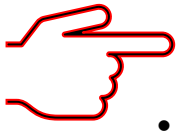
Proposal for changes in the start-up protocol to enable segment-by-segment training – Part 2: Details

Adee Ran, Cisco
Kent Lusted, Intel
Leon Bruckman, Huawei
Anil Mehta, Broadcom
Mike Dudek, Marvell
Ali Ghiasi, Ghiasi Quantum/Marvell
Matt Brown, Alphawave Semi
Zvi Rechtman, NVIDIA

Start-up protocol variable definitions (1)

(Each interface of a PMA, e.g. egress and ingress, has a separate set of these variables)

- mr_training_enable
  - Boolean variable that is set by management. When it is true, training is enabled on the interface. When it is false, training is not enabled on the interface.
- mr_restart
  - Boolean variable used by system management to restart the start-up protocol.



NOTE — To avoid live-lock situations, the start-up protocol should not be restarted based on a timeout on any of the segments, unless there is an indication of an unrecoverable fault.

- local_tf_lock<i> (Abbreviated as local_TFL in state diagrams)
  - Boolean variable that is true when mr_training_enable is true and the training frame marker positions have been identified on lane i of the PMA interface and is false otherwise. The value of this variable is encoded as the “training lock” bit in the status field of transmitted training frames.
- local_rx_ready<i> (Abbreviated as local_RR in state diagrams)
  - Boolean variable that is set to true when the receiver on lane i of the PMA interface has determined that the segment partner’s transmitter is not disabled, that the remote transmit and local receive equalizers have been optimized, and that no further adjustments are required for normal data transmission. It is set to false otherwise. The exact criteria for setting this variable to true are implementation specific. When mr_training_enable is true, the value of this variable is encoded as the “receiver ready” bit in the status field of transmitted training frames.
- remote_tf_lock<i> (Abbreviated as remote_TFL in state diagrams)
  - Boolean variable that indicates the value of local_tf_lock on lane i of the segment partner. If mr_training_enable is true, it is derived from the “receiver frame lock” bit of the status field of received training frames on lane i of the PMA interface. Otherwise (*if mr_training_enable is false*) it is unspecified.
- remote_rx_ready<i> (Abbreviated as remote_RR in state diagrams)
  - Boolean variable that indicates the value of local_rx_ready on lane I of the segment partner. If mr_training_enable is true, it is derived from the “receiver ready” bit of the status field of received training frames on lane i of the PMA interface. Otherwise (*if mr_training_enable is false*) it is set to true.

Start-up protocol variable definitions (2)

(Each interface of a PMA, e.g. egress and ingress, has a separate set of these variables)

- local_RTS
  - Boolean variable that indicates that a PMA interface is ready to send and receive normal data. It is set by the RTS state diagram. The logical-NOT of this variable is encoded as the “extend training” bit in the status field of transmitted training frames.
- remote_RTS
  - Boolean variable that indicates the value of local_RTS in the segment partner. If mr_training_enable is true, it is the logical-NOR of the “extend training” bit of the status field of received training frames on all lanes of the PMA interface (*i.e., true only if the bit is 0 on all lanes*). Otherwise (*if mr_training_enable is false*) it is set to true.
- adjacent_remote_RTS
  - Boolean variable that is set to the value of remote_RTS on the other interface of the PMA.
- segment_ready
  - Boolean variable that is set to true when local_rx_ready<i> and remote_rx_ready<i> are true for all lanes of the interface, and to false otherwise.
- adjacent_segment_ready
  - Boolean variable that is set to the value of segment_ready on the other interface of the PMA.
- client_is_PCS
  - Boolean variable that is true for a PMA interface when its other interface is attached to a PCS or a DTE XS, and false otherwise.

start-up protocol variable definitions (3)

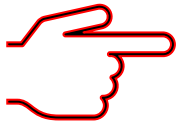
(Each interface of a PMA, e.g. egress and ingress, has a separate set of these variables)

- tx_mode<i>

  Enumerated variable that controls the content of the transmitter's output on lane i of the PMA interface. It is set by the start-up state diagram to one of the values:

  - TRAINING: transmit training frames.
  - LOCAL_PATTERN: transmits a pattern from a valid pattern generator for the PMA, such as PRBS31Q.
  - DATA: transmit data received from the other interface, after processing by the PMA's data path functions.

- tx_disable<i>
  - Boolean variable that controls the transmitter's output on lane i of the PMA interface. It is set by the start-up state diagram. When it is true, the transmitter's output on lane i is disabled.



NOTE—It is recommended that disabling the PMA output results in disabling the output of an attached PMD, such that it will be detectable by the link partner's PMD using the signal detect function. Other behavior is possible, but is beyond the scope of this standard.

The variable list may not be comprehensive.

Functions and timers for the start-up protocol

(Each interface of a PMA, e.g. egress and ingress, has a separate set)

- Function USE_TX_CLOCK(source)

  Selects the clock source for the transmitter. The source parameter takes one of the values:

  - local: use a local clock to drive the transmitter output.
  - recovered: use a clock recovered from the other interface's receiver (see <PMA clock recovery subclause>) to drive the transmitter output.

  NOTE—The details of the clock recovery and forwarding architecture are beyond the scope of this standard.

- forward_RTS_timer

  This timer is started when the RTS state diagram enters the SWITCH_CLOCK state. The terminal count of this timer is between 10 ms and 90 ms.

- quiet_timer<i>

  This timer is started when the start-up state diagram on lane i enters the QUIET state. The terminal count of this timer is between 100 ms and 200 ms.

- recovery_timer<i>

  This timer is started when the start-up state diagram on lane i enters the RECOVERY state. The terminal count of this timer is between 20 ms and 30 ms.

- propagation_timer<i>

  This timer is started when the start-up state diagram on lane i enters the LINK_READY state. The terminal count of this timer is between 100 ms and 200 ms.

Start-up state diagram

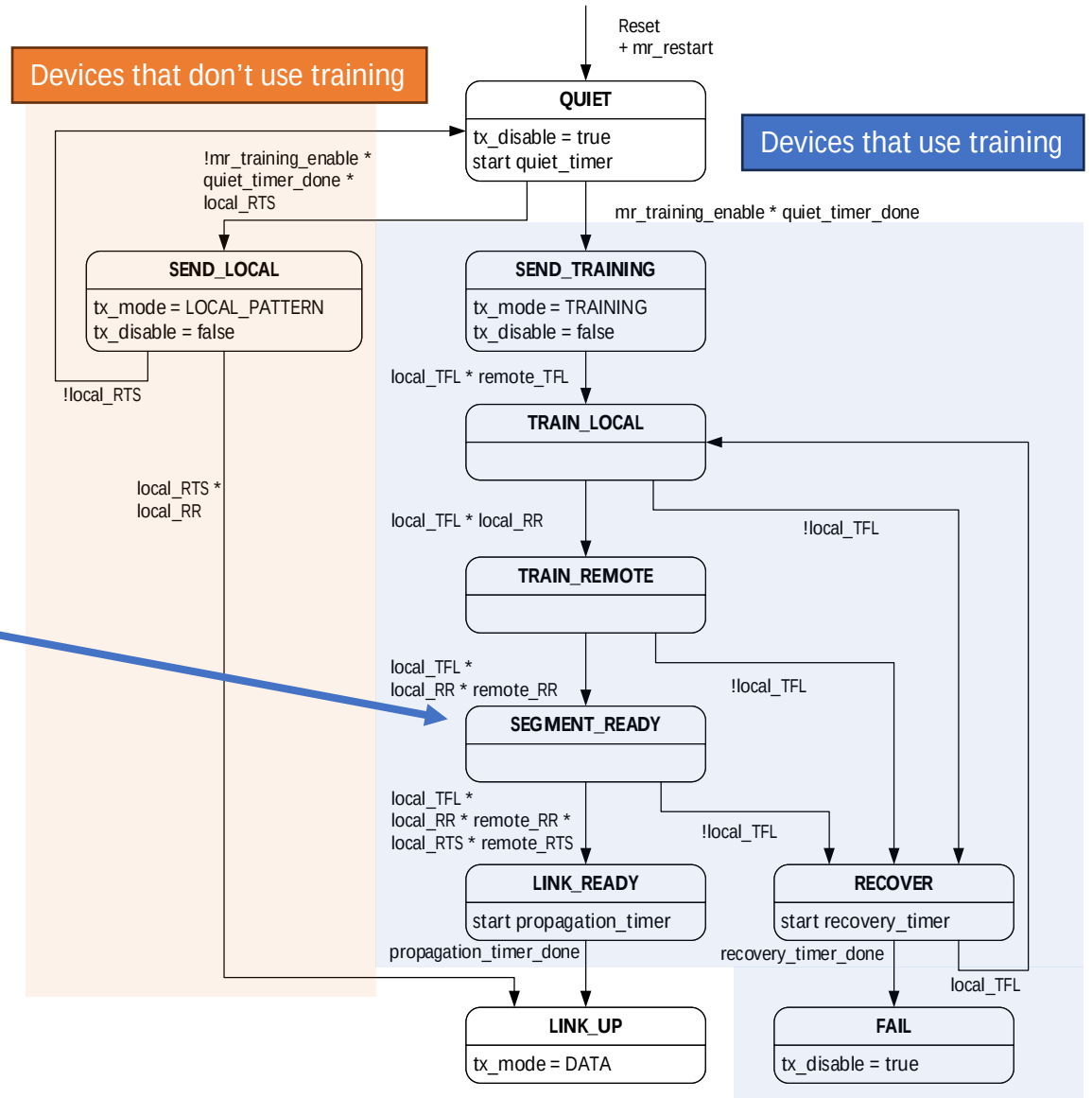
(instance on each lane of each interface)

This diagram is based on the PMD control state diagram in Figure 136–7, with some state re-ordering (start with QUIET).

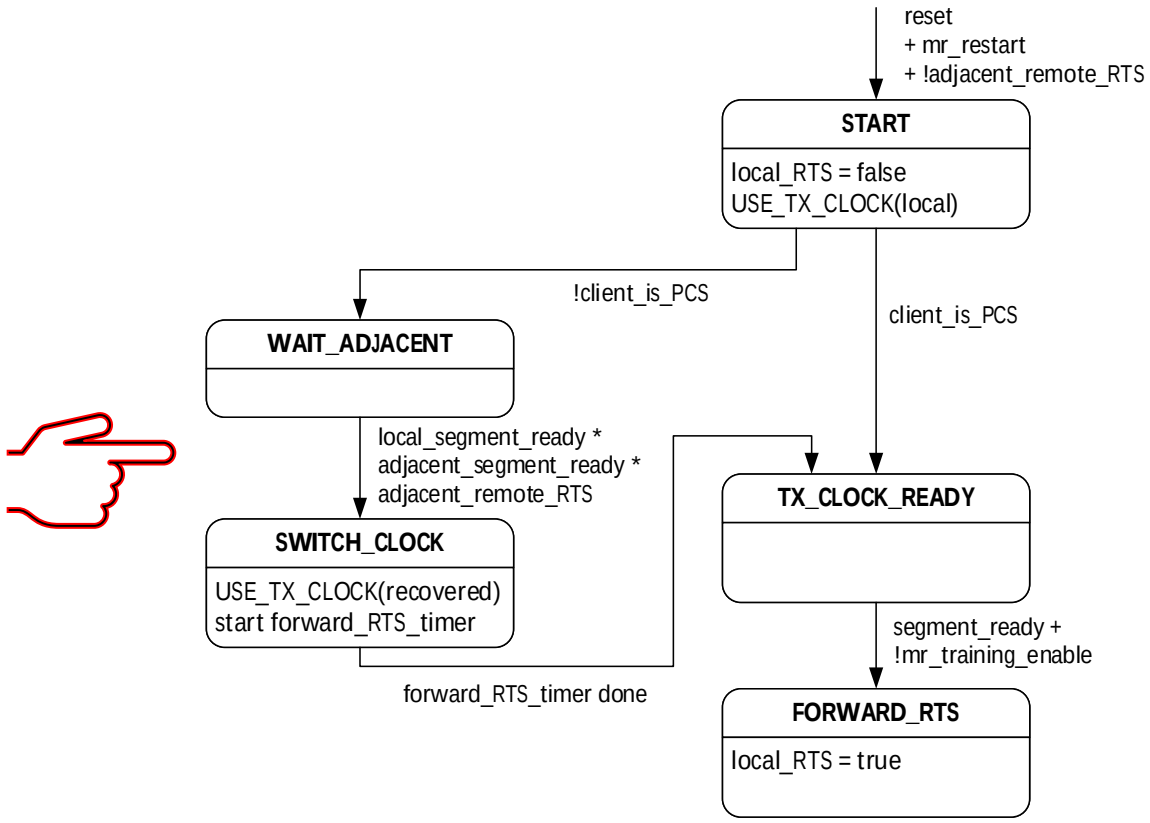
The new state SEGMENT_READY enables extending the exchange of training frames until the whole link can be brought up.

The condition for switching to DATA mode is that RTS is both **sent (local)** and **received (remote**). This indicates that the PCSs on both ends are “seen” on this interface.

Since RTS is propagated (see RTS state diagram), all PMAs will switch to DATA mode at about the same time.



RTS state diagram
(one instance per interface)



Suggested changes to the status field

Current status field structure (Clause 162)

14:12	Reserved	Transmit as 0, ignore on receipt
11:10	Modulation and precoding Status	11 10 1 1 = PAM4 with precoding 1 0 = PAM4 0 1 = Reserved 0 0 = PAM2
6	Reserved	Transmit as 0, ignore on receipt
5:3	Coefficient select echo	Mirror of Coefficient select
2:0	Coefficient status	<values>

Proposed change

14	One	Transmit as 1
13	Reserved	Transmit as 0, ignore on receipt
12:10	Pattern status	12 11 10 1 1 1 = PAM4 free-running PRBS31 with precoding 1 0 1 = Reserved 0 1 1 = PAM4 free-running PRBS31 0 0 1 = PAM2 free-running PRBS31 1 1 0 = PAM4 PRBS13 with precoding 1 0 0 = PAM4 PRBS13 0 1 0 = PAM4 free-running PRBS13 0 0 0 = PAM2 PRBS13
6	Extend training	1 = No data is available, continue training 0 = Switch to data when training is completed
5:3	Coefficient select echo	Mirror of Coefficient select
2:0	Coefficient status	<values>

Subject of a separate proposal

$= \overline{\text{RTS}}$

xMII Extenders

- A PMA below a DTE XS is the same as a PMA below a PCS.
- A PMA above a PHY XS uses SIGNAL_OK from PHY_XS:IS_SIGNAL.indication (see 173.5.8.2) as a condition for local_RTS on its physical interface (local_RTS=1 requires SIGNAL_OK=OK from the PHY XS).
- A PMA above a PHY XS sets SIGNAL_OK in PHY_XS:IS_SIGNAL.request (see 173.5.8.2) based on remote_RTS on its physical interface.

Additional notes

The content of the following slides may provide additional clarify, but it is not required for inclusion in a draft, if this proposal is adopted.

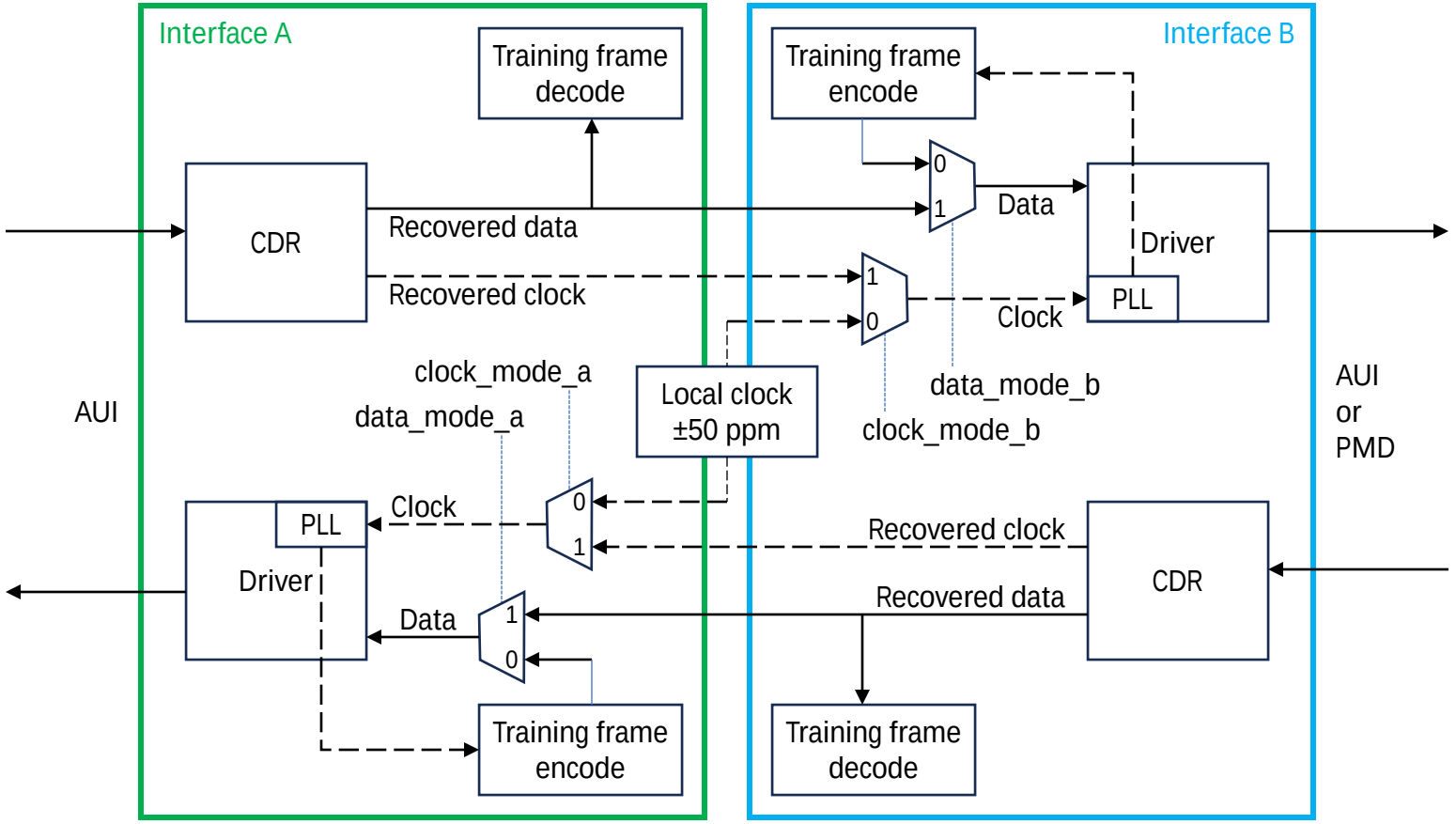
Clocking …

The figure to the right is a schematic representation with an AUI on the left side and either an AUI or a PMD on the other.

Clock control is set according to the RTS state diagram:
clock_mode_a/b = 0 is equivalent USE_TX_CLOCK(local)
clock_mode_a/b = 1 is equivalent USE_TX_CLOCK(recovered)

In this figure, selection of the clock source is simplified. In a real implementation switch clock sources may entail several housekeeping activities, e.g., buffer realignment.

Data control is set according to the Start-up state diagram as follows:
data_mode_a/b = 0 is equivalent to tx_mode = TRAINING
data_mode_a/b = 1 is equivalent to tx_mode = DATA



Behavior with and without training

If training is available on an interface

- The local_* variables are sent to the segment partner via the training frames. Remote_* variables are received from the segment partner.
- When both local and remote RTS are 1, after a specified delay, the PMA switches to its normal functionality, forwarding data between its interfaces in both directions.
  - If there are multiple lanes, all lanes switch within this time.
- There is no specified timeout when waiting for either RR or RTS.
  - While waiting for RR/RTS, losing frame lock and not recovering after a specified recovery time would cause training to fail and squelch. Training can be restarted by management after an unspecified time.
  - Management can access devices and restart training on a specific segment if desired.
- The SIGNAL_OK status variable of the interface is assigned according to local and remote RTS (OK when both are 1).

If training is not available on an interface (disabled, or not defined for the interface type)

- The following definitions are based on functionality assumed for retimers today (with auto-squelch) – no new implementation is required, but it should be standardized in 802.3.
- **local_RR** is generated on each lane based on internal criteria – similar to the signal indication logic (SIL) in existing PMAs
- **remote_RR** is set to 1 (no way to communicate it from the partner)
- **local_RTS** is independent of local_RR and is generated only from the variables of *the other interface of the PMA*. It is communicated to the partner by squelching (0) or un-squelching (1) the output
- **remote_RTS** is set to 1 (no way to communicate it from the partner)
- The variables **local_TFL** and **remote_TFL** are undefined and not used
- The SIGNAL_OK status variable of the interface is assigned according to local and remote RTS (OK when both are 1).

Training in retimers (including modules)

- Training protocol transmission starts with local clock and transitions to recovered clock when available.
- Local_RTS is set to true on the egress interface only after the transmit clock is derived from the local PCS clock; on the ingress interface, only after the transmit clock is derived from the remote PCS clock
  - This is a result of the RTS state diagram
  - The transition between clock sources occurs while sending local_RTS=false. This ensures that the whole link is running with the correct clocks before retimers go to “mission mode”.
- Propagation of RTS across a retimer:
  - Exchanging the RTS between the two PMA interfaces (i.e., copying remote_RTS to adjacent_remote_RTS) may be implemented in various ways. It may be done either autonomously inside the PMA, or using external management (e.g., CMIS).
  - When remote_RTS=1 is received on an interface that sends local_RTS=0, the propagation to the other interface does not need any timing requirements.
  - However, when remote_RTS=1 is received in on an interface that sends local_RTS=1, **it should be propagated to the other interface within a reasonable time** (e.g., 100 ms) to prevent unnecessary delay in bringing up the link (other retimers may have already transitioned to data mode).

Clause 136 compatibility considerations

- It is possible that a CR/KR link is built with 100G/lane PMDs.
  - The suggested method can be used in this scenario, but one of the PMDs may only support clause 136 training.
- Existing (“legacy”) devices always transmit 0 in bit 14, new devices always transmit 1 in bit 14.
- Usage of the Extend training bit:
  - Legacy devices always transmit 0 in bit 6 and switch to data when training is completed. This requires that their PCS be already active when training starts.
  - If a CR/KR PHY includes an AUI-C2C and the link partner has “legacy” training (i.e., KR/CR at 100 Gb/s per lane), then training on the medium must occur only after training on the AUI is completed. This way, transition to data mode will be successful.
    - If AN is used, the local device can send null next pages until the AUI is ready.
    - If AN is not used and training on the medium starts before the AUI is ready, the presence of a “legacy” partner can be detected by bit 14 – if it is 0, then training should be stopped and deferred until the AUI is ready.