X.X.X.X Constants

The PSE state diagrams use the following constants:

CC_DET_SEQ

A constant indicating the sequence in which the PSE performs Connection Check and Detection.

Values:

0: Connection Check is followed by staggered Detection for a Single-signature PD and parallel detection for a Dual-Signature PD.

1: Detection on a pairset is followed by Connection Check and then Detection on the other pairset for a Single-signature PD and both pairsets for a Dual-Signature PD.

2: Connection Check and Detection on both pairsets are performed within a single Tdet window.

3: Connection Check is followed by staggered Detection.

X.X.X.X Variables

The PSE state diagrams use the following variables:

alt_done_pri

A variable used to coordinate the main Single-signature state diagram with the Pseudo-independent Dual-signature state diagram for the Primary Alternative.

Values:

FALSE: The Pseudo-independent state diagram is not ready to return to global IDLE within the Single-signature state diagram.

TRUE: The Pseudo-independent state diagram is ready to return to global IDLE within the Single-signature state diagram.

alt_done_sec

A variable used to coordinate the main Single-signature state diagram with the Pseudo-independent Dual-signature state diagram for the Secondary Alternative.

Values:

FALSE: The Pseudo-independent state diagram is not ready to return to global IDLE within the Single-signature state diagram.

TRUE: The Pseudo-independent state diagram is ready to return to global IDLE within the Single-signature state diagram.

alt_pri

A variable used to select which Alternative assumes the role of Primary in the state diagram.

Values:

a: Alternative A is assigned Primary, and Alternative B is assigned Secondary.

b: Alternative B is assigned Primary, and Alternative A is assigned Secondary.

alt_pri_pwrd

A variable that controls the circuitry that the PSE uses to power the PD over the Alternative that has been assigned as Primary.

Values:

FALSE: The PSE is not to apply power to the Primary Alternative.

TRUE: The PSE has detected, classified, and will power a PD on the Primary Alternative; or power is being forced on the Primary Alternative in TEST_MODE.

alt_sec_pwrd

A variable that controls the circuitry that the PSE uses to power the PD over the Alternative that has been assigned as Secondary.

Values:

FALSE: The PSE is not to apply power to the Secondary Alternative.

TRUE: The PSE has detected, classified, and will power a PD on the Secondary Alternative; or power is being forced on the Secondary Alternative in TEST_MODE.

class_num_events

A variable indicating the maximum number of classification events performed by the PSE. A variable that is set in an implementation-dependent manner.

Values:

0: PSE does not perform Physical Layer classification.

1: PSE performs Single-Event Physical Layer classification or Multiple-Event Physical Layer classification with a maximum of 1 class event.

2: PSE performs Multiple-Event Physical Layer classification with a maximum of 2 class events.

4: PSE performs Multiple-Event Physical Layer classification with a maximum of 4 class events.

5: PSE performs Multiple-Event Physical Layer classification with a maximum of 5 class events.

det_start_pri

A variable that indicates to the Secondary Alternative that the Primary Alternative is between START_DETECT and POWER_UP.

Values:

FALSE: The Primary Alternative is not between START_DETECT and POWER_UP.

TRUE: The Primary Alternative is between START_DETECT and POWER_UP.

det_start_sec

A variable that indicates to the Primary Alternative that the Secondary Alternative is between START_DETECT and POWER_UP.

Values:

FALSE: The Secondary Alternative is not between START_DETECT and POWER_UP.

TRUE: The Secondary Alternative is between START_DETECT and POWER_UP.

det_temp

A temporary variable that indicates whether a 4-pair PSE has completed Detection on only one alternative.

Values:

0: The PSE has not completed a Detection on only one Alternative.

1: The PSE has completed a Detection on only one Alternative.

dll_4PID

A variable that indicates whether the PSE and PD have negotiated 2-pair or 4-pair power.

Values:

0: 2-pair power negotiated.

1: 4-pair power negotiated.

error_condition

A variable indicating the status of implementation-specific fault conditions or optionally other system faults that prevent the PSE from meeting the specifications in Table 33–11 and that require the PSE not to source power. These error conditions are different from those monitored by the state

diagrams in Figure 33–10.

Values:

FALSE: No fault indication.

TRUE: A fault indication exists.

IInrush-2P

Output current per pairset during POWER_UP (see Table 33–11 and Figure 33–13).

IPort-2P-pri

Total output current sourced by Primary Alternative (see 33.2.7.6).

IPort-2P-sec

Total output current sourced by Secondary Alternative (see 33.2.7.6).

mr_force_pwr_pri

This variable indicates if the Primary Alternative is to apply power to the link while in TEST_MODE (see Table 33-2). This variable is provided by a management interface that may be mapped to the PSE Control register Pair Control bit (11.6) or other equivalent function.

Values:

FALSE: The Primary Alternative is not powered.

TRUE: The Primary Alternative is powered.

mr_force_pwr_sec

This variable indicates if the Secondary Alternative is to apply power to the link while in TEST_MODE (see Table 33-2). This variable is provided by a management interface that may be mapped to the PSE Control register Pair Control bit (11.7) or other equivalent function.

Values:

FALSE: The Secondary Alternative is not powered.

TRUE: The Secondary Alternative is powered.

mr_mps_valid

The PSE monitors the Maintain Power Signature (MPS, see 33.2.9.1). This variable indicates the presence or absence of a valid MPS.

Values:

FALSE: MPS is absent.

TRUE: MPS is present.

mr_mps_valid_pri

The PSE monitors the Maintain Power Signature (MPS, see 33.2.9.1) on the Primary Alternative. This variable indicates the presence or absence of a valid MPS.

Values:

FALSE: MPS is absent.

TRUE: MPS is present.

mr_mps_valid_sec

The PSE monitors the Maintain Power Signature (MPS, see 33.2.9.1) on the Secondary Alternative. This variable indicates the presence or absence of a valid MPS.

Values:

FALSE: MPS is absent.

TRUE: MPS is present.

mr_pse_alternative

This variable indicates which Pinout Alternative the PSE uses to apply power to the link (see Table 33-2). This variable is provided by a management interface that may be mapped to the PSE Control register Pair Control bits (11.3:2) or other equivalent function.

Values:

a: The PSE uses PSE pinout Alternative A.

b: The PSE uses PSE pinout Alternative B.

both: The PSE uses both Alternative A and Alternative B.

mr_pse_enable

A control variable that selects PSE operation and test functions. This variables is provided by a management interface that may be mapped to the PSE Control register PSE Enable bits (11.1:0), as described below, or other equivalent functions.

Values:

disable: All PSE functions disabled (behavior is as if there was no PSE functionality). This value corresponds to MDIO register bits 11.1:0 = '00'.

enable: Normal PSE operation. This value corresponds to MDIO register bits 11.1:0 = '01'.

force_power: Test mode selected that causes the PSE to apply power to the PI when there are no detected error conditions. This value corresponds to MDIO register bits 11.1:0 = '10'.

mr_pse_ss_mode

A variable that controls whether the PSE 4-pair or 2-pair powers a Class 0-4 Single-signature PD.

0: Single-signature PD is 2-pair powered

1: Single-signature PD is 4-pair powered

option_detect_ted

This variable indicates if detection can be performed by the PSE during the ted_timer interval.

Values:

FALSE: Do not perform detection during ted_timer interval.

TRUE: Perform detection during ted_timer interval.

option_vport_lim

This optional variable indicates if V_{PSE} is out of the operating range during normal operating state.

Values:

FALSE: V_{PSE} is within the VPort_PSE-2P operating range as defined in Table 33–11.

TRUE: V_{PSE} is outside of the VPort_PSE-2P operating range on at least one pairset as defined in Table 33–11.

ovld_det_pri

A variable indicating if the PSE output current over the Primary Alternative has been in an overload condition (see 33.2.7.6) for at least T_{CUT} of a one second sliding time.

Values:

FALSE: The PSE has not detected an overload condition on the Primary Alternative.

TRUE: The PSE has detected an overload condition on the Primary Alternative.

ovld_det_sec

A variable indicating if the PSE output current over the Secondary Alternative has been in an overload condition (see 33.2.7.6) for at least T_{CUT} of a one second sliding time.

Values:

FALSE: The PSE has not detected an overload condition on the Secondary Alternative.

TRUE: The PSE has detected an overload condition on the Secondary Alternative.

pd_4pair_cand

This variable is used by the PSE to indicate that a connected PD is a candidate to receive power on both Modes. This variable is a function of the results of Detection, Connection Check, and 4PID.

Values:

FALSE: The PD is not a candidate to receive power on both Modes.

TRUE: The PD is a candidate to receive power on both Modes.

pd_dll_power_type

A control variable output by the PSE power control state diagram (Figure 33–27) that indicates the type of PD as advertised through Data Link Layer classification.

Values:

1: PD is a Type 1 PD (default)

2: PD is a Type 2 PD

3: PD is a Type 3 PD

4: PD is a Type 4 PD

Editor's Note: Mutual identification will require a variable pd_power_type similar to pd_dll_power_type.

pism

A variable used by the Single-signature state machine to kick off the Pseudo-independent Dual-signature state machines.

Values:

FALSE: Single-signature state machine has control of the Alternatives.

TRUE: Single-signature state machine has passed control of the Alternatives to the Pseudo-independent Dual-signature state machines.

power_not_available

Variable that is asserted in an implementation-dependent manner when the PSE is no longer capable of sourcing sufficient power to support the attached PD. Sufficient power is defined by classification; see 33.2.6.

Values:

FALSE: PSE is capable to continue to source power to a PD.

TRUE: PSE is no longer capable of sourcing power to a PD.

power_not_available_pri

Variable that is asserted in an implementation-dependent manner when the PSE is no longer capable of sourcing sufficient power on the Primary Alternative to support the attached PD. Sufficient power is defined by classification; see 33.2.6.

Values:

FALSE: PSE is capable to continue to source power to a PD.

TRUE: PSE is no longer capable of sourcing power to a PD.

power_not_available_sec

Variable that is asserted in an implementation-dependent manner when the PSE is no longer capable of sourcing sufficient power on the Secondary Alternative to support the attached PD. Sufficient power is defined by classification; see 33.2.6.

Values:

FALSE: PSE is capable to continue to source power to a PD.

TRUE: PSE is no longer capable of sourcing power to a PD.

pse_avail_pwr

This variable indicates the highest power PD Class that could be supported. The value is determined in an implementation-specific manner.

Values:

0: Class 1
1: Class 2
2: Class 0 or Class 3
3: Class 4
4: Class 5
5: Class 6
6: Class 7
7: Class 8

pse_avail_pwr_pri

This variable indicates the highest power PD Class that could be supported on the Primary Alternative. The value is determined in an implementation-specific manner.

Values:

0: Class 1
1: Class 2
2: Class 0 or Class 3
3: Class 4
4: Class 5

pse_avail_pwr_sec

This variable indicates the highest power PD Class that could be supported on the Secondary Alternative. The value is determined in an implementation-specific manner.

Values:

0: Class 1
1: Class 2
2: Class 0 or Class 3
3: Class 4
4: Class 5

pse_dll_capable

This variable indicates whether the PSE is capable of performing optional Data Link Layer classification. See 33.6 for a description of Data Link Layer functionality. This variable is provided by a management interface that may be mapped to the PSE Control register Data Link Layer Classification Capability bit (11.5), as described below, or other equivalent functions. A variable that is set in an implementation-dependent manner.

Values:

FALSE: The PSE's Data Link Layer classification capability is not enabled.
TRUE: The PSE's Data Link Layer classification capability is enabled.

pse_dll_enabled

A variable indicating whether the Data Link Layer classification mechanism is enabled. See 33.6.

Values:

FALSE: Data Link Layer classification is not enabled.
TRUE: Data Link Layer classification is enabled.

pse_ready

Variable that is asserted in an implementation-dependent manner to probe the link segment.

Values:

FALSE: PSE is not ready to probe the link segment.
TRUE: PSE is ready to probe the link segment.

NOTE—Care should be taken when negating this variable in a PSE performing detection using Alternative A after an invalid signature is detected due to the delay it introduces between detection attempts (see 33.2.4.1).

pse_reset

Controls the resetting of the PSE state diagram. Condition that is TRUE until such time as the power supply for the device that contains the PSE overall state diagrams has reached the operating region. It is also TRUE when implementation-specific reasons require reset of PSE functionality.

Values:

FALSE: Do not reset the PSE state diagram.

TRUE: Reset the PSE state diagram.

pwr_app_pri

A variable indicating that the PSE has begun steady state operation on the Primary Alternative by having asserted alt_pri_pwrd, completed the ramp of voltage, is not in a current limiting mode, and is operating beyond the POWER_UP requirements of 33.2.7.5.

Values:

FALSE: The PSE is either not applying power or has begun applying power but is still in POWER_UP on the Primary Alternative.

TRUE: The PSE has begun steady state operation on the Primary Alternative.

pwr_app_sec

A variable indicating that the PSE has begun steady state operation on the Secondary Alternative by having asserted alt_sec_pwrd, completed the ramp of voltage, is not in a current limiting mode, and is operating beyond the POWER_UP requirements of 33.2.7.5.

Values:

FALSE: The PSE is either not applying power or has begun applying power but is still in POWER_UP on the Secondary Alternative.

TRUE: The PSE has begun steady state operation on the Secondary Alternative.

short_det_pri

A variable indicating if the PSE output current has been in a short circuit condition on the Primary Alternative.

Values:

FALSE: The PSE has not detected a short circuit condition on the Primary Alternative.

TRUE: The PSE has detected qualified short circuit condition on the Primary Alternative.

short_det_sec

A variable indicating if the PSE output current has been in a short circuit condition on the Secondary Alternative.

Values:

FALSE: The PSE has not detected a short circuit condition on the Secondary Alternative.

TRUE: The PSE has detected qualified short circuit condition on the Secondary Alternative.

temp_var

A temporary variable used to store the value of the state variable mr_pd_class_detected.

PSEs shall meet at least one of the allowable variable definition permutations described in Table 33–3.

Table 33–3—Allowed PSE variable definition permutations

PSE Type	class_num_events
Type 4	1, 2, 4, 5
Type 3	1, 2, 4
Type 2	1, 2
Type 1	0, 1

PSEs shall issue no more Class events than the Class they are capable of supporting. For example, this would apply to a PSE that is oversubscribed and in power management mode or a PSE that has a hardware limitation.

X.X.X.X Timers

All timers operate in the manner described in 14.2.3.2 with the following addition: a timer is reset and stops counting upon entering a state where "stop x_timer" is asserted.

Editor's note: Timers to be added for Autoclass.

tcc_timer
: A timer used to monitor the duration of Connection Check.

tcc2det_timer
: A timer used to limit the time between Connection Check and Detection when CC_DET_SEQ = 0. See Table 33-3a.

tcle2_timer
: A timer used to limit the second classification event time in Multiple-Event classification; see TCLE2 in Table 33–10.

tcle3_timer
: A timer used to limit the third through fifth classification event time in Multiple-Event classification; see TCLE3 in Table 33–10.

tdbo_timer
: A timer used to regulate backoff upon detection of an invalid signature; see Tdbo in Table 33–11.

tdet_timer
: A timer used to limit an attempt to detect a PD; see Tdet in Table 33–11.

tdet_timer_pri
: A timer used to limit an attempt to detect a PD on the Primary Alternative; see Tdet in Table 33–11.

tdet_timer_sec
: A timer used to limit an attempt to detect a PD on the Secondary Alternative; see Tdet in Table 33–11.

tdet2det_timer
: A timer used to limit the time between the completion of a detection on one pairset and the beginning of a detection on the other. See Table 33-3a.

ted_timer

A timer used to regulate a subsequent attempt to power a PD after an error condition causes power removal; see Ted in Table 33–11. The default state of this timer is ted_timer_done.

ted_timer_pri

A timer used to regulate a subsequent attempt to power a PD after an error condition causes power removal from the Primary Alternative; see Ted in Table 33–11. The default state of this timer is ted_timer_pri_done.

ted_timer_sec

A timer used to regulate a subsequent attempt to power a PD after an error condition causes power removal from the Secondary Alternative; see Ted in Table 33–11. The default state of this timer is ted_timer_sec_done.

tinrush_pri_timer

A timer used to monitor the duration of the inrush event on the Primary Alternative; see $T_{Inrush-2P}$ in Table 33-11.

tinrush_sec_timer

A timer used to monitor the duration of the inrush event on the Secondary Alternative; see $T_{Inrush-2P}$ in Table 33-11.

tlcf_timer

A timer used to limit the first classification event time in Multiple-Event classification; see TLCF in Table 33–10.

tme1_timer

A timer used to limit mark event times for all but the last the first mark event time in during Multiple-Event classification; see TME1 in Table 33–10.

tme2_timer

A timer used to limit the second final mark event time in Multiple-Event classification; see TME2 in Table 33–10.

tmpdo_timer

A timer used to monitor the dropout of the MPS; see TMPDO in Table 33–11.

tmpdo_timer_pri

A timer used to monitor the dropout of the MPS on the Primary Alternative; see TMPDO in Table 33–11.

tmpdo_timer_sec

A timer used to monitor the dropout of the MPS on the Secondary Alternative; see TMPDO in Table 33–11.

tpdc_timer

A timer used to limit the classification time; see Tpdc in Table 33–10.

tpon_timer

A timer used to limit the time for power turn-on; see Tpon in Table 33–11.

tpon_timer_pri

A timer used to limit the time on the Primary Alternative for power turn-on; see Tpon in Table 33–11.

tpon_timer_sec

A timer used to limit the time on the Secondary Alternative for power turn-on; see Tpon in Table 33–11.

X.X.X.X Functions

do_cxn_chk

This function initiates the Connection Check in Section 33.2.5.0a. This function returns the following variables:

sig_type: This variable indicates the type of PD signature connected to the PI, with respect to 4-pair operation.

Values:

open_circ: The PSE has detected an open circuit on both pairsets.

single: The PSE has determined there is a single-signature PD configuration connected to the PI.

dual: The PSE has determined there is a dual-signature PD configuration connected to the PI.

do_classification

This function returns the following variables:

pd_cls_4PID: This variable indicates that 4PID has been established.

Values:

FALSE: PD is not a candidate for 4-pair power.

TRUE: PD is a candidate for 4-pair power.

pd_req_pwr: This variable indicates the power class requested by the PD. When a PD requests a higher class than a PSE can support, the PSE shall assign the PD Class 3, 4, or 6, whichever is the highest that it can support. See Section 33.2.6.

Values:

0: Class 1

1: Class 2

2: Class 0 or Class 3

3: Class 4

4: Class 5 (mr_pd_class_detected will have a value of 4 for the first two class events and a value of 0 for any subsequent class events.)

5: Class 6 (mr_pd_class_detected will have a value of 4 for the first two class events and a value of 1 for any subsequent class events.)

6: Class 7 (mr_pd_class_detected will have a value of 4 for the first two class events and a value of 2 for any subsequent class events.)

7: Class 8 (mr_pd_class_detected will have a value of 4 for the first two class events and a value of 3 for any subsequent class events.)

Editor's Note: DS PD classification must be taken into account here.

mr_pd_class_detected: The PD classification signature seen during a classification event; see Table 33-7 and 33.2.6.

Values:

0: Class 0

1: Class 1

2: Class 2

3: Class 3
4: Class 4
5: Class 5
6: Class 6
7: Class 7
8: Class 8

do_classification_pri

This function returns the following variables for the Primary Alternative:

pd_cls_4PID_pri: This variable indicates that 4PID has been established.
Values:

FALSE: PD is not a candidate for 4-pair power.
TRUE: PD is a candidate for 4-pair power.

pd_req_pwr_pri: This variable indicates the power class requested by the PD. When a PD requests a higher class than a PSE can support, the PSE shall assign the PD Class 3, 4, or 6, whichever is the highest that it can support. See Section 33.2.6.
Values:

0: Class 1
1: Class 2
2: Class 0 or Class 3
3: Class 4
4: Class 5 (mr_pd_class_detected_pri will have a value of 4 for the first two class events and a value of 0 for any subsequent class events.)

Editor's Note: DS PD classification must be taken into account here.

mr_pd_class_detected_pri: The PD classification signature seen during a classification event; see Table 33-7 and 33.2.6.
Values:

0: Class 0
1: Class 1
2: Class 2
3: Class 3
4: Class 4

do_classification_sec

This function returns the following variables for the Secondary Alternative:

pd_cls_4PID_sec: This variable indicates that 4PID has been established.
Values:

FALSE: PD is not a candidate for 4-pair power.
TRUE: PD is a candidate for 4-pair power.

pd_req_pwr_sec: This variable indicates the power class requested by the PD. When a PD requests a higher class than a PSE can support, the PSE shall assign the PD Class 3, 4, or 6, whichever is the highest that it can support. See Section 33.2.6.
Values:

0: Class 1
1: Class 2
2: Class 0 or Class 3
3: Class 4

4: Class 5 (mr_pd_class_detected_sec will have a value of 4 for the first two class events and a value of 0 for any subsequent class events.)

Editor's Note: DS PD classification must be taken into account here.

mr_pd_class_detected_sec: The PD classification signature seen during a classification event; see Table 33-7 and 33.2.6.
Values:
0: Class 0
1: Class 1
2: Class 2
3: Class 3
4: Class 4

do_detect_pri
This function returns the following variables (see 33.2.5):

sig_pri: This variable indicates the presence or absence of a valid PD detection signature on the Primary Alternative.
Values:
open_circuit: The PSE has detected an open circuit.
valid: The PSE has detected a PD requesting power.
invalid: Neither open circuit nor valid PD detection signature has been found.

mr_valid_sig_pri: This variable indicates that the PSE has detected a valid signature.
Values:
FALSE: No valid signature detected.
TRUE: Valid signature detected.

do_detect_sec
This function returns the following variables (see 33.2.5):

sig_sec: This variable indicates the presence or absence of a valid PD detection signature on the Secondary Alternative.
Values:
open_circuit: The PSE has detected an open circuit.
valid: The PSE has detected a PD requesting power.
invalid: Neither open circuit nor valid PD detection signature has been found.

mr_valid_sig_sec: This variable indicates that the PSE has detected a valid signature.
Values:
FALSE: No valid signature detected.
TRUE: Valid signature detected.

do_mark
This function produces the classification mark event voltage. This function does not return any variables.

set_parameter_type
This function is used by a PSE to evaluate the type of PD connected to the link based on Physical Layer classification or Data Link Layer classification results. The PSE's PI electrical requirements defined in Table 33–11 are set to values corresponding to either a Type 1, or Type 2, Type 3, or Type 4 PSE. This function returns the following variable:

parameter_type: A variable used by a PSE to pick between Type 1, and Type 2, Type 3 and Type 4 PI electrical requirement parameter values defined in Table 33–11.
Values:

1: Type 1 PSE parameter values (default)
2: Type 2 PSE parameter values
3: Type 3 PSE parameter values
4: Type 4 PSE parameter values

When a Type 2 PSE powers a Type 2, Type 3 or Type 4 PD, the PSE may choose to assign a value of '1' to parameter_type if mutual identification is not complete (see 33.2.6) and shall assign a value of '2' to parameter_type if mutual identification is complete.

Editor's Note: This paragraph requires further study.