



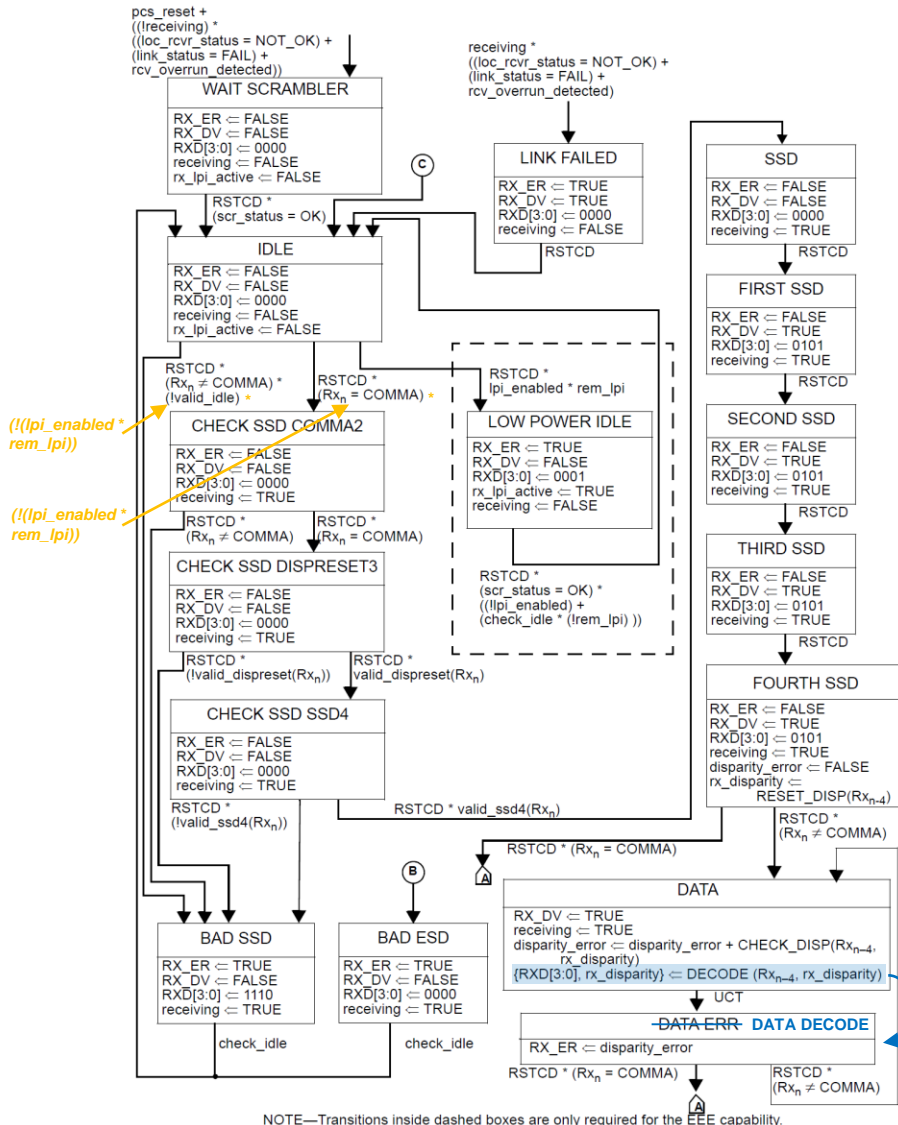
10 Mb/s Single Twisted Pair Ethernet 10BASE-T1L PHY State Diagram Changes

Steffen Graber
Pepperl+Fuchs

Content

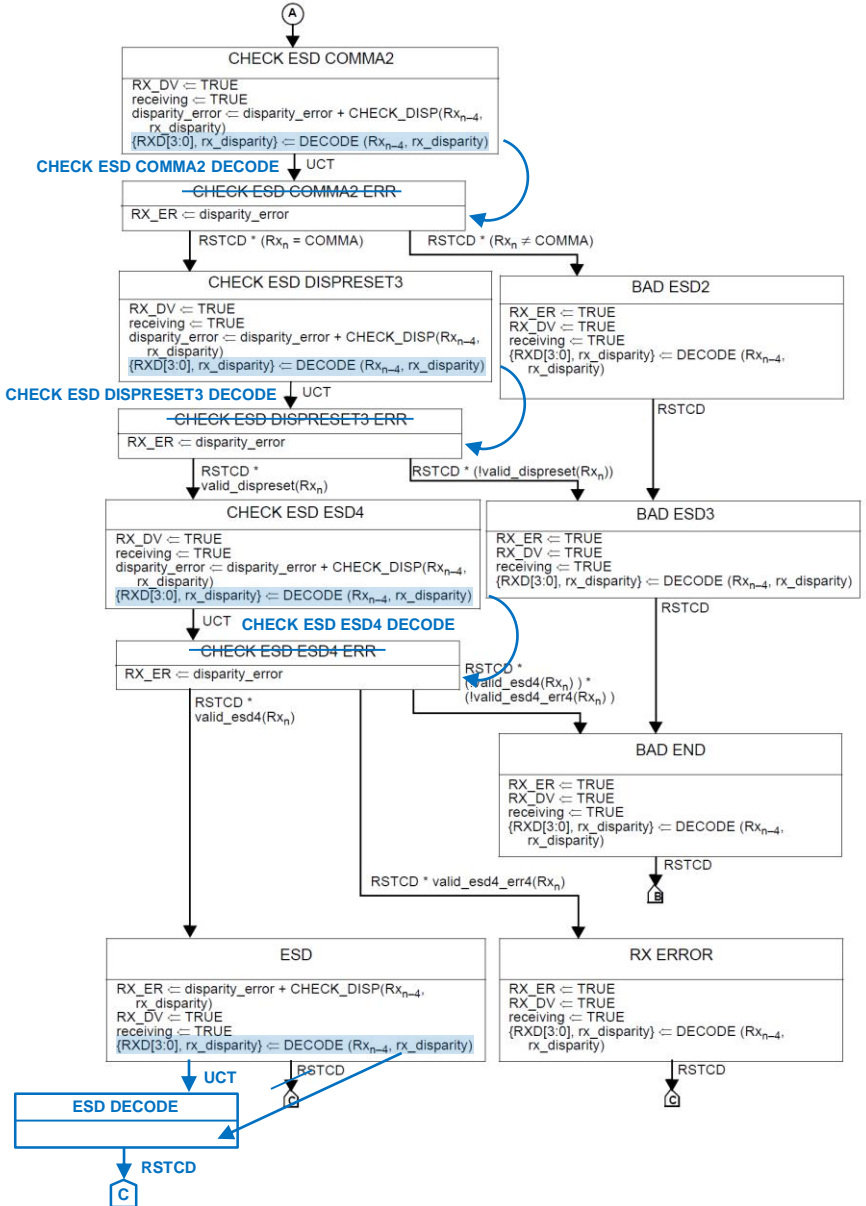
- PCS Receive State Diagram (part a) Changes (*Comments r01-61 and r01-62*)
- PCS Receive State Diagram (part b) Changes (*Comment r01-62*)
- PHY Control Training Synchronization (*Comment r01-64*)
- PHY Control LPI Timer Synchronization (*Comment r01-66*)
- PHY Control State Diagram (part a) Changes (*Comments r01-64, r01-65, r01-66, and r01-67*)
- PHY Control State Diagram (part b) Changes (*Comment r01-68*)
- PHY Control State Diagram (part c) Changes (new, suggestion to add this to Comment r01-68 in Vienna)
- Revised Comment Resolution for r01-68

PCS Receive State Diagram (part a) Changes



- Change exit conditions of IDLE state to be mutually exclusive (comment r01-61), marked in **orange and italic**.
- Avoid ambiguity when a variable (rx_disparity) is set and used in the same state (comment r01-62), marked in **blue**.
- This change is only for clarification, there is no change in the behavior of the state diagram.

PCS Receive State Diagram (part b) Changes



- Avoid ambiguity when a variable ($rx_disparity$) is set and used in the same state (comment r01-62), marked in blue.
- This change is only for clarification, there is no change in the behavior of the state diagram.

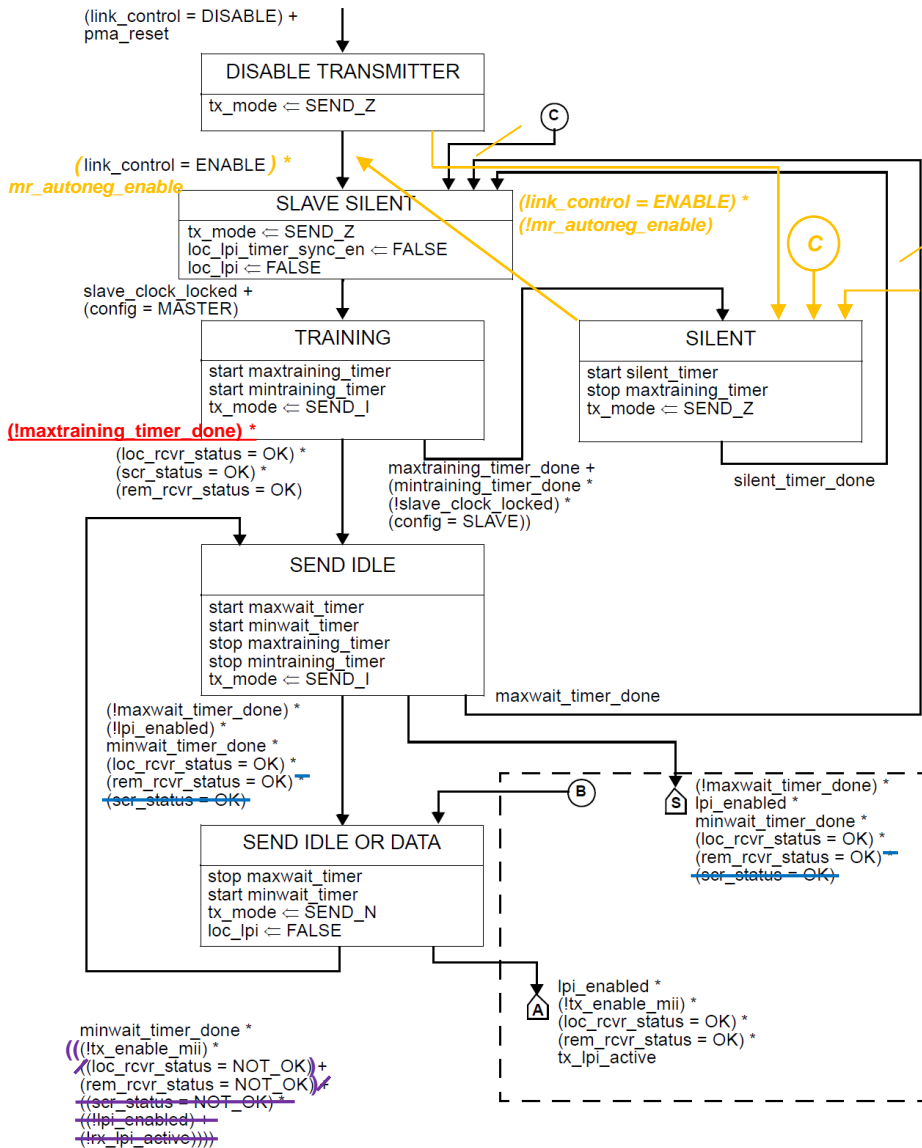
PHY Control Training Synchronization

- When a reset occurs for one PHY while Auto-Negotiation is not present, both PHYs will not reliably break their link, which subsequently may lead to training issues.
- A simple solution is to add a silent period for both PHYs in case Auto-Negotiation is disabled or not present.
- This time needs to be longer than the 200 ms `maxwait_timer` period (e.g. 245 ms +/- 5 ms).
- If Auto-Negotiation is enabled, then Auto-Negotiation will synchronize the training start, so that the silent time is not needed during the first startup of the PHY.
- If the PHY is in TRAINING state and the slave PHYs clock is lost, or if the PHY is in SEND IDLE state or LPI timer synchronization and the `maxwait_timer` expires, then going back to SILENT state breaks the link reliably before doing a retraining or, if Auto-Negotiation is active, restarts the Auto-Negotiation process by setting `tx_mode = SEND_Z`.

PHY Control LPI Timer Synchronization

- The transition from SEND IDLE OR DATA back to SEND IDLE may not occur fast enough on both link partners to allow a resynchronization of the LPI timers in case one PHY is actively transmitting data.
- Changing the exit condition of the SEND IDLE OR DATA state as shown on the next page allows the local PHY to go to SEND IDLE state as soon as the `rem_rcvr_status` is `NOT_OK`, even if it is currently transmitting data.
- In such a situation, where the remote PHY is not able to receive data, there is no need to finish the actual data transmission, so that also the local PHY can quickly follow the remote PHY to SEND IDLE state.

PHY Control State Diagram (part a) Changes



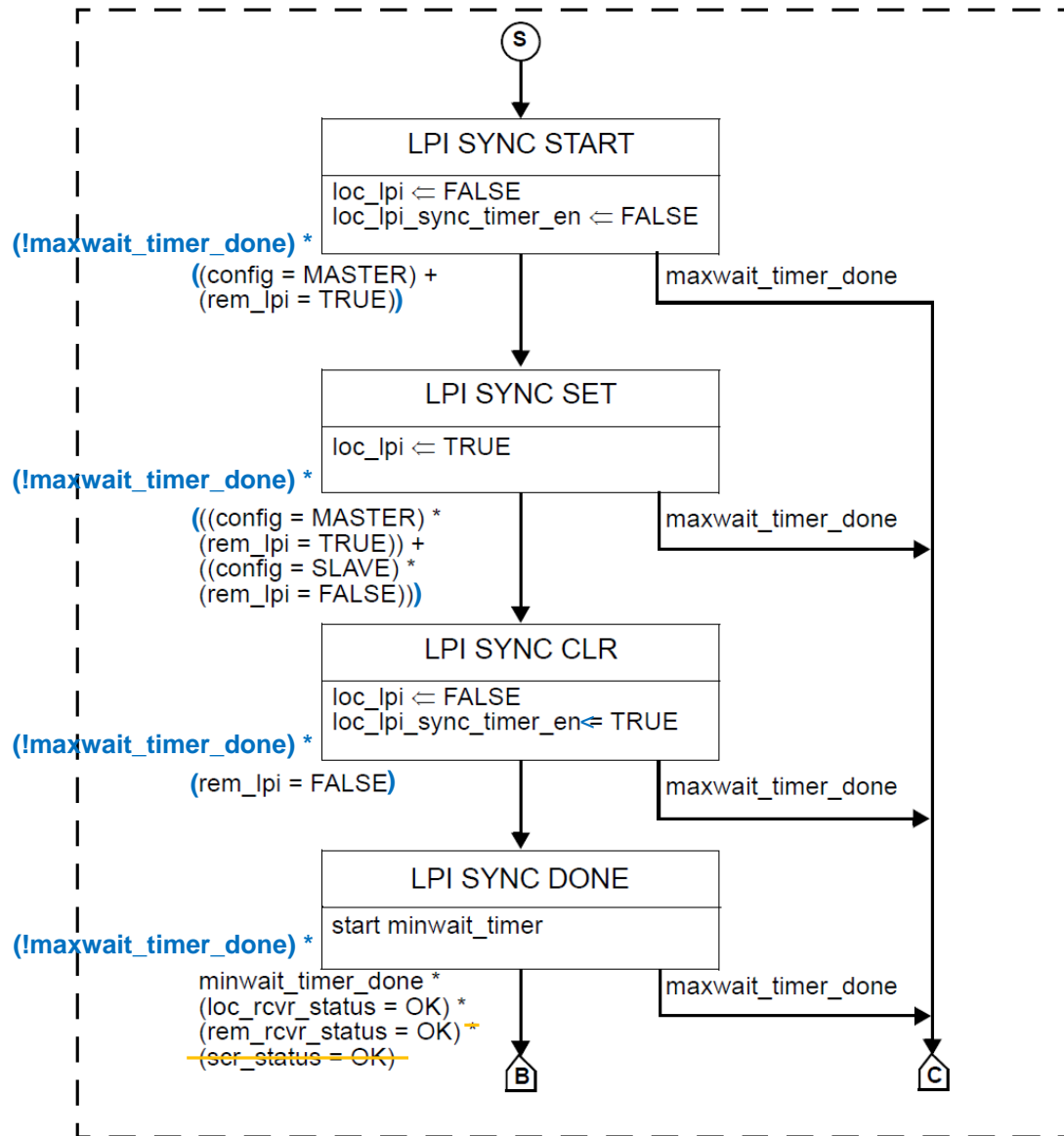
The **orange and italic** marked changes around SILENT and SLAVE SILENT state are necessary to resolve the problem of not reliably breaking the link, as described on page “PHY Control Training Synchronization” (comment r01-64).

Adding the **(!maxtraining_timer_done)** condition **(in red and underline)** to the arc going from TRAINING to SEND IDLE provides a mutual exclusive condition in relation to the second exit condition of the TRAINING state (comment r01-65).

The changes at the bottom left of the state diagram **(in violet)** are related to solving the issue described on page “PHY Control LPI Timer Synchronization” (comment r01-66).

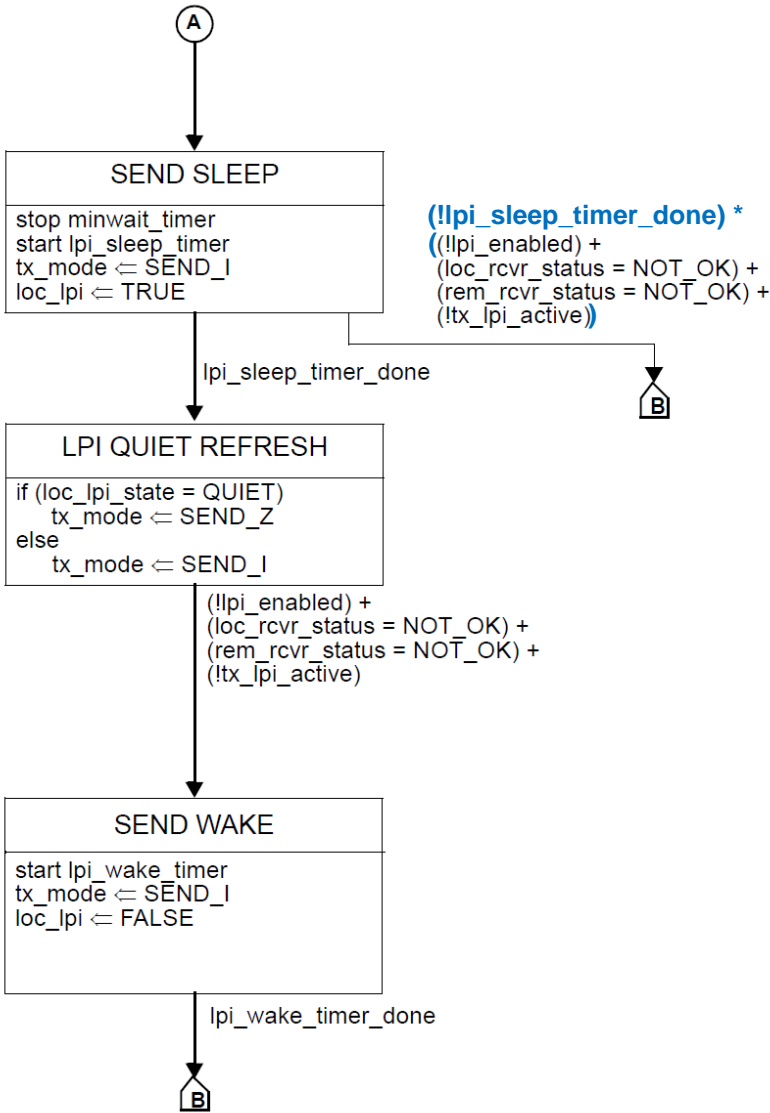
Besides an initial checking of “scr_status” variable after TRAINING, in all other states this variable is not checked any more, as this would already assume some implementation specific details, which are out of the scope of the standard (marked in **blue**, comment r01-67).

PHY Control State Diagram (part b) Changes



- Add `(!maxwait_timer_done)` conditions to generate mutually exclusive exit conditions of the states (comment r01-68), marked in **blue**.
- Remove `(scr_status = OK)` checking at the exit condition of state LPI SYNC DONE, as this is implementation dependent and out of the scope of the standard (comment r01-68), marked in **orange**.

PHY Control State Diagram (part c) Changes



- Add (!lpi_sleep_timer_done) condition to generate mutually exclusive exit conditions of SEND SLEEP state (in blue).
- This necessary change has been missed by the supplied comments for D3.1.
- Suggestion is to add this change to the proposed resolution of comment r01-68 during the meeting in Vienna (see the full suggested text for comment r01-68 on the next slide).

Revised Comment Resolution for r01-68

- Add the change described on the previous slide to comment r01-68 (P153, L8).
- The full text of the comment resolution for r01-68 will then read as:

P152, L8: On arc from LPI SYNC START to LPI SYNC SET, change condition from: "(config = MASTER) + (rem_lpi = TRUE)" to: "(!maxwait_timer_done) * ((config = MASTER) + (rem_lpi = TRUE))"

P152, L14: On arc from LPI SYNC SET to LPI SYNC CLR, change condition from: "((config = MASTER) * (rem_lpi = TRUE)) + ((config = SLAVE) * (rem_lpi = FALSE))" to: "(!maxwait_timer_done) * (((config = MASTER) * (rem_lpi = TRUE)) + ((config = SLAVE) * (rem_lpi = FALSE)))"

P152, L22: On arc from LPI SYNC CLR to LPI SYNC DONE, change condition from: "rem_lpi = FALSE" to: "(!maxwait_timer_done) * (rem_lpi = FALSE)"

P152, L27: On arc from LPI SYNC DONE to exit tag "B", change condition from: "minwait_timer_done * (loc_rcvr_status = OK) * (rem_rcvr_status = OK) * (scr_status = OK)" to: "(!maxwait_timer_done) * minwait_timer_done * (loc_rcvr_status = OK) * (rem_rcvr_status = OK)"

P153, L8: On arc from SEND SLEEP to exit tag "B", change condition from: "(!lpi_enabled) + (loc_rcvr_status = NOT_OK) + (rem_rcvr_status = NOT_OK) + (!tx_lpi_active)" to: "(!lpi_sleep_timer_done) * ((!lpi_enabled) + (loc_rcvr_status = NOT_OK) + (rem_rcvr_status = NOT_OK) + (!tx_lpi_active))"

Thank You