802.1as MD layer specification for multidrop busses

A Leading Provider of Smart, Connected and Secure Embedded Control Solutions

Michael Rentschler

Sep 22, 2020

802.1as MD layer specification for multidrop busses

Michael Rentschler

MICROCHIP

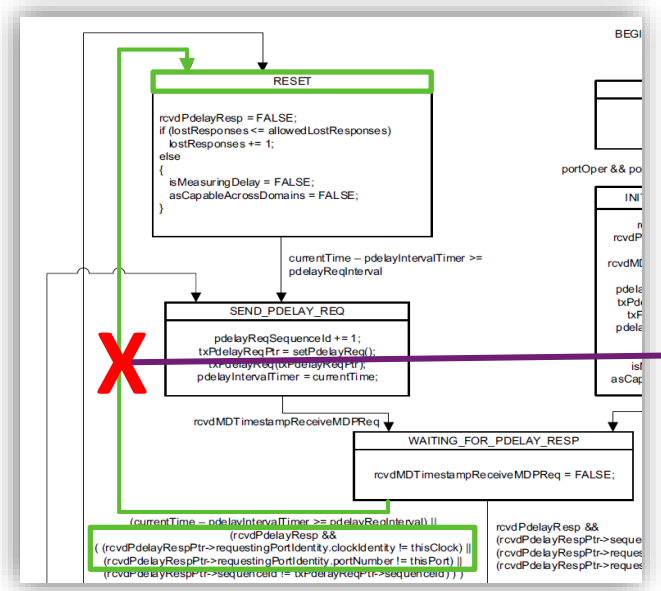
Problem

- **802.1as-2020 (gPTP) defines media-dependent layer specifications for…**
  - Full-duplex point-to-point links
  - IEEE 802.11 links (WLAN)
  - IEEE 802.3 Ethernet passive optical network links (EPON)
  - Coordinated Shared Network links (CSN)
  - Multimedia over Coax links (MoCA)

- **Problem: 802.1as lacks the specification of a media-dependent layer for multidrop busses (i.e. 10Base-T1S/10SPE)**
  - Pdelay peer-to-peer delay mechanism fails
  - 802.1as Signaling messages may not work as intended

Consequences As Of Today

- **If Pdelay is used on a multidrop bus (as defined by 802.1as), that implies...**
  - Each node on the bus will receive PdelayReq/Resp/FwUp from all others
  - State machine would fail in case of multiple Pdelay responses
    - Related port will change its state to "*asCapable=false*"
    - There is already a proposal from RUETZ that will fix this behavior
  - Heavy Pdelay traffic, if each of N nodes keeps sending Pdelay requests as required: $(2N^2-N)$ messages per PdelayReqInterval



Proposal:
Modification of the MDPdelayReq state machine such 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.

MICROCHIP

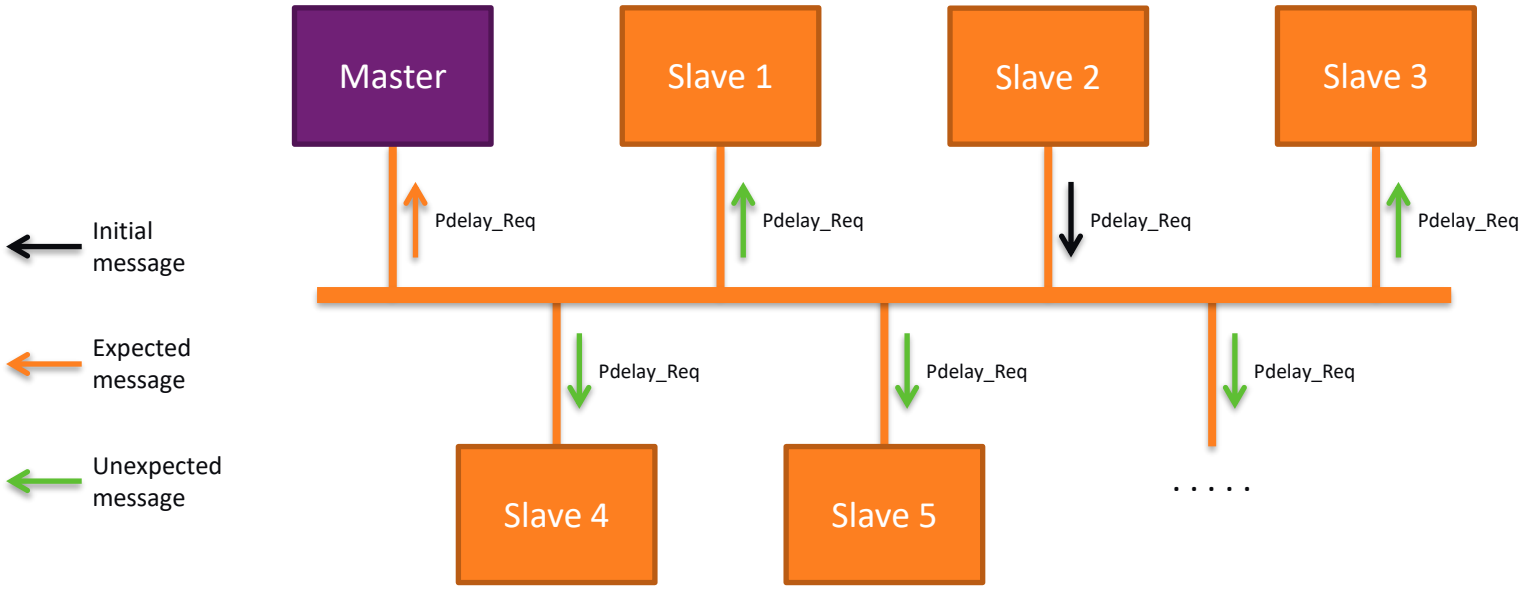
Available Solutions As Of Today

- **Disable Pdelay signaling**
  - 802.1dg automotive profile defines static gPTP values that would allow to disable Pdelay for AED-Es connected to a multidrop bus
    - "*Static gPTP values are configured prior to system startup, stored in non-volatile storage, and are not expected to change during system operation.*"
    - "*asCapable*" can be set to *true*, so there is no need to run Pdelay to detect the edge of time-aware networks
    - "initial/operLogPdelayReqInterval" can be used to disable sending Pdelay requests

- **That implies...**
  - AED-E is not able to determine propagation delay on the link
    → static configuration value must replace this value (not always acceptable)
  - AED-E is not able to determine *neighborRateRatio* from Pdelay
    - Nevertheless, GM ratio is accumulated *neighborRateRatio* (peer-to-peer)
      → breaks cumulative rate ratio mechanism of gPTP
    - Instead, only ingress-timestamps of gPTP Sync messages can be used to determine clock frequency ratio between AED-E and GM clock (end-to-end)
    - *neighborRateRatio* can be reconstructed for AED–E slave ports, if needed

Short Synopsis Of RUETZ Proposal

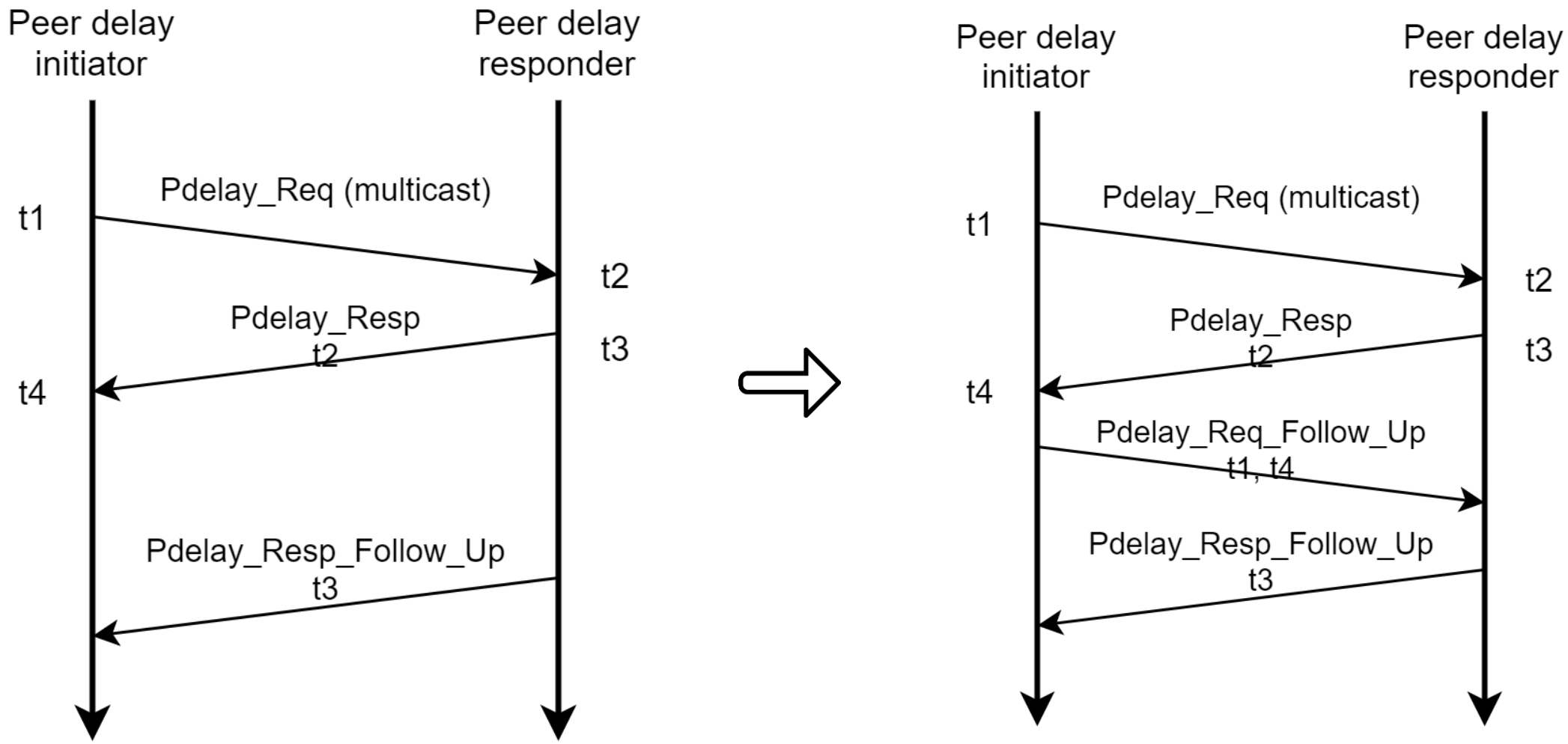
- **Modification of the MDPdelayReq state machine**
- **Inhibit slaves to respond to Pdelay_Req messages**
  - Only master port responds to Pdelay to avoid flooding the bus



Every time-aware network participant receives Pdelay_Req and would **have to respond to it**.
→ A requesting slave would have to distinguish between the relevant responses from the master and the rather useless responses from the other slaves.

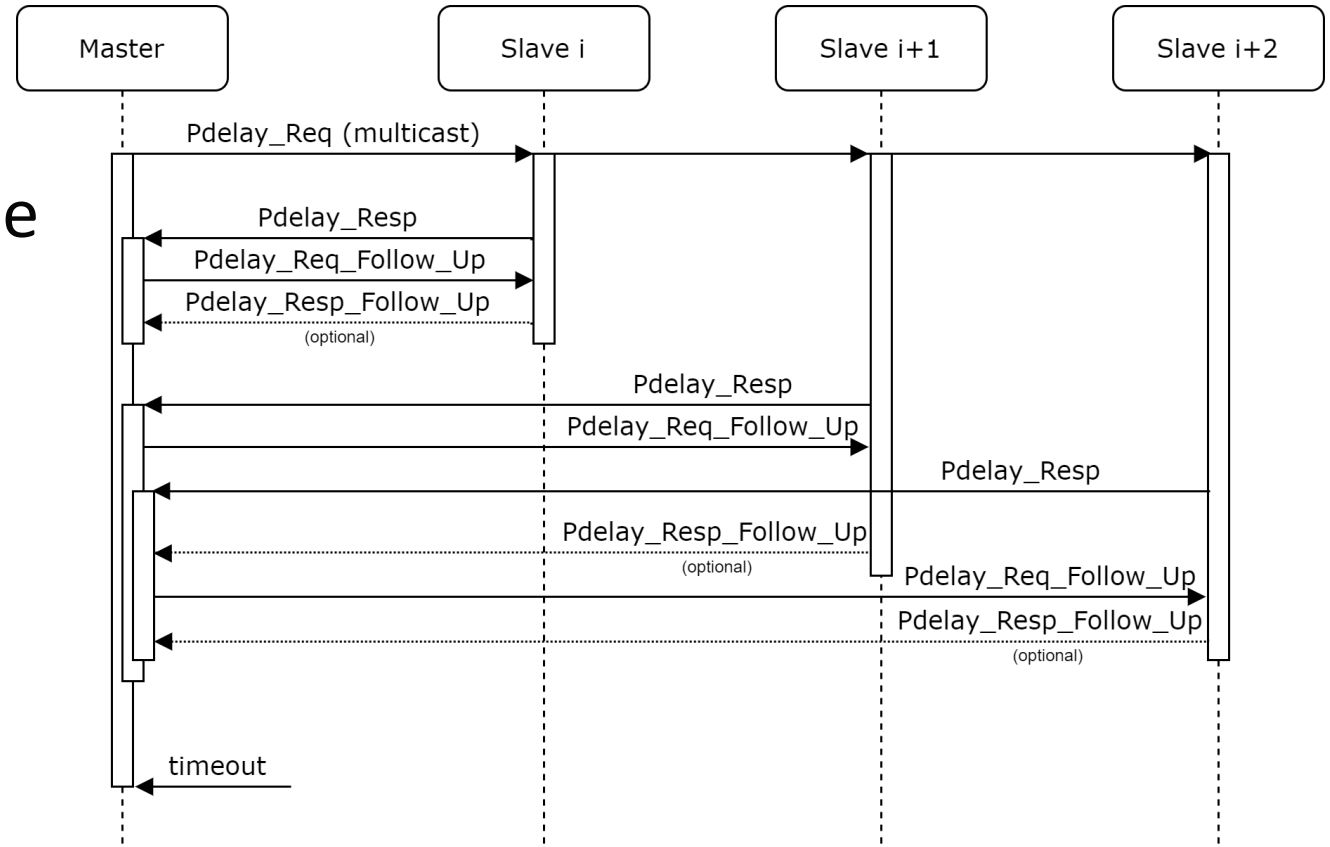
Alternative Proposal (1)

- **Introduce Pdelay_Req_Follow_Up message**



Alternative Proposal (2)

- **Allows any port to initiate Pdelay requests, if desired**
  - Typically, only master ports will initiate Pdelay requests, to reduce bandwidth, but not limited to!
  - Pdelay requests in the opposite direction are of no use anymore, except for enumeration
  - Each node can now enumerate all connected time-aware nodes using Pdelay requests, at any time this is required



"Slave i+n" simply demonstrate how responses to Pdelay_Req of multiple ports can overlap. Pdelay_Resp_Follow_Up is optional, if the slave port will never change its role to a master port. A 10Base-T1S slave node will most likely never be a GM instance, as long as there is a 100Base or faster network connected, for better clock quality. But distribution direction can change, i.e. in case of redundant network paths.

MICROCHIP

Comparison Of Proposals

- **Proposal from RUETZ comes with smallest impact to the standard**
  - Requires extra configuration to selectively disable nodes for Pdelay, if not fixed to T1S master nodes
  - Slave nodes can never become clock source w/o changing this configuration first
  - Slave nodes can't detect other time-aware nodes through Pdelay (*gPTP domain*)
    - If they still need to enumerate, nodes might have to listen for Pdelay requests from yet unknown ports, instead
- **Pdelay traffic bandwidth for a multidrop bus with N nodes…**
  - RUETZ proposal requires (3N-3) and alt. proposal (3N-2) or (2N-1) [1] Pdelay messages per PdelayReqInterval
- **Pdelay RX/TX symmetry and queue size considerations**
  - RUETZ proposal requires master to receive (N-1) and transmit (2N-2) messages
  - With the alt. proposal, the master node must transmit (N) messages and receive (2N-2) or (N-1)[1]
    - Ports must be able to receive (N-1) messages back-to-back, but in case of alt. proposal only after a request was initiated
    - Processing of multiple responses can be serialized to save resources, but may become more challenging to fulfill timings
- **Consider Unicast Pdelay responses for CPU offloading**
  - In case of multicast addressing, each multidrop node receives all other Pdelay traffic, passing them to gPTP stack just to drop most of them there: (3N-6) dropped frames per PdelayReqInterval
  - Can be avoided, if all responses use source MAC address from origin Pdelay request/response as destination
  - MAC-filter will then eliminate those frames, offloading this from the CPU
  - With proposal from RUETZ, not all can be offloaded: still (N-2) requests must be dropped at gPTP stack

[1] Only if optional Pdelay_Resp_Follow_Up gets disabled, preventing slave nodes from becoming clock source

802.1as Announce And Signaling Messages

- **BMCA Announce messages**
  - Announce message is only sent by master ports
  - How to perform BMCA on multidrop networks?

- **Signaling messages**
  - Defined for P2P links only
  - How to apply those to multidrop networks?

Conclusion

- **Specify new media-dependent layer for Multidrop Links**
  - Pdelay Behavior
  - BMCA and Announce message
  - Signaling messages

- **Add this to 802.1dg automotive profile first?**
- **Standardize new MD layer with next 802.1as revision**

Thank You