TSN compatibility of TDD PHY with a focus on gPTP operation

IEEE 802.3dm

July 27, 2025

Guy Hutchison
Aviva Links

Venkat Arunarthi
Broadcom

Supporters

Kamal Dalmia – Aviva Links

Claude Gauthier - NXP

Steve Gorshe – Microchip

Kirsten Matheus - BMW

Mehmet Tazebay - Broadcom

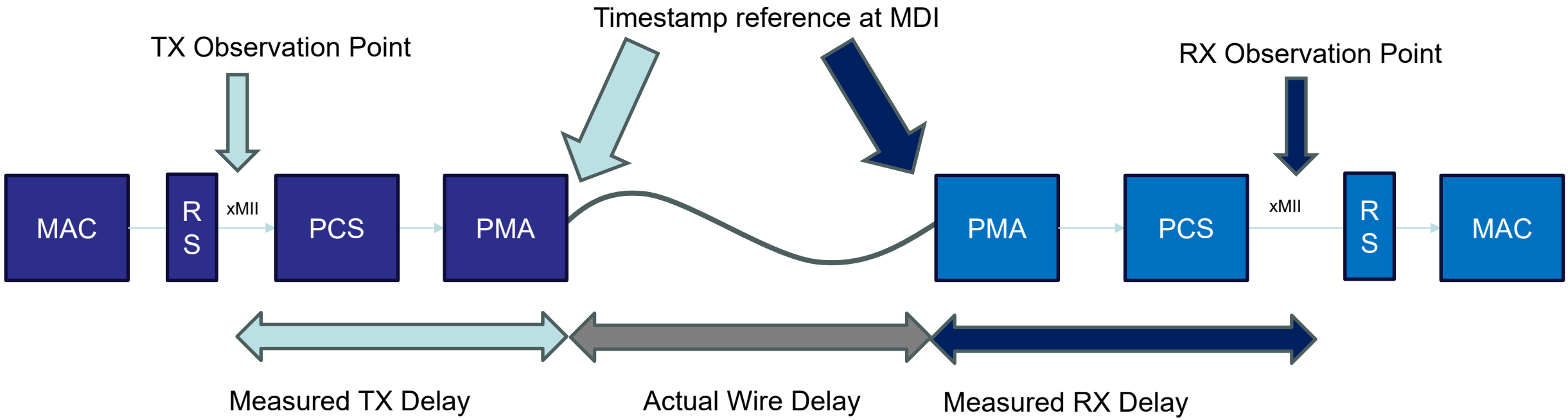
Overview/ Background

- TSN support is an essential element for automotive Ethernet PHYs
- TDD PHYs have quiet gaps that appears to make the operation not as obvious as constant delay PHYs
- Some 802.3dm participants may have the desire to learn how a TDD PHY works in a TSN environment
- This contribution explains how the TDD quiet gaps become redundant in the overall context of TSN and how a TDD based 802.3dm would be fully compatible with upper layers in 802
- This contribution also brings to attention 802.3 Clause 90 that has addressed this topic well in the past.
  - 802.3dj has also raised the same issue with regard to cyclical delays

Requirements to support gPTP

- Support for gPTP requires a local reference clock, and recording the transmission time of selected packets on the local reference clock
- As the actual time of transmission is not observable, practical methods of implementation involve recording the time of passage of selected packets at an *observation point*, past which point the transmission time is predictable.
- Predicting transmission time is straightforward when the delay from the observation point is fixed, in a particular configuration and direction

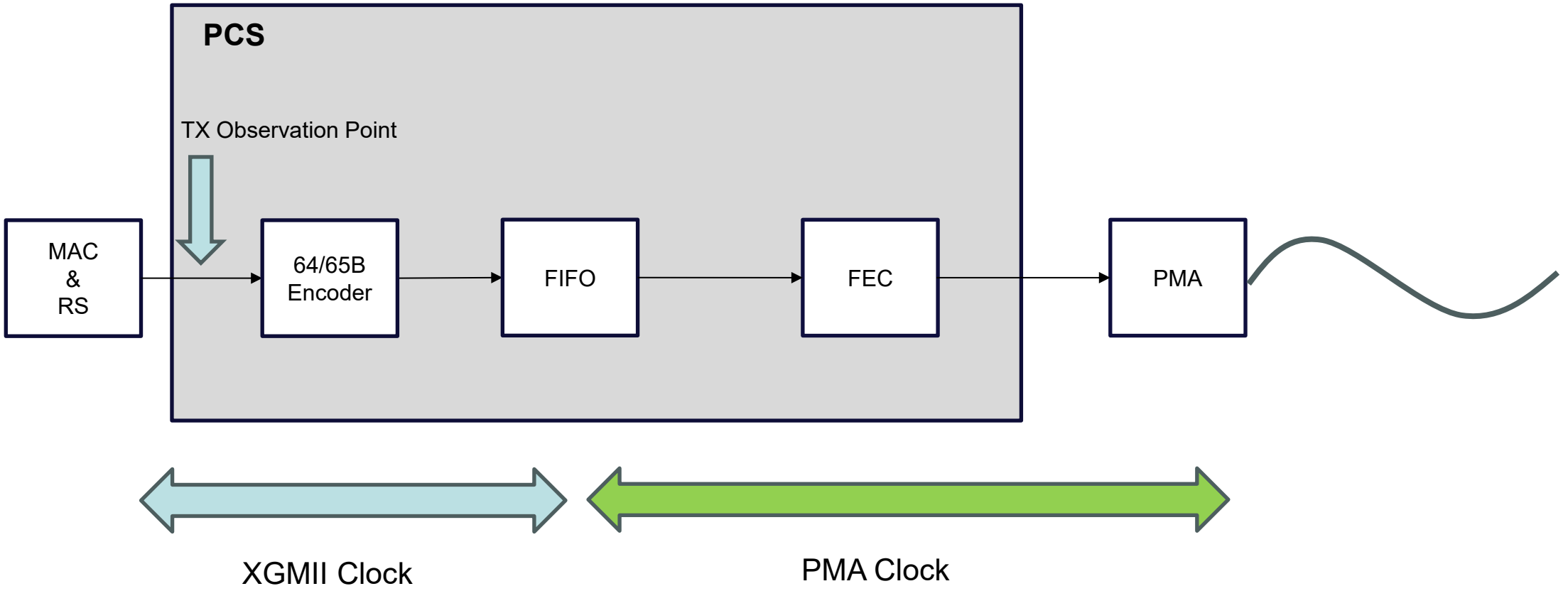
Pictorial View



Timestamp Adjustment

- Transmitter and Receiver Adjust Timestamps
  - Both, transmitter and receiver adjust reported timestamps
  - Transmitter *adds measured TX delay* to observation to create reported timestamp
  - Receiver *subtracts measured RX delay* from observation to create reported timestamp
  - This technique is commonly used in several modern gPTP implementations for very high timestamp accuracy

PCS Details


PCS
TX Observation Point
MAC
&
RS
64/65B
Encoder
FIFO
FEC
PMA
XGMII Clock
PMA Clock

FIFO

- Conceptually needed for :
  - Buffering data during TDD TX Transmission Gaps
  - Buffering PCS codewords (both TDD and ACT) for FEC encoder/decoder

- FIFO cycles from high to low watermark
  - Through each TDD cycle
  - Through each FEC block
  - This cycling exhibits *mirrored cyclical delay*
- High and low watermark fixed by TDD cycle length and size of the FEC block

- TDD presents *constant data rate* over length of TDD cycle

- The FIFO converts data rate from MAC to instantaneous data rate of PCS/PMA

IEEE 802.3 Clause 90

IEEE Std 802.3-2022, IEEE Standard for Ethernet
SECTION SIX

90. Ethernet support for time synchronization protocols

90.1 Introduction

This clause specifies the optional Time Synchronization Service Interface (TSSI). The TSSI can be used to support protocols that require knowledge of packet egress and ingress time.

The TSSI is defined for the full-duplex mode of operation only. It supports MAC operation at various data rates. The MII (Clause 22), GMII (Clause 35), XGMII (Clause 46), 25GMII (Clause 106), XLGMII (Clause 81), CGMII (Clause 81), 50GMII (Clause 132), 200GMII (Clause 117), and 400GMII (Clause 117) specifications are all compatible with the generic Reconciliation Sublayer (gRS) defined in 90.5.

90.2 Overview

The goal of this clause is to provide an accurate indication of the time at which any packet is transmitted or received as required to support various time synchronization protocols, e.g., IEEE Std 1588 [B41], and IEEE Std 802.1AS [B39].

The specific goals are to:

a) Define the Time Synchronization Service Interface (TSSI), with associated service primitives;
b) Add management registers to indicate the maximum and minimum path delays for estimation of link latency at the Time Synchronization Protocol (TimeSync) Client.

- Many existing 802.3 PHYs have variable delay between xMII and MDI
- TSN for variable delay PHYs was addressed several years ago in Clause 90
  - Amended by 802.3cx
- Examples of such Phy Layers are
  - Multi lane PHYs (lane deskew delay is variable). Clause 86, Clause 119
  - LDPC PHYs (decode delay can be variable). Clause 55.
  - Implementations using XGXS (variable delay due to lane sync)

Clause 90.7.1

- Clause 90.7.1 provides explicit instructions as to how to calculate the path data delay:
  - “it is recommended that the transmit and receive path data delays be reported as if the DDMP is at the start of the FEC codeword and/or at the start of the PCS lane distribution sequence”
- Annex 90A.7 generalizes the rule to any PHY function that has mirrored Tx/Rx cyclical delays:
  - “recommended to … allocate the maximum value of the intrinsic delay to the transmit PHY and the minimum value of the intrinsic delay to the receive PHY.”

Source: 802.3cx amendment, dekoos_3dj_01a_2407

Time Measurement with Transmission Gaps

- As shown on previous slides, IEEE 802.3 Clause 90 addresses variable PHY delay.

- TDD transmission gaps are a form of variable PHY delay
  - Delays for blocks with FEC compensate as if the delay was from the start of the FEC block (this applies to ACT as well)
  - Varying delay from this point in transmission are compensated by identical inverse cyclical delay at the receiver
  - A constant value of pipeline delay is used on both ends
  - This method is in use by many existing IEEE PHYs today

- 802.3dm would specifically leverage existing attributes of 90.7
  - Delays measured from first transmission unit of burst, which has maximum delay

Simulations of TDD Delays

Overview

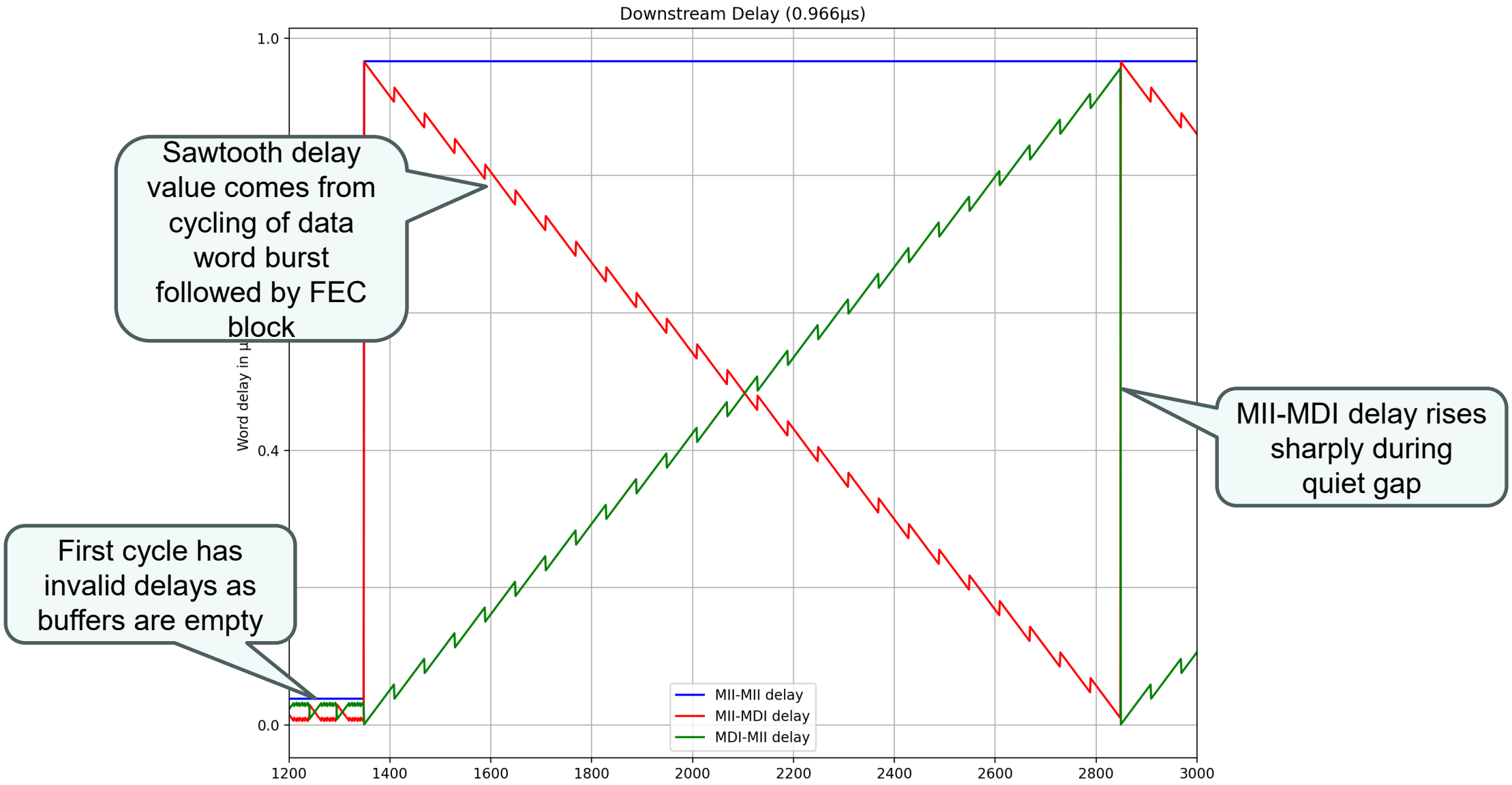
- Simulation of unit delays created using discrete event simulation
  - Built in SimPy environment
  - TDD cycle parameters from Chini_3dm_01a_0125

- Simulation demonstrates constant latency between transmit and receive Reconciliation Sublayers (XGMII to XGMII)

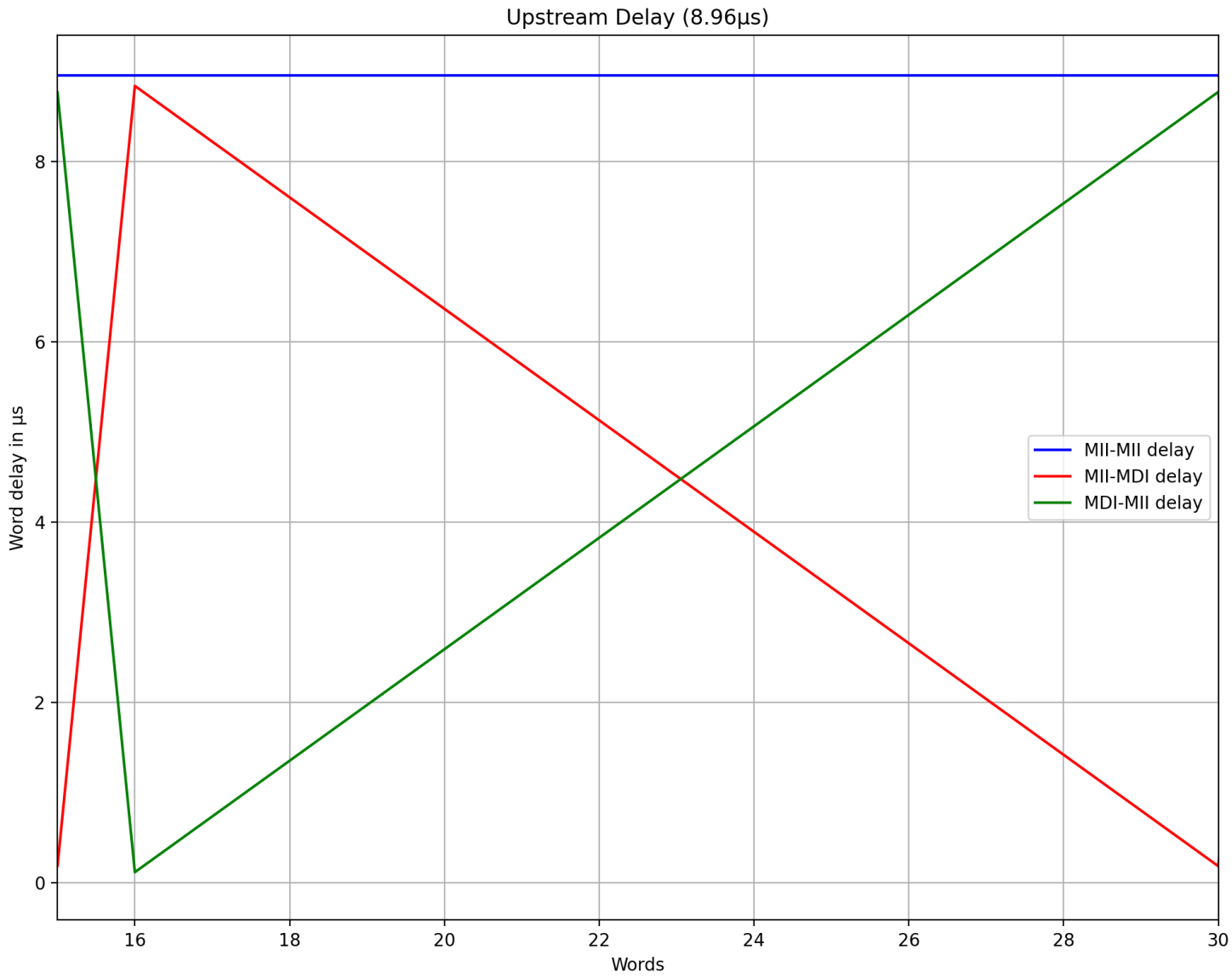
Simulation Methodology

- Simulation unit is a single 65-bit code word
  - Built in SimPy environment
  - Transmission is modelled as a blocking process which takes time equivalent to the time needed to transmit a code word
- Source and Sink MACs produce and consume at constant rate
- TDD PHY transmits blocks on a burst schedule
  - TDD PHY transmits a burst of code words, followed by a gap representing FEC block
  - A sequences of blocks is separated by a longer gap modelling the quiet gap
  - TDD cycle parameters from Chini_3dm_01a_0125
- Simulation demonstrates constant latency between transmit and receive Reconciliation Subsystem (XGMII interface)

Downstream Delay



Upstream Delay



Conclusion

- This contribution explains how the TDD quiet gaps become redundant in the overall context of TSN and how a TDD based 802.3dm would be fully compatible with upper layers in 802
- This contribution also brings to attention that **802.3 Clause 90 that has already addressed this topic well in the past**.
- Many existing PHYs use Clause 90 method to address variable delay.
- TDD PHY leverages this existing capability for seamless compatibility with TSN functions

References

- https://www.ieee802.org/3/dj/public/24_07/dekoos_3dj_01a_2407.pdf

Thank You!

Comments, questions and feedback are welcome.