802.3az January 09 Interim: LLDP's Use in EEE

Carlson, Steve – HSD
Booth, Brad – AMCC
Diab, Wael William – Broadcom
Dietz, Bryan – Alcatel-Lucent
Dove, Dan – HP
Law, David – 3COM
Vetteth, Anoop – Cisco

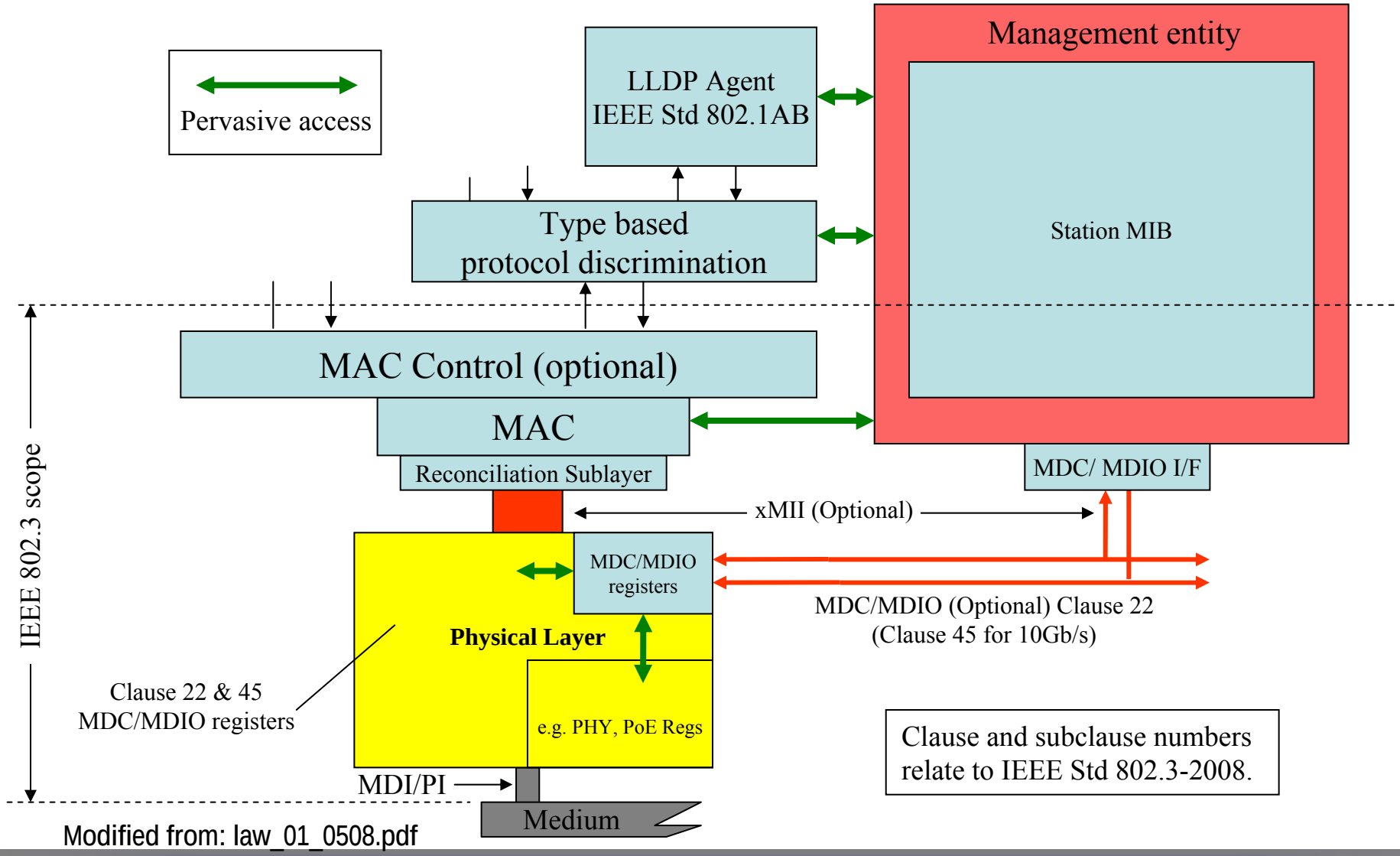
…Other Supporters Welcome…

Background and Overview

- Current usage of LLDP to re-negotiate system wake-up times in 78.4.2.5 of D1.1 does not match how LLDP works
- Purpose of presentation
  - Review how LLDP works
    - What it can and cannot do
  - Summarize and agree on functionality required in 78.4.2.5
  - Review work done in .3at (PoEP) for dynamic power negotiation/allocation and feedback on SM from 802.1
  - Propose a framework based on the above and process as a starting point for D1.2

LLDP Review

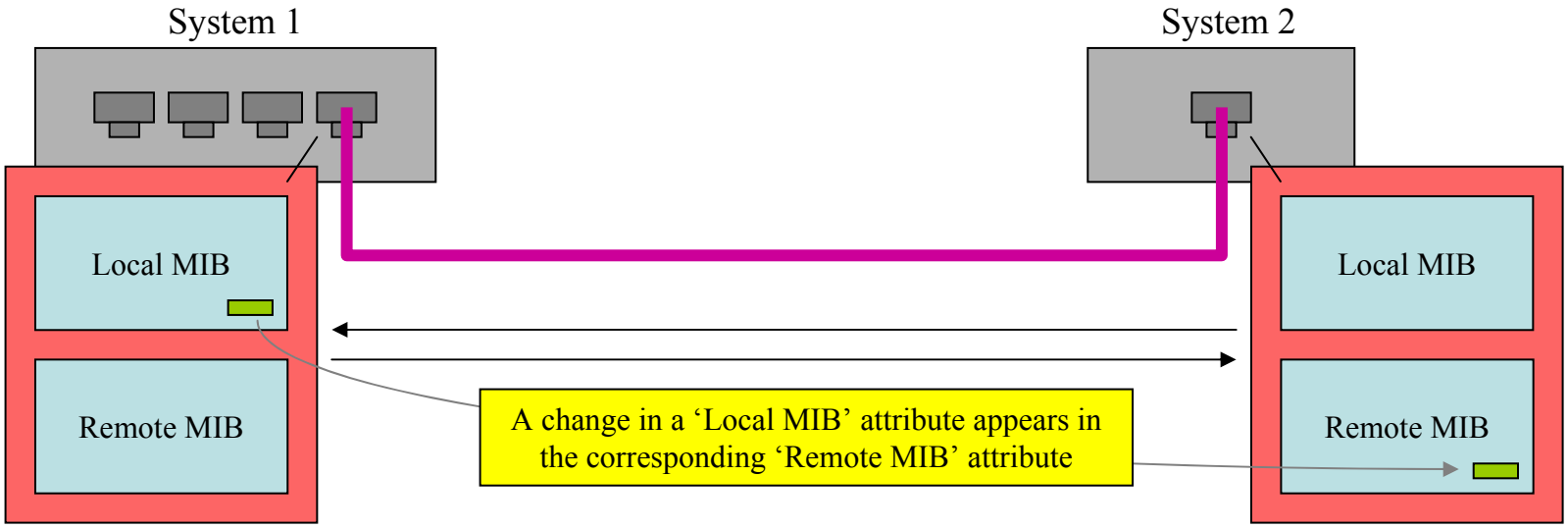
Mgmt Access (LLDP vs. MDC/MDIO)



Modified from: law_01_0508.pdf

LLDP Overview

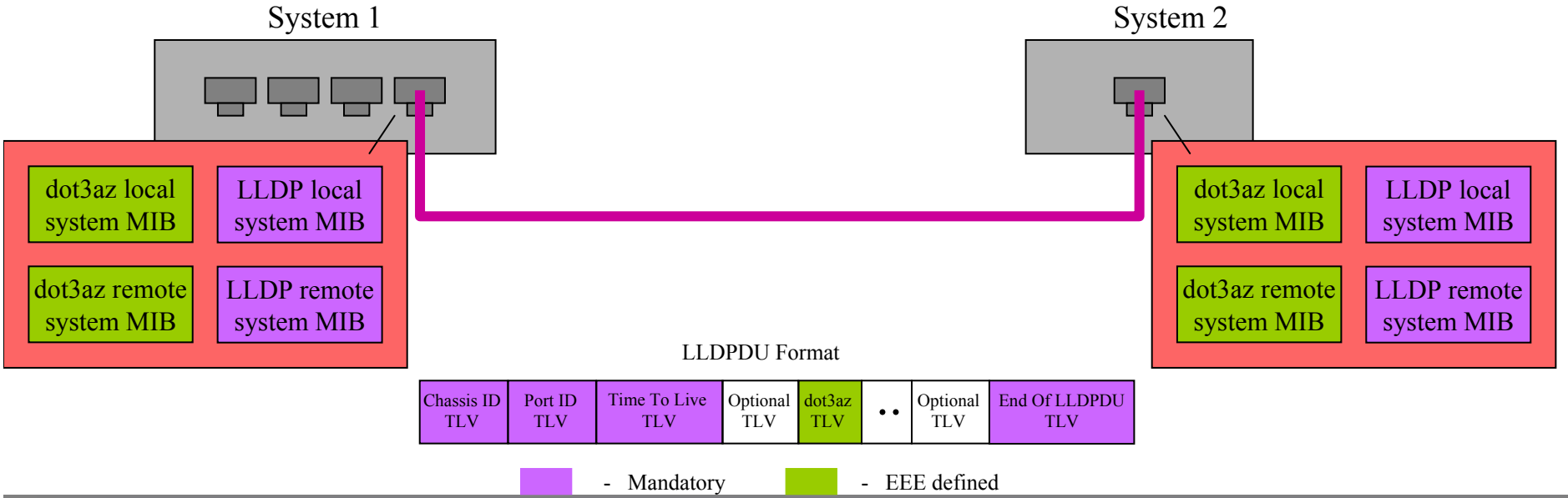
- Operates over a point to point link
- Completely enclosed protocol
  - We define data, it gets transported
    - We don't get to make changes to the protocol
- Data in ‘Local MIB’ transported to ‘Remote MIB’
  - Transported by TLVs (type, length, value)



Source: law_01_0508.pdf

LLDPDU and Associated TLVs

- The LLDP frame consists of an LLDPDU (LLDP Data Unit)
  - LLDPDU is constructed from mandatory TLVs and optional TLVs
  - Mandatory TLVs are chasis ID, Port ID and TTL
  - Optional TLVs can be management TLVs or organizationally specific TLVs
  - Selection of optional TLVs used in the LLDPDU is under network management's control
- TLVs are associated with a station's MIB
  - Mandatory basic LLDP MIB: Associated with basic TLVs
  - Optional LLDP MIB extensions: Associated with optional TLVs
  - IEEE P802.3az needs to define an LLDP MIB extension and associated TLV
- **Consequence: LLDPDU contains more than EEE's TLV and exchange may be triggered by other TLV changes**



To Summarize…

- What LLDP *can* do
  - **Transport** parameters defined in TLVs across a link
  - **Keep** a copy of the remote and local value of a parameter
  - **Automatically** initiate an update upon a change in the local variable's value and/or notify the local agent of a remote change
  - **Support** SM (See guidance from 802.1 in following section)
  - Offers benefits of a **packet based protocol**: CRC protection, so there is no need to worry about this
- What LLDP *cannot* do
  - **Force** specific number of LLDPUs to go out for a single change
  - **Rely** on a fixed rate of exchange
    - There are mechanisms to ensure "quick" updates but there are system interactions that may not be under the control of EEE
  - **Assume** that a LLDPU received or transmitted is due to a specific change in a specific TLV
    - Changes elsewhere in a station's MIB or MIB extension can trigger an LLDPDU exchange. Delay timers to consolidate TLV changes into fewer LLDPUs exist but do not eliminate the issue and their use may be constrained by other system requirements

EEE's Desired Functionality with Respect to LLDP

Functionality w.r.t LLDP

- There seems to be 2 basic requirements:
  - Initial capability exchange of Tw_sys
    - Upon initialization exchange the Tw_sys parameters to allow for the resolution process to occur
  - Dynamic negotiation of Tw_sys
    - At any time during operation, allow either link partner in either path to initiate a change its Tw_sys to allow for power savings / performance dynamic optimization
- Any other high-level functionality missing?

Requirements w.r.t LLDP

- The resolution of Tw_sys gates when data will appear on the link upon exit from LPI
  - This is true during the initial exchange (startup) and during a subsequent exchange during operation (dynamic)
  - Hence, it is important that the following is maintained so the data integrity is not compromised
    - Both link partners perform the Tw_sys resolution based on the *most recently advertised capability*
    - Both link partners are confident that the remote side has completed its resolution prior to changing its own behavior
    - Avoid deadlocks
- As with the control policy, the rate of change of Tw_sys for a system is outside the scope of .3az
  - There may be other system constraints like LLDP itself
  - As noted, measuring and enforcing a rate may be difficult
  - Limits may need to be set by EEE but to the extent possible avoid
- A simple SM similar to PoEP L2 negotiation may suffice

Review of 802.3at's LLDP Work and .1 Guidance on SM

PoEP's L2: Motivation and SM

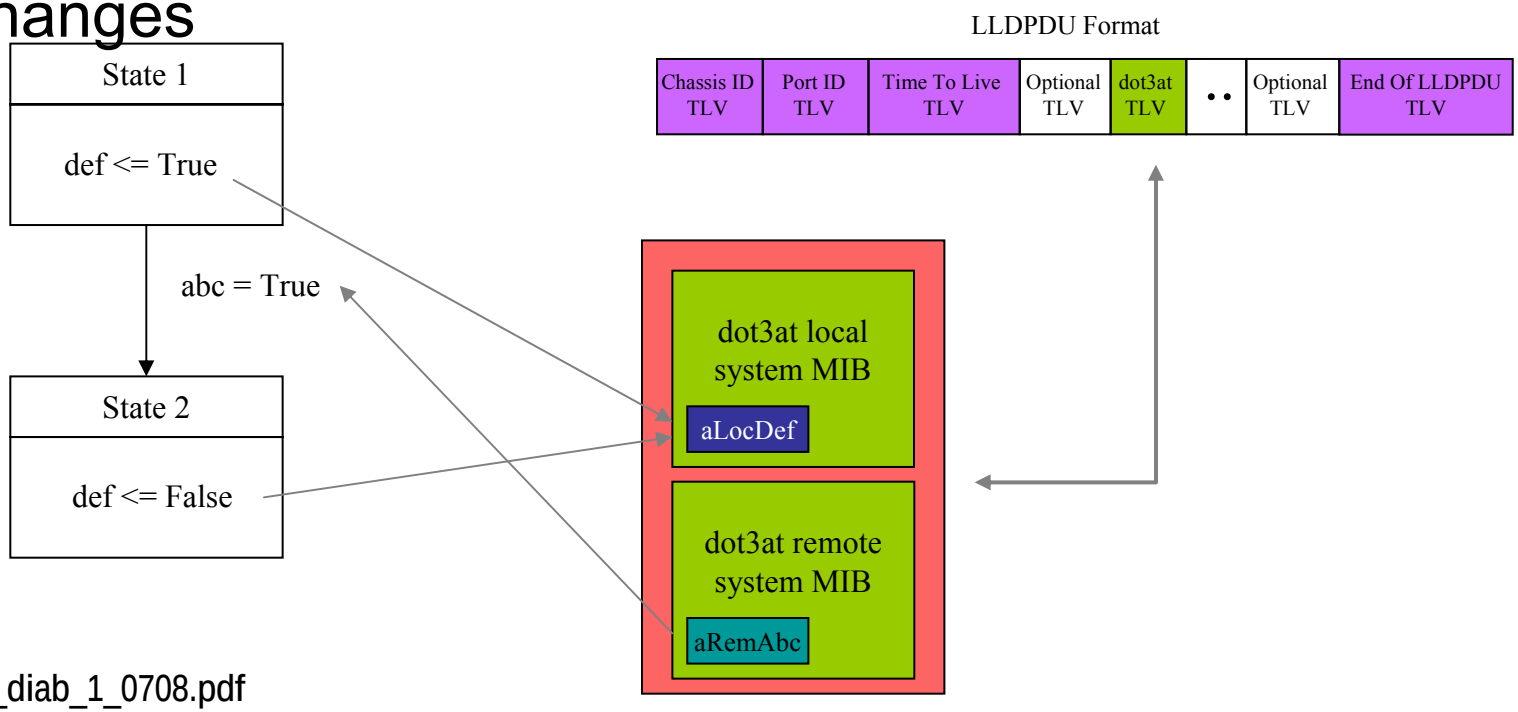
- PoEP wanted to use LLDP for a dynamic power allocation between the PD and PSE
  - In an end-span configuration PSE and PD are link partners
  - PD may request new power allocation. PSE responds to PD's request and/or initiates re-allocation
- Similarities between PoEP's L2 and EEE
  - Desire to use LLDP as it is a widely deployed protocol
  - Value of parameters exchanged set by "upper layers"
  - Startup and dynamic requirements on solution
  - Decision on allocation requires assurance that information being acted on is most recent
    - Avoid deadlocks
  - Prior to changing behavior, PSE/PD needs to be confident other side is ready – random power change may have drastic effects!
  - One side has to enforce the decision (PSE for PoEP, TX for EEE)
- Differences between PoEP's L2 and EEE
  - Behavior for PSE and PD unidirectional. EEE has duplex
    - EEE instantiates a TX and an RX for each station

PoEP's L2: Adopted Mechanism

- Basic variables used for power negotiation
  - PSE allocates power using PSEAllocatedPowerValue
  - PD requests power using PDRequestedPowerValue
- Provisions to ensure accuracy in resolution
  - Both the PSE and the PD echo back what they believe the latest value the remote side sent them. For example, the PD echo's back the PSE's Allocated Power Value it has in its MIB
  - Both sides retain memory of the power value set for each review as the review may not be conducted real time. i.e. an updated value may have been sent since
  - State machine to ensure above occurs
    - SM intended to avoid deadlocks while providing an ACK
    - SM does not constrain choice of values – that is for system

Review: LLDP and State diagrams

- Can't map directly to TLV contents
  - Map through objects in dot3at local and remote MIB
  - Define MIB attribute to variable mapping
  - Allows .3 layers to take action based on variable changes

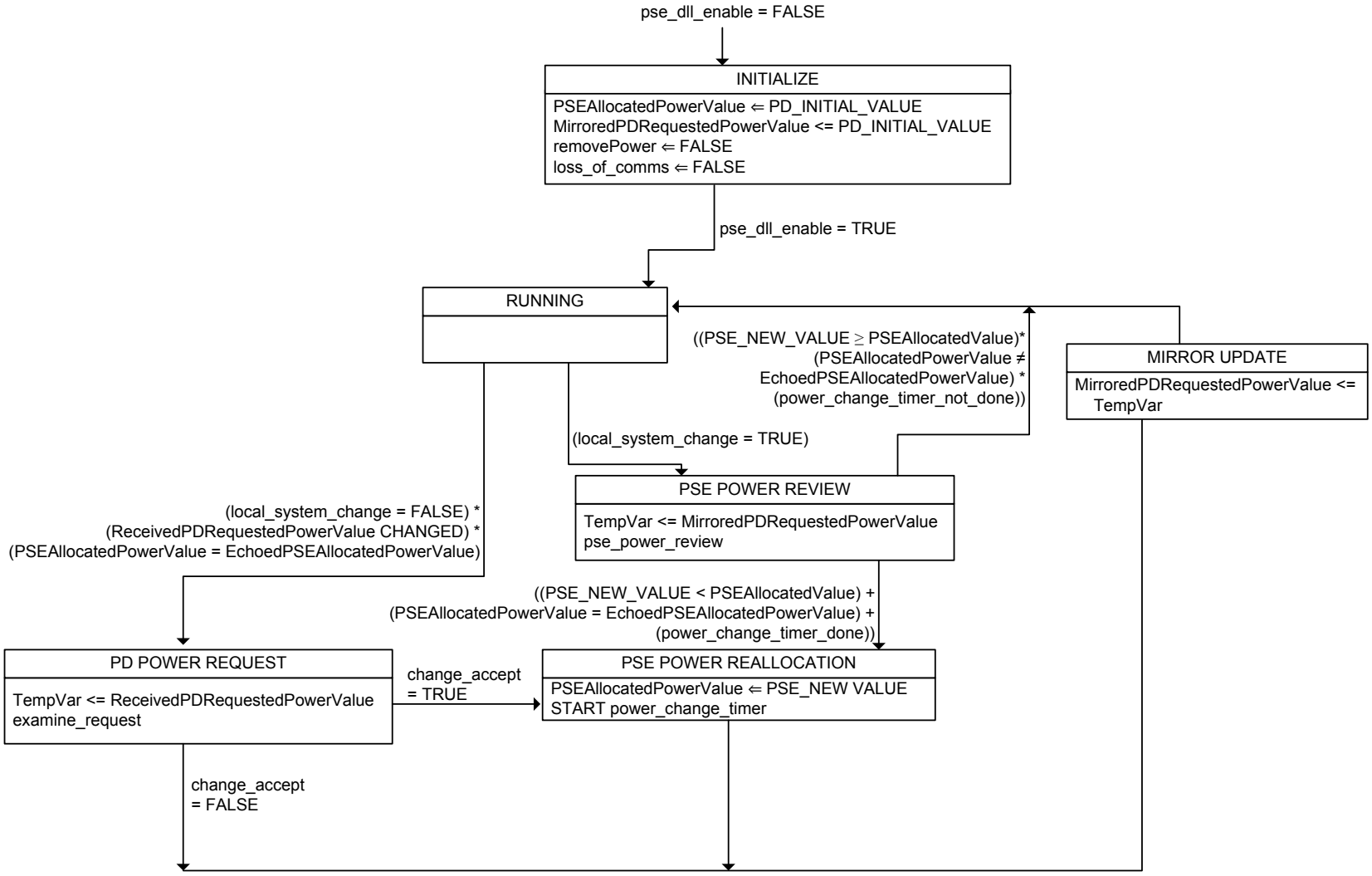


Source: joint_diab_1_0708.pdf

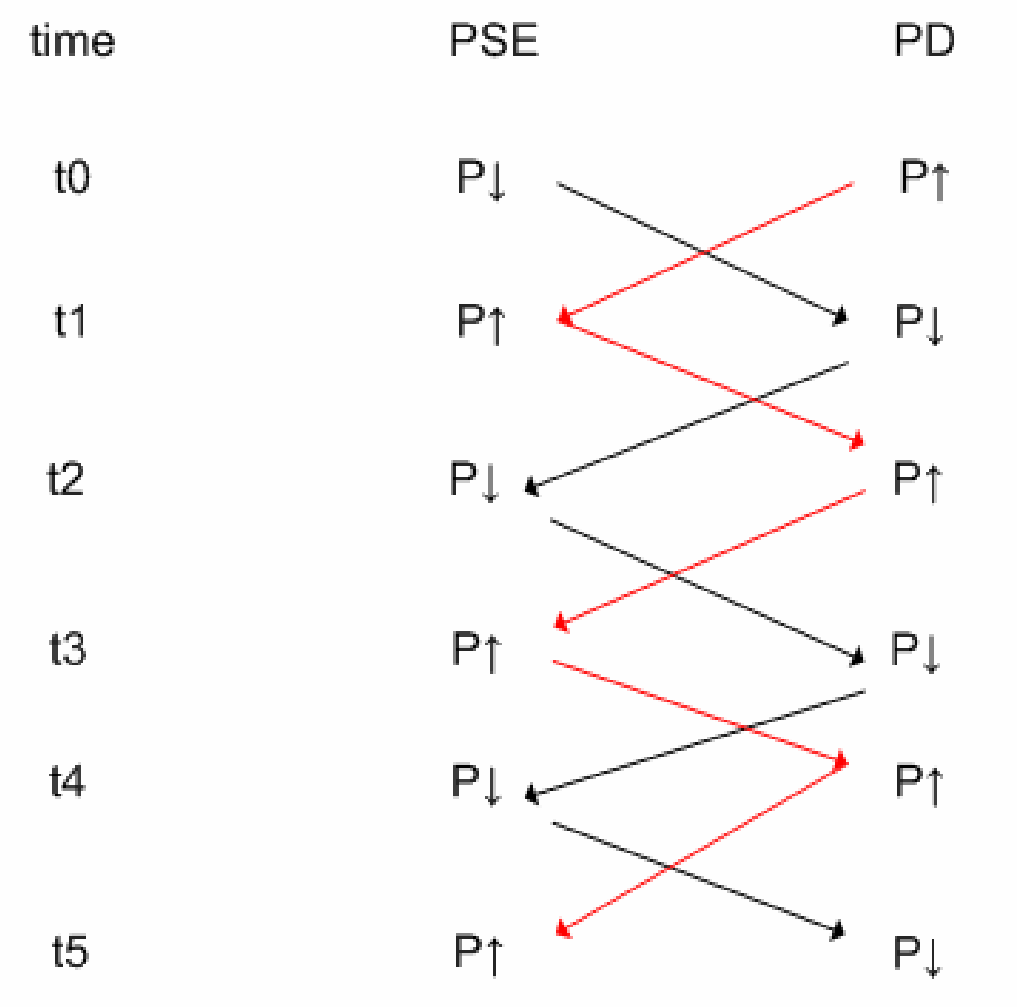
802.1's Guidance for use of a SM w/LLDP

- No fundamental problem to do State Machine
- Preferably don't do ACK/NACKs, if you do, you need serial numbers
- Don't make it too chatty
  - LLDP may be running other protocols
  - Minimize the number of frames transmitted
- 802.1 expertise may be available to help
- Opportunity for 802.1 members to ballot in WG on 802.3at
  - Request based system
  - Same for 802.3az

Example 802.3at PSE SM



Keeping Track of The Value Set



Keeping Track of The Value Set

- Value advertised by the local partner in some part may depend on the value being advertised by the remote
- Since LLDP's agents and review process may not be real time, a review process may be operating on stale information and/or out of synch information. This can cause unwanted positive feedback
- To ensure this does not happen
  - The Mirrored value is the received value corresponding to which the power review is conducted
  - If a PSE receives a PDU where the echoed value does not match the Allocated Power Value, it ignores the PDU
  - If a PD receives a PDU where echoed value does not match Requested Power Value, it continues to treat the PDU as valid
- EEE: Control policy may react to a change in buffering on the remote side, then theoretically this could occur
  - Can put in a similar mechanism into the SM or chose to ignore if its not a practical concern

Baseline Proposal for EEE's LLDP as a Starting Point for D1.2

Document structure

- IEEE Std 802.1AB subclause 9.6 Organizationally Specific TLVs

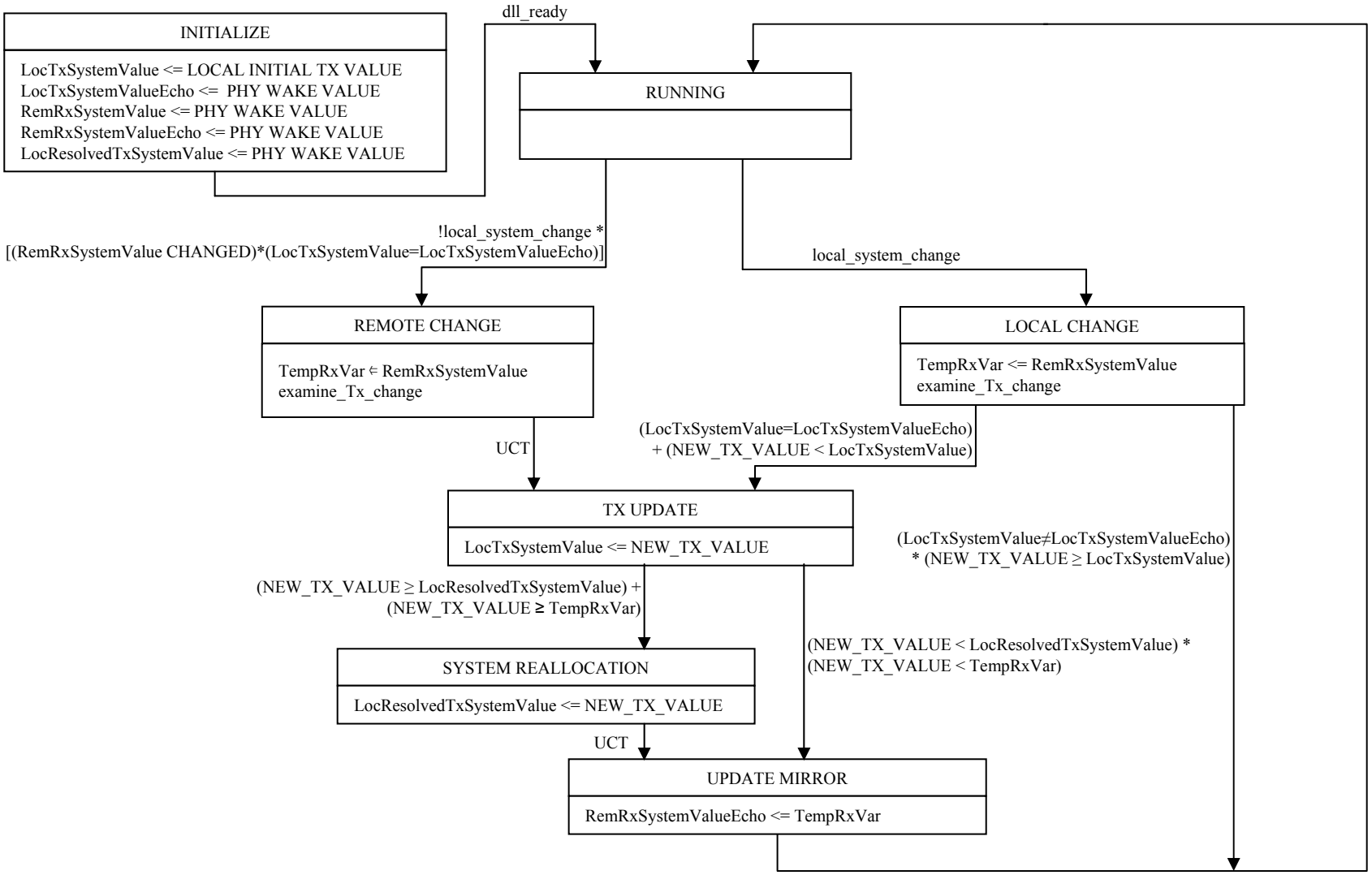
  'Each set of Organizationally Specific TLVs shall include associated LLDP MIB extensions and the associated TLV selection management variables and MIB/TLV cross reference tables (for example, see F.6 and G.6).'

  – For 802.3, this is being moved to C77 as part of P802.3bc
- Hence to use LLDP IEEE P802.3az has to define
  – LLDP TLV selection management variables
  – LLDP MIB extensions
  – MIB/TLV cross reference table
- IEEE P802.3az also needs to define
  – MIB to state diagram cross reference table
  – State diagram using MIB derived variables

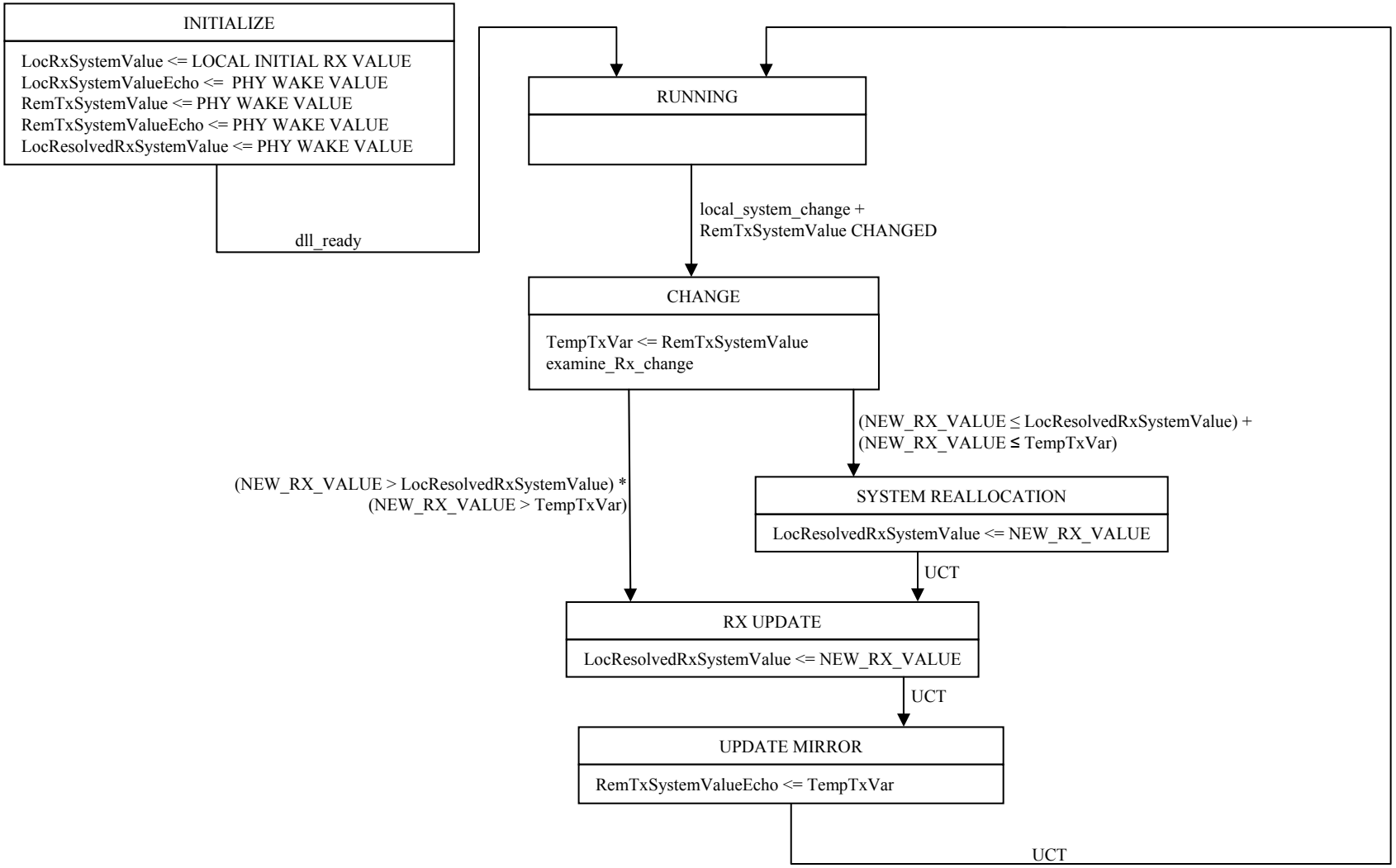
Summary of Basic Behaviour

- TX enforces the Tw_sys
  - Once TX allocates a Tw_sys, the RX has to live within it
  - Tw_sys_tx represents the system wakeup time the transmitter can support
  - Tw_sys_rx represents the receiver desired system wakeup time
  - Goal is to get an optimal Tw_sys that the TX side can support and allows for the RX to save energy
- Cases
  - Tw_sys_rx = Tw_sys_tx
    - Unlikely but ideal
  - Tw_sys_rx < Tw_sys_tx
    - The transmitter can support the receiver's requirement
    - Transmitter is free to retain excess allocation or reduce to RX
  - Tw_sys_rx > Tw_sys_tx
    - The transmitter cannot support the receiver's requirement
    - RX can either live within the TX allocation or only save PHY power
- Similar to PD-PSE power allocation

Initial Stab at TX SM



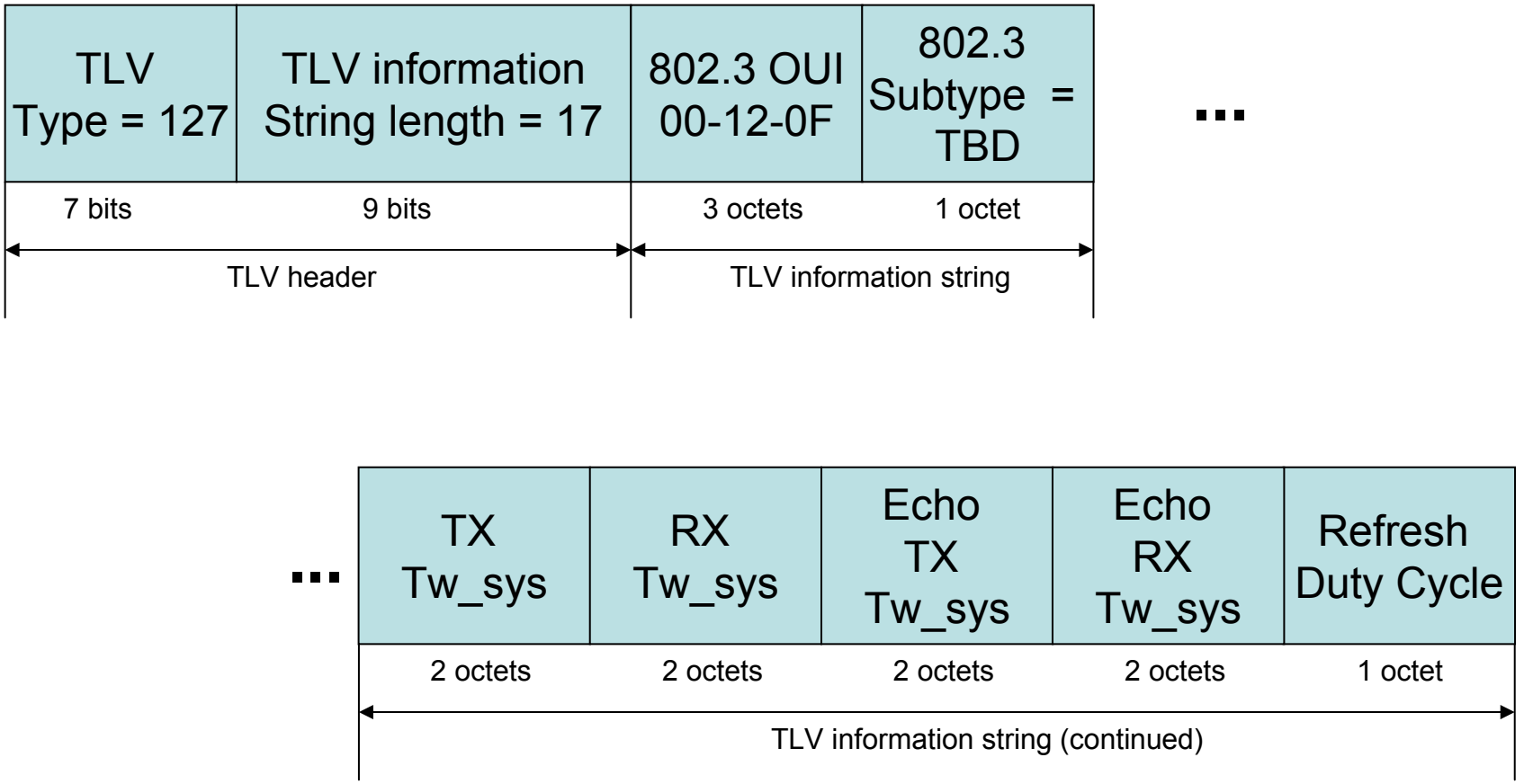
Initial Stab at RX SM



Definitions

- Initial values
  - These are the values used upon initialization
  - For the local side it's the initial value the system wants
  - For the remote and resolved variables it's the PHY negotiated times
- Rem
  - These are the values that show up in the MIB for the link partner (i.e. the remote partner's information)
  - E.g. Tw_sys TX will be the local partner's TX value and the mirror is derived from the remote partner's value that appears in the MIB
- Echo
  - The local partner's reflection (echo) of the remote's values
    - Echoing a value reflects its receipt but not necessarily that new resolved levels
  - When this value doesn't match the local value then the link partner knows the request was based on stale information
- Resolved
  - This is the actual value allocated for each direction (Tx and Rx)

Energy Efficient Ethernet TLV



Initial Stab at TLV to MIB mapping

Energy Efficient Ethernet TLV to EEE object class cross-references

TLV name	TLV variable	Clause 30 attribute
Energy Efficient Ethernet	Tx Tw_sys	aEEELocTxTwSys
	Rx Tw_sys	aEEELocRxTwSys
	Echo Tx Tw_sys	aEEERemTxTwSysEcho
	Echo Rx Tw_sys	aEEERemRxTwSysEcho
	Refresh Duty Cycle	aEEERefreshDutyCycle

Initial Stab at MIB to Variable Mapping

Attribute to state diagram variable cross-reference

Object	Attribute	Mapping	State diagram variable
oEEE managed object class	aEEELocTxTwSys	<=	LocTxSystemValue
	aEEELocRxTwSys	<=	LocRxSystemValue
	aEEELocTxTwSysEcho	=>	LocTxSystemValueEcho
	aEEELocRxTwSysEcho	=>	LocRxSystemValueEcho
	aEEERemTxTwSys	=>	RemTxSystemValue
	aEEERemRxTwSys	=>	RemRxSystemValue
	aEEERemTxTwSysEcho	<=	RemTxSystemValueEcho
	aEEERemRxTwSysEcho	<=	RemRxSystemValueEcho

Other Startup Conditions

- A time is required for the initial LLDPDU to be sent upon start-up
- As a placeholder suggest using the following text
  - An EEE link partner shall send an LLDPDU containing an EEE TLV within 10 seconds of the Link Layer capability exchange being enabled as indicated by the variable dll_enabled (ref)
- The variable that kick starts the DLL (dll_enabled) machine needs to be kicked off at the end of the PHY auto-negotiation process for EEE
- Each system shall instantiate both RX and TX SMs
  - Similar to DCB TG

Other Reaction Conditions

- Reaction time to receiving an updated LLDPDU
- A system reaction time for processing a change (based on receiving a change) and sending out an updated TLV is required
- As a placeholder suggest using the following text
  - Under normal operation, an LLDPDU containing an EEE TLV with an updated value for the “Echo TX Tw_sys” field shall be sent within 10 seconds of receipt of an LLDPDU containing an EEE TLV when the corresponding “TX Tw_sys”
  - Under normal operation, an LLDPDU containing an EEE TLV with an updated value for the “Echo RX Tw_sys” field shall be sent within 10 seconds of receipt of an LLDPDU containing an EEE TLV when the corresponding “RX Tw_sys”

Proposed Next Steps

- Adopt a framework for the draft as outlined in the document structure
- Discuss the behaviour desired from the L2 protocol
- Refine the SM based on the behaviour
- Refine the MIB, variables etc. based on above
- Procedurally
  - Propose basic framework included in D1.2 with an editor's note outlining the work TBD
  - Suggest an L2-adhoc to refine text for draft and bring into March for TF review

Motion

- Move that:
- The IEEE 802.3az TF adopt diab_02_0109.pdf (pages 21 – 29) as the baseline for inclusion in D1.2, with an editor's note outlining the work to be done, and to charter an ad hoc to progress this work.
- Move: S. Carlson
- Second: A. Vetteth
- Technical 75%
- Y__18__ N__0__ A__2__
- Passes
- 13th January, 2009 5:23PM