P802.1ASds – Solution Implications

Presenter: Don Pannell, NXP

Co-Authors:
Olaf Mater, Marvell
Silvana Rodrigues, Huawei – as an Avnu Subject Matter Expert
Woojung Huh, Microchip
Vlad Lyalikov, Microchip

March 2023 802.1 Plenary

AVNU®
RELIABLE SYNCHRONIZED NETWORKING

Administration

This presentation is the collaborative work of the Presenter and Co-Authors that was performed in Avnu Automotive Work Group meetings.

This is not an official publication from Avnu.org, but permission was granted to use the Avnu template to show multi-company consensus on the items enclosed.

Consider this a starting point for discussing the needed gPTP mechanisms to support 10BASE-T1S half-duplex links.

OVERVIEW

- Summary of past proposals and what we know so far
- Initial evaluations of latest two proposals
- A new Proposal to simplify the ClockID filtering database
- Common Impacts to 802.1AS due to ASds Regardless of the accepted solution
- Future work and summary

SUMMARY OF PAST PROPOSALS AND WHAT WE KNOW SO FAR

Summary of proposed solutions to date:

[1] Rentschler, proposed a new Pdelay_Req_Follow_Up message:

- https://www.ieee802.org/1/files/public/docs2020/dg-rentschler-802-1as-MD-multidrop-0920-v01.pdf

[2] Janker et al., proposed modifications to MDPdelayReq state machine:

- https://www.ieee802.org/1/files/public/docs2021/dg-janker-timesync-in-10BASE-T1S-networks-0521.pdf

[3] Pannell et al., presented a simplified 10BASE-T1S Use-Case model:

- https://www.ieee802.org/1/files/public/docs2022/ds-pannell-Avnu-Automotive-UseCase-Requirements-0922-v01.pdf

[4] Rodrigues et al., presented a shared media filtering mechanism using gPTP ClockID:

- https://www.ieee802.org/1/files/public/docs2022/ds-Rodrigues-Lv-10BASE-T1S-time-sync-1122-v00.pdf

Pro's & Con's of [1] Michael Rentschler (Microchip)

Proposal:

- Only timeTransmitter (Master) ports initiate Pdelay measurements
- Introduces a new Pdelay_Req_Follow_Up message to provide t1 and t4 to the timeReceiver (Slave) ports

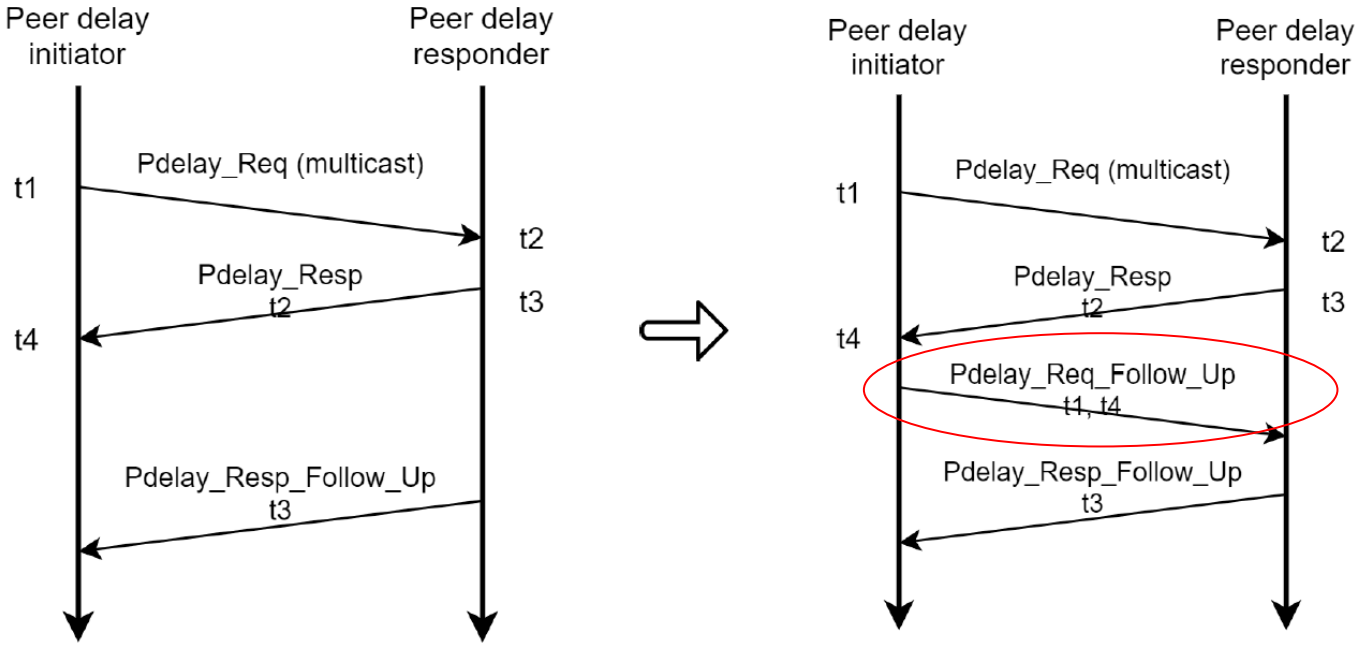
- Pro's:
  - Solves the issue of multiple responses of a sent Pdelay_Req message by changes in the state machines
  - Appears to saves wire bandwidth (4 frames per Pdelay sequence vs. 6)
- Con's:
  - Requires a new PTP message in both IEEE 802.1 and IEEE 1588
  - Significant changes in MDPdelayReq & MDPdelayResp state machines
  - timeReceivers get multiple Pdelay_Req_Follow_Up messages (filtering still needed)

This proposal needs more work & appears to break compatibility w/1588

- We propose to end consideration on this proposal until more contributions are submitted

Summary of [1] Michael Rentschler (Microchip)

- **Introduce Pdelay_Req_Follow_Up message**



MICROCHIP

Pro’s & Con’s of [2] Georg Janker, et. al

- Proprosal:
  - Change MDPdelayReq state machine to ignore unexpected Pdelay_Resp messages
  - Enable/disable nodes to respond to Pdelay_Req messages
- Pro‘s:
  - Solves the issue of multiple responses for a sent Pdelay_Req message by changes in the state machines
  - No filtering or shim layer needed
- Con‘s:
  - Changes in MDPdelayReq state machine are needed
  - Doesn‘t support multiple Domains nor BMCA
  - Problem with overlapped Pdelay_Req messages from multiple nodes

Proposal needs more work & doesn‘t support the use case model defined in [3]

- We propose to end consideration on this proposal until more contributions are submitted

Summary of [2] Georg Janker, et. al

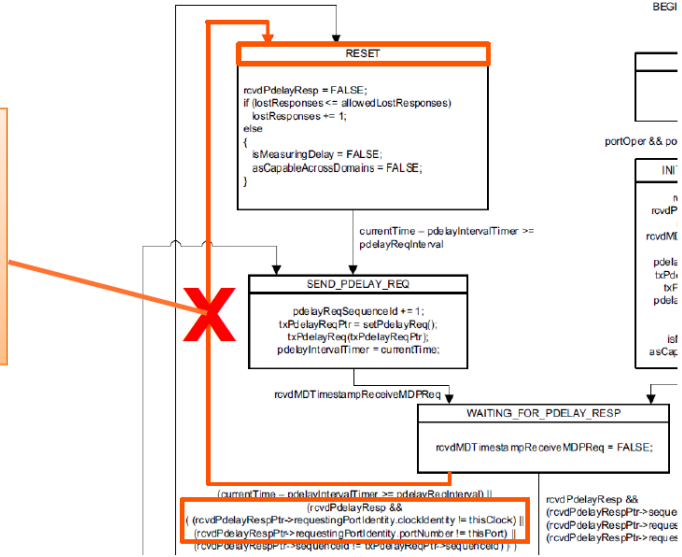
RUETZ SYSTEM SOLUTIONS

Proposal

Modification of the MDPdelayReq state machine

Proposal:
Modify the MDPdelayReq state machine so that Pdelay_Resp messages with deviating requestingPortIdentity are **ignored** instead of triggering a RESET of the state machine.

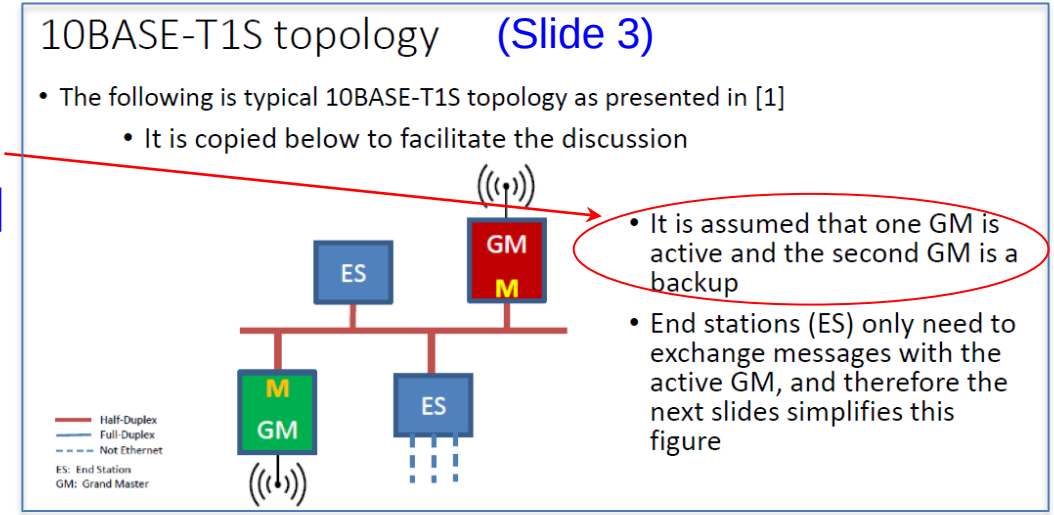
Instead, only Pdelay_Resp messages with a matching requestingPortIdentity, but with deviating sequenceId should trigger a RESET.

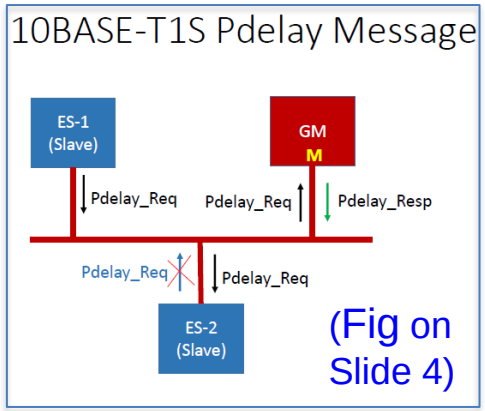


Alignment with Silvana's Nov 22 presentation

Silvana's Slide 3 & 4 [4]:

- Red oval indicates a concern as Hot Standby was assumed in [1] with the two GMs





Confirmed with Silvana (Jan '23) that her Slide 4 would then be per Domain in Hot Standby case

- Means ASds needs to filter on Domain as well as ClockID

Summary of What we Know

1) Due to the shared media of 10BASE-T1S a frame filtering mechanism for Pdelay is required. Options are:
   a) Filter on DA MAC address, i.e., using Unicast Pdelay messages [3]
   b) Filter on some fields in the gPTP frames [4]

2) Any filtering mechanism needs to know what to compare against (i.e., a database of acceptable DA MACs or ClockIDs). Options are:
   a) It knows, i.e., its configured at boot-up (works for engineered networks)
   b) Auto learned / informed (required for plug-and-play networks)

   Note: Solutions with a small database are preferred

Summary of What we Know (2)

1) Filtering has two components to consider:
   a) Performing filtering on frame reception (called receive filtering here)
   b) Creating the correct frame content on frame transmission (called transmit filtering here)

2) The database has two components as well:
   a) Database creation
   b) Database access

INITIAL EVALUATIONS OF PROPOSALS [3] VS. [4]

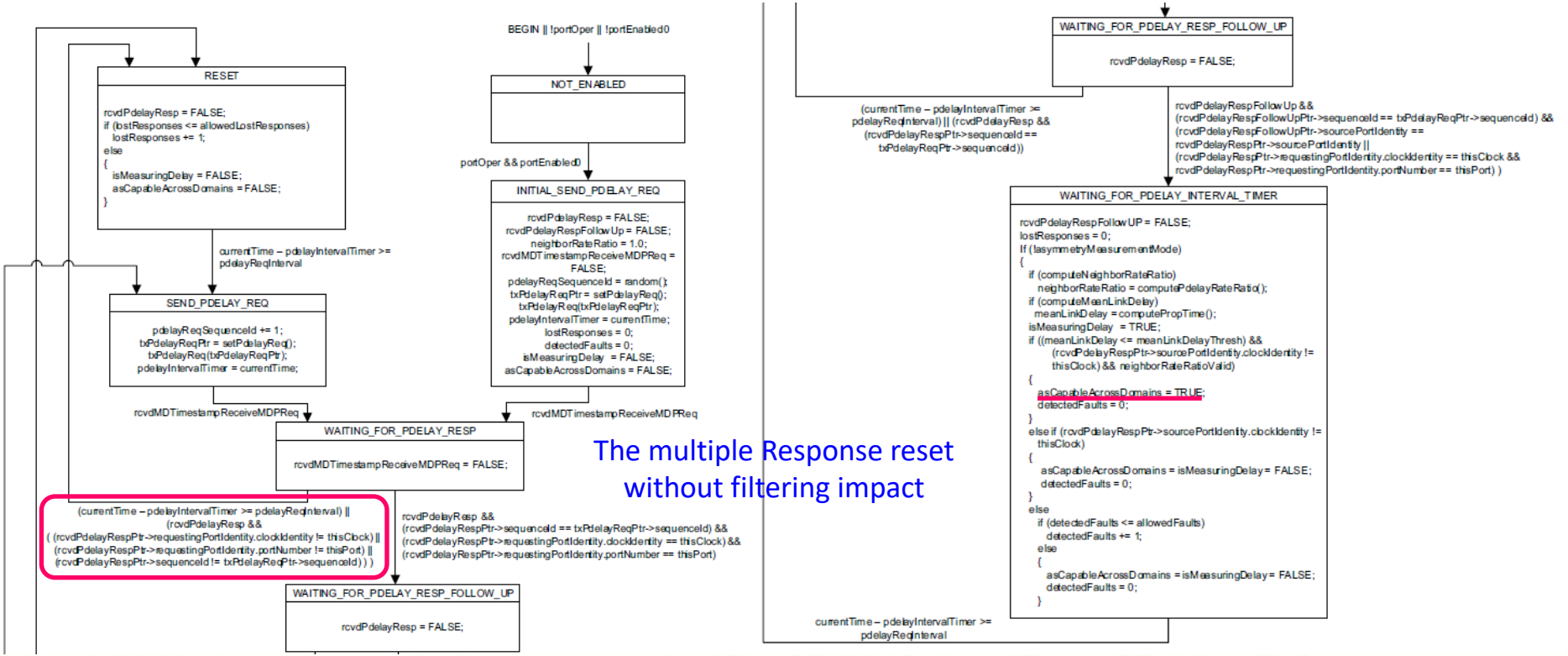
Pro's and Con's of the Filtering Proposals

- Bits / Bandwidth "on the wire"
  - Frame sizes are identical – so both are equivalent in this area (for non-PnP)
- CPU overhead for gPTP on Receive Filtering
  - Unicast filtering is a standard hardware feature of all End Station NIC / Bridges
    - The End Station's Unicast address is assumed to be there, so no additional resources are needed
    - DA address filtering only forwards appropriate frames saving unnecessary interrupts and processing
    - In IEEE 1588, what defines gPTP is the frame's Ethertype (or UDP Dest Port) not the frame's DA
  - Domain+ClockID filtering would be in the gPTP stack
    - If the gPTP stack is in software (virtually all are) this increases the CPU overhead for every Pdelay
    - Pdelay exchanges at once per second per node for the expected 8 nodes isn't that much overhead
    - But if the node count or the Pdelay rate increases (as in Industrial) this could start being a problem
  - At once per second Pdelay with 8 nodes there is a slight Pro for Unicast filtering

Pro's and Con's of the Filtering Proposals

- gPTP Receive Filtering State Machine Changes
  - Unicast filtering does not require any Pdelay state machine changes
    - As the gPTP stack only gets the Pdelay frames directed to the target End Station
    - The problem of dropping asCapable on receipt of multiple pDelayResp's can stay in the state machine, as this won't happen
    - In fact, leaving the test in the state machine indicates a misconfiguration problem, so it is best left in
    - Every End Station / Bridge supports "is this frame for me" unicast filtering in H/W, a must for shared media (standard Bridge mapping prevents 10BASE-T1S local unicast frames from escaping the Bridge)
  - Domain+ClockID filtering requires Pdelay state machine changes
    - Changing the existing MDPdelayReq state machines in Clause 11 is risky and not clean
    - A low-risk proposal is to insert a "shim" state machine between the LLC & MD layers that only forwards the desired frames to MDPdelayReq – i.e., this "shim" does the appropriate filtering.
    - With the "shim" in the right place it appears the Pdelay state machines can be unchanged – as in the Unicast filter case
  - Assuming ClockID filtering is done in S/W gives a Pro for Unicast filtering

Clause 11 MDPdelayReq State Machine
Impact areas if a Shim Receive Filter is not used



The multiple Response reset without filtering impact

Receive Filtering Locations

AS Standard

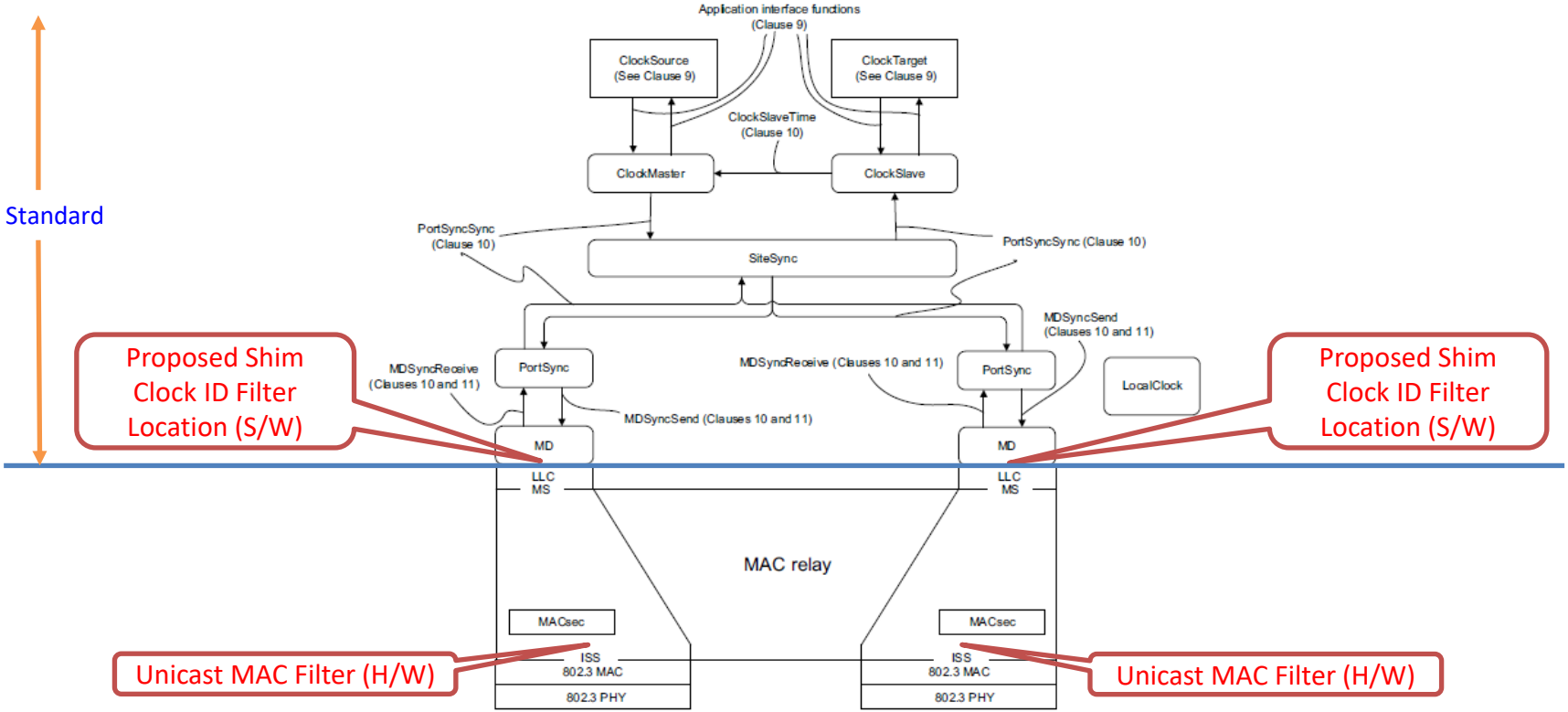


Figure 11-3—Model for a PTP Instance of a time-aware system with full-duplex point-to-point links

A NEW PROPOSAL TO SIMPLIFY THE CLOCK_ID FILTERING DATABASE

Unidirectional pDelay_Req

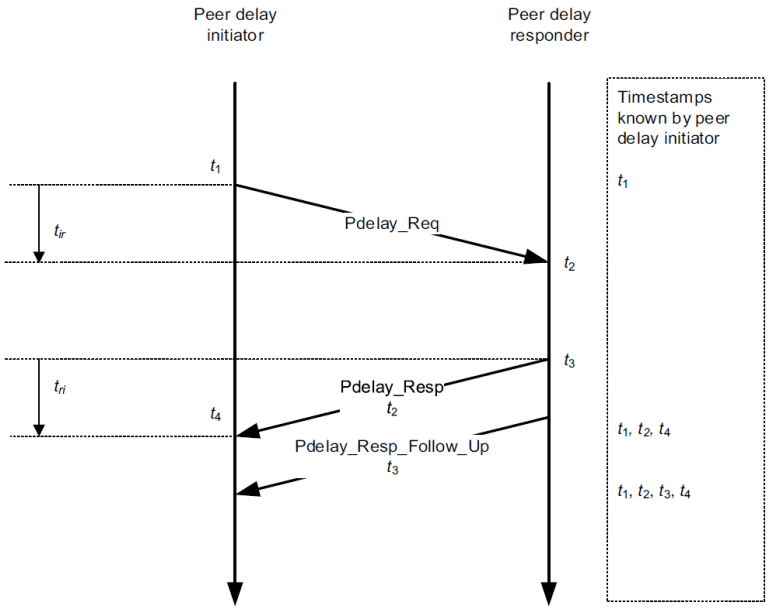
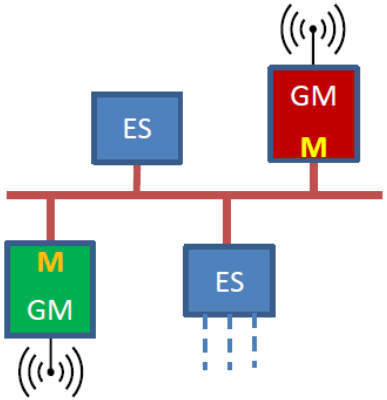


Figure 11-1—Propagation delay measurement using peer-to-peer delay mechanism

$$t_{ir} = t_2 - t_1$$

$$t_{ri} = t_4 - t_3$$

$$D = \frac{t_{ir} + t_{ri}}{2} = \frac{(t_4 - t_1) - (t_3 - t_2)}{2}$$



- In IEEE 802.1 AS, either side of the link can be the initiator and the responder
  - The initiator has all the timestamps necessary to calculate the mean link delay (D) relative to the GM timebase
- For 10BASE-T1S, we can restrict the initiator and only allow the End Station to send Pdelay_Req
  - This will avoid the need for the master to keep a database of all End Stations.

COMMON IMPACTS TO 802.1AS DUE TO ASds REGARDLESS OF THE ACCEPTED SOLUTION

Common Impacts to AS-2020 due to ASds

- “half-duplex” is not in AS-2020 but “halfduplex” (typo):
  - Is used once in 16.2, the Coordinated Shared Network clause & it is a typo nobody caught. It shows up in the document as “half-duplex” (the desired term) but can’t be found that way as the hyphen is due to a line break
- “fullduplex” (typo):
  - Is used once in 10.7.2.1, the Media Independent clause & it is a typo nobody caught. It shows up in the document as “full-duplex” (the desired term) but can’t be found that way as the hyphen is due to a line break
- “full-duplex”:
  - Is used 98 times: Clause 5.5 x 2, 7.2 x 1, 7.3.2 x 3, 7.3.4 x 4, 7.5 x 3, 10 x 8, 11 x 20, 12 x 4, 14 x 17, 15 x 22, 16 x 2, A x 5, F x 3, rest in ToC
  - All of these appear to need a new term to support ASds

Common Impacts to AS-2020 due to ASds

- Does the “full-duplex” term’s appearance 98 times mean a new “half-duplex” Media Dependent clause is needed?
  - All of the 98 appearances would still need to be examined to see which remain
  - At first look, many would still need to be updated
  - Changing the terms is a low-risk change as no state machines need to be changed
  - Creating a new Media Dependent clause could be 100% the same as the current Clause 11, with the added ClockID filter shim (if that approach is used)
  - Having so much identical text twice in the STD! is a high-risk maintenance, document reviewing, & reader problem (what’s different and where?)
  - Low-risk vs. high-risk gives a clear Pro for Changing the terms & not a new Clause

FUTURE WORK AND SUMMARY

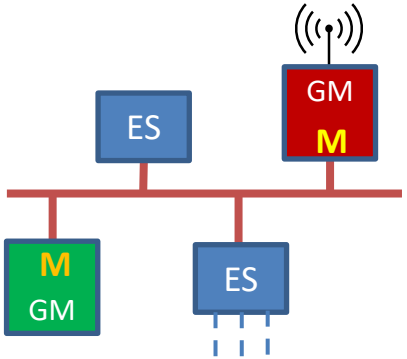
Future Work

- This has been an initial comparison between [3] Unicast & [4] Domain+Clock ID filtering
  - It covers the Receive Filtering function only
- The comparison between [3] & [4] needs to be completed
  - To compare Transmit frame creation
  - To compare Database creation
  - To compare Database access
- Unidirectional pDelayReq: Are there any side effects of this approach and is this an acceptable solution to 802.1AS?
  - 1588 does not require Bi-directional pDelayReq
- Handling of Signaling Messages on 802.3 Clause 4 half-duplex media
- Table 11-1 meanLinkDelayThresh needs to be updated

Summary of Pro's & Con's: Unicast vs. Clock ID

	Unicast - Clock ID				
Bits / Bandwidth "on the wire"					
CPU overhead for gPTP Receive Filtering					
gPTP Receive Filtering State Machine Changes					
gPTP Transmit Filtering State Machine Changes	?	?	?	?	?
gPTP Filtering Database Access	?	?	?	?	?
gPTP Filtering Database Creation	?	?	?	?	?
Others?	?	?	?	?	?

SUMMARY



- Use case model from [3] is still valid:
- Drop consideration of [1] & [2] until more contributions are submitted
- Assume a new Media Dependent Clause is not needed
- Known changes needed in Clause 11 so far:
  - Change “full-duplex IEEE 802.3 links” or “full-duplex point-to-point links” to “IEEE 802.3 Clause 4 MAC links”
- More comparison analysis on [3] vs. [4] is still needed

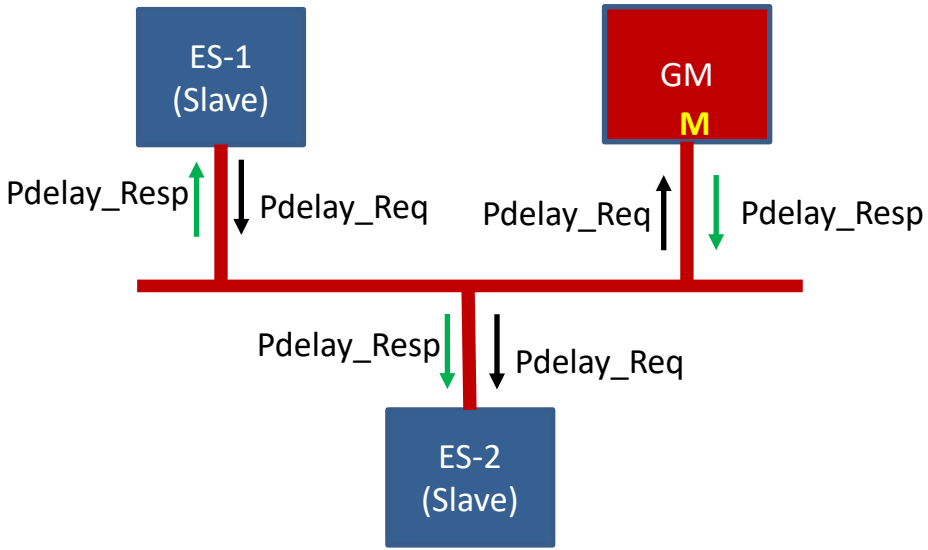

QUESTIONS?
COMMENTS?
CONCERNS?


Thanks!

BACKUP SLIDES

For unidirectional pDelayReq messages originating only from timeReceiverPorts to the timeTransmitterPort:

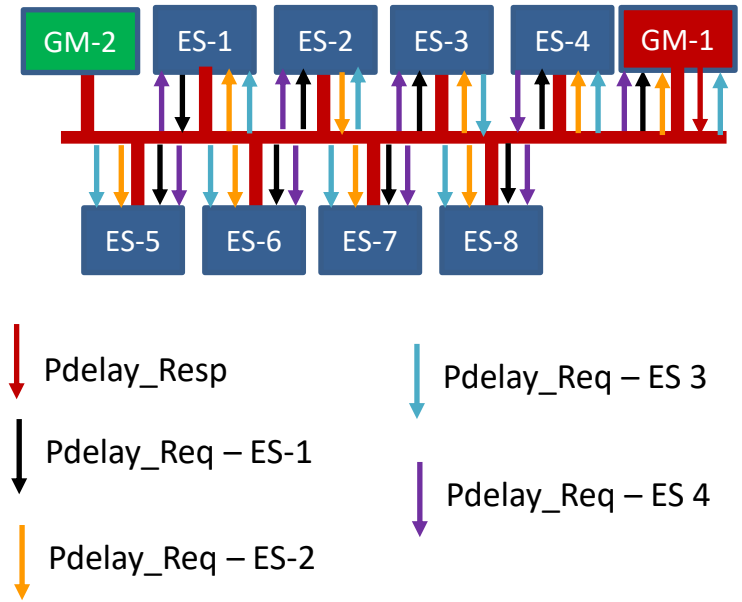
10BASE-T1S Pdelay Messages using clockID



Note that if Announce messages are not used, then sync messages can still be used to identity the GM clockID, even though sourcePortIdentity field of the PTP common header identifies the upstream master port, and in this case it is the GM.

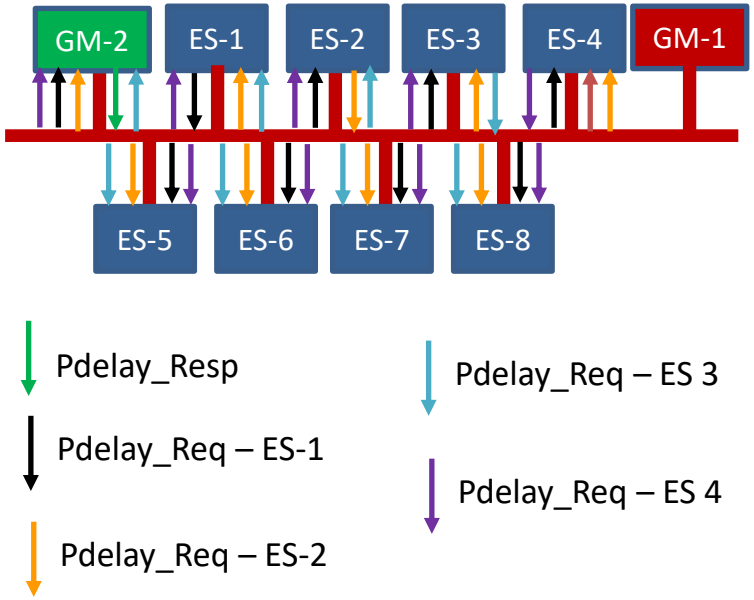
- GM sends Announce and sync messages to ES-1 and ES-2
  - ES-1 and ES-2 learns the GM clockID
- ES-1 sends Pdelay_Req
- ES-2 received Pdelay_Req from ES-1 and checks the clockID. The clockID does not match the GM clocID, and therefore ES-2 does not reply Pdelay_Resp due to the Pdelay_Req message
- GM receives the Pdealy_Req and replies with Pdelay_Resp and set the requestingPortIdentity to the sourcePortIdentity field of the corresponding Pdelay_Req message from ES-1
- ES-2 receives Pdelay_Resp and it does not act on it, as the requestingPortIdentity field does not correspond to its PortIdentity
- Finally ES-2 ignores Pdelay messages associated with ES-1
- ES-2 can also exchange messages with the GM using the same principle

10BASE-T1S Pdelay Messages using clockID



- GM-1 sends Announce and sync messages to all End Stations on its domain
  - ES-1 and ES-2 learns the GM clockID
- ES-1 sends Pdelay_Req
- ES-2-ES-8 receive Pdelay_Req from ES-1 and checks the clockID. The clockID does not match the GM clocID, and therefore ES-2-ES-8 do not reply Pdelay_Resp due to the Pdelay_Req message
- GM-1 receives the Pdealy_Req and replies with Pdelay_Resp and set the requestingPortIdentity to the sourcePortIdentity field of the corresponding Pdelay_Req message from ES-1
- ES-2-ES-8 receive Pdelay_Resp and it does not act on it, as the requestingPortIdentity field does not correspond to its PortIdentity
- Finally ES-2 ignores Pdelay messages associated with ES-1-ES-8
- ES-2-ES-8 can also exchange messages with the GMs using the same principle

10BASE-T1S Pdelay Messages using clockID



- GM-2 sends Announce and sync messages to all End Stations on its domain
  - ES-1 and ES-2 learns the GM clockID
- ES-1 sends Pdelay_Req
- ES-2-ES-8 receive Pdelay_Req from ES-1 and checks the clockID. The clockID does not match the GM clocID, and therefore ES-2-ES-8 do not reply Pdelay_Resp due to the Pdelay_Req message
- GM-1 receives the Pdealy_Req and replies with Pdelay_Resp and set the requestingPortIdentity to the sourcePortIdentity field of the corresponding Pdelay_Req message from ES-1
- ES-2-ES-8 receive Pdelay_Resp and it does not act on it, as the requestingPortIdentity field does not correspond to its PortIdentity
- Finally ES-2 ignores Pdelay messages associated with ES-1-ES-8
- ES-2-ES-8 can also exchange messages with the GMs using the same principle