Considerations for IEEE 802.3 PHY Delays in gPTP

Richard Tse

TSN meeting

Sept 2026

Outline

- Reference presentations
- Conceptual considerations
- Practical considerations
- What can be done?

Reference presentations

- https://www.ieee802.org/1/files/public/docs2026/as-Turner-PHYdelay-dot3-0826-V0.pdf
- https://www.ieee802.org/1/files/public/docs2026/as-Turner-PHYdelay-dot3-0526.pdf

Conceptual considerations 1/2

- IEEE 802.3/802.3cx/802.3de/802.3df use the MDI as the reference plane for timestamping, per IEEE 1588 and IEEE 802.1AS
- IEEE 802.3 PHY datasheet values enable functions outside of a PHY to compensate for the PHY's static delay
  - The PHY delay covers the functions between the gRS and the MDI
  - PHY functions that have mirrored Tx/Rx dynamic delays (e.g., interleaving/deinterleaving, multi-lane distribution/merging) are represented by static delay values, per IEEE 802.3cx
    - Tx side uses the maximum intrinsic delay of the mirrored function
    - Rx side uses the minimum intrinsic delay of the mirrored function
  - Other dynamic functional delays (e.g., idle insertion/removal) are not given by a PHY's datasheet
    - These might automatically be compensated by the PHY or, otherwise, could contribute to timestamping uncertainty
  - Implementation-specific PHY delays must be static since the xMII to xMII delay across a link is constant

Conceptual considerations 2/2

- Other functions (higher layer H/W, S/W, etc) could exist between the reference plane and the timestamp measurement plane, depending on where the timestamp capture is performed in an 802.1AS implementation
  - These delays are not covered by the 802.3 PHY management objects (i.e., delay registers, datasheet values)
  - These delays could be covered by implementation-specific registers/datasheet values or could be automatically compensated for by an implementation
- Delays already compensated by a PHY should not be included in the PHY datasheet for external compensation

Practical considerations 1/4

- Different types of delays in a PHY
  - Physical delays (e.g., wire delays, logic ramp-time delays) are static or pseudo-static
  - Intrinsic functional delays of some standardized functions (e.g., mirrored functions like interleaving/deinterleaving, lane distribution/merging) are static, per IEEE 802.3cx
  - Functional delays of some standardized functions (e.g., idle insertion/removal) are dynamic
  - Implementation-based functional delays (e.g., flop pipeline stages, FIFOs) are (typically) pseudo-static
- The above delays are (typically) given in values that are based on ideal time (e.g., 10.3ns delay = 10.3ns TAI, 3 clock-cycles of delay with a 1GHz clock = 3ns TAI)

Practical considerations 2/4

- Should PHY delays be considered to be part of the link delay?
  - IEEE 802.3 already considers the MDI as its reference plane for timestamping
  - Each endpoint is expected to compensate for its own PHY delays
  - The Tx+Rx PHY delay in one direction is (typically) not the same as the Tx+Rx PHY delay in the other direction
    - These delays are, often, significantly different
      - Different devices with different implementations at each endpoint
      - Different start-up conditions at each endpoint
    - The Tx and Rx delays of the PHY at one endpoint are not known by the other endpoint so the asymmetry cannot be compensated for in gPTP calculations
      - Some dynamic delays (e.g., Idle insertion/removal) cannot be known outside a PHY and, thus, can only be compensated for by the PHY
  - No, PHY delays should not be considered to be part of the link delay

Practical considerations 3/4

- Should PHY delays be multiplied by some rateRatio value?
  - Estimated PHY delay is (typically) based on an ideal clock (e.g., TAI)
  - Actual Tx PHY delay is (typically) based on its line clock
    - This line clock might not be related to the gPTP local clock
  - Actual Rx PHY delay is (typically) based on its recovered line clock, which is the same as the line clock of the Tx PHY at the other end of the link
  - To determine the actual PHY delay in the gPTP time base, each PHY could multiply its own TAI-based delay estimate by the offset ratio between its line clock rate and the GM's clock rate
    - A PHY's line clock is not part of the gPTP specification but could be described as a non-normative concept

Practical considerations 4/4

- What timing accuracy is required for an application?
  - A PHY with a (very large) 10µs TAI-based delay estimate with a line clock running at 100ppm offset has a delay estimation error of 1ns
    - Is this sufficiently accurate for the given application?
  - A PHY with a 100ns TAI-based delay estimate with a line clock running at 100ppm offset has a delay estimation error of 10ps
    - This should be sufficiently accurate for (most/all?) current applications

What can be done?

- Since there are many functional specifications and implementation styles for PHYs, there does not appear to be any universal solution that resolves PHY delay estimation errors
- An informative annex could be added to IEEE 802.1AS:
  - Refer to relevant IEEE 802.3 specifications on PHY delays
  - Explain how PHY delay estimates could add error to timestamps
    - Show the relevant clocks that are present in example PHY implementations
  - Give examples of the magnitude of these errors (PHY delay magnitude, ppm offsets of relevant clocks)
  - Give examples of how these errors could be reduced

Questions?