ILT local_pattern specification

(in support of comments 415, 416, 417)

Adee Ran, Cisco

Overview

- The ILT state diagrams support path start-up either with or without a training protocol (exchange of training frames).
- When training is not used, the “readiness” status (RTS) should be communicated by transmission of a locally generated signal (tx_mode = local_pattern).
- The local pattern is currently not specified anywhere.
- This presentation provides some clarification of the concept and suggests adding specific local patterns for the PMDs / AUIs that use PAM4 signaling.

Note on terminology

- In this presentation I use the term “ILT” to denote the functionality specified by Annex 178B. It includes both the **training protocol** and the **path start-up procedure**.
  - I make a distinction between the two concepts when required.

Motivation for addressing the case of no training protocol

- Before we had ILT, the behaviors of the AUI and PMD were independent
- Other specifications (e.g. SFF and CMIS) required that the module's "transmitter output" (PMD) is **auto-squelched** when the "transmitter input" (AUI) has a fault (loss of lock, loss of signal)
  - Also in the receive direction, in some cases
  - This was never strictly required by Ethernet but that's what everyone did… (or should have done)
  - If auto-squelch is not used, path start-up is undefined and can cause "interesting problems"
- ILT recognizes the common existing behavior and formalizes the auto-squelch function using the QUIET state (tx_disable=true)
  - This enables existing modules to participate in the **path start-up procedure** even though it isn't formally defined
- ILT also allows operation with the training protocol disabled, even when it is defined
  - Some use cases may choose to disable the **training protocol** (mr_training_enable=false) on specific ISLs, for various reasons
  - Users who choose that must use management to set the training-controlled parameters (Tx equalization and precoding) on both sides
  - This choice was available in previous "training" specifications
  - Possible example for usage: a module can be configured to use the training protocol only when initially plugged into a host, and in subsequent activations the same settings are used without re-training.

What happens in ILT when training is not used

- The left-hand part of the diagram is used when mr_training_enable=false.
- While local_rts=false, state is QUIET
  - Auto-squelch
- When local_rts becomes true, it is signaled to the peer by enabling the transmitter
  - But what signal should be transmitted?

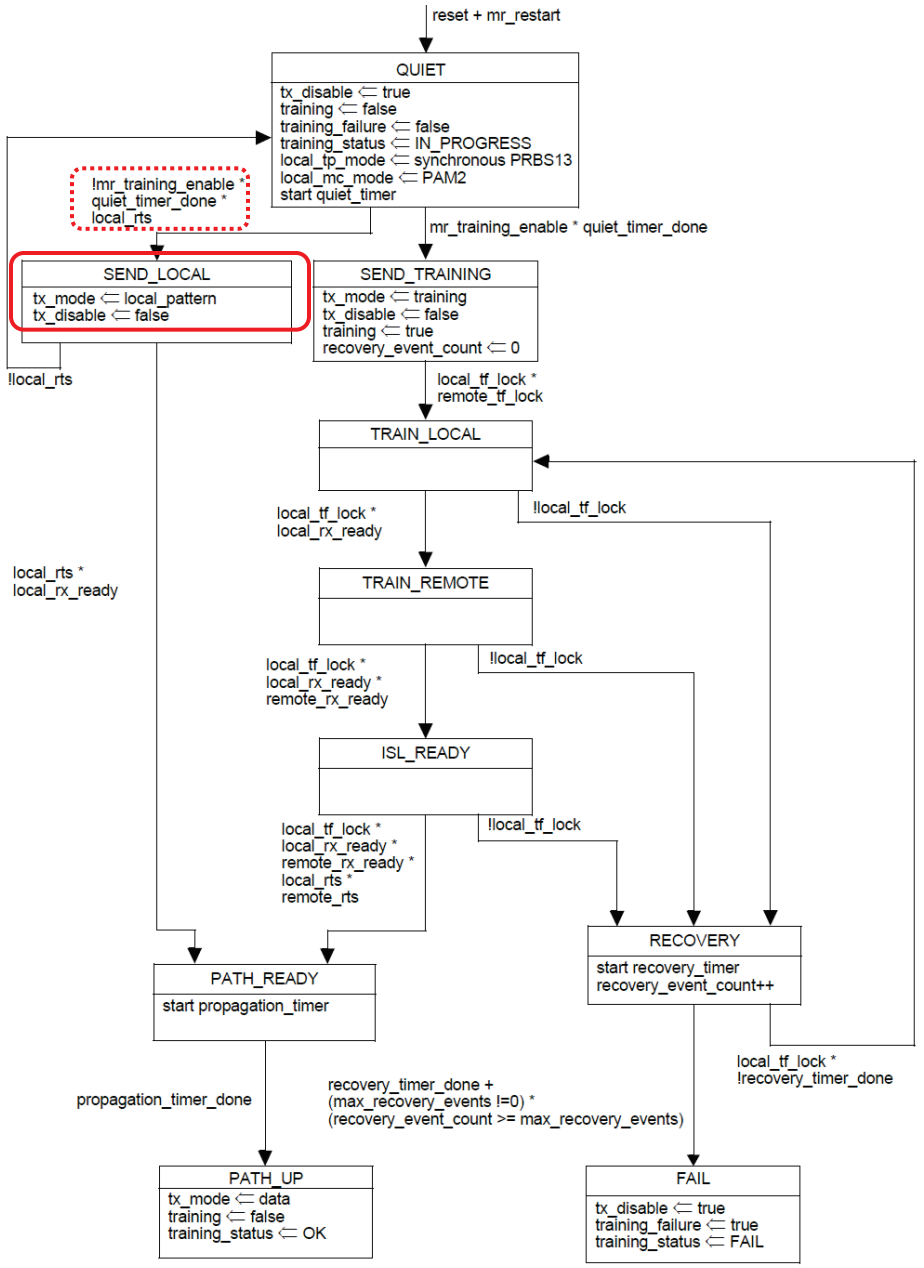
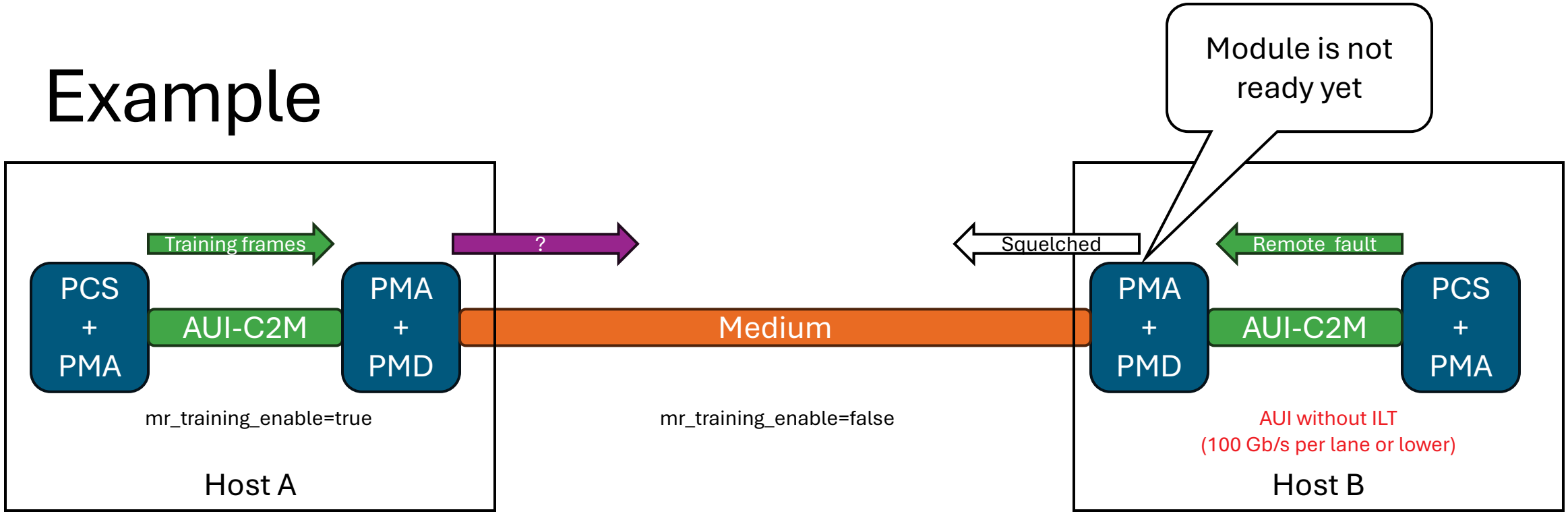


Figure 178B–8—Training control state diagram

Behavior with local_rts=true

- Existing modules go into “DATA mode” when their AUI becomes functional
  - For example, “DPActivated” state in CMIS
  - In this state the data stream received by the AUI is transmitted by the PMD and vice versa
- What happens in a new module (with ILT capability) if the AUI uses the training protocol (mr_training_enable=true), but the PMD does not?
  - If the PMD has the same “DATA mode” behavior, then training frames from the AUI will be sent by the PMD to the link partner
  - This can lead to “interesting problems” again, as shown in the next slide

Example



What should the PMD on host A send towards host B when it has local_rts=true?

- If it forwards the training frames from the AUI, host B may receive “synchronous PRBS13Q”
- This may not be a good pattern for receiver adaptation (and it can’t be changed)
- The PMD and/or AUI receiver on host B probably assume/expect a scrambled remote fault at link startup

To avoid that, **training frames should not be forwarded**

- Something else should be sent instead ➔ tx_mode = local_pattern

When is local_pattern sent?

- Only when local_rts=true (condition to enter SEND_LOCAL)
- This means USE_TX_CLOCK(recovered) has already occurred
  - The transition from tx_mode=local_pattern to tx_mode=data occurs later, in PATH_UP state, after local_rx_ready is asserted
- **tx_mode=local_pattern is a way to communicate local_rts when training is not used**

What's missing? (comments 415, 416, 417)

- The pattern sent when tx_mode=local_pattern is not well defined
  - Currently only "a pattern from a valid pattern generator"
  - It could be, e.g., a square wave… not what was intended
- PRBS31Q is a good pattern for training and is implemented everywhere…
  - Except PHYs with inner FEC
  - And it might not be appropriate for other interfaces (e.g., coherent)
- Outline of proposal:
  - Make the pattern used in local_pattern specified by each PMD/AUI that uses Annex 178B (comment #415).
  - Define it as **PRBS31Q** for all AUIs and for PMDs that use PAM4 without inner FEC (comment #416, except that it wrongly included clause 182).
  - Define it as **PRBS31Q encoded by inner FEC** for PMDs that use PAM4 with inner FEC (comment #417, except that it wrongly omitted clause 182).
- Note that the generators for the proposed patterns are defined in the sublayer above (PMA or inner FEC). This should not be a practical problem.

Detailed proposal

- In 178B.14.3.1 (definition of per-lane ILT variables), in the definition of tx_mode, change as follows

  local_pattern: transmits a pattern ~~from a valid pattern generator~~ specified by the AUI component or PMD that uses the ILT function.

- In 178.8.9, 179.8.9, 180.5.12, and 181.5.12 (ILT function of PMDs), add the following statement:

  When mr_training_enable=false and tx_mode=local_pattern (see 178B.14.3.1), the PMD transmits PRBS31Q (see 176.7.4.2).

  - This also covers the AUIs because they are defined as functionally equivalent to the electrical PMDs.

- In 182.5.12 and 183.5.12 (ILT function of PMDs with inner FEC), add the following statement:

  When mr_training_enable=false and tx_mode=local_pattern (see 178B.14.3.1), the PMD transmits PRBS31 encoded by Inner FEC (see 177.6.1.1).

That’s all

Questions?