case where a Type 2 PSE that supports 1-event physical layer classification, Data Link Layer Classification, and chooses the option of setting the parameter_type variable to 1 in the set_parameter_type function if mutual identification is not complete, is connected to a Type 2 PD, which will result in two different values for pd_dll_power_type from these two sources.

After a successful detection, Figure 33-13 'Type 1 and Type 2 PSE state diagram' will transition in to the DETECT_EVAL state and then to the ONE_EVENT_CLASS state (arrow B), since the PSE supports 1-event physical layer classification (class_num_events = 1). The state diagram will then call the do_classification function which will result in the pd_requested_power variable being set to 3 and the mr_pd_class_detected variable being set to 4. The state diagram will then proceed to the CLASSIFICATION_EVAL and, assuming sufficient power, to the POWER_UP state.

Once power up has been completed successfully, since this is a TYPE 2 PSE (PSE_TYPE = 2), the state diagram will transition from the POWER_UP state to the SET_PARAMETERS state calling the set_parameter_type function. Since only 1-event physical layer classification has taken place mutual identification is not complete however a Type 2 PD has been detected since the mr_pd_class_detected variable is set to 4. The PSE therefore has the option of setting the parameter_type variable to 1 (see page 72, line 54, 'When a Type 2 PSE powers a Type 2 PD, the PSE may choose to assign a value of ‘1’ to parameter_type if mutual identification is not complete ...'). I will assume this option is taken.

The state diagram will therefore transition to the POWER_ON state. At some point later, since Data Link Layer Classification is supported, the pse_dll_ready variable becomes TRUE and the aLldpXdot3RemPowerType attribute will return a bit string indicating a Type 2 PD. This, according to Table 33–41 'Attribute to state diagram variable cross-reference', also results in pd_dll_power_type being set to 2. The problem is that, according to the Figure 33-46 'PSE power control state diagram', when pse_dll_ready becomes TRUE the value of parameter_type is latched on to pd_dll_power_type, and at that point intimeitis1.

Now it seems that the intent was that when pd_dll_power_type became 2 due to Data Link Layer Classification, the equation on the transition from the POWER_ON state to the SET_PARAMETERS state became true ((PSE_TYPE = 2) * (pd_dll_power_type = 2) * (parameter_type = 1)) resulting in the set_parameter_type function being called for a second time. The parameter_type variable would then be set 2 enabling the PSE to increase the power it supplies from Type 1 to Type 2 limits.

The problem is there are two values of pd_dll_power_type once Data Link Layer Classification is in operation, the one based on the Table 33–41 mapping which in this case would be set to a value of 2, and the one output by the Figure 33-46 state diagram, which in this case would be set to a value of 1. As well as the statement that 'State diagrams take precedence over text.' incorporated by the reference to subclause 21.5 in subclause 33.2.5.2 the definition of the pd_dll_power_type variable in subclause 33.2.5.4 'Type 1 and Type 2 variables' for Figure 33-13 state that it is 'control variable output by the PSE power control state diagram (Figure 33–46) ...'. Based on this it would seem that the latter value of 1 should be used, however the problem with that is the second call to SET_PARAMETERS state will then never happen, and the PSE will have to continue using Type 1 limits.

It would seem a better approach would be to remove the assignment of parameter_type to pd_dll_power_type in the INITIALIZE state of Figure 33–46 'PSE power control state diagram' and just use the Table 33–41 'Attribute to state diagram variable cross-reference' mapping for Figure 33-13. This is the only use of the parameter_type and pd_dll_power_type variables in Figure 33–46 so they can also be removed from the associated variable definition lists.

The variable pd_dll_power_type however has to gated while pse_dll_ready is FALSE, since at that time aLldpXdot3RemPowerType is undefined and therefore the mapping of Table 33–41 'Attribute to state diagram variable cross-reference' is undefined. There4 also needs to be some qualification based on DLL being implemented for the case of a Type 2 PSE with 2-event physical layer classification but no Data Link Layer Classification.

Based on this the use of pd_dll_power_type on the POWER_ON to SET_PARAMETERS transition should be qualified with pse_dll_capable = TRUE and pse_dll_ready = TRUE, so the equation would become (PSE_TYPE = 2) * (pd_dll_power_type = 2) * (parameter_type = 1) * pse_dll_capable * pse_dll_ready.

SuggestedRemedy

[1] The equation on the transition from the POWER_ON state to the SET_PARAMETERS state in Figure 33-13 'Type 1 and Type 2 PSE state diagram' be changed to read '(PSE_TYPE = 2) * (pd_dll_power_type = 2) * (parameter_type = 1) * pse_dll_capable * pse_dll_ready'.

[2] The assignment 'pd_dll_power_type <= parameter_type' in the INITIALIZE state in Figure 33–46 'PSE power control state diagram' be removed.

[3] The definition of parameter_type be removed from 33.5.3.3 'Single-signature system Variables'.

[4] The definition of pd_dll_power_type be removed from 33.5.3.3 'Single-signature system Variables'.

[5] In definition of pd_dll_power_type in subclause 33.2.5.4 'Type 1 and Type 2 variables' change the text 'A control variable output by the PSE power control state diagram (Figure 33–46) that indicates ...' to read 'A variable mapped from the aLldpXdot3RemPowerType as defined in Table 33-41 that indicates ...'.

Proposed Response *Response Status* **O**

Cl **33** *SC* **33** *P* *L* # 9

Jones, Chad
Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of David Law (1307)

There is an assignment to the pse_dll_power_type variable in the INITIALIZE state of Figure 33–49 'PD power control state diagram' as well as a mapping to it in Table 33–41 'Attribute to state diagram variable cross-reference' so effectively there are two sources to this variable. There is a case where a Type 2 PD is connected to a Type 2 PSE that supports 1-event physical layer classification, Data Link Layer Classification which will result in two different values for pd_dll_power_type from these two sources.

On entry to the DO_DETECTION state of Figure 33–31 'Type 1 and Type 2 PD state diagram' the pse_power_type variable is set to 1. As a result of the 1-event physical layer classification that this PSE will perform, the state diagram will then progress to the DO_CLASS_EVENT1 state and then, assuming that the PSE starts supplying power, will progress to the MDI_POWER1 state once the power_received variable becomes TRUE.

The pd_max_power variable will be set to 0 (4 modulo 4), allowing the PD to draw up to Class 0 power (13.0W). Since pse_power_type has been set to 1 the state diagram will then progress to the DLL_ENABLE state setting the pd_dll_enabled variable to TRUE enabling Data Link Layer Classification for the PD. At this point however pse_power_type is still set to 1 so the state diagram will transition back to the MDI_POWER1 state where it will remain as pd_dll_enabled is now TRUE.

Since the PSE supports Data Link Layer Classification the aLldpXdot3RemPowerType attribute within the oLldpXdot3RemSystemsGroup managed object class will return a bit string indicating a Type 2 PSE at some point afterwards when the pd_dll_ready variable becomes TRUE. This, according to Table 33–41 'Attribute to state diagram variable cross-reference', also results in pd_dll_power_type being set to 2. The problem is that, according to the Figure 33-49 'PD power control state diagram', when pd_dll_ready becomes TRUE the value of pse_power_type is latched on to pse_dll_power_type, and at that point in time it is 1.

Now it seems that the intent was that when pse_dll_power_type became 2 due to Data Link Layer Classification, the equation on the transition from MDI_POWER1 to MDI_POWER_DLY state became true (pse_power_type = 2) + (pse_dll_power_type = 2) causing, after a delay, entry to the MDI_POWER2 state. At that point the pd_max_power variable will be increased from 0 (class_sig modulo 4) to 4 due to the assignment pd_max_power <= class_sig enabling the power drawn to increase from Type 1 to Type 2 limits.

The problem is there are two values of pse_dll_power_type once Data Layer Classification is in operation, the one based on the Table 33–41 mapping which in this case would be set to a value of 2, and the one output by the Figure 33-49 state diagram, which in this case would be set to a value of 1. As well as the statement that 'State Diagrams take precedence over text.' the definition of the pse_dll_power_type variable in subclause 33.3.3.4 'Type 1 and Type 2 Variables' for Figure 33-31 states 'A control variable output by the PD power control state diagram (Figure 33–49) that ...'. . Based on this it would seem that the latter value of 1 should be used, however the problem with this is that the MDI_POWER2 state will then never be reached, and the PD will have to continue draw power within the Type 1 limits.

It would seem a better approach would be to remove the assignment of pse_power_type to pse_dll_power_type in the INITIALIZE state of Figure 33–49 'PD power control state diagram' and just use the Table 33–41 'Attribute to state diagram variable cross-reference' mapping for Figure 33-31. This is the only use of the pse_power_type and pse_dll_power_type variables in Figure 33–49 so they can also be removed from the associated variable definition lists.

The variable pse_dll_power_type however has to gated while pd_dll_ready is FALSE, since at that time aLldpXdot3RemPowerType is undefined and therefore the mapping of Table 33–41 'Attribute to state diagram variable cross-reference' is undefined. Based on this the use of pse_dll_power_type on the MDI_POWER1 to MDI_POWER_DLY transition should be qualified with pse_dll_ready = TRUE, so the equation would become (pse_power_type = 2) + (pse_dll_power_type = 2 * pd_dll_ready).

SuggestedRemedy

[1] The equation on the transition from the MDI_POWER1 state to the MDI_POWER_DLY state in Figure 33-31 'Type 1 and Type 2 PD state diagram' be changed to read '(pse_power_type = 2) + (pse_dll_power_type = 2 * pd_dll_ready)'.

[2] The assignment 'pse_dll_power_type <= pse_power_type' in the INITIALIZE state in Figure 33–49 'PD power control state diagram' be removed.

[3] The definition of pse_power_type be removed from 33.5.3.3 'Single-signature system Variables'.

[4] The definition of pse_dll_power_type be removed from 33.5.3.3 'Single-signature system Variables'.

[5] In definition of pse_dll_power_type in subclause 33.3.3.4 'Type 1 and Type 2 Variables' change the text 'A control variable output by the PD power control state diagram (Figure 33–49) that ...' to read 'A variable mapped from the aLldpXdot3RemPowerType as defined in Table 33-41 that indicates ...'.

Proposed Response *Response Status* **O**

Cl **33** *SC* **33.4.9.1.4** *P* *L* # 10

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of Valerie Maguire (1310)

An explanation of Midspan PSE and how it is implemented within a link segment is needed. Align with the resolution to comment #119 against the draft 2.4 Working Group ballot recirculation of P802.3bt. I believe that this change can be implemented as part of the 802.3bt revision project.

SuggestedRemedy

Replace "Replacing the work area or equipment cable with a cable that includes a Midspan PSE should not alter the requirements of the cable. This cable"
with "A Midspan PSE replaces an element in a link segment and"

Proposed Response *Response Status* **O**

Cl **33** *SC* **33.4.9.1** *P* *L* # 11

Jones, Chad Cisco

Comment Type **TR** *Comment Status* **X**

Filed on behalf of Valerie Maguire (1311)

An explanation of Connector Midspan PSE and how it is implemented within a link segment is needed. Align with the resolution to comment #116 against the draft 2.4 Working Group ballot recirculation of P802.3bt. I believe that this change can be implemented as part of the 802.3bt revision project.

SuggestedRemedy

Replace "The Midspan PSE equipment to be inserted as “connector” or “telecom outlet” shall meet the following transmission parameters"
with "A connector Midspan PSE replaces one of the connectors in the link segment and shall meet the following transmission parameters"

Proposed Response *Response Status* **O**