# Path data delay through the n:m SM-PMA

Related to comment #131 against 802.3dj D3.1

Andras de Koos – Microchip Technology

IEEE 802.3dj Task Force

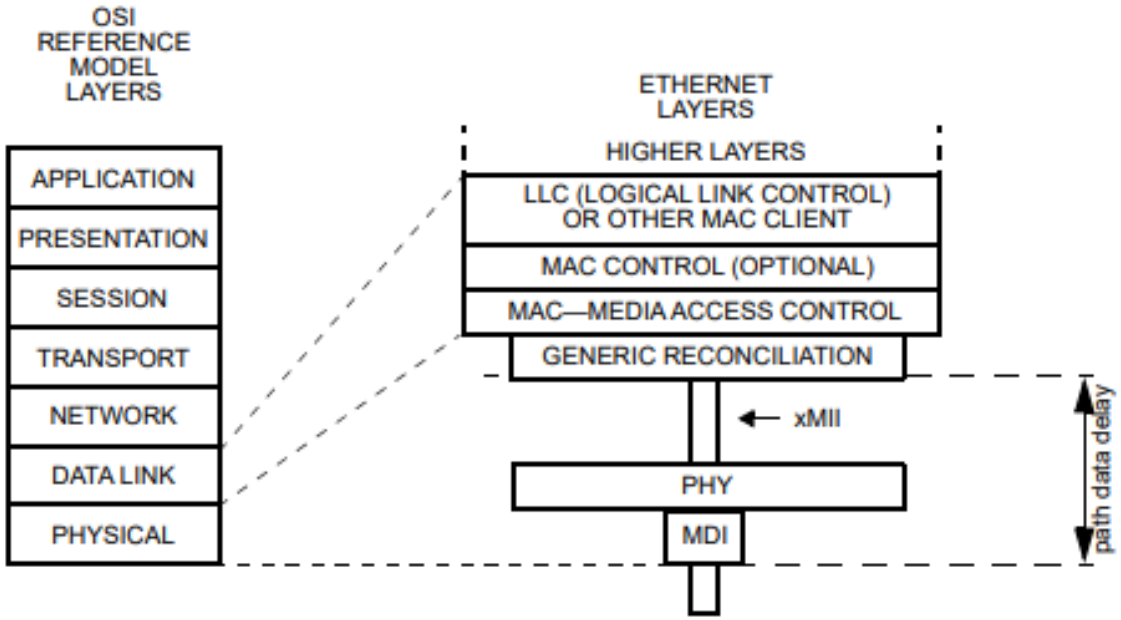
# Introduction – path data delay

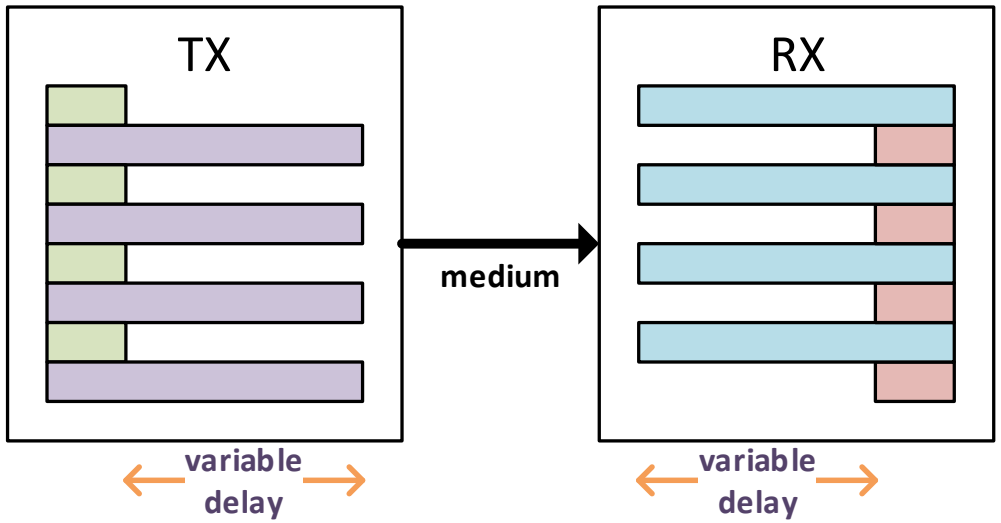
The path data delay is meant to reflect a PHY layer's in-to-out delay. The arrival/departure time at the MDI can thus be inferred from the arrival/departure time at the MII.

Departure timestamp = Tx MII timestamp + Tx path data delay
Arrival timestamp = Rx MII timestamp - Rx path data delay

The sum of the reported receive path data delay values through all the PHY sublayers should reflect the actual MDI -> MII delay.
The sum of the reported transmit path data delay values through all the PHY sublayers should reflect the actual MII -> MDI delay.

There are also guidelines in Clause 90 (and Annex 90A) about how to report the path data delay when the the delay is not constant. e.g. half the PCS lanes get delayed on the transmit side, while the other half get delayed on the receive side.
➔ Allocate **all** the 'variable delay' to the transmitter's path data delay, and **none** to the receiver's path data delay.
(See 90A.7 in 802.3cx)



# Delay Through a PHY with an n:m PMA

When a PHY has an n:m PMA sublayer, it **always** exists in series with an m:n PMA sublayer.

Transmit :

- m:n delays odd PCS lanes
- n:m delays even PCS lanes

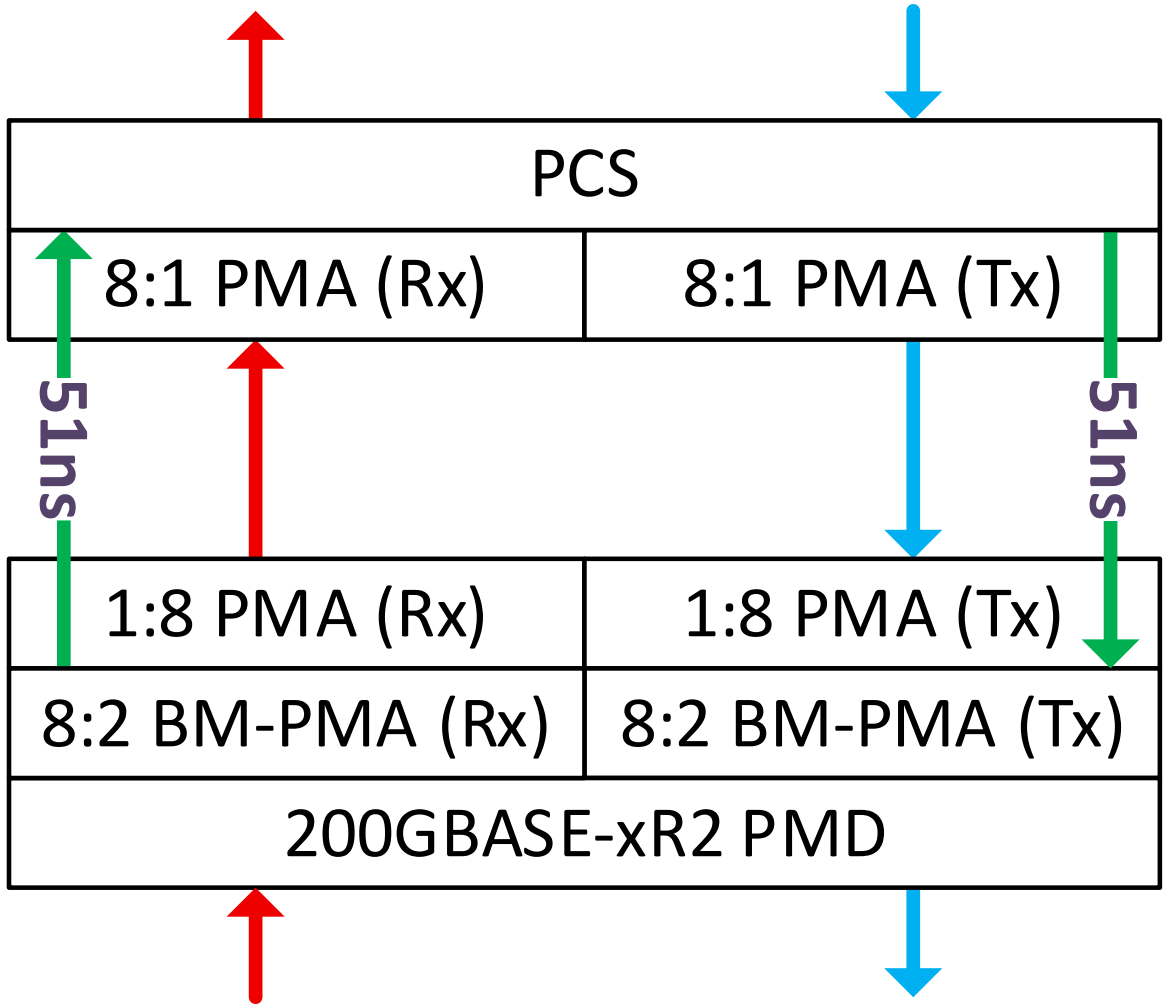
Receive:

- n:m delays odd PCS lanes
- m:n delays even PCS lanes

Overall : the **variable delays** from the m:n and n:m in series add up to **real delay.**

- No different than if the m PCS lanes were to go through a FIFO with a fixed delay.
- The sum of the path data delay values of the PHY sublayers should reflect this.

Example with 200GBASE-R
(ignoring additional implementation delays & AUI delay)
Total Tx path data delay through n:m and m:n = 51ns
Total Rx path data delay through n:m and m:n = 51ns



# What 176.10 Actually Says…

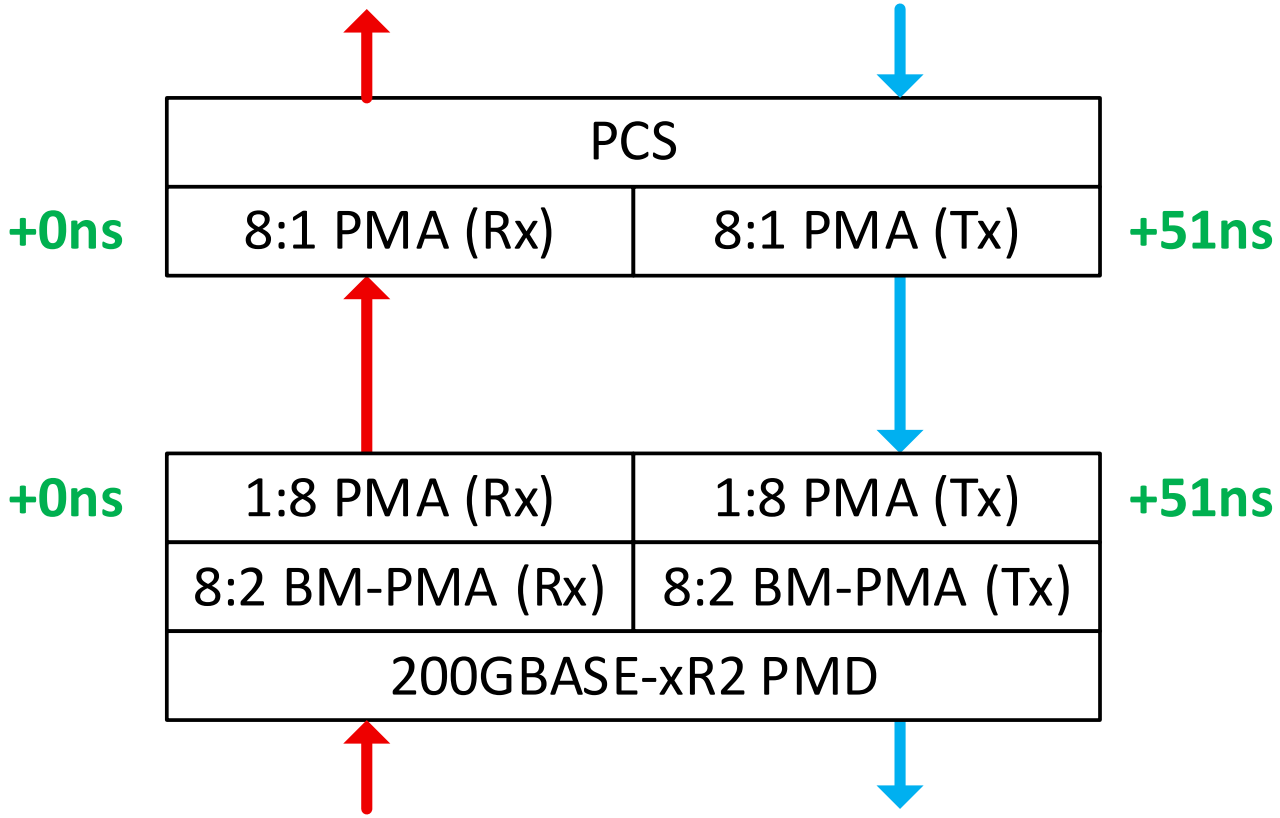
Currently, 176.10 makes no distinction between m:n and n:m PMAs:

The transmit path data delay is reported as the delay incurred by the first bit of the PCS alignment marker of the PCS lane with the largest delay through the PMA. The receive path data delay is reported as the delay incurred by the first bit of the PCS alignment marker of the PCS lane with the smallest delay through the PMA.

BOTH m:n and n:m Tx PMAs report their path data delay with the largest in-to-out PCS lane delay
BOTH m:n and n:m Rx PMA : report their path data delay with the smallest in-to-out PCS lane delay

200GBASE-R example (ignoring additional implementation delays) :
8:1 PMA Tx path data delay = 51ns
1:8 PMA Tx path data delay = 51ns
Total Tx path data delay = 102ns : 51ns too large!

8:1 PMA Rx path data delay = 0ns
1:8 PMA Rx path data delay = 0ns
Total Rx path data delay = 0ns : 51ns too small!



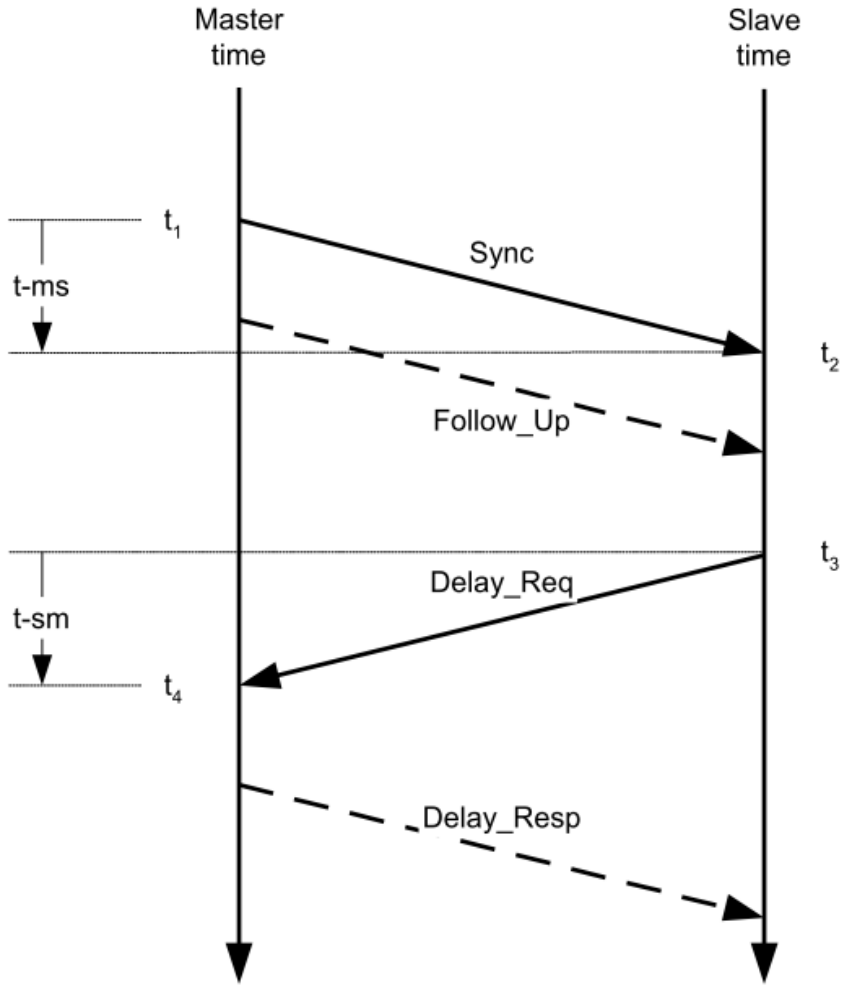
# Does this really need to be fixed? YES!

The inaccurate path data delay values look like a link asymmetry, which directly affects time-sync:

If the TimeTransmitter's (master's) PHY reports its path data delay incorrectly:

- $t_1$ is 51ns too large (Tx MII time + Tx path data delay)
  - **t-ms** appears **smaller** than it should
- $t_4$ is 51ns too large (Rx MII time - Rx path data delay)
  - **t-sm** appears **larger** than it should

Time-of-Day Correction =
= $[(t_4-t_3)-(t_2-t_1)]/2$
= [(51ns)-(-51ns)]/2
= 51ns.
It appears as if the TimeTransmitter (master) is 51ns ahead of the TimeReceiver (slave).
The TimeReceiver (slave) would **erroneously** adjust its Time-of-Day by +51ns!



# Possible Remedy (Comment #131)

The suggested Remedy attached to Comment #131 was to specify that, since m:n Tx is functionally identical to n:m Rx (and vice versa):
- All “Rx” management objects for the n:m PMA actually apply to the Tx datapath of the n:m PMA
- All “Tx” management objects for the n:m PMA actually apply to the Rx datapath of the n:m PMA

This way, the ‘Tx path data delay’ for the n:m PMA would be reported using the instructions for the ‘Rx path data delay’ of the m:n PMA.

But, speaking to the editorial team, this is NOT the intent of the standard.
- Rx management objects always point to the Rx datapath
- Tx management objects always point to the Tx datapath

=> True for n:m, m:n or n:n PMAs

The suggested remedy attached to comment #131 is thus not appropriate. ‘Reject’ is justified.

This absolutely needs to be clarified, especially for the PMA_delay status variables for TimeSync. There, the 'Rx' and 'Tx' delays are calculated differently. The delays would be reported incorrectly if approach A were to be used!

*SuggestedRemedy*

In sections 176.5.2, change the sentence to:
The transmit function of the n:m PMAs is identical to the receive function of the m:n PMAs (see 176.4.3), and uses the 'RX' PMA Management variables.

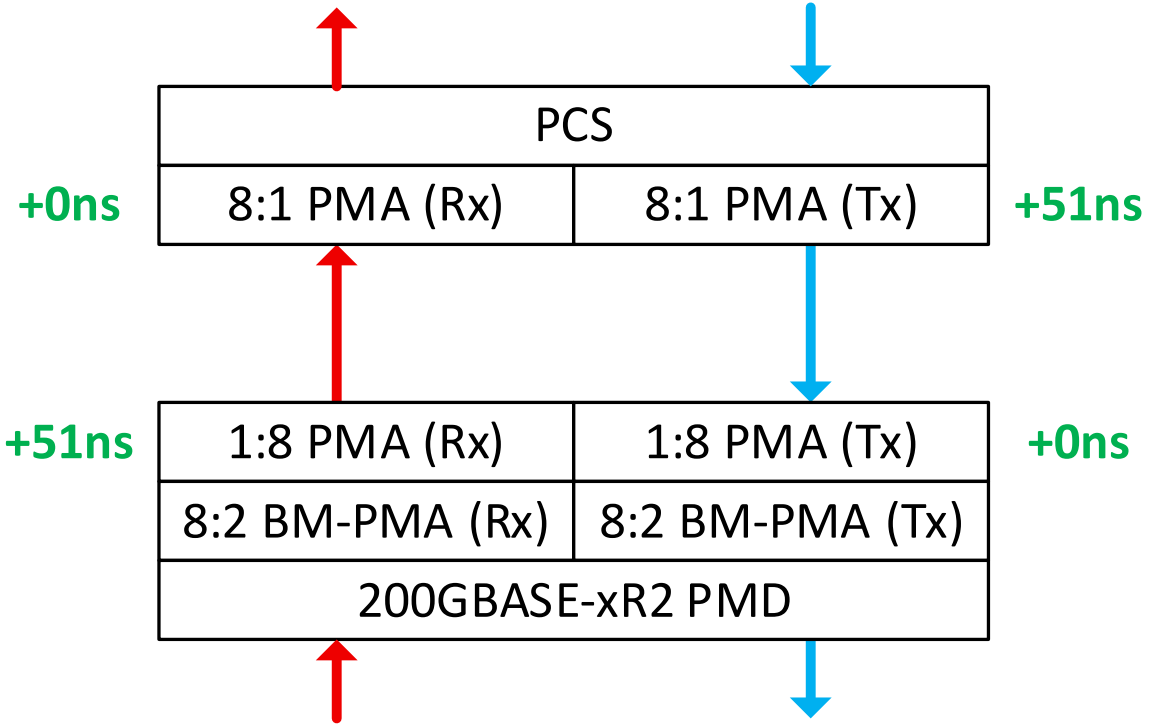
(and a similar update for section 176.5.3)

# Alternate Remedy to Comment #131

A more complete remedy is to specify which PMAs use the largest in-to-out PCS lane delay, and which use the smallest in-to-out PCS lane delay, rather than just blanket ‘receive’ and ‘transmit’ instructions in 176.10.

Change:
The transmit path data delay is reported as the delay incurred by the first bit of the PCS alignment marker of the PCS lane with the largest delay through the PMA.
To:
The transmit path data delay of m:n and n:n PMAs, and the receive path data delay of n:m PMAs, are reported as the delay incurred by the first bit of the PCS alignment marker of the PCS lane with the largest delay through the PMA.

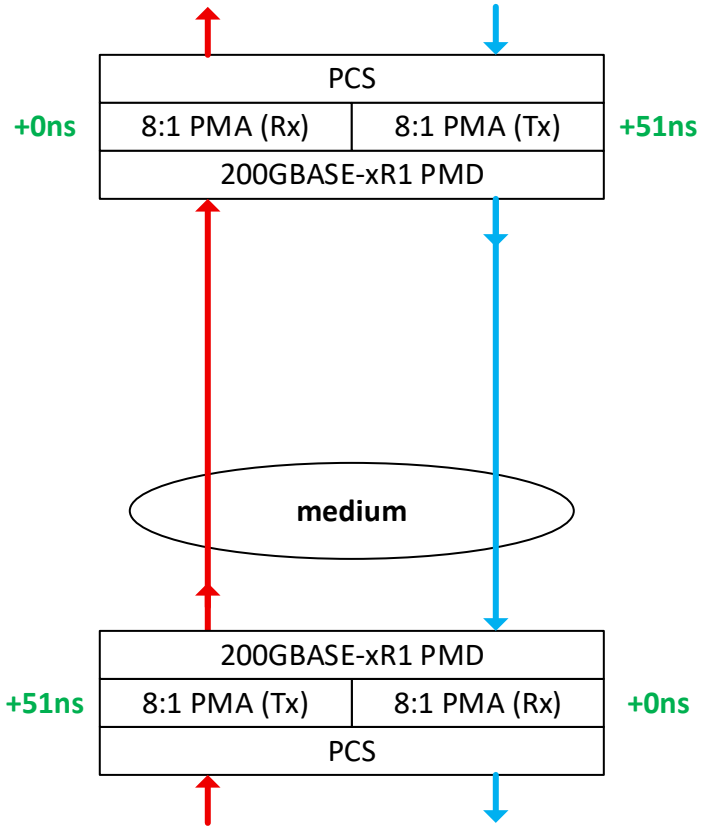
Change:
The receive path data delay is reported as the delay incurred by the first bit of the PCS alignment marker of the PCS lane with the smallest delay through the PMA.
To:
The receive path data delay of m:n and n:n PMAs, and the transmit path data delay of n:m PMAs, are reported as the delay incurred by the first bit of the PCS alignment marker of the PCS lane with the smallest delay through the PMA.



# Thanks!

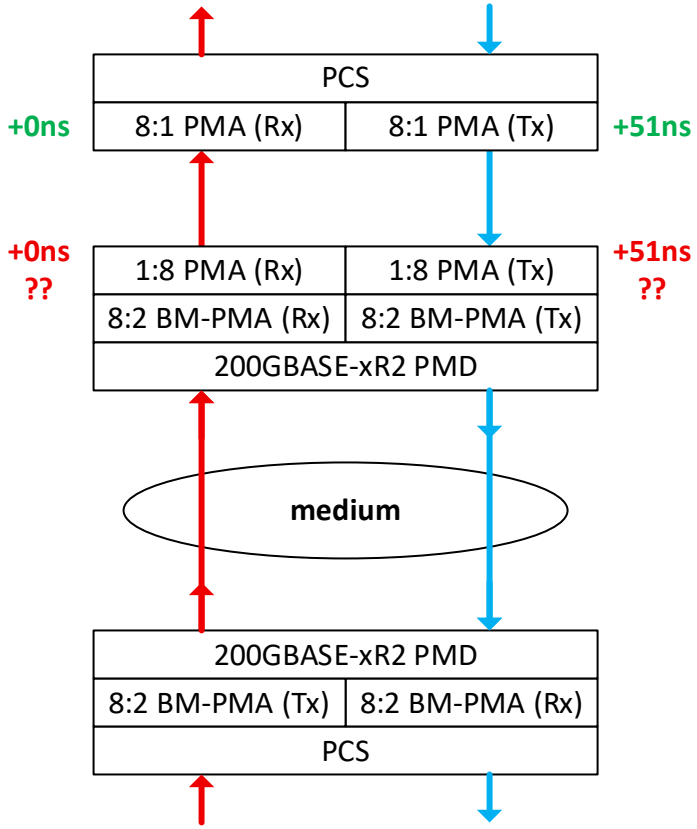
# Backup : Incorrect path data delay causes perceived Asymmetry

200Gbps/lane PMD:




For a 200Gbps link, the allocation of the variable delays is symmetric in the upstream and downstream directions.

100Gbps/lane PMD where the local PHY has a 200Gbps AUI, and an n:m PMA reporting its path data delays incorrectly:




If the PHY with the n:m PMA reports its path data delay incorrectly, the upstream and downstream directions appear asymmetric.