Further Considerations on Multi-PCS Distribution Delay

Xiang He (Huawei)

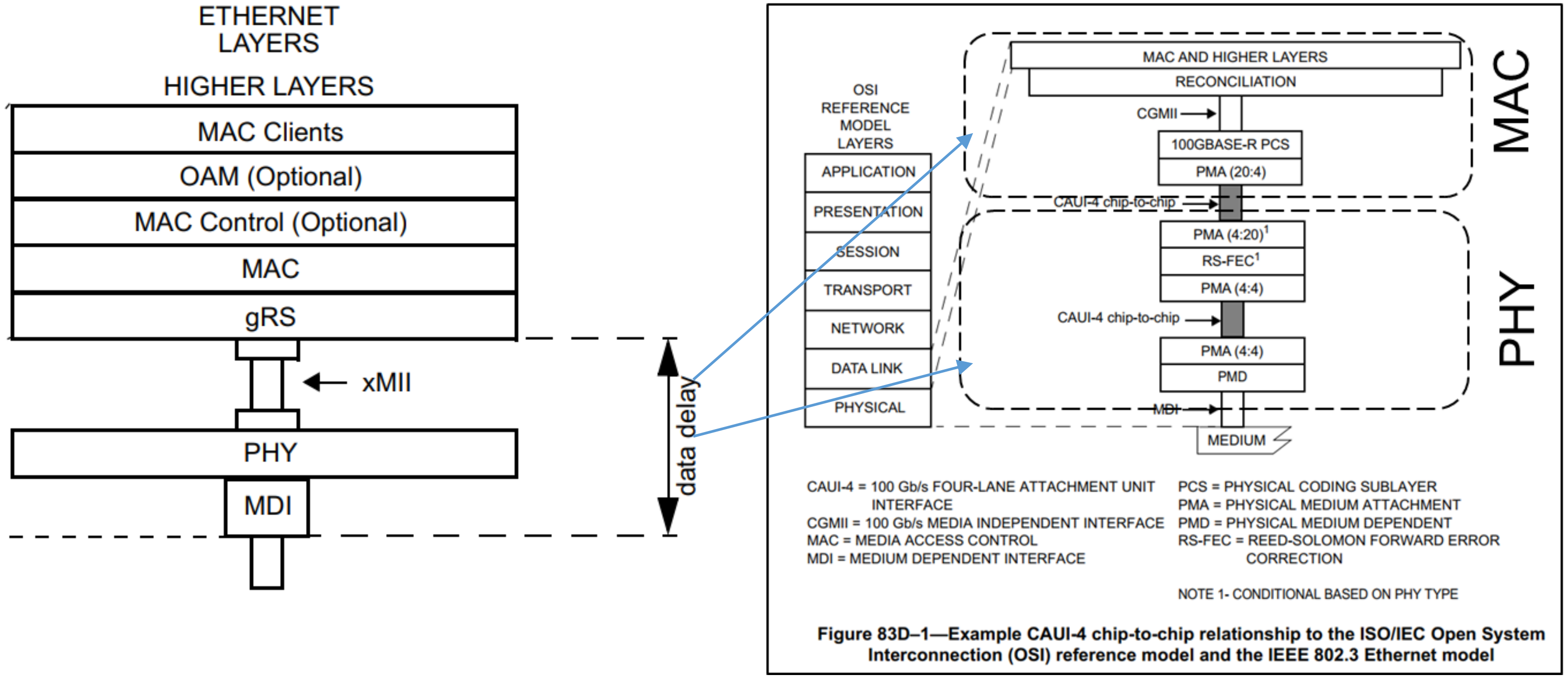
Jingfei Lv (Huawei)

Silvana Rodrigues (Huawei)

802.3cx ad hoc – 11/17/2020

Clarification on Delay Reporting

- Questions were raised on how to report the delay introduced by PCS lane distribution for each PTP message.
- Dynamic delay (caused by PCS lane distribution) will not be reported through the registers.
- Fixed delay shall be reported through the registers that are already defined in Clause 45.
- Both dynamic delay and fixed delay are compensated in the PTP messages.
- Taking separated MAC and PHY as an example:



CAUI-4 = 100 Gb/s FOUR-LANE ATTACHMENT UNIT INTERFACE
CGMII = 100 Gb/s MEDIA INDEPENDENT INTERFACE
MAC = MEDIA ACCESS CONTROL
MDI = MEDIUM DEPENDENT INTERFACE

PCS = PHYSICAL CODING SUBLAYER
PMA = PHYSICAL MEDIUM ATTACHMENT
PMD = PHYSICAL MEDIUM DEPENDENT
RS-FEC = REED-SOLOMON FORWARD ERROR CORRECTION

NOTE 1- CONDITIONAL BASED ON PHY TYPE

Figure 83D–1—Example CAUI-4 chip-to-chip relationship to the ISO/IEC Open System Interconnection (OSI) reference model and the IEEE 802.3 Ethernet model

- Data delay could be separated as two parts:
  - Dynamic delay – which can be estimated (by the MAC chip) and get compensated to minimize time error.
  - Fixed delay – which shall be reported through registers as defined in IEEE 802.3.

Definitions of Timestamp and Reference Plane

- According to IEEE 802.3 Figure 90-3, the timestamp is generated at gRS layer, and after the path data delay is reported to the gRS layer and compensated, the timestamp reference plane would be the MDI.

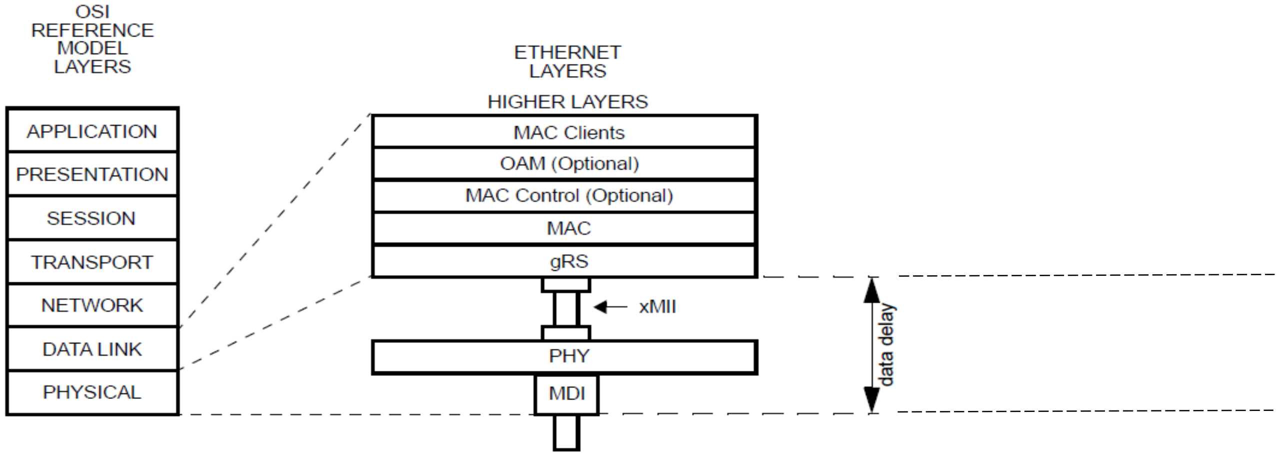
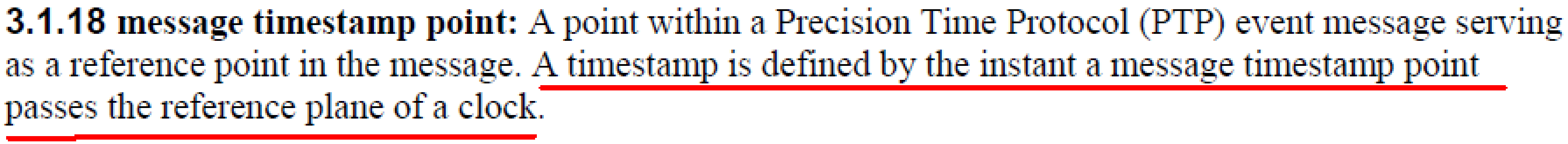


Figure 90–3—Data delay measurement

Timestamp generation point

Timestamp reference plane

- The Clause 3.1.18 of IEEE 1588-2008 provides the definition of timestamp, which should be the time, when a timestamp point (the first symbol after SFD) passes the reference plane (MDI as defined in IEEE 802.3-2018).



3.1.18 message timestamp point: A point within a Precision Time Protocol (PTP) event message serving as a reference point in the message. A timestamp is defined by the instant a message timestamp point passes the reference plane of a clock.

- **Option A/B + Method 1 (tse_3cx_02_0520): Timestamp is the time when the 1st symbol after SFD passes the reference plane (MDI).**

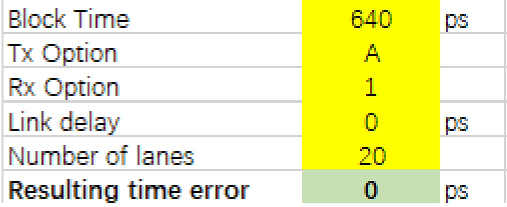
Interoperation Between Different Methods

- tse_3cx_02_0520 lists three options to generate timestamps at Tx:
  - Option A: 66B blocks and timestamps are not aligned at NxPCS lane transmitter
  - Option B: 66B blocks and timestamps are aligned at NxPCS lane transmitter
  - Option C: 66B blocks are aligned but timestamps are not aligned at NxPCS lane transmitter

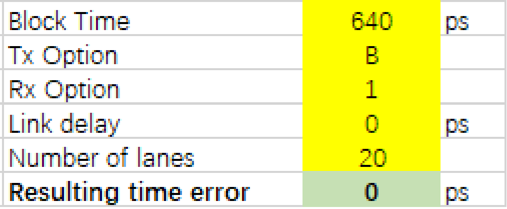
- And two methods for handling multi-PCS lane distribution delay at Rx:
  - Method 1: Account for the delay between the MII and the lane that carries the message timestamp point of the PTP message
  - Method 2: Use a constant delay regardless of which lane carries the message timestamp point, because the Tx+Rx lane distribution delay is a constant for every lane.

- Using the spreadsheet tse_multilane_TE_analysis , 0 time error can be achieved by three approaches.
  - Rx Method 1 (accurate compensation) can work with Tx Option A & B.
  - Rx Method 2 (inaccurate compensation) works with Tx Option C.

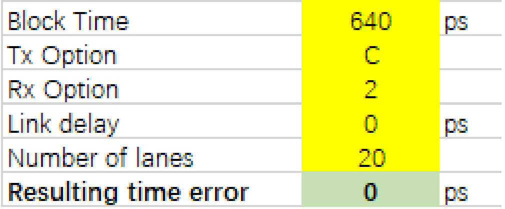
TX option	RX method	Time Error
A	1	0
B	1	0
C	2	0



Block Time	640	ps
Tx Option	A	
Rx Option	1	
Link delay	0	ps
Number of lanes	20	
Resulting time error	**0**	ps



Block Time	640	ps
Tx Option	B	
Rx Option	1	
Link delay	0	ps
Number of lanes	20	
Resulting time error	**0**	ps



Block Time	640	ps
Tx Option	C	
Rx Option	2	
Link delay	0	ps
Number of lanes	20	
Resulting time error	**0**	ps

Multi-PCS Lane Distribution vs FEC Parity Bits

- It was argued that FEC parity bits were handled in a similar way as “Option C + method 2”.
  - FEC parity caused timestamp error is not compensated on either side.
  - Time errors due to parity insertion/deletion on Tx and Rx cancel each other out.
- The method above was introduced when there were only Class A/B applications.
  - RS(544,514) FEC has 300 parity bits, which could introduce a maximum of 2.82 ns timestamp error on a 100GE link. This is trivial compared with the requirements (100ns/70ns).
  - Time error caused by PCS lane distribution is huge compared with Class C/D requirements (30ns/5ns) – and even non-negligible for Class B for 100GE. Extra care shall be taken when choosing the options.

Ethernet Rate	PDDV_max caused by FEC parity bits	Percentage of Class B max|TE|	PDDV_max caused by PCS lane distribution	Percentage of Class D max|TE|
50GE	5.65	8%	3.84	76.8%
100GE(w/o FEC)	0	0%	12.16	243.2%
100GE(w/KP4 FEC)	2.82	4%	12.16	243.2%
200GE	2.82	4%	0.33	6.6%
400GE	1.41	2%	0.35	7%

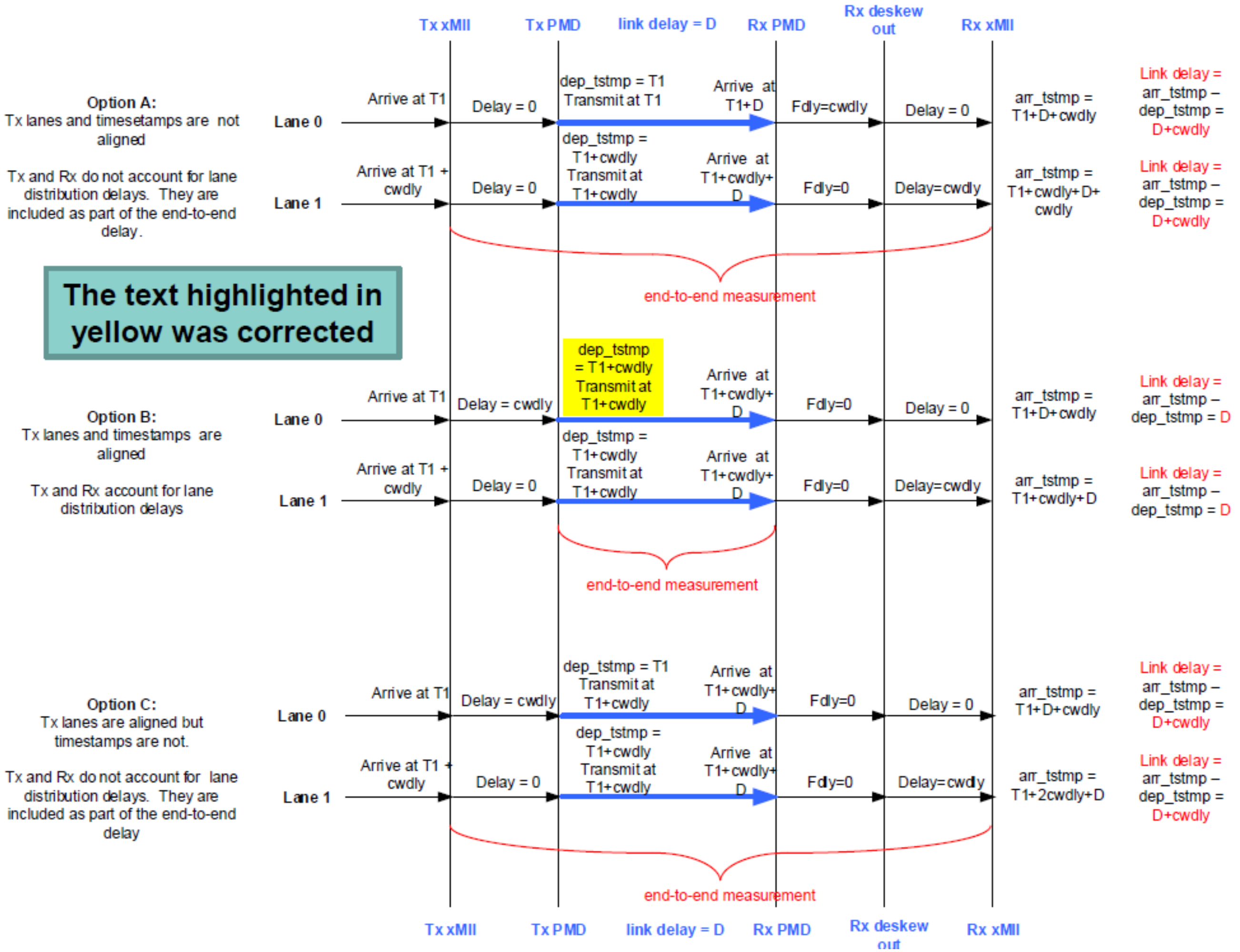
Backward Compatibility?

- It is highly likely huge amount of equipment complying with the current IEEE 802.3-2018 is in service and meets Class C/D requirements when 802.3cx is released.
- How shall we provide backward compatibility if we do not compensate the PCS distribution delay?
  - A register ***(X)*** can be used to let the upper management know how it handles the PCS lane distribution delay, but will **NOT** solve the interop issue – it only broadcasts its own capability and relies on the other end to cooperate.
    - X = 1, PCS lane distribution delay is cancelled by the Rx side;
    - X = 0, 802.3-2018 compliant.
  - Upper layer management could decide how to use this register.
    - Beyond the scope of 802.3cx.

THANK YOU!

Background

- tse_3cx_02_0520 lists three possible solutions to compensate timestamp error caused by multi-PCS lane distribution.



Tx xMII
Tx PMD
link delay = D
Rx PMD
Rx deskew out
Rx xMII
Option A:
Tx lanes and timesetamps are not aligned
Tx and Rx do not account for lane distribution delays. They are included as part of the end-to-end delay.
Lane 0
Lane 1
Arrive at T1
Delay = 0
dep_tstmp = T1 Transmit at T1
Arrive at T1+D
Fdly=cwdly
Delay = 0
arr_tstmp = T1+D+cwdly
Link delay = arr_tstmp – dep_tstmp = D+cwdly
Arrive at T1 + cwdly
Delay = 0
dep_tstmp = T1+cwdly Transmit at T1+cwdly
Arrive at T1+cwdly+D
Fdly=0
Delay=cwdly
arr_tstmp = T1+cwdly+D+cwdly
Link delay = arr_tstmp – dep_tstmp = D+cwdly
end-to-end measurement
Option B:
Tx lanes and timestamps are aligned
Tx and Rx account for lane distribution delays
Lane 0
Lane 1
Arrive at T1
Delay = cwdly
dep_tstmp = T1+cwdly Transmit at T1+cwdly
Arrive at T1+cwdly+D
Fdly=0
Delay = 0
arr_tstmp = T1+D+cwdly
Link delay = arr_tstmp – dep_tstmp = D
Arrive at T1 + cwdly
Delay = 0
dep_tstmp = T1+cwdly Transmit at T1+cwdly
Arrive at T1+cwdly+D
Fdly=0
Delay=cwdly
arr_tstmp = T1+cwdly+D
Link delay = arr_tstmp – dep_tstmp = D
end-to-end measurement
Option C:
Tx lanes are aligned but timestamps are not.
Tx and Rx do not account for lane distribution delays. They are included as part of the end-to-end delay
Lane 0
Lane 1
Arrive at T1
Delay = cwdly
dep_tstmp = T1 Transmit at T1+cwdly
Arrive at T1+cwdly+D
Fdly=0
Delay = 0
arr_tstmp = T1+D+cwdly
Link delay = arr_tstmp – dep_tstmp = D+cwdly
Arrive at T1 + cwdly
Delay = 0
dep_tstmp = T1+cwdly Transmit at T1+cwdly
Arrive at T1+cwdly+D
Fdly=0
Delay=cwdly
arr_tstmp = T1+2cwdly+D
Link delay = arr_tstmp – dep_tstmp = D+cwdly
end-to-end measurement
Tx xMII
Tx PMD
link delay = D
Rx PMD
Rx deskew out
Rx xMII

The text highlighted in yellow was corrected

tse_3cx_02_0520