FREEDOM TO INNOVATE

Path delay variance from multi-PCS lane distribution

Richard Tse, Steve Gorshe

IEEE 802.3 Geneva, Switzerland January 2020

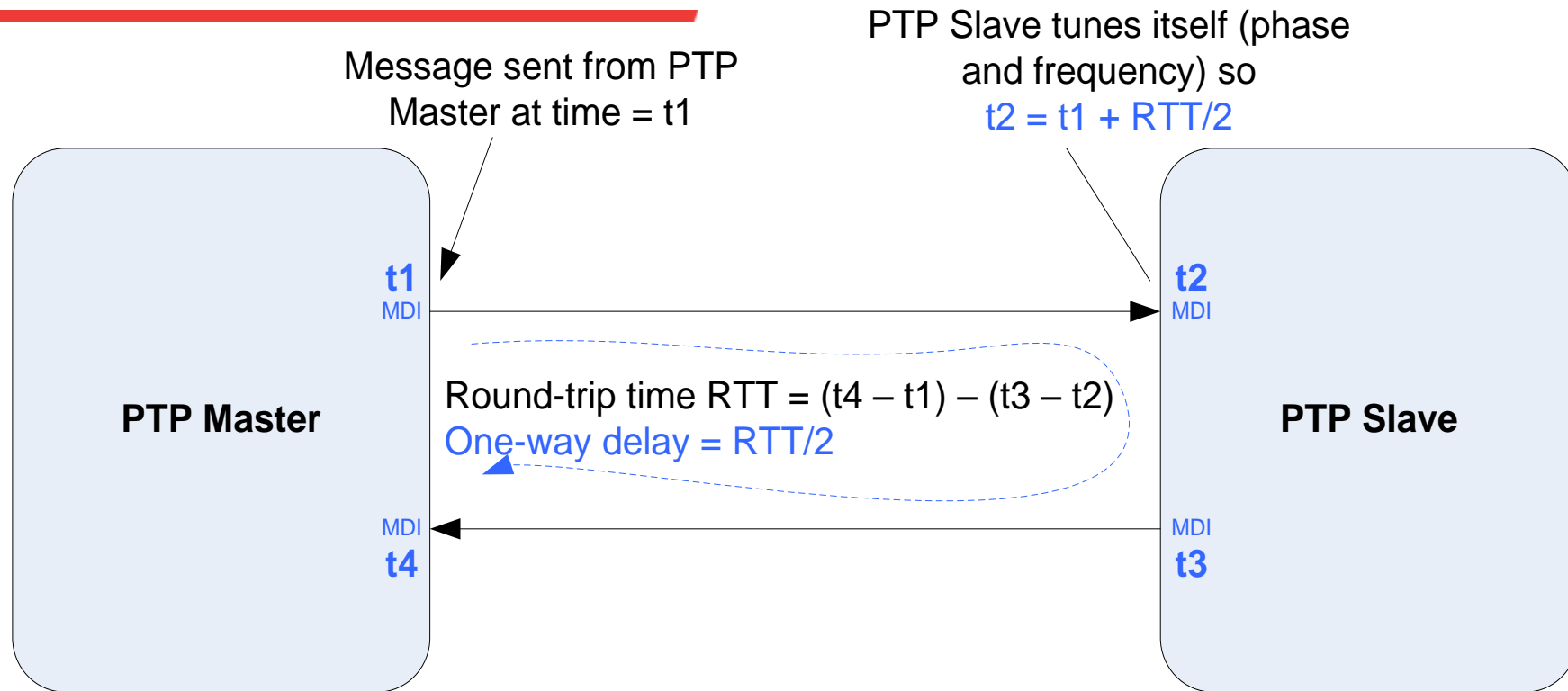microchip.com

MICROCHIP

Agenda

- Background
  - PTP Time Distribution Mechanism
  - Time Error Measurement Model
  - PTP Timestamp Generation Model
  - Current IEEE 802.3 Support for Time Synchronization
  - Why Can't High Accuracy Time Transport be Achieved Now?
- Problem: Multi-PCS lane distribution
  - Transmitter alignment behavior
  - Lane distribution delays
- Potential Solutions

MICROCHIP

Background

MICROCHIP

PTP Time Distribution Mechanism

Message sent from PTP Master at time = t1

PTP Slave tunes itself (phase and frequency) so
$t2 = t1 + RTT/2$

Because round-trip measurement is used, delay symmetry affects performance

PTP Master

t1
MDI

MDI
t4

Round-trip time RTT = (t4 – t1) – (t3 – t2)
One-way delay = RTT/2

t2
MDI

PTP Slave

MDI
t3

-Timestamps **t1** and **t4** (corresponding to MDI) are captured at the PTP Master
-Timestamps **t2** and **t3** (corresponding to MDI) are captured at the PTP Slave
-All timestamps are given to the PTP Slave so it can:
- calculate RTT
- do adjustments to make t2 = t1 + RTT/2

MICROCHIP

Time Error Measurement Model (for Boundary Clock)

- PTP Master and PTP Slave are ideal (no timestamping errors, perfectly stable clocks)
- Boundary Clock's time error (TE) is affected by timestamping errors on messages to/from Master and to/from Slave
  - other sources of TE are ignored for this discussion
- $|TE_{BC}| = 0.5*(|t1_{err\_bc}| + |t2_{err\_bc}| + |t3_{err\_bc}| + |t4_{err\_bc}|) = (|Tx_{timestamp\_error}| + |Rx_{timestamp\_error}|)$
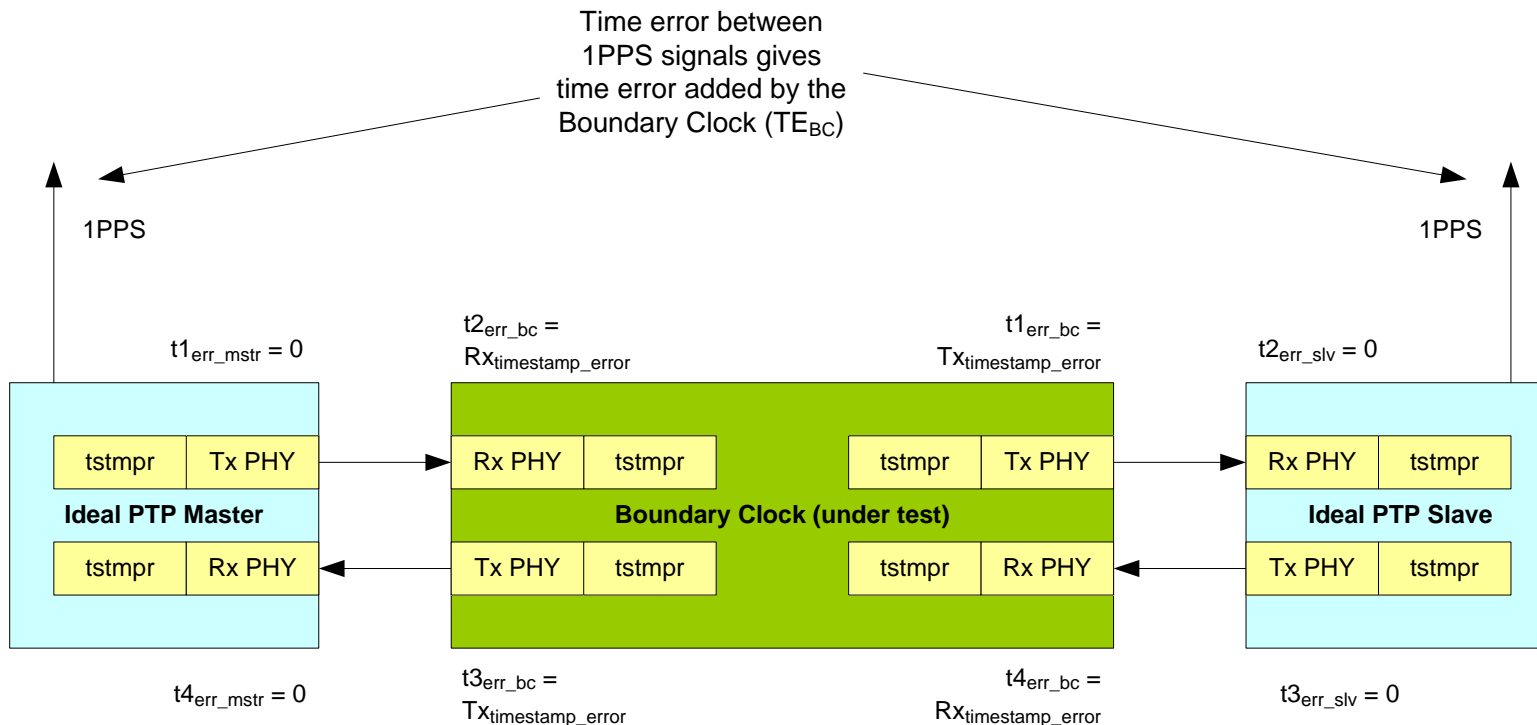
PTP Timestamp Generation Model

- A timestamp is generated at the time the "message timestamp point" crosses "reference plane", which is the intersection between the network (i.e. the medium) and the PHY

- Timestamp capture is implemented at the "timestamp measurement plane", which, in practice, occurs at point A and must be moved back to the reference plane

- *Good estimate of the PHY delay* ("path data delay", the time between the reference plane and the timestamp measurement plane) *is needed → varying delays should be compensated for*

- *Every endpoint needs to have the same understanding of the above concepts and how compensation is done*
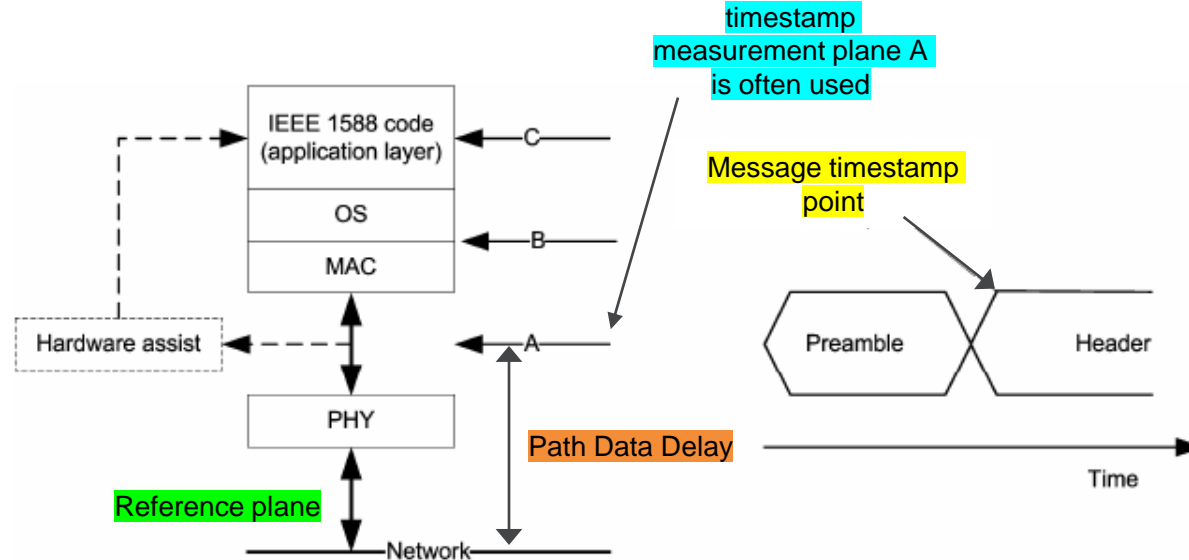
Current IEEE 802.3 Support for Time Synchronization (1)

- IEEE 802.3 Clause 90 provides support for a TimeSync Client
  - The optional Time Synchronization Service Interface (TSSI) supports protocols that require knowledge of packet egress and ingress time
  - Timestamping is done in the gRS, where the timestamp is captured when the message timestamp point crosses the xMII
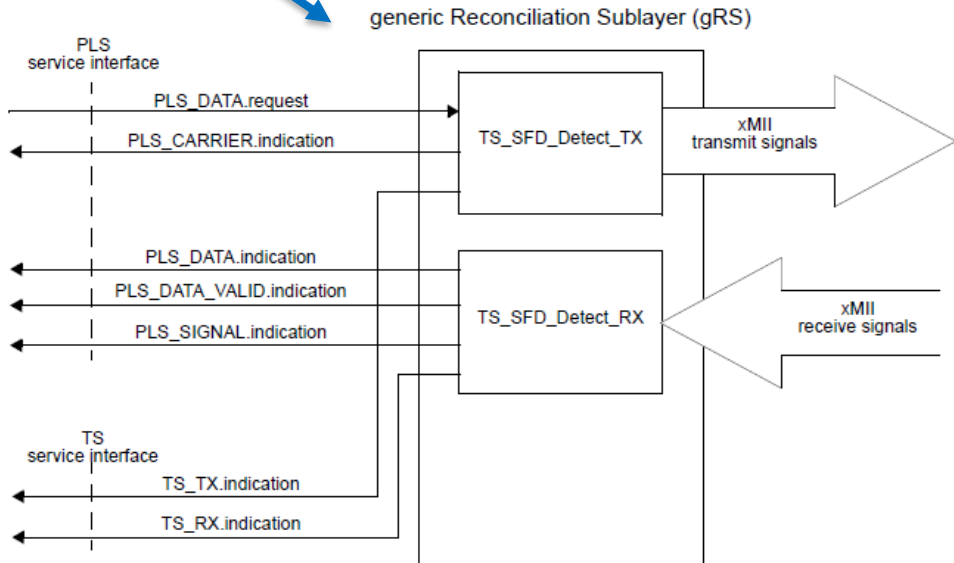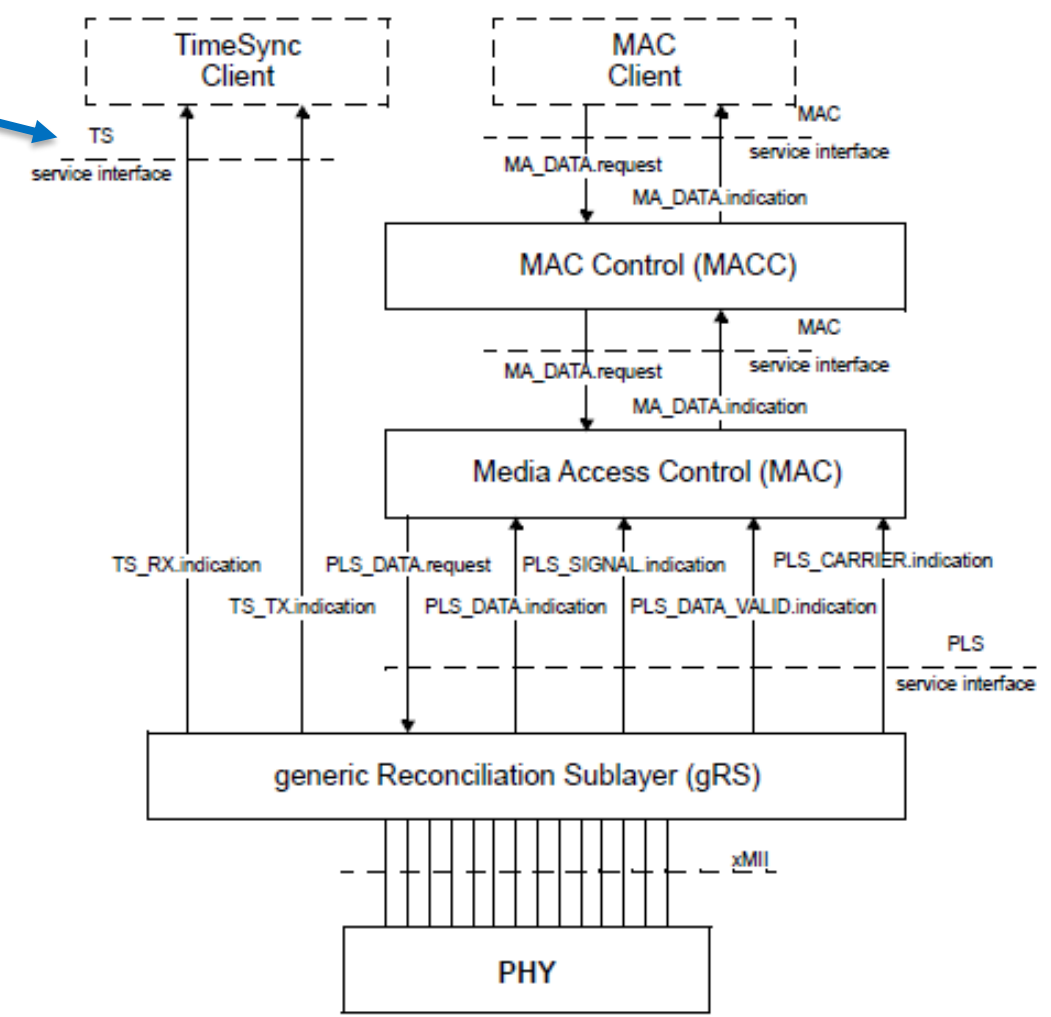

Figure 90–2—TS_SFD_Detect_TX and TS_SFD_Detect_RX functions within the generic Reconciliation Sublayer (gRS)

MICROCHIP

Current IEEE 802.3 Support for Time Synchronization (2)

- TSSI allows for "PHY" delay measurement to be done by TimeSync Client(s)
  - The **transmit path data delay is measured** from the beginning of the SFD at the xMII input to the beginning of the SFD at the MDI output.
  - The **receive path data delay is measured** from the beginning of the SFD at the MDI input to the beginning of the SFD at the xMII output.

- The obtained path data delay measurement is reported in the form of a quartet of values as defined for the TimeSync managed object class.
  - maximum transmit path data delay
  - minimum transmit path data delay
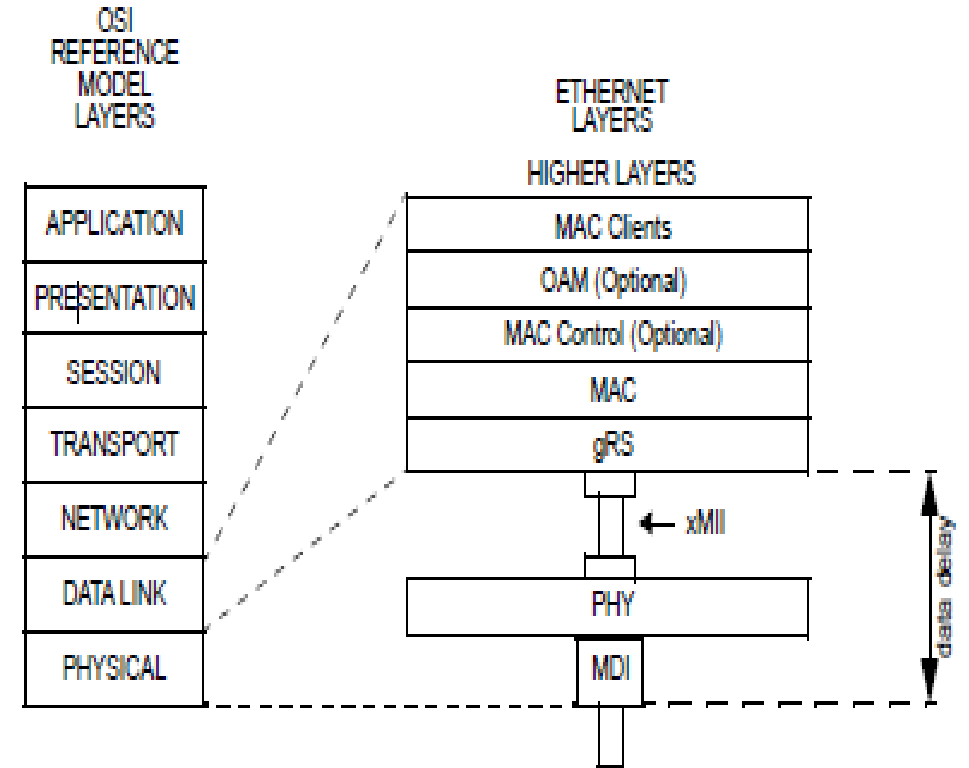 - maximum receive path data delay
  - minimum receive path data delay

Figure 90-3—Data delay measurement

Current IEEE 802.3 Support for Time Synchronization (3)

- **Multi-Lane – clause 90.7 (added in 2016):**

  - "The receiver of a multi-lane PHY is expected to include a buffer to compensate for skew between the lanes. This buffer selectively delays each lane such that the lanes are aligned at the buffer output. The earliest arriving lane experiences the most delay through the buffer and the latest arriving lane experiences the least delay through the buffer. The receive path data delay for a multi-lane PHY is reported as if the beginning of the SFD arrived at the MDI input on the lane with the smallest buffer delay."

- **FEC – clause 90.7 (added in 2018):**

  - "For a PHY that includes an FEC function, the transmit and receive path data delays may show significant variation depending upon the position of the SFD within the FEC block. However, since the variation due to this effect in the transmit path is expected to be compensated by the inverse variation in the receive path, it is recommended that the transmit and receive path data delays be reported as if the SFD is at the start of the FEC block."

MICROCHIP

Why Can't High Accuracy Time Transport be Achieved Now with IEEE 802.3?

- PTP timestamping is done at the MDI
- IEEE 802.3's timestamping is done at the xMII (per clause 90 of IEEE 802.3)
- PHY path data delay must be known for the PTP message to move the timestamp from xMII to MDI
- Many newer 802.3 PHYs have fundamental dynamic variations in their path data delay
- But
  - Path data delay variations in the PHY are not inherently visible at the xMII
- Thus
  - IEEE 802.3's current timestamping mechanism does not inherently support high accuracy on PHYs with path data delay variations
  - Specifications are needed on how to deal with each path data delay variation
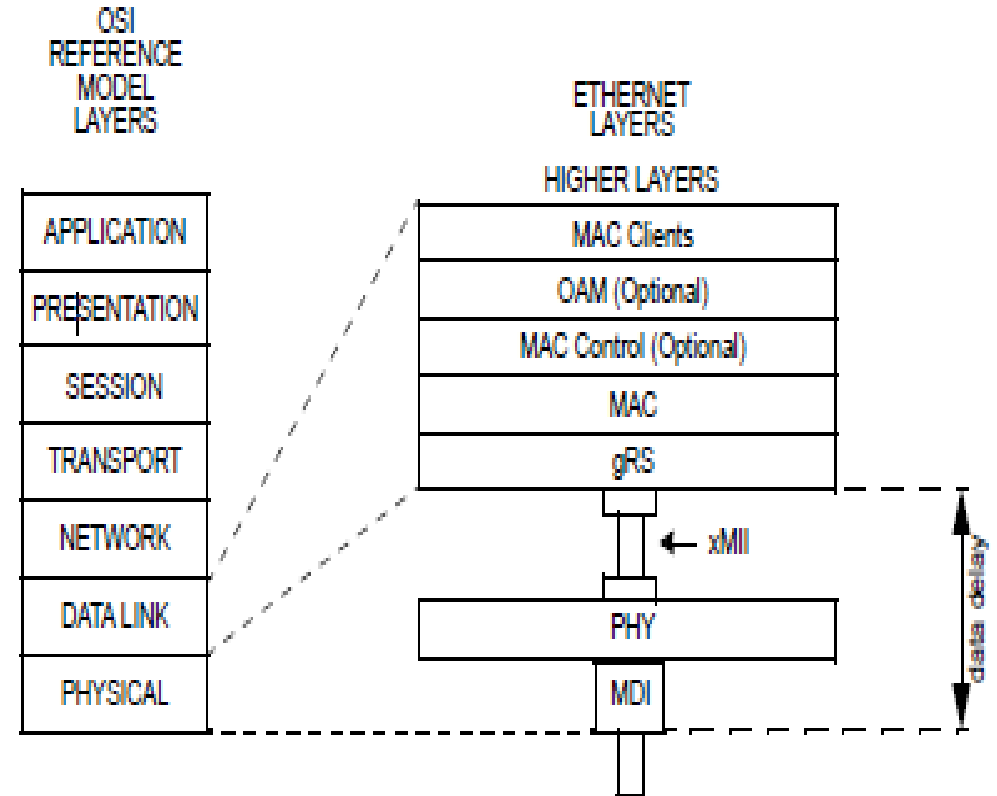
Figure 90–3—Data delay measurement

Path Data Delay Variations in 100GE PHY

Timestamps are captured at xMII

Block distribution to multi-PCS lanes, Alignment Marker insertion/removal (and their corresponding Idles), and FEC all inherently cause dynamic path data delay variation

Timestamps should correspond to the time at MDI
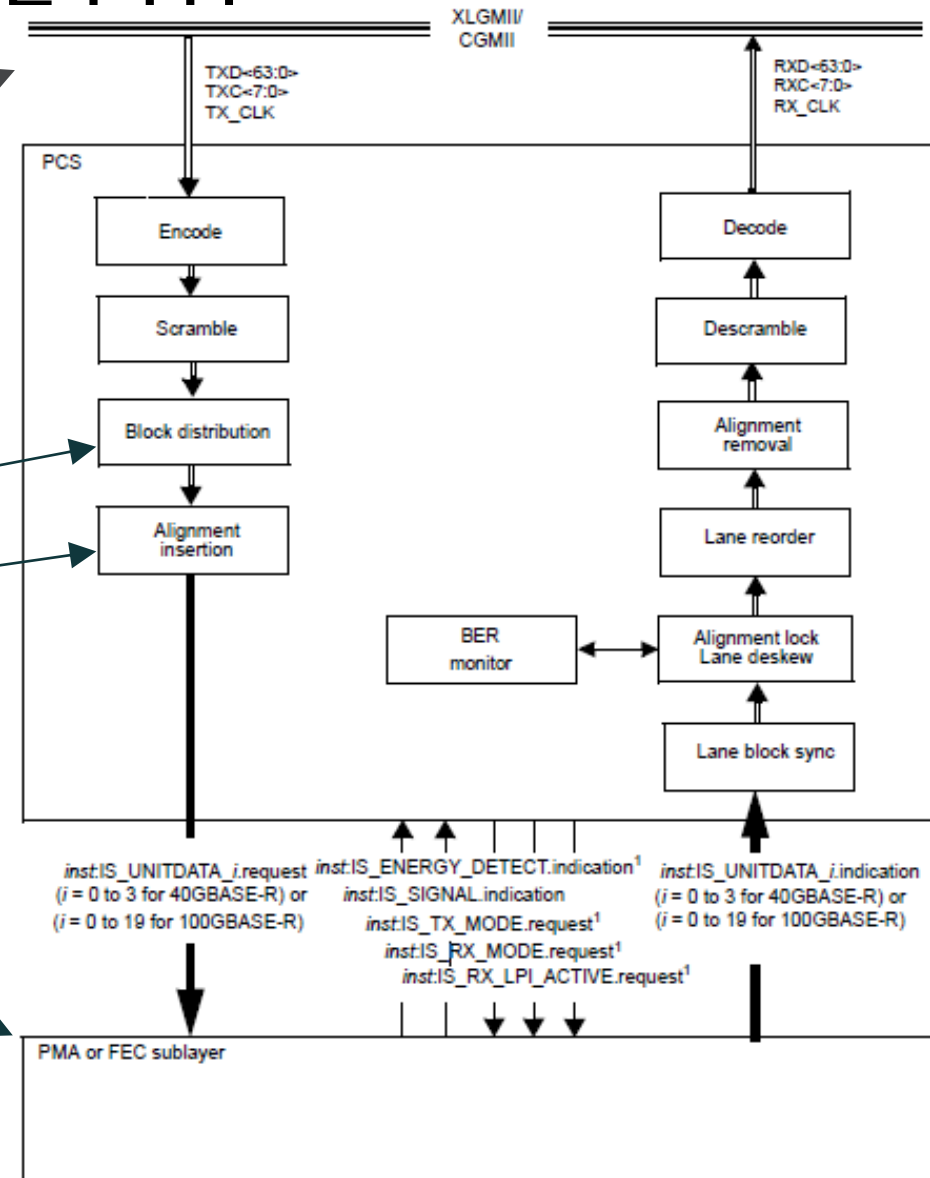
Figure 82–2—Functional block diagram

Problem: Path Data Delay (PDD) variance from multi-PCS lane distribution function needs to be accounted for in a standardized manner

- The characteristics of PHY path data delay, PDD_x, (PDD_1 + PDD_2), and (PDD_3 + PDD_4), must be specified to allow consistency between interworking PHYs so an accurate RTT can be measured

PHY MLD Block Distribution
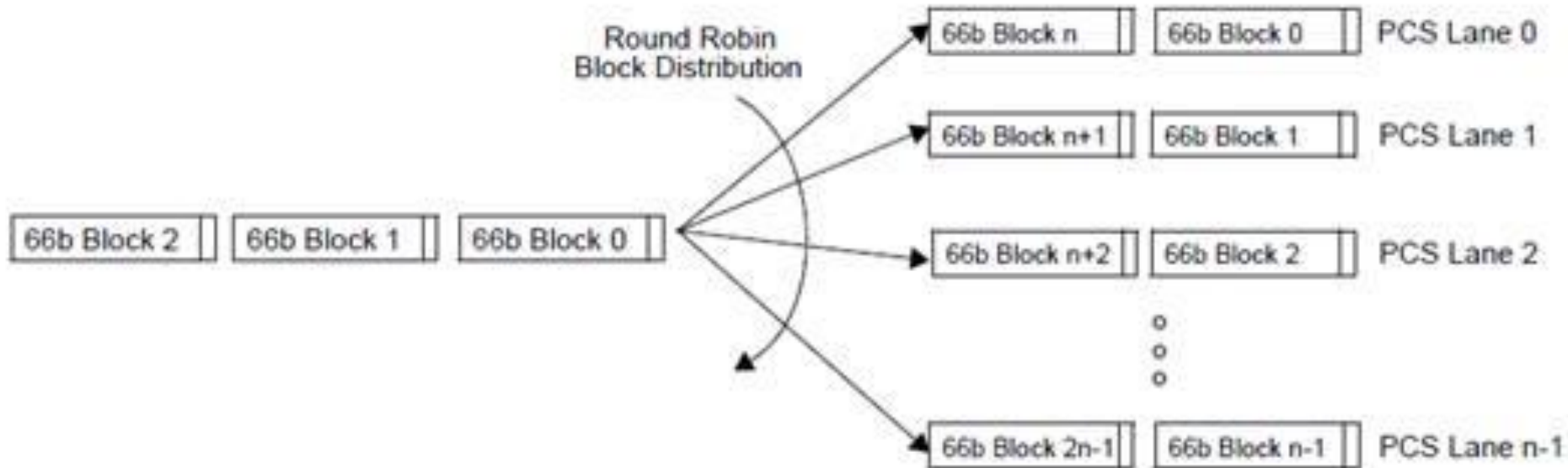
Figure 82–6—PCS Block distribution

PCS-Lane Distribution Interpretation Option Details (1)

Ambiguities in IEEE 802.3 affect path data delays.

No instructions are given in IEEE 802.3 on how to handle these deterministic but varying delays:

- NxPCS lane Transmitter Interpretation Options
  - A. 66B blocks and timestamps are not aligned at NxPCS lane transmitter
    - xMII to MDI has constant path data delay for every lane
      - Data for Lane 0 arrives first at xMII and is transmitted first at MDI
      - Data for Lane N arrives last at xMII and is transmitted last at MDI
    - 66B blocks on each lane have a different timestamp because they cross the reference plane at different times
      - Timestamper at Tx xMII uses the same xMII-to-MDI constant data path delay for every lane
    - Lane-to-lane skew of 66B blocks at the transmitter is removed by Rx deskew buffers

MICROCHIP

PCS-Lane Distribution Interpretation Option Details (2)

- NxPCS lane Transmitter Interpretation Options (continued)
  - B. 66B blocks and timestamps are aligned at NxPCS lane transmitter
    - xMII to MDI path has different path data delay for each lane
      - Data for Lane 0 arrives first at xMII and is transmitted at the same time as lane N at MDI, causing largest path data delay
      - Data for Lane N arrives last at xMII and is transmitted at the same time as Lane 0 at MDI, causing smallest path data delay
    - 66B blocks on every lane have the same timestamp because they cross the reference plane at the same time
      - Timestamper at Tx xMII uses appropriate xMII-to-MDI path data delay for each lane
    - No lane-to-lane skew of 66B blocks

MICROCHIP

PCS-Lane Distribution Interpretation Option Details (3)

- NxPCS lane Transmitter Options (continued)
  - C. 66B blocks are aligned but timestamps are not aligned at NxPCS lane transmitter
    - xMII to MDI path has different path data delay for each lane
      - Data for Lane 0 arrives first at xMII and is transmitted at the same time as lane N at MDI, causing largest path data delay
      - Data for Lane N arrives last at xMII and is transmitted at the same time as Lane 0 at MDI, causing smallest path data delay
    - Timestamps assume a constant data path delay for all lanes
      - Timestamper at Tx xMII uses the same xMII-to-MDI constant path data delay for every lane
    - No lane-to-lane skew of 66B blocks

MICROCHIP

PCS-Lane Distribution Interpretation Option Details (4)

- NxPCS lane Receiver Options:
  - After deskew buffers, all lanes are aligned
    - For N-lane transmitter type "A", intrinsic lane-to-lane skew of 66B blocks is "moved into the medium" by the deskew function
    - For N-lane transmitter types "B" and "C", there is no skew of 66B blocks between lanes
  - MDI to xMII multiplexer causes varying path data delay
    - All lanes are deskewed and are ready to go to xMII
    - Data for Lane 0 goes to xMII first and has smallest path data delay
    - Data for Lane N goes to xMII last and has largest path data delay
  - How is this lane-to-lane delay variation handled?

MICROCHIP

PCS-Lane Distribution Interpretation Options Details (4)

- Figure shows examples of the 3 Options

- Arrival times at each stage are shown (Arrive at, Transmit at)

- The delays through each functional stage are shown (Delay, Fdly, link delay)
  - Constant delays are assumed to be 0 where the actual values don't matter

- The departure timestamps at Tx (dep_tstmp) and arrival timestamps at Rx (arr_tstmp) are shown

- The calculated link delay (Link_delay) is shown for the span (end-to-end measurement)
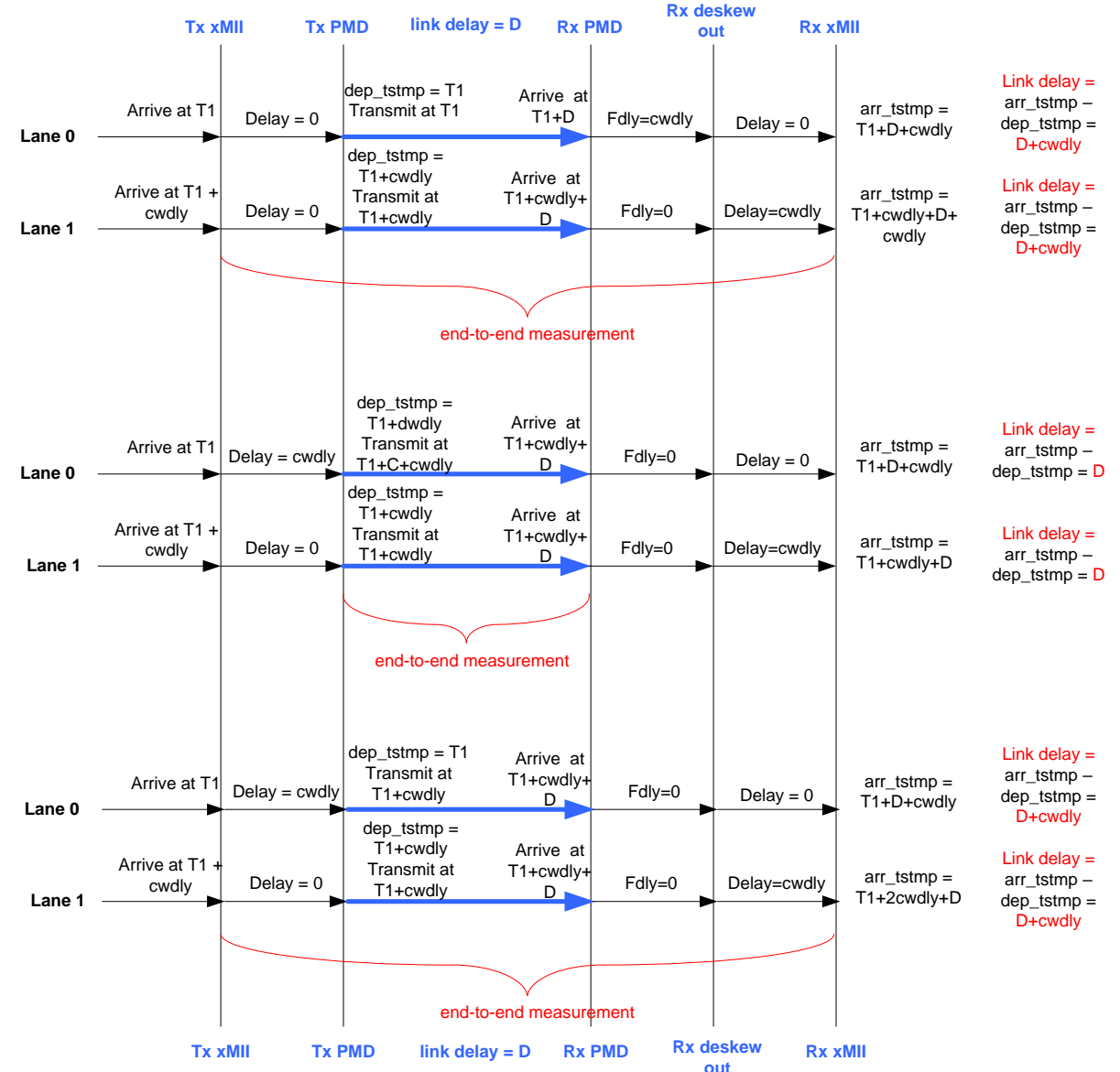
PCS-Lane Distribution Delays – Constant vs per-Lane

- There are two inherent approaches for determining the xMII-to-MDI delay on multi-PCS lane PHYs

  1. Method 1 – Account for the delay between the MII and the lane that carries the message timestamp point of the PTP message.

  2. Method 2 – Because the Tx + Rx lane distribution delay is a constant for every lane, use this constant delay regardless of which lane carries the message timestamp point.

     - This is like how IEEE 802.3 handles FEC delays

MICROCHIP

PCS-Lane Distribution Delays: Method 1

- For a multilane PHY, after deskew delays are accounted for appropriately and since timestamping is at the MDI, would the timestamps be the same regardless of which lane the message's timestamp reference point is transmitted on (or received on)?

  - Since all lanes are transmitted at the same time and received at the same time (after deskew) at the MDI, it would seem this is a valid conclusion.

90.7 Data delay measurement

The TimeSync capability requires measurement of data delay in the transmit and receive paths, as shown in Figure 90–3. The transmit path data delay is measured from the beginning of the SFD at the xMII input to the beginning of the SFD at the MDI output. The receive path data delay is measured from the beginning of the SFD at the MDI input to the beginning of the SFD at the xMII output.

PCS-Lane Distribution Delays: Method 1 (continued)
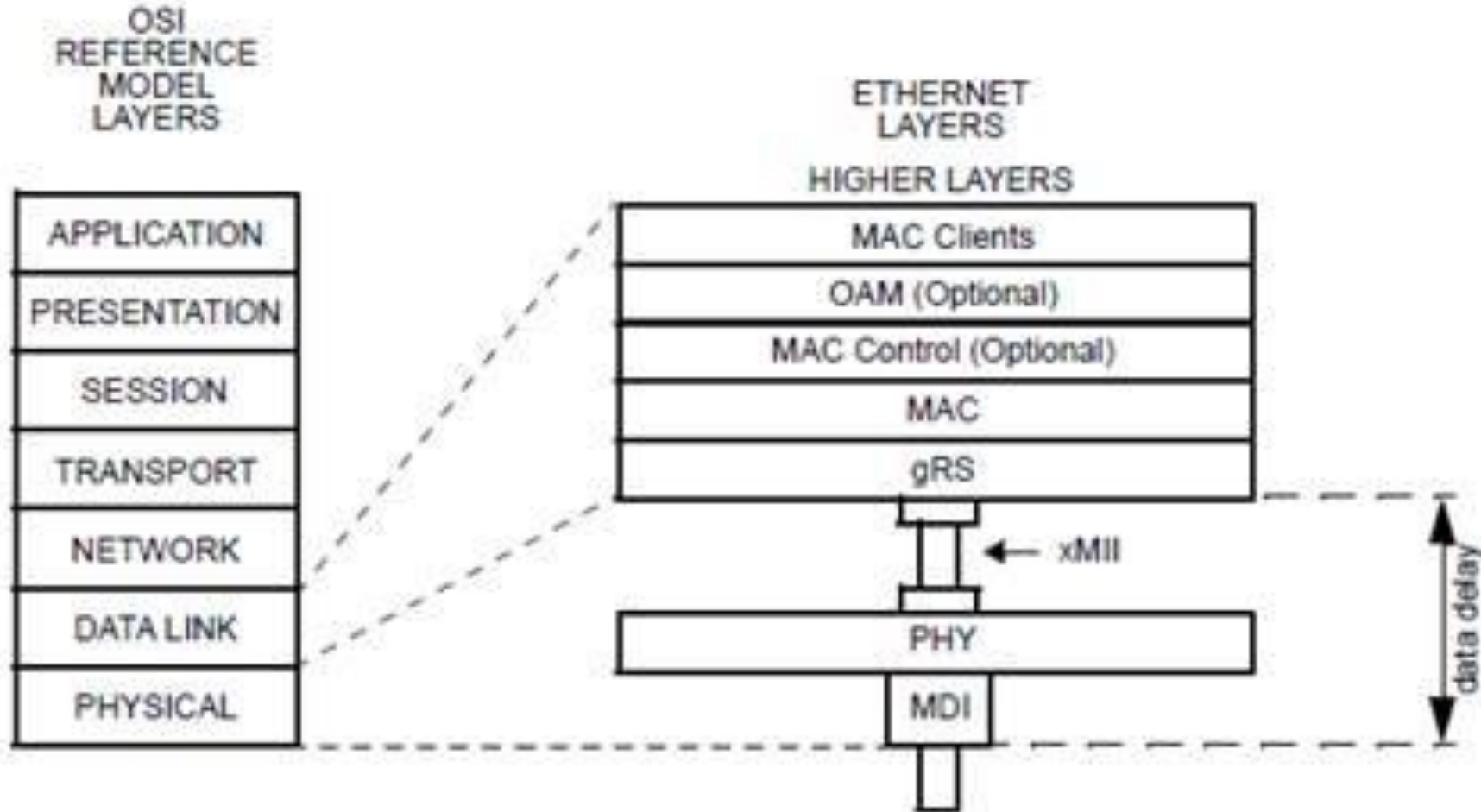
Figure 90–3—Data delay measurement

PCS-Lane Distribution Delays: Method 1 (continued)

- However, this means that PHY path data delay (between xMII and MDI, as per Figure 90-3 above) is not the same for every lane because the MDI-to-xMII multiplexing delay (for Rx) and xMII-to-MDI demultiplexing delay (for Tx) is different for each lane (as shown in Figures 82-3 and 82-4 below). In the Tx direction, 66B blocks going to lane 0 have the most delay and 66B blocks going to lane 3 have the least delay. In the Rx direction, the opposite is true. To capture an accurate timestamp at the xMII (as per the IEEE 802.3 model), the lane-based intrinsic delay must be included as part of the PHY path data delay.

  - Was this the intent?

MICROCHIP

PCS-Lane Distribution Delays: Method 1 (continued)

xMII

Other PCS functions

1 Arrival at distribution function follows #0, #1, #2, then #3 ordering

distribution waiting buffer

Lane distribution

2 All lanes depart the interleave function at same time thus,
Lane #0 waits the longest time
Lane #3 waits the shortest time

| Lane #0 Tx port | Lane #1 Tx port | Lane #2 Tx port | Lane #3 Tx port |

Tx PHY Data Delay

Tx PHY Data Delay is not the same for all lanes

3 PTP timestamp represents when message timestamp point crosses the PMA here. If no Tx skew, **all lanes have the same timestamp.**

xMII

Rx PHY Data Delay is not the same for all lanes

Other PCS functions

departure from multiplex function follows #0, #1, #2, then #3 ordering thus,
Lane #0 waits the shortest time
Lane #3 waits the longest time

6

Lane multiplex

multiplex waiting buffer

All lanes depart deskew function at the same time

5

Lane deskew

| Lane #0 Rx port | Lane #1 Rx port | Lane #2 Rx port | Lane #3 Rx port |

Rx PHY Data Delay

4 PTP timestamp represents when message timestamp point crosses the PMA here. If no Rx skew, **all lanes arrive at the same time and have the same timestamp.** Deskew function makes all lanes look like they arrived at the same time as the latest arriving lane.

23

PCS-Lane Distribution Delays: Method 2

- These multi-PCS lane PHY path data delays could also be designated to be a constant value for all lanes if the principle that is used for FEC's varying intrinsic delays is applied for multilane's multiplexing/demultiplexing varying intrinsic delays.

  - i.e., the Tx intrinsic demultiplexing delay is balanced by the Rx multiplexing intrinsic delay, making the aggregated demux/mux delay a constant.
  - Was this principle on anyone's mind when the multiplane PHY function was defined?

MICROCHIP

PCS-Lane Distribution Delays: Method 2 (continued)

xMII

Other PCS functions

1 Arrival at distribution function follows #0, #1, #2, then #3 ordering

distribution waiting buffer

Lane distribution

2 All lanes depart the interleave function at same time thus,
Lane #0 waits the longest time
Lane #3 waits the shortest time

Tx PHY Data Delay

| Lane #0 Tx port | Lane #1 Tx port | Lane #2 Tx port | Lane #3 Tx port |

Tx PHY Data Delay is the same for all lanes

3 Distribution function's delay variance is a defined to be a constant (actual variance is cancelled out by the peer Rx multiplex function).
Departure timestamps are defined to have a constant offset relative to timestamp at xMII.
Despite departing at the same time, all lanes have different timestamps.

xMII

Rx PHY Data Delay is the same for all lanes

Other PCS functions

departure from multiplex function follows #0, #1, #2, then #3 ordering thus,
Lane #0 waits the shortest time
Lane #3 waits the longest time

6

Lane multiplex

multiplex waiting buffer

All lanes depart deskew function at the same time

5

Lane deskew

Rx PHY Data Delay

| Lane #0 Rx port | Lane #1 Rx port | Lane #2 Rx port | Lane #3 Rx port |

4 Multiplex function's delay variance is defined to be a constant, and undoes the delay variance added by the peer Tx distribution function.
Arrival timestamps are defined to have a constant offset relative to timestamp at xMII.
After deskew, all lanes arrive at the same time but have different timestamps.

25

Potential Solutions

- Clarify how to handle lane distribution delays (methods 1 and 2 in this presentation)
  - Method 2 handles lane distribution delay in the same manner as 802.3's FEC delay and might be easier to implement
- Clarify the NxPCS lane transmitter's intended behavior for lane-to-lane alignment (options A, B, and C in this presentation)
  - Option B and C might have been the intended architecture
  - Option C also matches up with Method 2 for lane distribution

- Thus: Method 2 and Option C might be a good choice

Questions?

Backup Information

Application Timing Requirements

Classes C and D were added in 2018 for 5G transport applications

- From ITU-T Recommendation G.8273.2, Timing characteristics of telecom boundary clocks and telecom slave clocks
  - Specifies the max timing errors that can be added by a telecom boundary clock
  - cTE: constant time error
  - dTE_L: low-passed dynamic time error
    - MTIE: Maximum Time Interval Error
    - TDEV: Time Deviation
  - TE_L: constant time error + low-passed dynamic time error
  - TE: constant time error + unfiltered dynamic time error

Time Error Type	Class	Requirement (ns)
max|TE|	A	100
	B	70
	C	30
	D	for further study
max|TE_L|	A, B, C	not defined
	D	5

Time Error Type	Class	Requirement (ns)	Observation interval τ (s)
dTE_L	A and B	MTIE = 40	m < τ ≤ 1000 (for constant temp)
	A and B	MTIE = 40	m < τ ≤ 10000 (for variable temp)
	C	MTIE = 10	m < τ ≤ 1000 (for constant temp)
	D	MTIE = for further study	
	A and B	TDEV = 4	m < τ ≤ 1000 (for constant temp)
	C	TDEV = 2	
	D	TDEV = for further study	

Class	cTE Requirement (ns)
A	±50
B	±20
C	±10
D	for further study

MICROCHIP

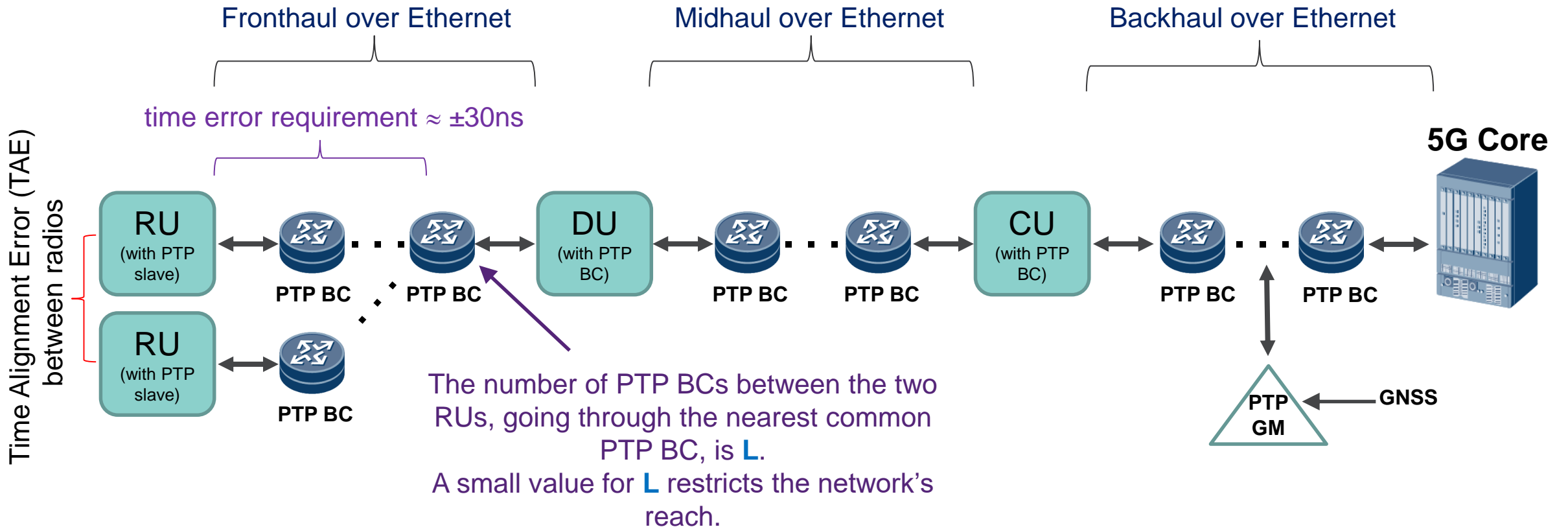# Resulting Performance vs Target Performance

- **Target Max|TE| = 30ns for class C Telecom Boundary Clock**
  - In a system, there are other sources of TE, in addition to those from timestamping, that use up the allowance

Ethernet Rate	Path Data Delay Variation per Tx/Rx Interface (ns)				Total TE per Tx or Rx Interface (ns)	Path Data Delay Variation Contribution to Max|TE|, per PTP Boundary Clock (ns)
	mismatched SFD timestamp point	Idle insert/remove (per Idle)	AM insert/remove	Lane Distribution		
GE	8	16	N/A	N/A	24	48
10GE	0.8	3.2	N/A	N/A	4	8
25GE	0.32	1.28	2.56	N/A	4.16	8.32
40GE	0.2	1.6	6.4	4.8	13	26
100GE	0.08	0.64	12.8	12.16	25.68	51.36
200GE	0.04	0.32	2.56	2.24	5.16	10.32
400GE	0.02	0.16	2.56	2.4	5.14	10.28

100GE is very important for C-RAN

Microchip

Transport Timing for 5G Centralized-RAN (C-RAN)

- C-RAN separates the BBU into "centralized" elements (Distributed Units (DUs) and Central Units (CUs)), allowing their resources to be efficiently shared between the Remote Units (RUs, radios)
- 5G mmWave NR (New Radio) has short reach (i.e. are densely packed) and high capacity
  - These qualities cause a need for a substantial fronthaul network (i.e. more timing hops) to connect RUs to their DUs

Fronthaul over Ethernet

Midhaul over Ethernet

Backhaul over Ethernet

time error requirement ≈ ±30ns

5G Core

Time Alignment Error (TAE) between radios

RU (with PTP slave)

PTP BC

PTP BC

DU (with PTP BC)

PTP BC

PTP BC

CU (with PTP BC)

PTP BC

PTP BC

RU (with PTP slave)

PTP BC

The number of PTP BCs between the two RUs, going through the nearest common PTP BC, is **L**.
A small value for **L** restricts the network's reach.

PTP GM

GNSS

Microchip

Application Timing Consequences

- ITU Q13/SG15 WD13-25 shows why improved PTP performance is needed:
  - For radio time alignment error (TAE) of 260ns (see "TAE" in the figure on slide 9):
    - With all Class B Boundary Clocks everywhere, including in the RUs,
      $L = 1$ (only direct connect can satisfy requirements!)
    - With all Class C Boundary Clocks in network and class B Slave Clocks in the RUs,
      $L = 5$
    - With all Class C Boundary Clocks in network and "class C-like" Slave Clocks in the RUs,
      $L = 7$
    - If results were expanded to use class D Boundary Clocks in network and "class C-like" Slave Clocks in the RUs, $L > 17$

- To build a practical C-RAN network for 5G applications, PTP Clock performance should be Class C or better

MICROCHIP