

DLL Classification PSE Power Increase

David Tremblay
Hewlett Packard Enterprise
Rev 2.3

List of Supporters

Manfred Arndt	Hewlett Packard Enterprise co-author ANSI/TIA-1057 (LLDP-MED)
David Law	Hewlett Packard Enterprise
Wael Diab	Huawei
Yan Zhuang	Huawei
Yair Darshan	Microsemi

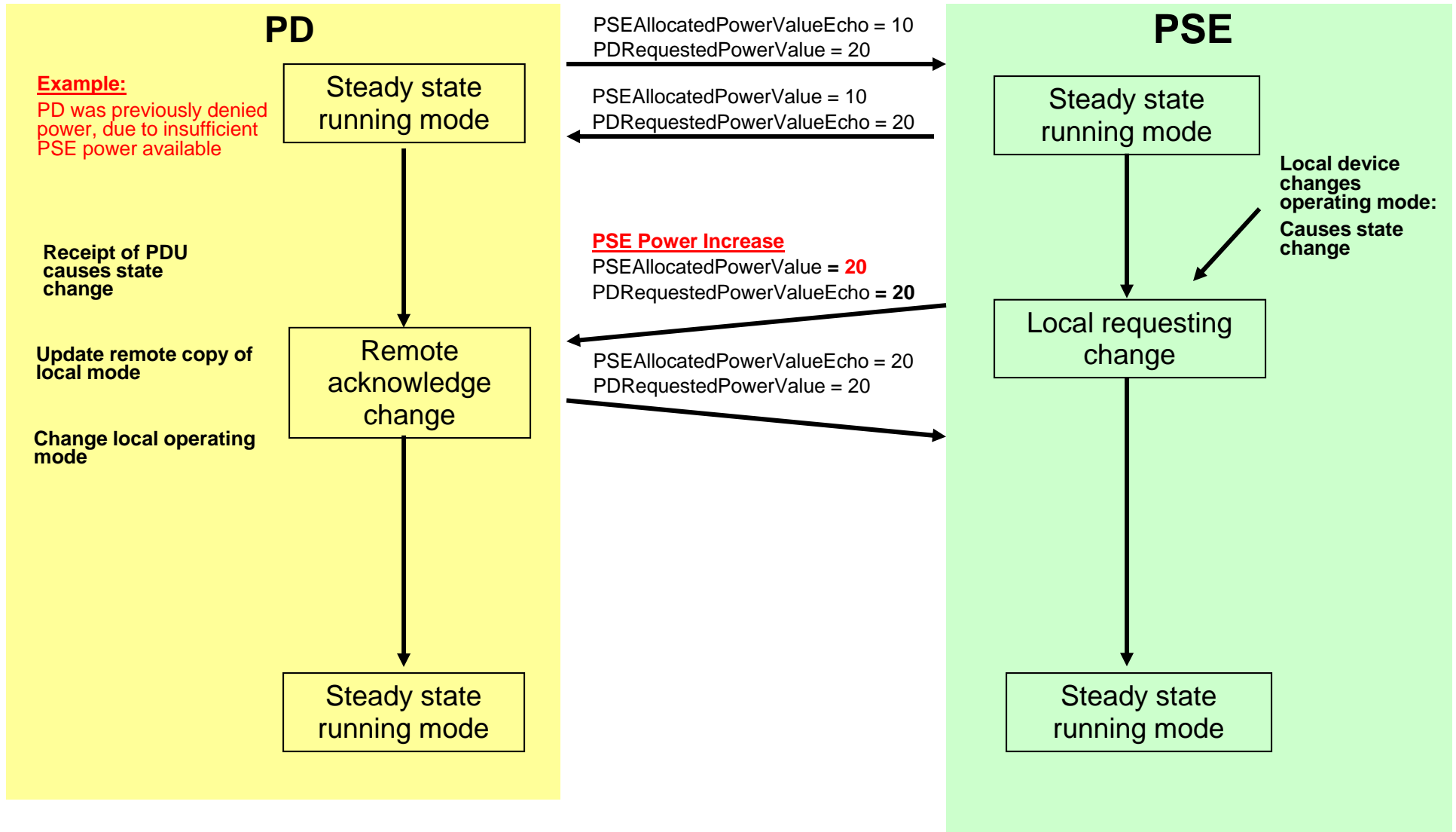
Scope

- Propose a flexible and efficient method for PSE to increase power using DLL:
 - Allow PSE to advertise a power increase within physical layer classification limits, when additional power budget becomes available
- Must be backwards compatible with legacy IEEE 802.3 Type 1 and Type 2 devices

DLL Usage Rules

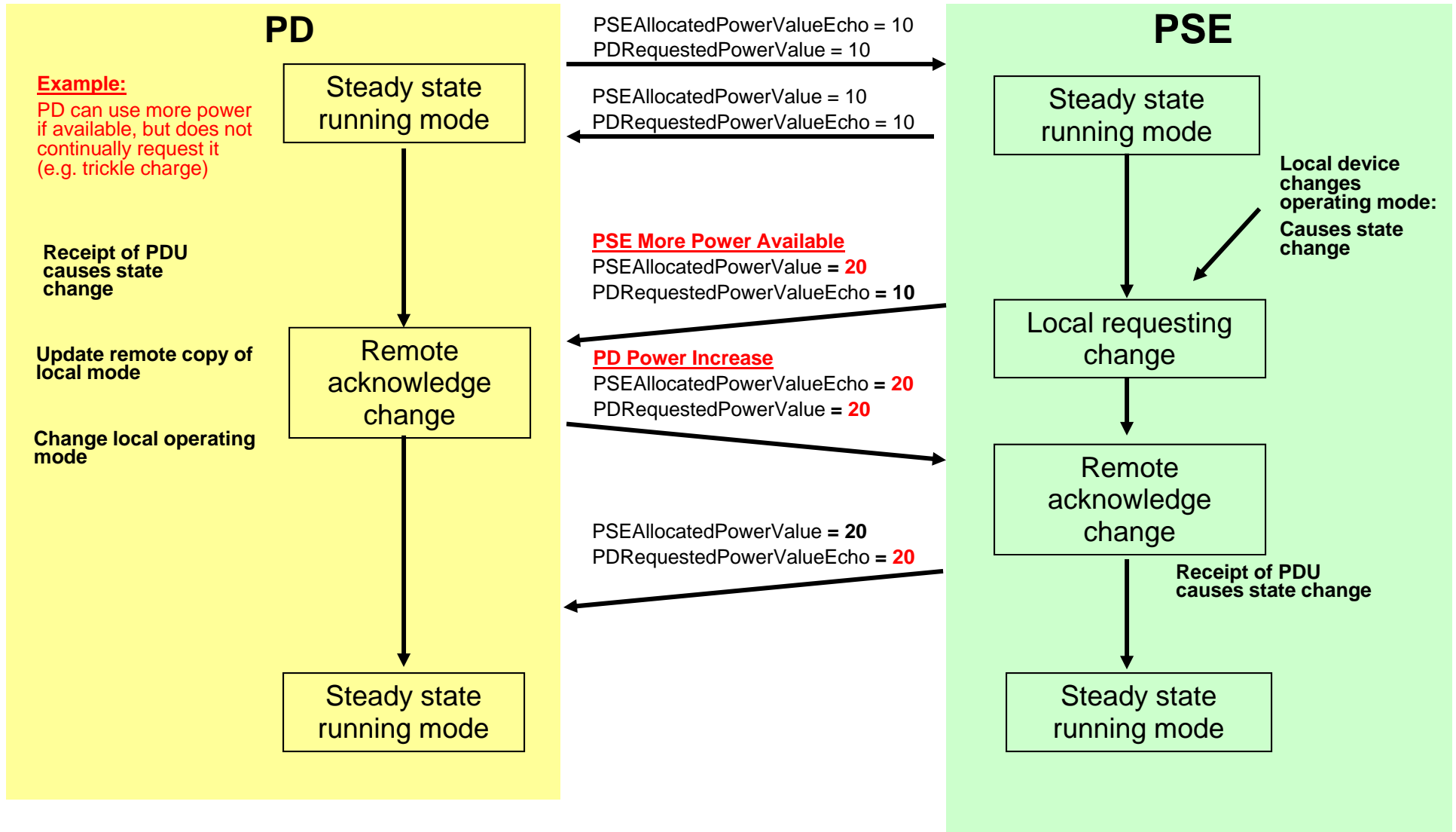
- PD shall never request more power than physical layer classification
- PD shall never draw more than max power advertised by PSE
- PSE may deny any PD drawing more power than max allowed by PSE
- PSE may request reduced power allocated to PD
- If PSE advertises additional power available, then PD can increase power immediately and advertise power usage

Successful PSE Power Increase - Case 1



Note: PSE may advertise more power available than PD requests (as long as sufficient power pool budget exists)

Successful PSE Power Increase - Case 2



DLL PSE State Change - D1.7

33.6.4.1 PSE state change procedure across a link

A PSE is considered to be in sync with the PD when the value of PSEAllocatedPowerValue matches the value of MirroredPSEAllocatedPowerValueEcho. When the PSE is not in sync with the PD, the PSE is only allowed to decrease its power allocation.

During normal operation, the PSE is in the RUNNING state. If the PSE wants to initiate a change in the PD allocation, the local_system_change is asserted and the PSE enters the PSE POWER REVIEW state, where a new power allocation value, PSE_NEW_VALUE, is computed. If the PSE is in sync with the PD or if PSE_NEW_VALUE is smaller than PSEAllocatedPowerValue, it enters the MIRROR UPDATE state where PSE_NEW_VALUE is assigned to PSEAllocatedPowerValue. It also updates PDRequestedPowerValueEcho and returns to the RUNNING state.

If the PSE sees a change to the previously stored MirroredPDRequestedPowerValue, it recognizes a request by the PD to change its power allocation. It entertains this request only when it is in sync with the PD. The PSE examines the request by entering the PD POWER REQUEST state. A new power allocation value, PSE_NEW_VALUE, is computed. It then enters the MIRROR UPDATE state where PSE_NEW_VALUE is assigned to PSEAllocatedPowerValue. It also updates PDRequestedPowerValueEcho and returns to the RUNNING state.

Text does not explicitly include a state change for PSE power increase

Proposed Baseline Text

33.6.4.1 PSE state change procedure across a link

18

19

A PSE is considered to be in sync with the PD when the value of PSEAllocatedPowerValue matches the value of MirroredPSEAllocatedPowerValueEcho. When the PSE is not in sync with the PD, the PSE is ~~only~~ allowed to ~~decrease~~ its power allocation.
~~change~~

20

21

22

23

During normal operation, the PSE is in the RUNNING state. If the PSE wants to initiate a change in the PD allocation, the local_system_change is asserted and the PSE enters the PSE POWER REVIEW state, where a new power allocation value, PSE_NEW_VALUE, is computed. If the PSE is in sync with the PD or if PSE_NEW_VALUE is ~~smaller~~ ^{different} than PSEAllocatedPowerValue, it enters the MIRROR UPDATE state where PSE_NEW_VALUE is assigned to PSEAllocatedPowerValue. It also updates PDRequestedPowerValueEcho and returns to the RUNNING state.

24

25

26

27

28

29

The general state change procedure for PSEs is shown in Figure 33–49.

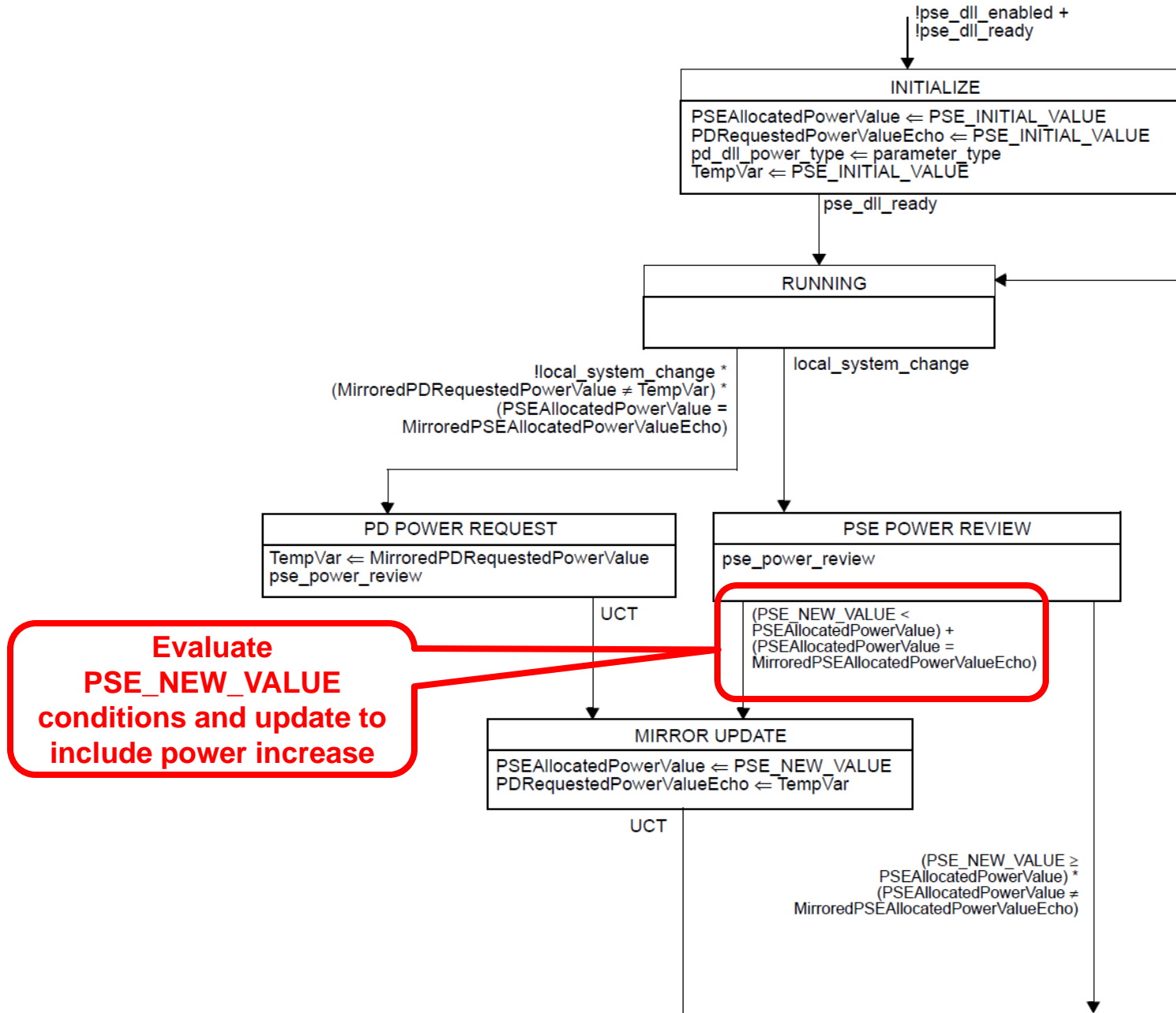


Figure 33–49—PSE power control state diagram

Backup Slides

Motion

Adopt proposed baseline text in tremblay_01_0516_rev02_03, evaluate PSE_NEW_VALUE conditions and update to include power increase in Fig 33-49

Mover: David Tremblay Second: Yair Darshan

Yes: 15 Abstain: 4 No: 0

Motion Passes