PLCA improvement for high node count

Wojciech Koczwara • | 1st May 2019

Rockwell Automation

PLCA improvement for high node count

Presenter: Wojciech Koczwara, Rockwell Automation

Presenter/Supporter: Piergiorgio Beruto, Canova Tech

1 Background

2 PLCA with small delay - no issues

3 PLCA with big delay (large number of nodes) - problem

4 PLCA with big delay (long to_timer) - problem

5 Text changes

Background

Collision handling at MAC vs. slotTime

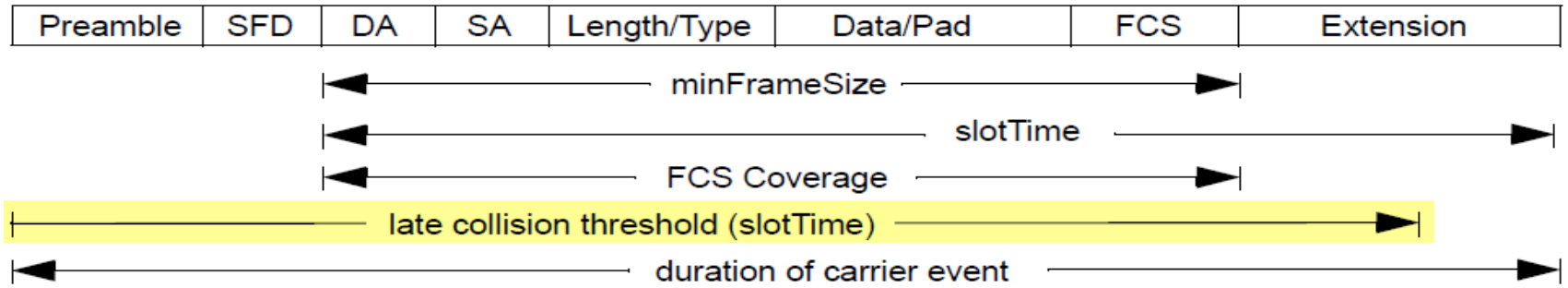


Figure 4–5—Frame with carrier extension

During sending, MAC reacts to PLS_SIGNAL.indication, reacting to the collisions and making new transmit attempts. This behavior is guaranteed until slotTime (512BT) has been reached during sending. After the slotTime has passed, MAC can report lateCollisionErrorStatus, and stop making further transmit attempts.

Late collisions should never happen on a properly functioning network.

Background

PLCA variable delay line

„The variable delay line is a small buffer that aligns a transmission with the transmit opportunity. The variable delay line length is no greater than to_timer × plca_node_count + beacon_timer.”

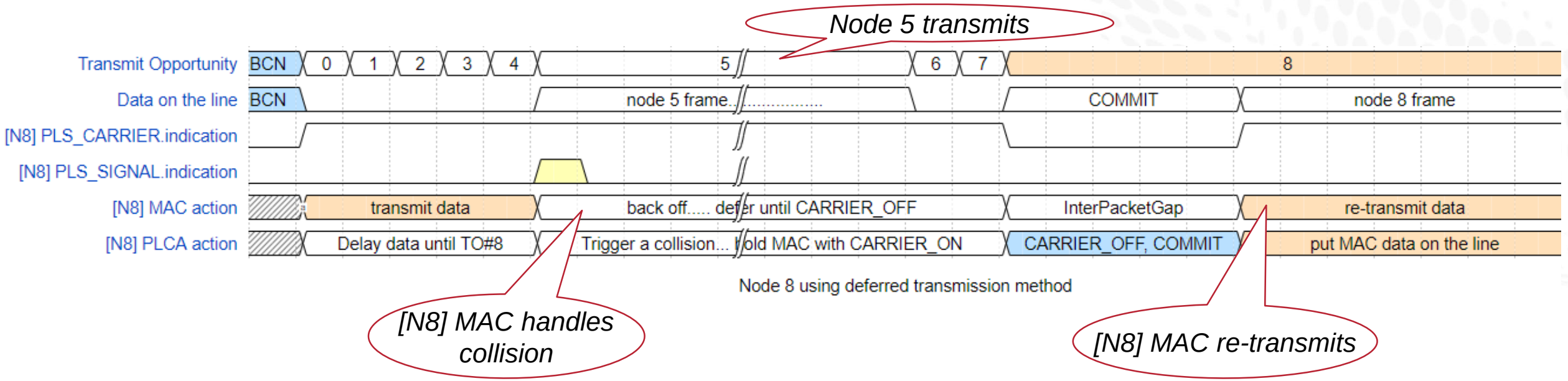
IF to_timer × plca_node_count + beacon_timer > slotTime

THEN the maximum delay line length can exceed the late collision threshold in MAC.

PLCA with small delay line – NO ISSUES

MAC of Node 8 starts sending, but node 5 sends data and delays Transmit Opportunity 8.

- When MAC of node 8 starts sending, PLCA uses the delay line to align transmission to Transmit Opportunity 8.
- If node 5 starts transmitting meanwhile, Transmit Opportunity 8 will come later than expected
- To limit the delay, node 8 PLCA asserts PLS_SIGNAL.indication to the MAC.
  - Node 8 MAC will back off for 0 or 512 BT, then make a new transmit attempt
  - PLCA uses PLS_CARRIER.indication to defer MAC re-transmission until Transmit Opportunity 8
  - At TO#8, PLS_CARRIER.indication is de-asserted. MAC defers for InterPacketGap, this time is filled with COMMIT, then MAC sends data which is put directly on the line.

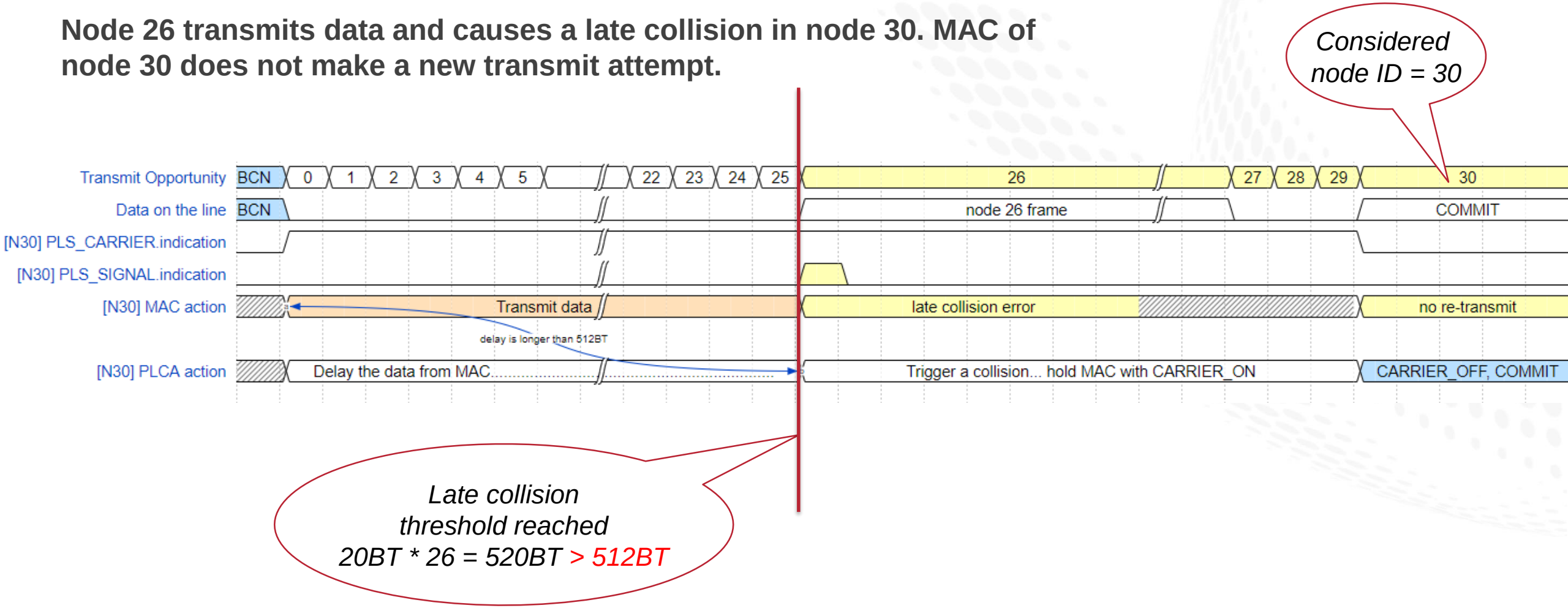


PLCA with big delay line (large number of nodes) - PROBLEM

Perspective of node 30, yellow fields highlight the problem

Due to excessive delay, late collision threshold (>=512BT) is reached in MAC.

Node 26 transmits data and causes a late collision in node 30. MAC of node 30 does not make a new transmit attempt.

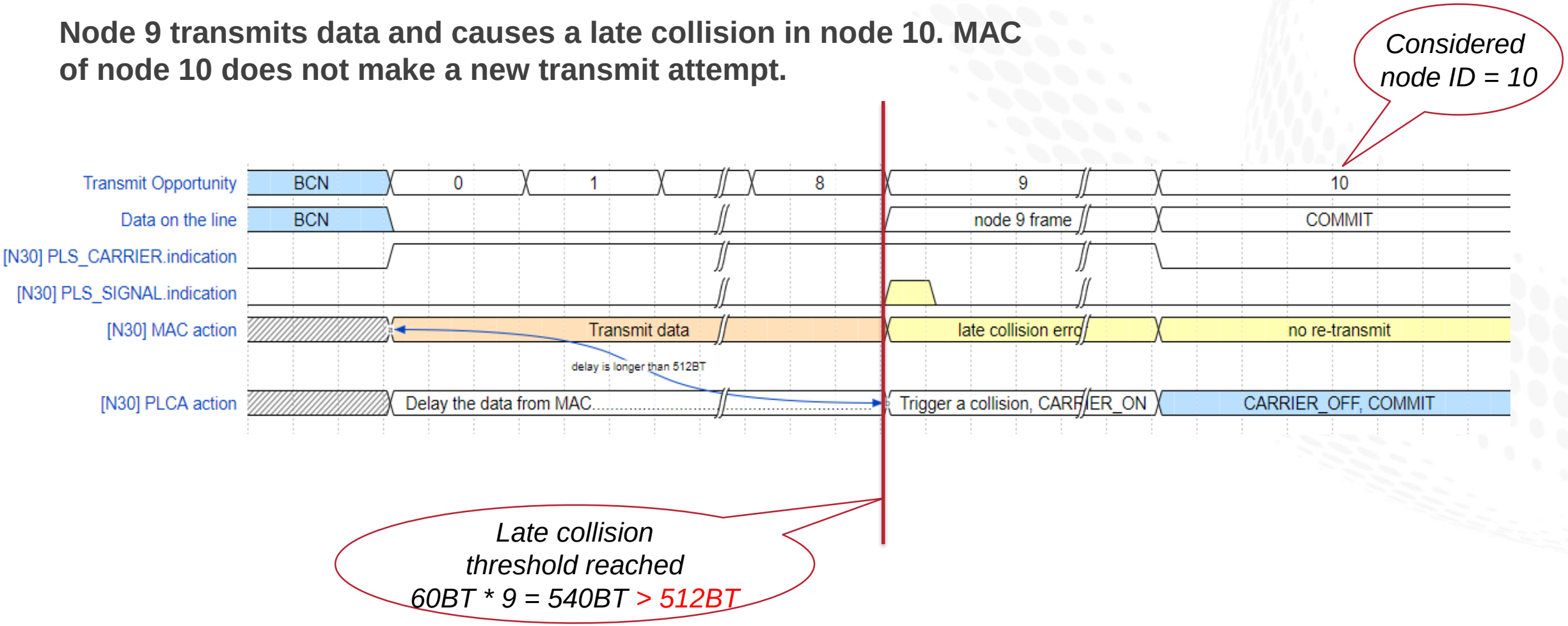


PLCA with big delay line (to_timer = 60BT) – PROBLEM

Perspective of node 10, yellow fields highlight the problem

Due to excessive delay, late collision threshold (>=512BT) is reached in MAC.

Node 9 transmits data and causes a late collision in node 10. MAC of node 10 does not make a new transmit attempt.



PLCA high node count

Text Changes

V1.2

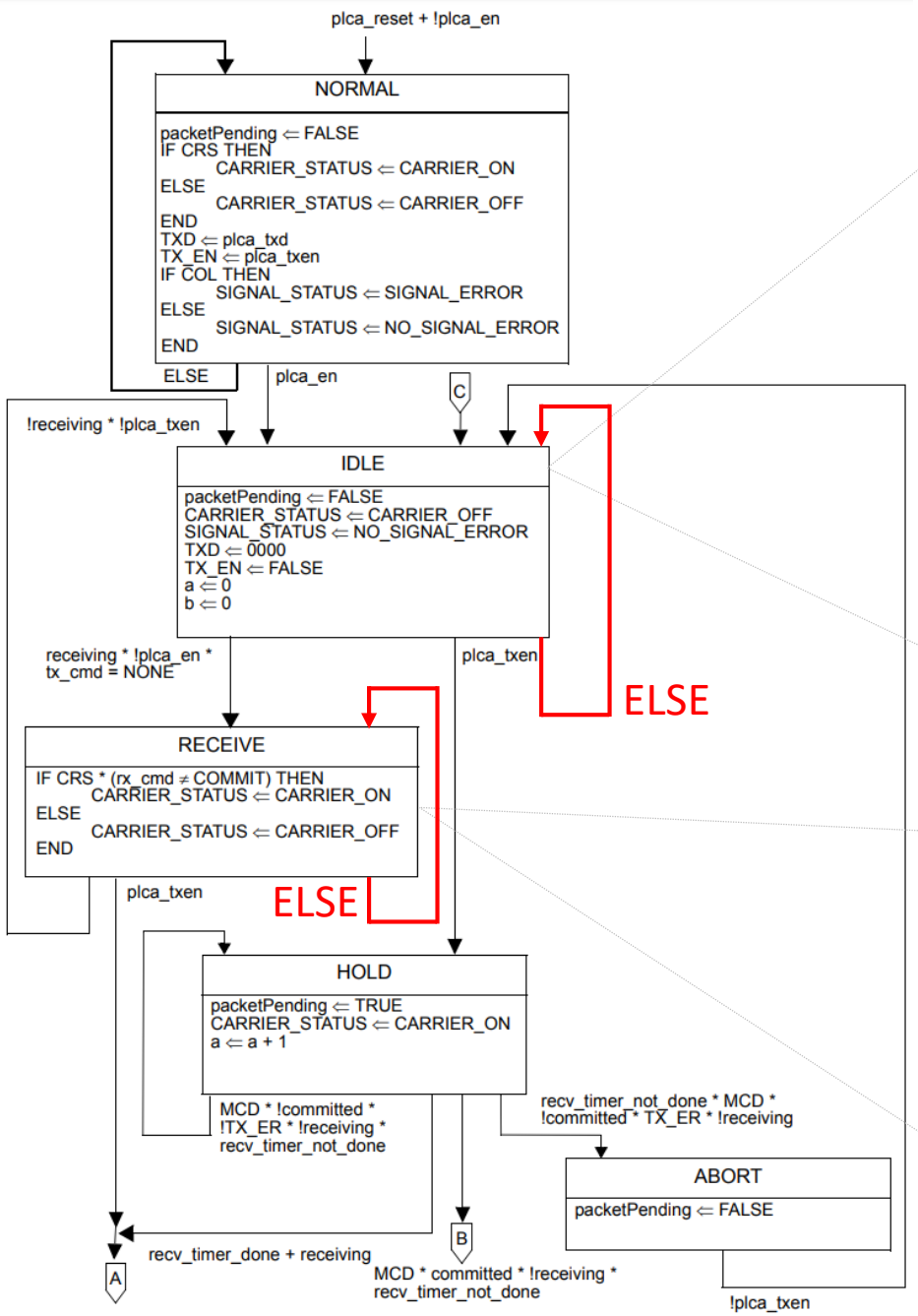


Figure 148–4—PLCA DATA state diagram

IDLE

```
packetPending <= FALSE
IF CARRIER_INTERDICTED(curID) THEN
   CARRIER_STATUS <= CARRIER_ON
ELSE
   CARRIER_STATUS <= CARRIER_OFF
END
SIGNAL_STATUS <= NO_SIGNAL_ERROR
TXD <= 0000
TX_EN <= FALSE
a <= 0
b <= 0
```

RECEIVE

```
IF CRS * (rx_cmd ≠ COMMIT) + CARRIER_INTERDICTED(curID) THEN
   CARRIER_STATUS <= CARRIER_ON
ELSE
   CARRIER_STATUS <= CARRIER_OFF
END
```

1. change text as indicated in the boxes
2. add recirculating arcs as indicated in red

Replace 148.4.6.3 content with:

ROUND_DOWN

returns the nearest integer less or equal than the argument

CARRIER_INTERDICTED

This function takes as its argument the curID parameter and returns TRUE if it falls within the range of node IDs where CARRIER_STATUS has to be reported as CARRIER_ON. It returns FALSE otherwise. The function behavior is defined by the following pseudo-code:

```
opening := ROUND_DOWN((delay_line_length - beacon_timer) / to_timer)

IF opening >= plca_node_count THEN
    return FALSE
END

closing := local_nodeID - curID;
IF closing < 0 THEN
    closing := closing + plca_node_count
END

return (closing > opening) + (closing <= 1)
```

148.4.5.4

recv_timer

This timer determines how much time to wait in the EARLY_RECEIVE state before performing a recovery procedure.

Duration: this timer is implementation dependent and should be greater than the PHY's total receive latency.

to_timer

The transmit opportunity timer maps to aPLCATransmitOpportunityTimer. The timer value should be long enough to allow the transmitting node to have the first nibble of its transmission (including the COMMIT request) to be received by all other PLCA RS before their own to_timer expires. This includes the worst case PHY TX and RX latency and the maximum MDI to MDI propagation delay. to_timer shall be set equal across the mixing segment in order for PLCA to work properly.

~~Duration: integer number between 1 and 255, expressed in bit times.~~

Duration: from 20 to 128 bit times

recv_beacon_timer

During recovery, recv_beacon_timer times the period that all nodes need to be silent

30.3.9.2.5 aPLCATransmitOpportunityTimer

ATTRIBUTE

APPROPRIATE SYNTAX:

INTEGER

BEHAVIOUR DEFINED AS:

This value is assigned to define the time between PLCA transmit opportunities for the node. aPLCATransmitOpportunityTimer maps to the duration of the timer to_timer. The value of aPLCATransmitOpportunityTimer represents the duration of to_timer in bit times. Valid range is

~~1 to 255~~ 20 to 128, inclusive. The default value is 20. See 148.4.5.4.;

add subclause 148.4.6.5 Constants

delay_line_length

This constant is implementation dependent and specifies the length of the PLCA RS delay line depicted in figure 148-2.
Value: 276 to 496 bit times

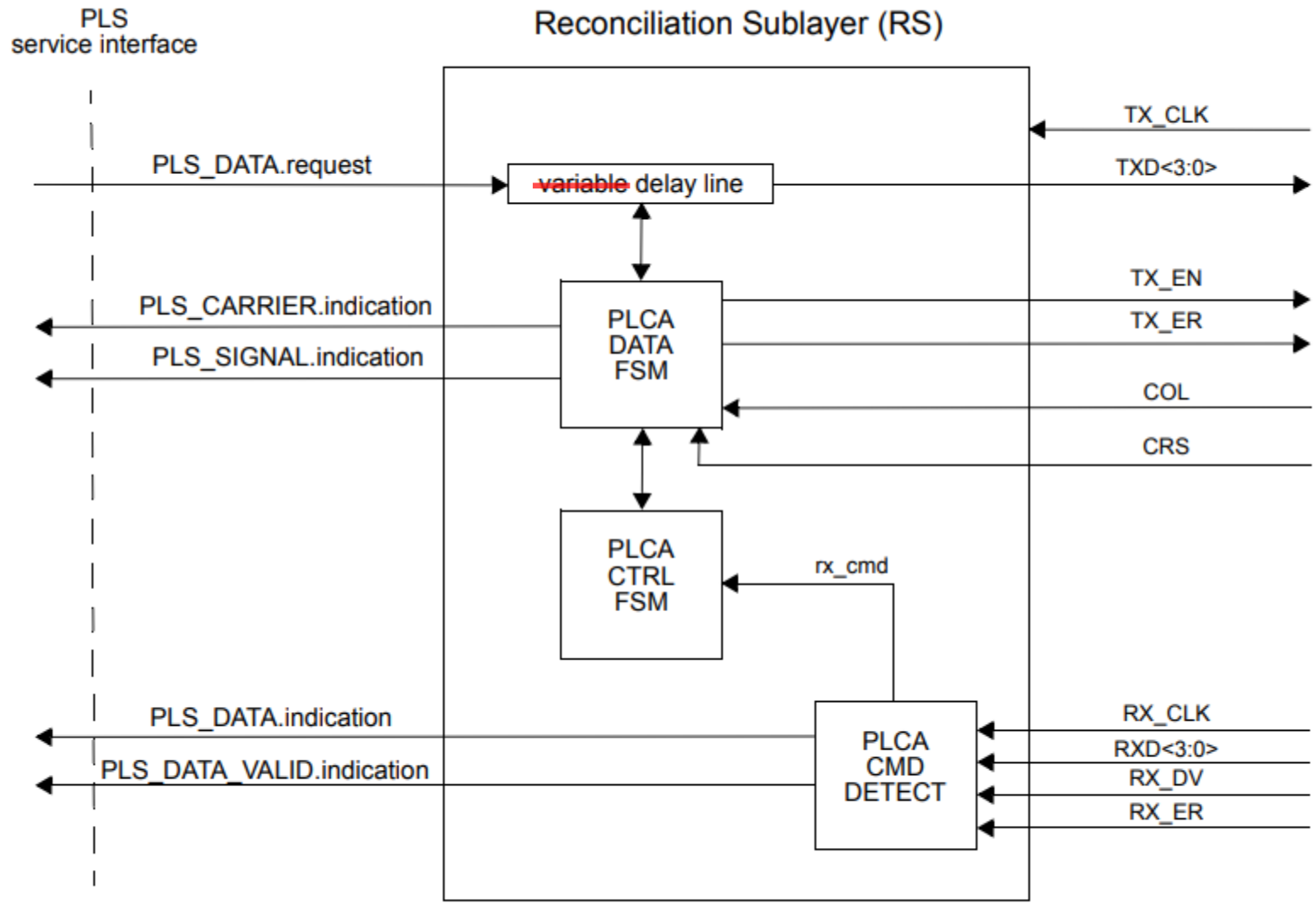


Figure 148–2—PLCA functions within the Reconciliation Sublayer (RS)

148.4.5.2

local_nodeID

ID representing the PLCA transmit opportunity number assigned to the node. This signal maps to aPLCALocalNodeID.

Values: integer value from 0 to 255.

plca_node_count

Maximum number of PLCA nodes on the mixing segment receiving transmit opportunities before the node with local_nodeID = 0 generates a new BEACON, reflecting the value of aPLCANodeCount. ~~This parameter is only meaningful for the node with local_nodeID = 0, otherwise it is ignored.~~

Values: integer number from 0 to 255

148.4.5.1 PLCA Control State Diagram

To achieve error free operation the PLCA node should be configured appropriately before transmit functions are enabled. Appropriate configuration includes:

a) each local_nodeID is unique to the local collision domain,

b) there is one and only one node with local_nodeID = 0 on the local collision domain,

c) the transmit opportunity timer (to_timer) is set equal across all the nodes on the local collision domain, and

d) plca_node_count is set on the node with local_nodeID = 0 to the number of nodes on the local collision domain

In case ROUND_DOWN((delay_line_length – beacon_timer) / to_timer) is less than the number of nodes on the local collision domain, additional configuration is required:

a) plca_node_count is set on all nodes with local_nodeID ≠ 0 to the number of nodes on the local collision domain, or

b) plca_node_count on nodes with local_nodeID ≠ 0 is set by dynamically learning the length of one cycle of transmit opportunities.

148.4.6.1 PLCA Data State Diagram

When PLCA functions are enabled, the PLCA Data state diagram transitions to the IDLE state and waits for the MAC to start a transmission ~~or the PHY to assert carrier sense~~.

~~In the former case, the d~~Data conveyed by the MAC through the PLS_DATA.request primitive is delayed by switching to HOLD state. ~~In the latter case, CARRIER_ON is signaled through the PLS_CARRIER.indication to have the MAC defer any new transmission, then the RECEIVE state is entered.~~

148.4.6.1 PLCA Data State Diagram

During the HOLD state, the PLCA Control state diagram is notified via the packetPending variable that data is available to be transmitted and the beginning of the transmission is held in the ~~variable~~ delay line. At the next transmit opportunity, the PLCA Control state diagram allows transmitting the delayed data by setting the "committed" variable to TRUE. In such a case, the PLCA Data state diagram switches to TRANSMIT state to actually deliver the data for the PHY to encode and transmit on the medium.

The ~~variable~~ delay line is a small buffer that aligns a transmission with the transmit opportunity. ~~The variable delay line length is no greater than to_timer × plca_node_count + beacon_timer.~~

148.4.2 Reconciliation Sublayer operation

Figure 148–2 depicts the RS interlayer service interfaces. The PLCA RS contains the Control and Data state diagrams, the ~~variable~~ delay line, and command detect logic.

Append text to 148.4.6.2 PLCA Data variables

curID

See 148.4.5.2.

local_nodeID

See 148.4.5.2.

plca_node_count

See 148.4.5.2.

Append text to 148.4.6.4 Timers

to_timer

See 148.4.5.4.

148.4.7.4 Timers

plca_status_timer

Represents the time plca_status is maintained in TRUE state when plca_active is FALSE while in HYSTERESIS state.

Duration: the duration of this timer is ~~130 090~~ 65 320 bit times, which is

2 × (max to_timer × max plca_node_count + beacon_timer).

Thank you

www.rockwellautomation.com

Rockwell Automation