

Consensus proposal for resolving comment #188

Adee Ran, Cisco

Jeff Slavick, Broadcom

Leon Bruckman, NVIDIA

Marco Mascitto, Nokia

Luz Osorio, Nokia

Sebastien Gareau, Ciena

Mike Dudek, Marvell

Supporters

- Paul Brooks, Viavi
- Kent Lusted, Synopsys
- Gary Nicholl, Cisco
- Xuebo Wang, Huawei
- Matt Brown, Qualcomm

The concern described in the comment

- A variable is defined as being set in a certain state
 - instead of setting it explicitly in the state diagram
 - Not clean

local_rx_ready

Boolean variable that is set to true when the receiver on a lane of the interface has determined that the peer interface transmitter is transmitting the expected signal format, that the remote transmit has been optimized if mr_training_enable is true, that the local receive equalizers have been optimized, and that no further adjustments are required for normal data transmission. The specific conditions for setting this variable to true are implementation dependent. It shall be set to false when an input signal corresponding to tx_disable = true in the peer is received for 100 ms (the minimum duration of the quiet_timer) or longer. When in the RECOVERY state (Figure 178B–10a) this variable is false.

Further thoughts

- Discussion in the consensus group revealed a deeper aspect of the problem
 - `local_rx_ready` is defined as implementation-dependent “receiver happiness” variable
 - As such, it can toggle between true and false (added explicitly by an accepted comment against D3.1)
 - But it is also sent to the peer in the **Receiver ready** bit of the training frame (178B.7.5.1)
- The version sent to the peer should be “debounced” to improve predictability and prevent potential deadlock scenarios in the current state diagram
 - Separating into “internal” and “debounced” variables enables solving that
- The consensus group strongly recommends making this separation

Current text in D3.1

local_rx_ready

Boolean variable that is set to true when the receiver on a lane of the interface has determined that the peer interface transmitter is transmitting the expected signal format, that the remote transmit has been optimized if mr_training_enable is true, that the local receive equalizers have been optimized, and that no further adjustments are required for normal data transmission. The specific conditions for setting this variable to true are **implementation dependent**. It shall be set to false when an input signal corresponding to tx_disable = true in the peer is received for 100 ms (the minimum duration of the quiet_timer) or longer. When in the RECOVERY state (Figure 178B–10a) this variable is false.

Equals
local_rx_ready
(no debounce)

178B.7.5.1 Receiver ready

The receiver ready bit is used to signal the local receiver state to the peer interface. When this bit is 1, it indicates that the local receiver has completed training and is prepared to receive data (local_rx_ready = true). When this bit is 0, it indicates that the local receiver is requesting that training continue (local_rx_ready = false).

remote_rx_ready

Boolean variable that indicates the value of **local_rx_ready** on a lane of the peer interface. If mr_training_enable is true and training is true, it is derived from the receiver ready bit of the status field of received training frames on the corresponding lane of the interface. Otherwise it is set to true.

Reflects the peer's
local_rx_ready

Core change

- Make local_rx_ready a state-diagram-assigned variable
 - Set to false in SEND_TRAINING and RECOVERY, and TRAIN_FAIL
 - Set to true in TRAIN_REMOTE
 - Debounced by the state diagram definition
 - Is sent to the peer using the training protocol
- Use another variable, rx_ok, as the implementation-dependent signal quality indication

Debouncing local_rx_ready

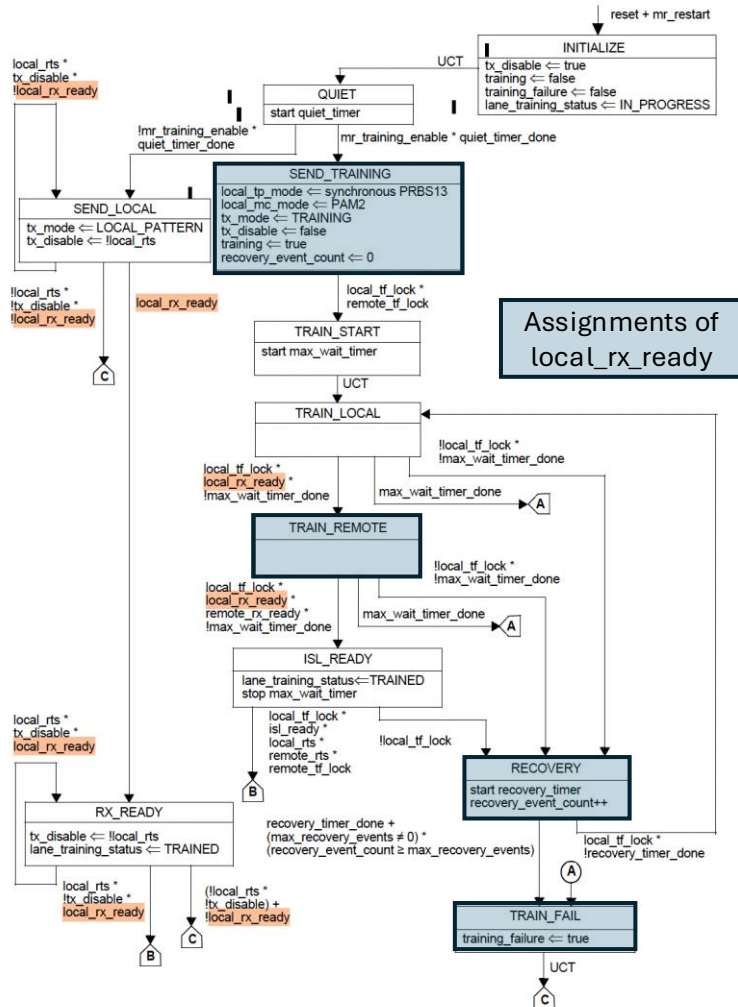


Figure 178B-10a—Training control state diagram, part 1

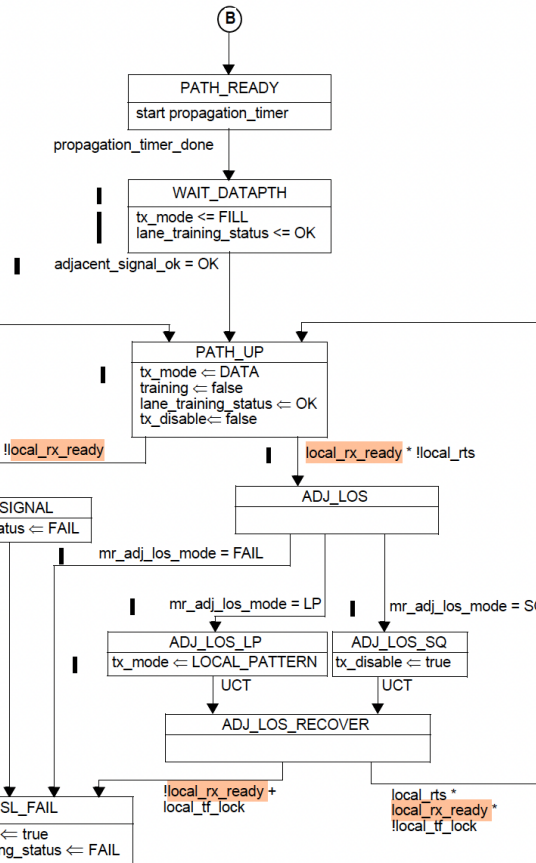


Figure 178B-10b—Training control state diagram, part 2

Rename the internal variable to rx_ok in the highlighted places

(Many text changes in the state diagram, but no change in state diagram behavior)

Keeps the current relationship between local_rx_ready, remote_rx_ready, and existing management variables

Details of the change

~~local_rx_ready~~ **rx_ok**

Boolean variable that is set to true when the receiver on a lane of the interface has determined that the peer interface transmitter is transmitting the expected signal format, that the remote transmit has been optimized if mr_training_enable is true, that the local receive equalizers have been optimized, and that no further adjustments are required for normal data transmission. The specific conditions for setting this variable to true are implementation dependent. It shall be set to false when an input signal corresponding to tx_disable = true in the peer is received for 100 ms (the minimum duration of the quiet_timer) or longer. ~~When in the RECOVERY state (Figure 178B-10a) this variable is false.~~

local_rx_ready

Boolean variable that indicates the state of the receiver. Its value is set by the training control state diagram (Figure 178B-10a).

In Figure 178B-10, change all instances of **local_rx_ready** to **rx_ok**, and add the assignments of local_rx_ready: set to false in SEND_TRAINING, RECOVERY, and TRAIN_FAIL; set to true in TRAIN_REMOTE. Add a new MDIO mapping for rx_ok.

(contingent on separating local_rx_ready and rx_ok)

Prevent a potential deadlock situation (when RECOVERY is entered from ISL_READY)

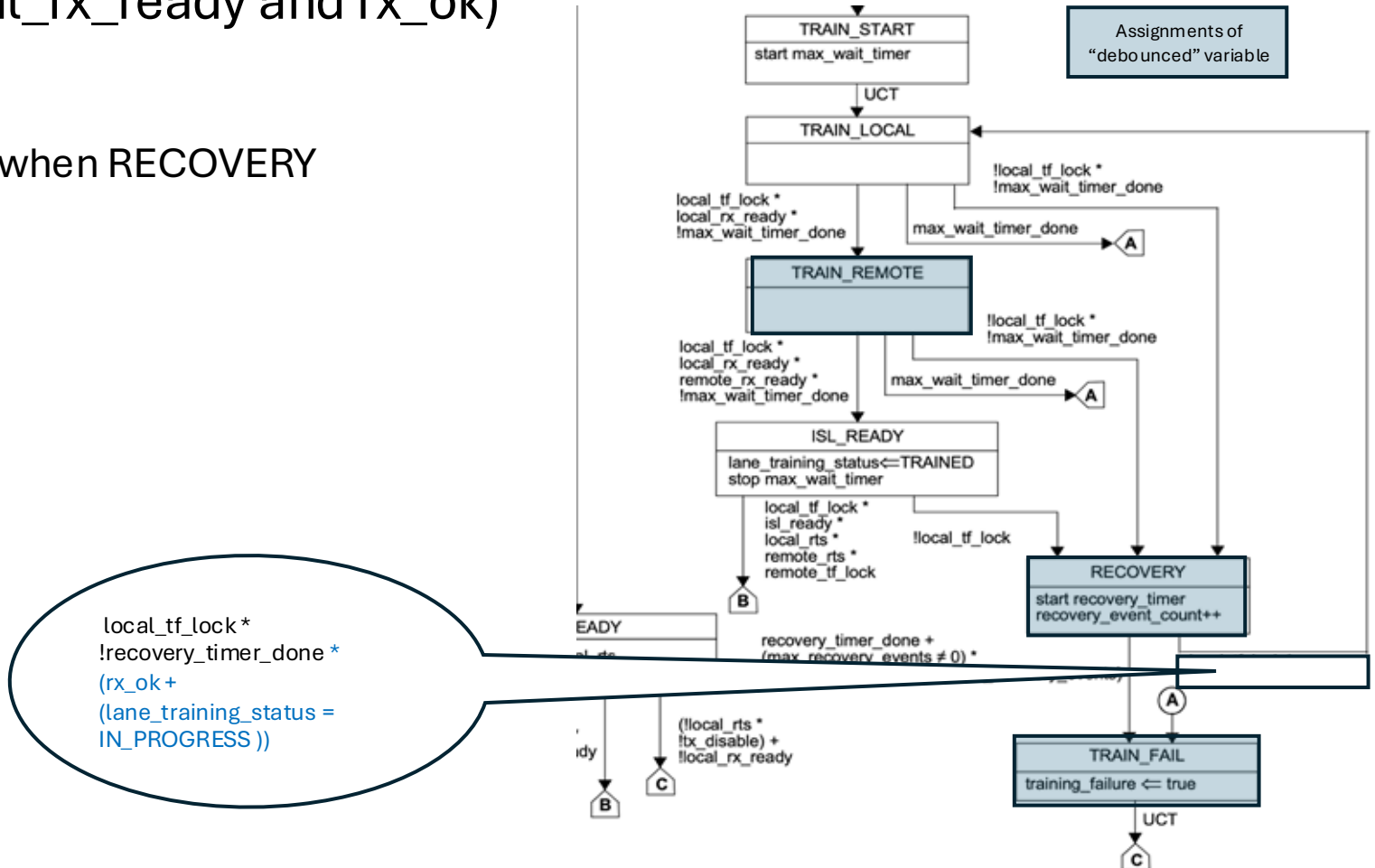


Figure 178B–10a—Training control state diagram, part 1

Summary

- The variable for the “internal received signal quality indication”, used in transition conditions in the training control state diagram, is renamed **rx_ok**.
- The training protocol uses **local_rx_ready**, which becomes a debounced version of the internal variable.
- A potential deadlock in the current state diagram is prevented.
- The consensus group recommends implementing the changes on slides 7, 8, and 9.