

July 16th, 2024

Buffer in TDD PHYs

Presenter: Max Turner (max.turner@ieee.org)

Project: IEEE P802.3dm - ISAAC

Event: 2024 July 802 Plenary in Montreal, CAN

Contribution to:  IEEE

IEEE P802.3dm Task Force - ISAAC

ETHERNOVIA

IEEE Std 802.3ch-2020: 44.1.3 Relationship of 10 Gigabit Ethernet to the ISO/OSI reference model

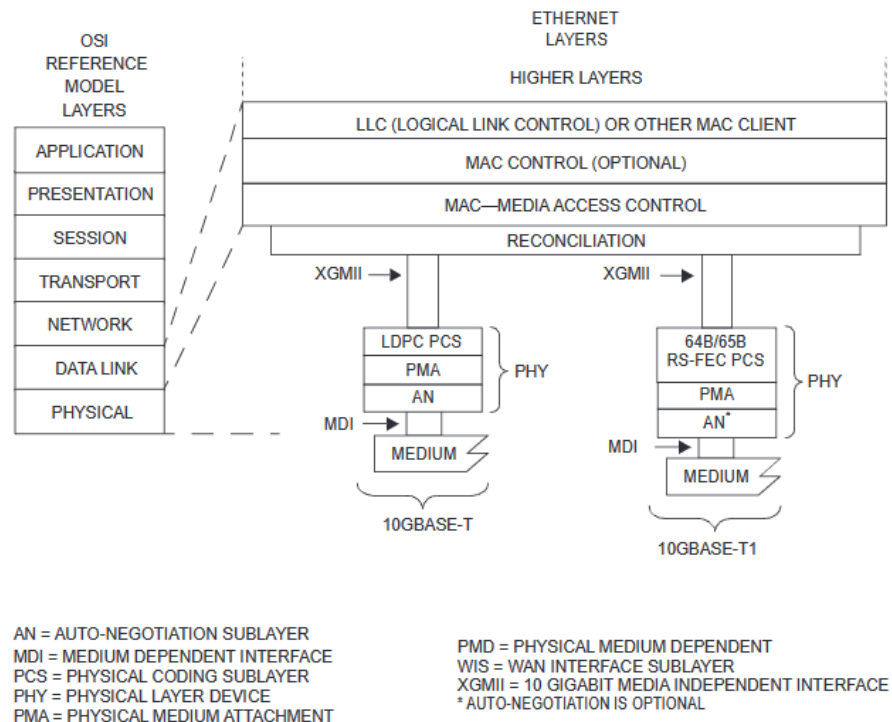
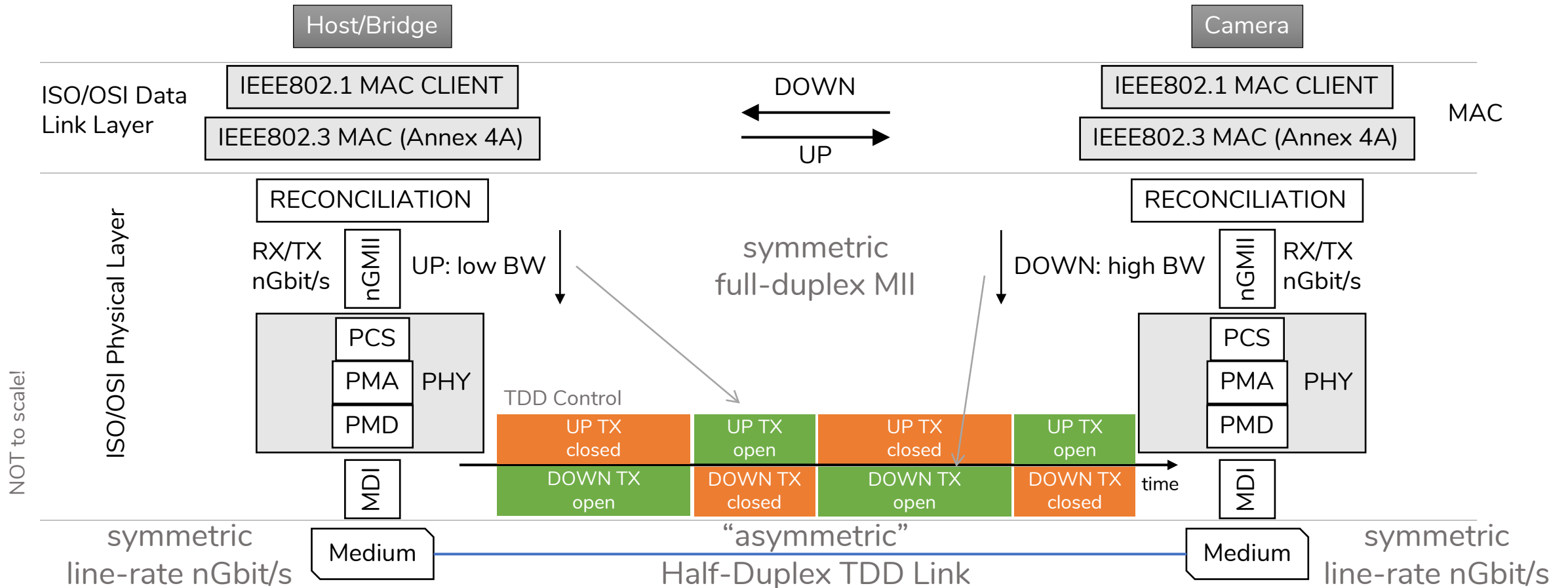


Figure 44-1—Architectural positioning of 10 Gigabit Ethernet

Fundamental challenges in TDD links

- This presentation points out two fundamental issues with time-division access schemes (TDD) to a medium
- The numbers given are examples only, tweaking those numbers will not solve the underlying problem
- The base assumptions are:
 - A link is operated at one specific line-rate in a half-duplex fashion, i.e. the line-rate in both directions is the same
 - The effective bandwidth available in any direction is controlled through the TDD access time only, i.e. one direction can get more time at the line-rate than the other
 - The MAC layer on each side uses a symmetric interface (MII) at line-rate, not at the effective bandwidth

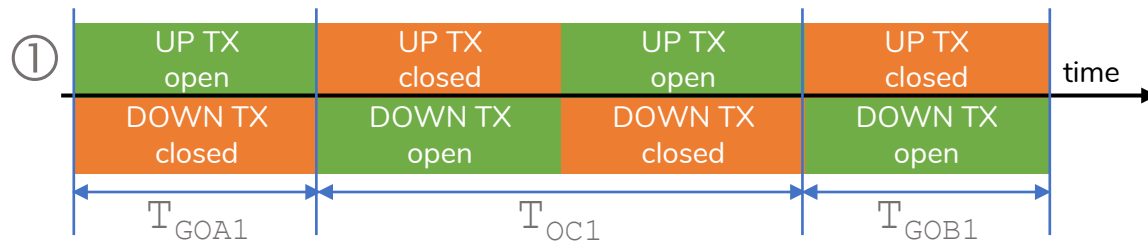
TDD with symmetric line-rate and symmetric MII



NOT to scale!

Time Aware Shaper (TAS)¹⁾: Delay vs. Bandwidth

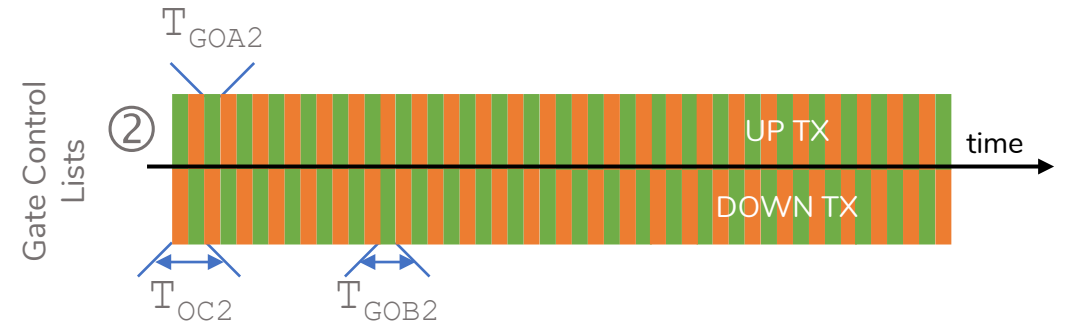
NOT to scale!



$$T_{OC1} = 1s$$

$$T_{GOA1} = 500ms \Rightarrow T_{LA1} \leq 500ms$$

$$T_{GOB1} = 500ms \Rightarrow T_{LB1} \leq 500ms$$



$$T_{OC2} = 100ms$$

$$T_{GOA2} = 50ms \Rightarrow T_{LA2} \leq 50ms$$

$$T_{GOB2} = 50ms \Rightarrow T_{LB2} \leq 50ms$$

Maximum available TAS bandwidth depends on the Ratio T_{GO}/T_{OC} ,
but TAS Delay (T_{LA}) depends on the absolute difference of $T_{OC}-T_{GO}$!

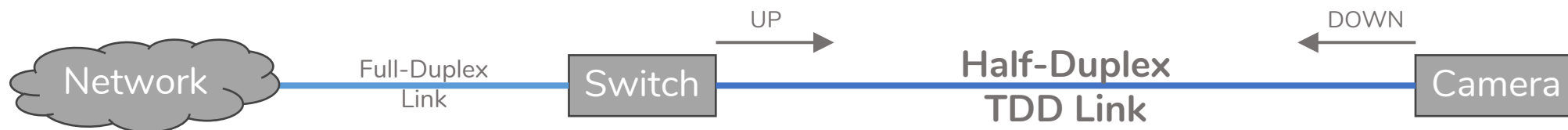
T_{OC} ... OperCycleTime
 T_{GO} ... GateOpenTime
 $T_{OC} = \text{gaps} + \sum T_{GOx}$

¹⁾ IEEE Std 802.1Qbv™-2015 Enhancements for scheduled traffic

A simplified TDD example

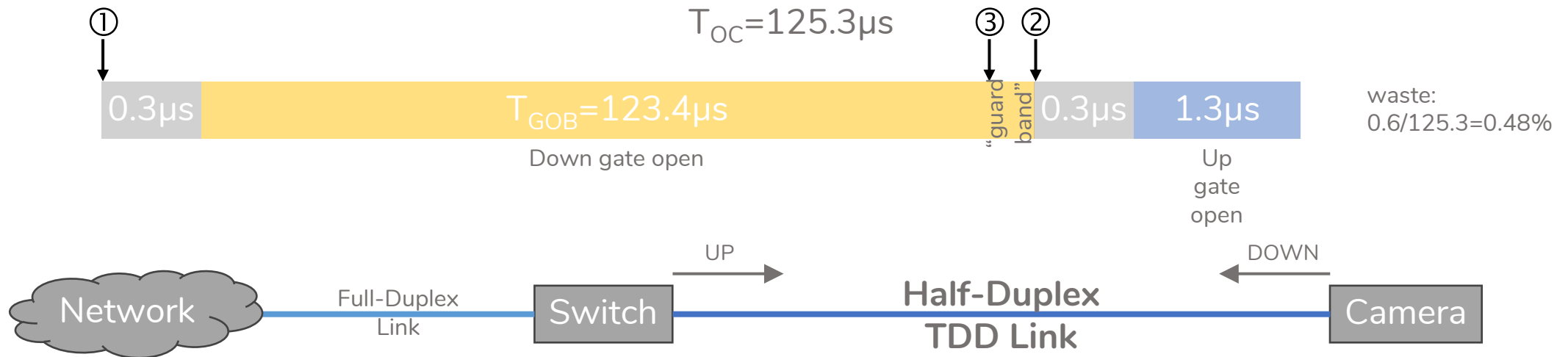
- Run the TDD-Link at 10Gbit/s in both directions.
- We want to allocate a 100× greater BW to the Down direction than to the Up direction.
- In order to not violate Ethernet fundamentals, the 1 time unit Up needs to support at least a 1542Byte Frame; so the Up gate needs to be open for at least $T_{GOA} = 1.2336\mu\text{s}$.
- Therefore: $100 \times 1.2336\mu\text{s} = 123.36\mu\text{s} = T_{GOB}$
- Adding re-sync gaps of $0.3\mu\text{s}$, one can round this to the following simplified TDD cycle:

$$\text{Up: } 123.4\mu\text{s} + \text{gap: } 0.3\mu\text{s} + \text{Down: } 1.3\mu\text{s} + \text{gap: } 0.3\mu\text{s} = 125.3\mu\text{s} = T_{OC}$$



Transmission delays in the simplified example

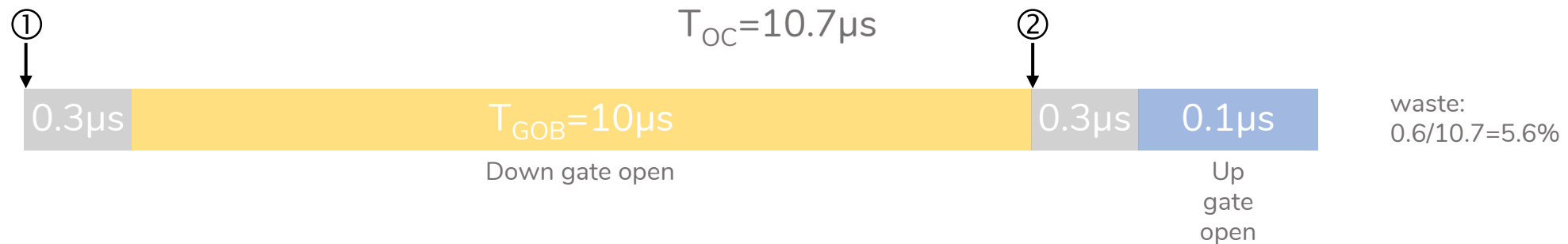
NOT to scale!



- An upstream Frame eligible for transmission on the Switch's TDD Link at instance ① of the TDD cycle must wait for 124 μs
- A downstream Frame eligible for transmission on the Camera's TDD Link at instance ② of the TDD cycle must wait for 1.9 μs
- Instance ③ of the TDD cycle lies the Frame length before instance ②. I.e. the Frame is too long to be transmitted during the gate open time for Down, adding a maximum of 1.23 μs to the delay at ②

Does the Gate have to be open for a full Frame? – No!

- If most upstream Frames are small, one could reduce the gate open time for Up to e.g. $0.1\mu\text{s}$
- Allowing for splitting of a Frame if it is longer than the gate open time
- Also send split Frames in the Down direction, while still keeping the ratio: $100 \times 0.1\mu\text{s} = 10\mu\text{s}$
- Unfortunately, the re-sync gaps will likely not change, therefore the wasted time goes up

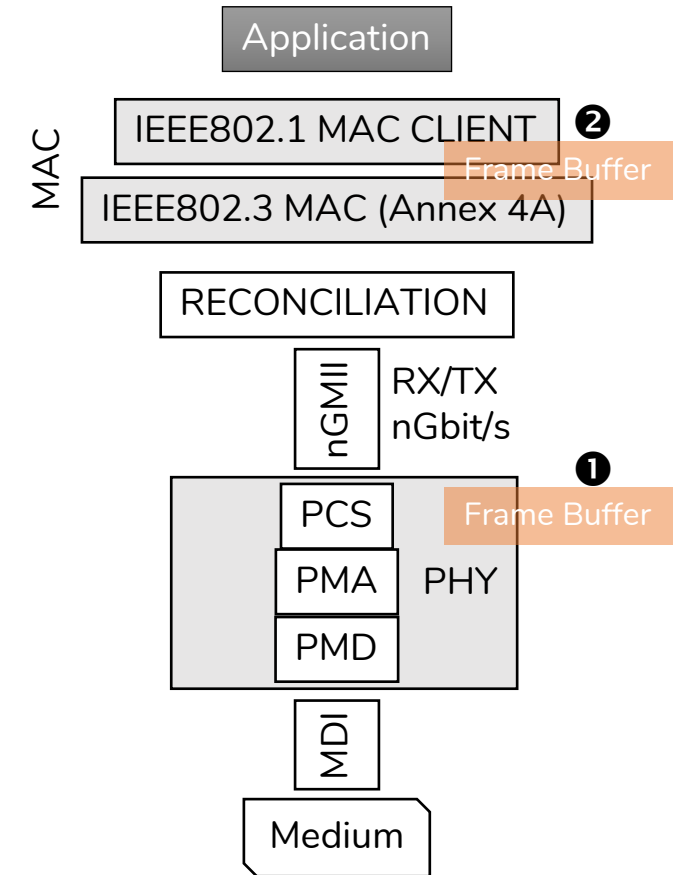


NOT to scale!

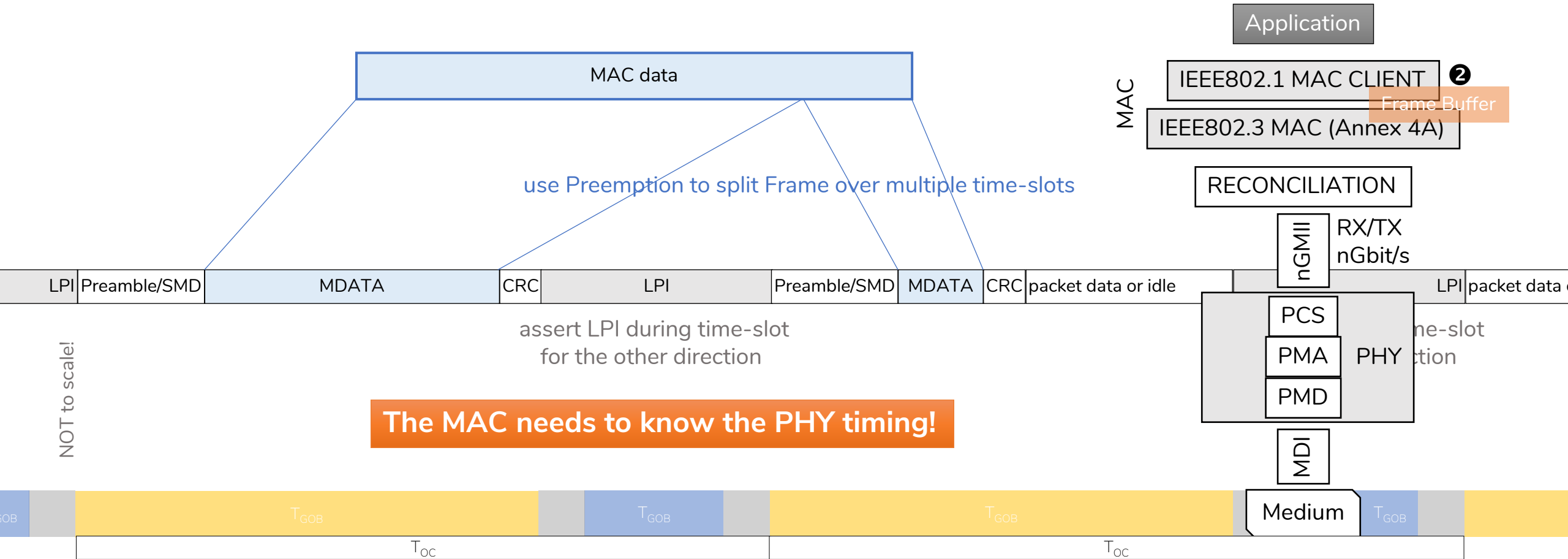
- Now a small upstream Frame eligible for transmission on the Switch's TDD Link at instance ① of the TDD cycle must wait for only $10.6\mu\text{s}$ – roughly a factor of 10 less!
- A downstream Frame eligible for transmission on the Camera's TDD Link at instance ② of the TDD cycle must wait for $0.7\mu\text{s}$ – roughly a factor of 3 less
- The problem of the “guard band” is solved by not having to transmit full Frames, the full open time can be used

Need a Frame Buffer

- If the Medium is not available for transmission of a full Frame, its “fragments” must be buffered somewhere
- **Option 1:** Buffer in the PHY and use CSMA/CA half-duplex signalling from the PHY up to prevent new Frames to be transmitted across the MII
 - The PHY usually does not know about Frames
- **Option 2:** Buffer in the MAC and ...
 - Make sure only full Frames traverse the MII
 - No Idle-Symbols traverse the MII, when the Medium is not available

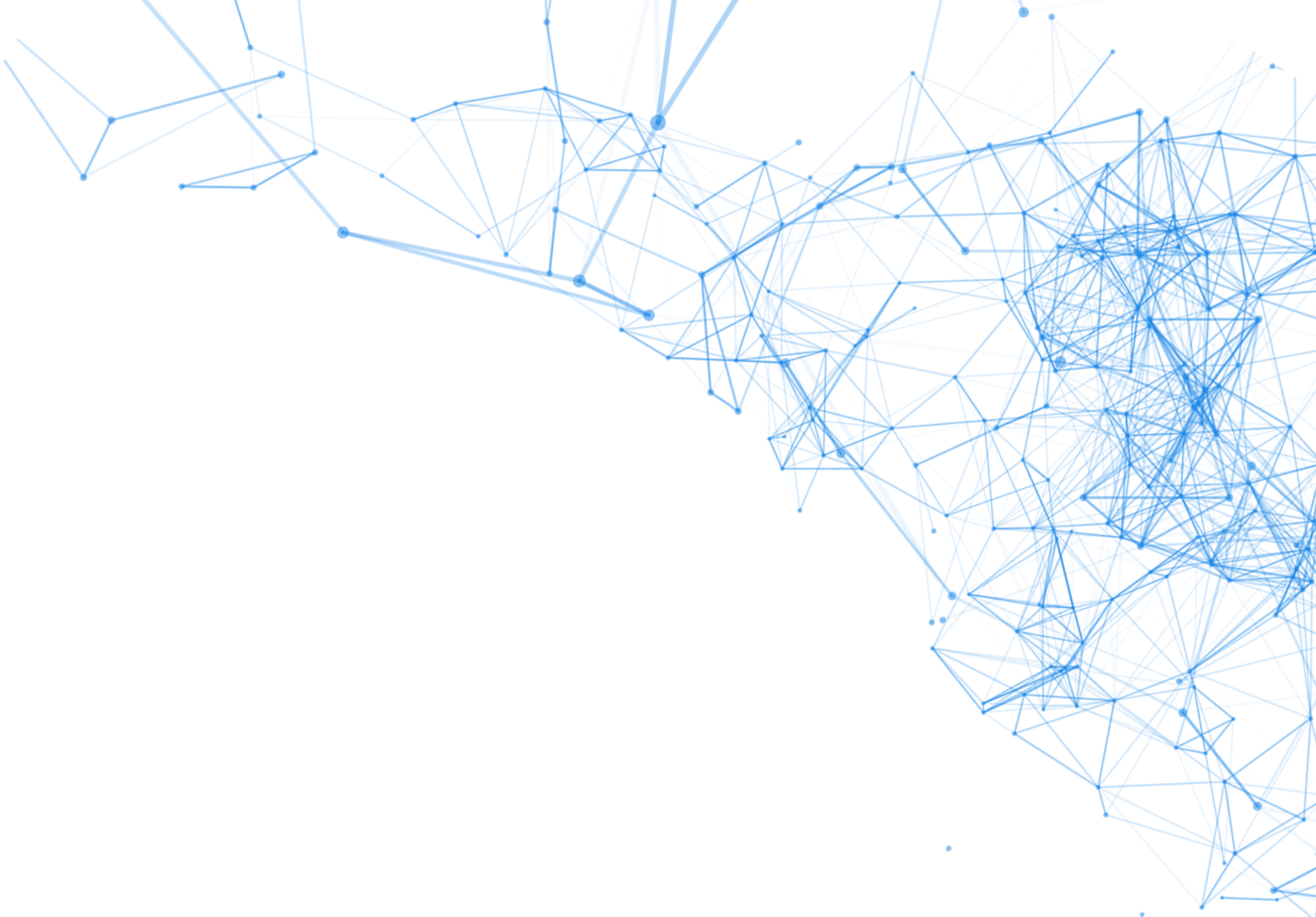


Using Preemption and EEE's LPI



Conclusion


- In TDD access schemes
 - The wait time of the lower effective bandwidth direction increases the lower the effective bandwidth is, if full Frames must be transmitted
 - If Frames are split between available open times, they need to be buffered somewhere and made available at the right time according to the TDD cycle
- Full-Duplex operation would be less complex
 - No wait times to access the half-duplex TDD controlled Medium
 - No Buffering of (split) Frames
 - No coordination effort with the TDD cycle



Max Turner

Utrechtseweg 75
NL-3702AA Zeist
The Netherlands
+49 177 863 7804

max.turner@ethernovia.com

Contribution to:  **IEEE**

IEEE P802.3dm Task Force - ISAAC



ETHERNOVIA

Thank You
for your attention!

Contribution to:  **IEEE**

IEEE P802.3dm Task Force - ISAAC