Changes/Additions Needed After 802.1ASbt/D0.4 (to Produce the Next Draft)

Geoffrey M. Garner
Consultant
gmgarner@alum.mit.edu

IEEE 802.1 TSN TG
2014.05.09

Outline

- Introduction
- Assumptions
- Changes/Additions to 802.1ASbt

Introduction - 1

- This presentation summarizes changes needed to produce the next draft of 802.1ASbt, after D0.4
- It is based on the following:
  - Comments on Draft D0.4, presented at the March, 2014 802.1 TSN TG meeting
  - Comments on the presentation “Multiple Timescale Feature for 802.1ASbt” from the March, 2014 802.1 TSN TG meeting
  - The above comments were discussed in the meeting and subsequent TSN calls, on 4/23, 4/30, and 5/7/2014

Introduction - 2

- This presentation also discusses the following items
  - Maintenance request on Pdelay_Req state machine submitted by Bob Noseworthy and Jeff Laird on May 7, 2014 [2] (based on [1])
  - Maintenance request on PortSyncSyncSend state machine submitted by Bob Noseworthy and Jeff Laird on May 7, 2014 [3]
  - Possible changing of 802.1AS title to eliminate the word “bridge,” and also change the terminology to not refer to a bridge unless that really is what is meant
  - Possible changing of 802.1ASbt from an amendment to a revision

Comments on Draft D0.4 – 1

- New definition 3.11A (note that it should be 3.10A)

3.11A maximum absolute value relative time error: The maximum absolute value relative time error, max|RTE|, measured between two clocks over a measurement interval of duration T, is defined as:

$$\max|\mathrm{RTE}| = \max_{t_0 \le t \le t_0 + T} |x_2(t) - x_1(t)|$$

where $x_1(t)$ and $x_2(t)$ are the time errors, relative to the timescale in use, of the two clocks, respectively, as a function of time, and t_0 is the start time of the measurement interval.

- Propose to remove the word "measured," i.e.,
  - The maximum absolute value relative time error, max|RTE|, measured between two clocks …"

Comments on Draft D0.4 – 2

- Request that the support of the message interval request TLV be made mandatory
  - If this change were made, it would be mandatory that every time-aware system be able to process and honor the request of a message interval request TLV
  - It was stated in the discussion that a device could still be considered compliant if it could not send as fast as requested (would only want it to not send faster than requested
  - Would every device also be required to be capable of making such a request, i.e., sending this TLV?
  - Right now, 802.1AS does not specify when a port would send this TLV
    - Would this still be the case?

Comments on Draft D0.4 – 3

- Request that the support of the message interval request TLV be made mandatory (Cont.)
  - Note that current PICs seems to refer to sending this TLV, but not receiving it, e.g.,
    - A.5, row 4: SIG Does the device transmit signaling messages?
    - A.7, row 4: MINTA Does the device port sending a Signaling message that contains a message interval request TLV adjust its syncReceiptTimeoutTimeInterval in compliance with the requirements of 10.5.4.3.7 and Table 10-12?
    - A.9, row 19: BMC-19 Does the device port sending a message interval request signaling message adjust its announceReceiptTimeoutTimeInterval in compliance with the requirements of10.5.4.3.8 and Table 10-13?

Comments on Draft D0.4 – 4

- Turn the Editor's Note on p.18 into a note:

10.3.8.2 selected: a Boolean array of length numberPorts+1 (see ~~8.6.1~~8.6.2.8). selected[j], where 0 ≤ j ≤ numberPorts, is set to TRUE immediately after the port roles of all the ports are updated. This indicates to the PortAnnounceInformation state machine (see 10.3.11) that it can update the portPriorityVector and other variables for each port.

<<Editor's note: array elements 0 of the reselect and selected arrays are not used, except that the function clearReselectTree() sets reselect[0] to FALSE when it sets the entire array to zero and the function setSelectedTree() sets selected[0] to TRUE when it sets the entire array to TRUE. This is done only for convenience, so that array element j can correspond to port j. Should a note be added to point all this out to the reader? Note also that, in contrast, selectedRole[0] is used (see 10.2.3.20).>>

NOTE – Array elements 0 of the reselect and selected arrays are not used, except that the function clearReselectTree() sets reselect[0] to FALSE when it sets the entire array to FALSE and the function setSelectedTree() sets seleced[0] to TRUE when it sets the entire array to TRUE. This is done for convenience, so that array element j can correspond to port j. Note also that, in contrast, selectedRole[0] is used (see 10.2.3.20).

Comments on Draft D0.4 - 5

- Discussion of redundancy topic with respect to best master selection and computation of redundant (maximally disjoint) paths – **So far no firm conclusions**
  - Should IS-IS (and SPB, with link-state protocol) be used?
  - What portions should be specified in 1588? What portions should be specified in 802.1AS (or perhaps the future 802.1 document that would be the media-dependent specification of 1588 for 802 media, if such a split of 1588 occurs)?
    - Suggested that 1588 would do best master selection; however, maximally redundant path computations might be media dependent because various decisions could depend heavily on the media type.
  - Should the redundant paths (and therefore the redundant GMs, since each GM would be the root of a different synch spanning tree) be identified by domain number, or a different (new?) identifier?
    - Suggested we could use domains for now, until such time as we find out it doesn't work
    - But note that if we also use domains for multiple timescales, we need to distinguish (carry information) on which domains are redundant paths of the same timescale, and which are for different timescales

Multiple Timescales Assumptions - 1

- Each time-aware system will support at most two domains
  - Each time-aware system shall support at least one domain
- Every time-aware system shall support domain 0
  - This is the Universal Time domain, and corresponds to gPTP Gen 1
- The working clock domain number shall be in the range 1 – 127 (it shall not be zero)
- The different domains correspond to different instances of gPTP
  - Whether or not a time-aware system supports a particular domain is reflected by whether asCapable is TRUE for that domain
  - For full-duplex 802.3, the Pdelay mechanism is used by a port in the normal way to determine if the port at the other end of the link supports that domain
  - If the neighboring port does not respond to Pdelay_Req (or if it responds but the mean propagation delay exceeds the respective threshold), then the port does not send any other PTP messages on that link for that domain

Multiple Timescales Assumptions – 2

- Note regarding neighborRateRatio
  - In the 802.1AS model, the neighborRateRatio is the ratio of the frequency of the LocalClock entity of the time-aware system at the far end of the link to the frequency of the LocalClock entity of this time-aware system
  - Since these two frequencies are free-running, local clock (oscillator) frequencies, the neighborRateRatio is the same in all domains
  - Nonetheless, for simplicity in the specifications, we can specify a separate per port neighborRateRatio variable in each domain
  - However, implementations are free to compute the neighborRateRatio in one domain and use it in all the domains

Multiple Timescales Assumptions - 3

- Domain 0 will use the PTP timescale (see 8.2.1 of 802.1AS and 7.2.1 of 1588 - 2008)
- The working clock domain (1 – 127) may use either the PTP or ARB timescale (see 7.2.1 of 1588 – 2008)
  - Note that in IEEE 1588 – 2008, some clockClasses are specific to the PTP or ARB epoch; however, this will not impact 802.1ASbt because 802.1AS does not explicitly call out or use these clockClasses (but they are not prohibited)
  - Add brief description of the ARB timescale to 802.1ASbt; from 1588 – 2008:

7.2 PTP timescale

7.2.1 General

The timescale for a domain is established by the grandmaster clock.

There are two types of timescales supported by PTP:

— The timescale PTP: In normal operation, the epoch is the PTP epoch and the timescale is continuous; see 7.2.4. The unit of measure of time is the SI second as realized on the rotating geoid.

— The timescale ARB (arbitrary): In normal operation, the epoch is set by an administrative procedure. The epoch is permitted to be reset during normal operation. Between invocations of the administrative procedure, the timescale is continuous. Additional invocations of the administrative procedure may introduce discontinuities in the overall timescale.

Multiple Timescales Feature Changes/Additions - 1

- Subclause 8.1 – gPTP domain
  - Modify the text to indicate that:
    - Domain 0 shall be supported (Universal time domain)
    - A second domain, with domain number in range 1 – 127, may be supported (working clock domain)
    - Should we have descriptive information on what is meant by “universal time domain” and “working clock domain” (or is this descriptive material application dependent and, if so, does it belong somewhere else)?
  - Add text that gives a general description of how multiple-domain information is organized in the remainder of the document
    - Unless otherwise specified in this standard, the operation of the protocol and the timescale in different domains are independent (this is also stated in IEEE 1588 – 2008)
    - Unless otherwise stated, information in the remainder of the document is per domain

Multiple Timescales Feature Changes/Additions - 2

- Subclause 8.2.1 – Introduction (of 8.2 Timescale)
  - Add description of the ARB timescale (following 7.2.1 of IEEE 1588, which describes both the PTP and ARB timescales)
  - Note that it will be necessary to go through 802.1AS and possibly add references to the ARB timescale in places where the PTP timescale is mentioned
  - Indicate in 8.2.4 that the epoch can be the PTP or ARB epoch
- Subclause 8.2.3 – UTC Offset
  - It must be indicated that currentUtcOffset shall not be used to compute UTC when the timescale is ARB (this was recently clarified by the P1588 Upkeep Subcommittee)

Multiple Timescales Feature Changes/Additions - 3

❑Clause 9 – Application interfaces

- This clause describes a ClockSourceTimeInterface, which provides external time to a time-aware system, and four Clock Target interfaces, which provide time from a time-aware system to an application
- It needs to be decided whether domainNumber needs to be added to these interfaces
  - If these interfaces are considered to be implicitly associated with a domain, then domain number is not needed
  - If these interfaces are considered to be associated with the time-aware system as a whole, then domain number is needed

Changes/Additions to 802.1ASbt - 4

- Clause 12 – Media-dependent layer specification for IEEE 802.11 links
  - A mechanism for a port attached to an 802.11 link to let its neighbor(s) know which domains it supports must be defined
  - Right now, asCapable is FALSE if the 802.11 timing measurement capability is not supported, otherwise it may be set to TRUE (12.3)
  - However, there is not currently a way for a node to determine if its neighbor supports gPTP if the link is 802.11
  - One possibility might be to define a supportedDomains TLV and attach this TLV to the Timing Measurement Action Frame and to the ACK

Changes/Additions to 802.1ASbt – 5

- The following should be investigated
  - Can 802.1AS define additional vendor-specific information (e.g., with a Type=1) that would signify the supportedDomains TLV?
  - Can this information be attached to ACK (currently the Timing Measurement Action Frame carries vendor-specific information with Type=0 (FollowUpInformation)?
- If the above is possible, it must be described, and respective processing of the information must be added to the master and slave state machines
  - If the above is not possible, alternative approaches must be examined, e.g., use of Signaling messages to carry the supportedDomains TLV

Changes/Additions to 802.1ASbt - 6

- Clause 12 – Media-dependent layer specification for IEEE 802.11 links
  - The content of clause 12 must be examined, so that the various aspects can be indicated as domain-independent or domain-specific
  - The FollowUpInformation TLV and aspects related to this (e.g., use of its parameters in state machines, and related local variables, shared variables, and functions) are domain-specific
  - While the 802.11 timing measurement capability is domain-independent, it is invoked separately by each domain
  - The master and slave state machines are domain-specific

Changes/Additions to 802.1ASbt - 7

- Clause 13 – Media-dependent layer specification for interface to IEEE 802.3 Ethernet passive optical network links
  - In EPON transport, the TIMESYNC message, transported using the organization-specific slow protocol, carries the correspondence between grandmaster time and MPCP counter. It is necessary to:
    - Add the domain number to the TIMESYNC message (subclause 13.3)
    - Add the domain number to the OSSPDU.request (subclause 13.6.1)
    - Add the domain number to the OSSPDU.indication (subclause 13.6.2)
    - The requestor and responder state machines are domain-specific (subclauses 13.8.1 and 13.8.2)
    - The TIMESYNC message transmission interval is domain-specific (subclause 13.9.1)

Changes/Additions to 802.1ASbt - 8

- Clause 13 – Media-dependent layer specification for interface to IEEE 802.3 Ethernet passive optical network links
  - A mechanism for a port attached to an 802.3 EPON link to let its neighbor(s) know which domains it supports must be defined
  - Possible solutions are to define a supportedDomainsTLV, and:
    - Carry the supportedDomains TLV in a Signaling message
    - Carry the supportedDomains TLV in a new message carried using the organization-specific slow protocol
    - Whatever solution is used, the relevant messages and state machines must be described

Changes/Additions to 802.1ASbt - 9

❑Clause 14 – Timing and synchronization management

- ▪After clauses 1 – 13 and Annex E are updated for multiple-domain support, each managed object of clause 14 must be indicated as domain-specific (i.e., one instance per domain) or domain-independent
  - •It is expected that the vast majority of the managed objects (and possibly all the managed objects) will be domain-specific

❑Clause 15 – Managed object definitions

- ▪The MIB must be generalized to allow for multiple domains, in accordance with the changes to clause 14

Changes/Additions to 802.1ASbt - 10

- Annex A – PICS Proforma
  - Appropriate PICS entries related to multiple domain support must be added
  - The feature is optional, but if it is implemented it shall be implemented as specified
- Annex B – Performance requirements
  - It must be indicated that if multiple domains are present, the requirements of Annex B apply to all the domains

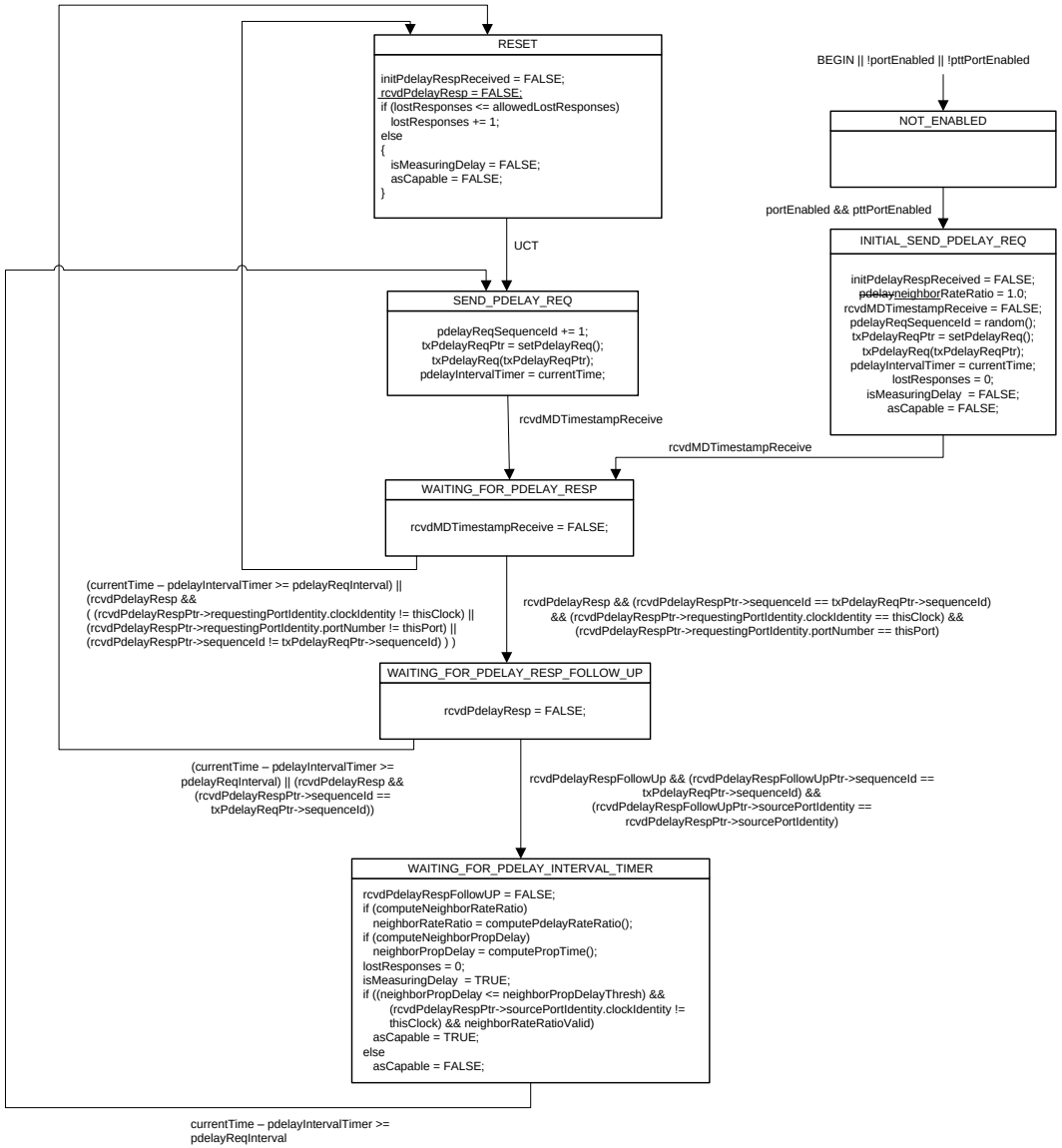
Changes/Additions to 802.1ASbt - 11

- Annex E – Media-dependent layer specification for CSN network (note that it has been agreed to make Annex E a numbered clause, following clause 15)
  - All computations of Annex E are modeled as domain-specific

Changes/Additions to 802.1ASbt - 12

- Annex F – PTP profile included in this standard
  - In F.2, item (a), it must be indicated that at least one domain, namely domain 0, is required, and that there may be a second domain with domain number in the range 1 – 127

Maintenance Request on MDPdelayReq SM - 1

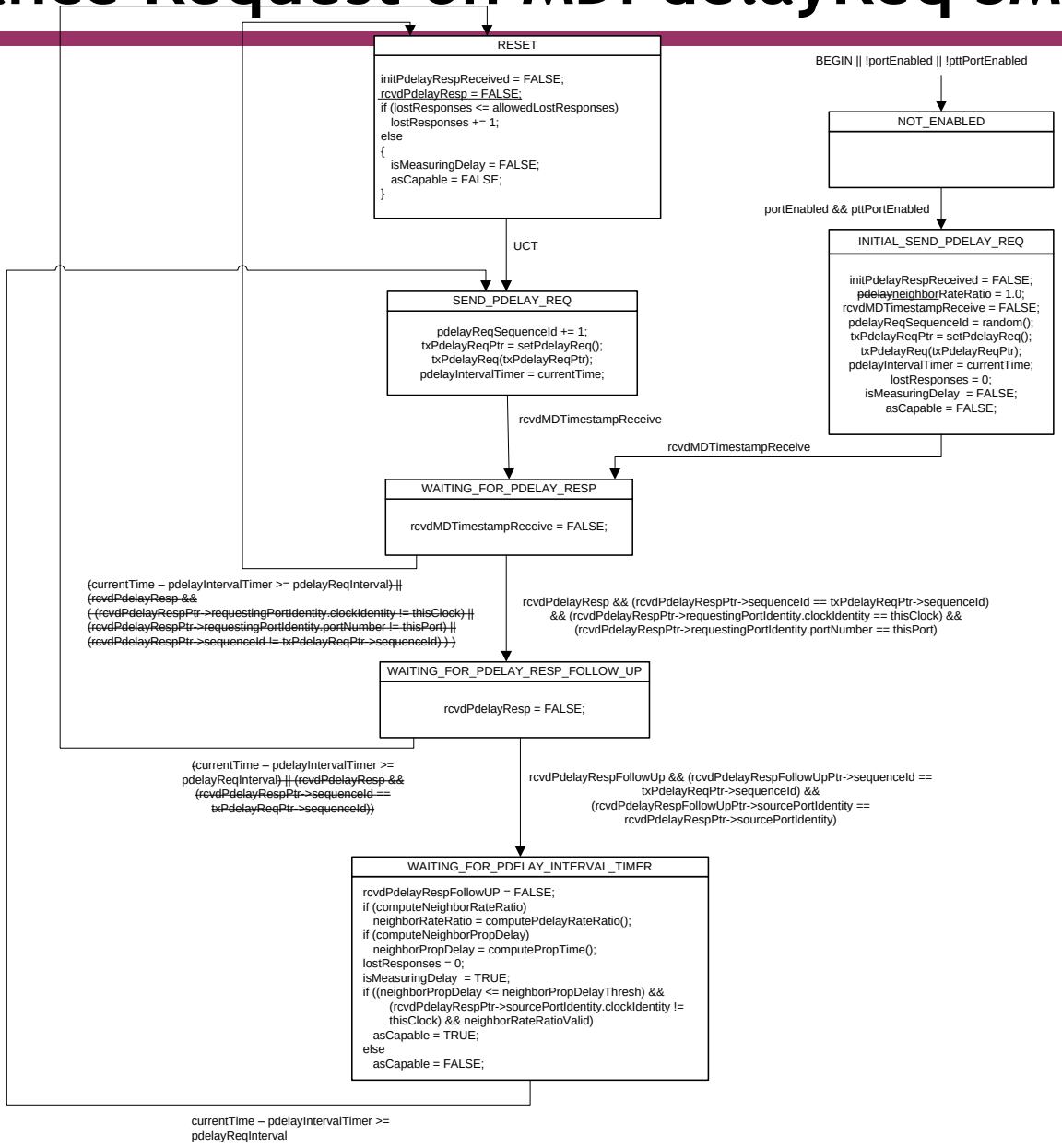


- Fig. 11-8 from 802.1AS-cor-1
- For detailed description, see the actual maintenance request [2]
- Transitions to RESET due to lost Pdelay_Resp or Pdelay_Resp_Follow_Up occur after timeout
- However, transitions due to erroneous frames are instantaneous
- This means that persistent erroneous frames sent by responder can cause the requestor to generate a storm of Pdelay_Req messages

Maintenance Request on MDPdelayReq SM - 2

- Since the conditions that check for erroneous Pdelay_Resp or Pdelay_Resp_Follow_Up frames are (or, at least, are intended to be) negations for conditions that cause transitions when correct frames are received, we can simply eliminate the check for erroneous frames, and simply wait for timeout
- This will cause a delay in entering the RESET state, and therefore a delay before the next Pdelay_Req is sent
- Proposed modifications to the MDPdelayReq state machine are shown on the next slide

Maintenance Request on MDPdelayReq SM - 3



Maintenance Request on MDPdelayReq SM - 4

- Additional notes:
  - It appears that the condition being deleted from the transition from WAITING_FOR_PDELAY_RESP_FOLLOW_UP to RESET is not what was intended (it corresponds to the transition from WAITING_FOR_PDELAY_RESP)
  - It appears that the variables rcvdPdelayResp and rcvdPdelayRespFollowUp are not initialized to FALSE in the state INITIAL_SEND_PDELAY_REQ
  - However, the descriptions of these variables (11.2.15.1.2 and 11.2.15.1.4) indicate “This variable is reset by the current state machine.”
    - It is not clear whether this means they are initialized to FALSE on entering the state machine, or something else (there is similar text for analogous variables of other state machines, and in some cases the respective variables are initialized and in some cases they are not
  - These points should be addressed here and in other state machines if necessary

Maintenance Request on PortSyncSyncSend SM - 1

- For detailed description, see the actual maintenance request [3]
- The issue was also discussed in the January, 2014 802.1 TSN TG meeting, and is briefly summarized in the Editor's Note at the beginning of Clause 10 (printed p. 16) of 802.1ASbt/D0.4
- We therefore only briefly summarize the issue and proposed solutions in the maintenance request here
- The branch from the SET_SYNC_RECEIPT_TIMEOUT_TIME state to the SEND_MD_SYNC state has the condition for the transition:
  - (((rcvdPSSync && (currentTime – lastSyncSentTime >= 0.5*syncInterval) && rcvdPSSyncPtr->localPortNumber != thisPort))

    || ((currentTime – lastSyncSentTime >= syncInterval) &&

    (lastRcvdPortNum != thisPort)))

    && portEnabled && pttPortEnabled && asCapable &&

    selectedRole[thisPort] == MasterPort

Maintenance Request on PortSyncSyncSend SM - 2

- Suppose the upstream system has nominally the same Sync interval, but in reality it is slightly larger due, e.g., to a slight frequency difference between the nodes
- With this, the current system's Sync interval could expire, and the system would send a Sync message, slightly before receiving a Sync message from upstream
- When the Sync message (call this sync1) is received from upstream, the current system will wait 0.5*syncInterval before sending the next Sync message
- The upstream system will then send the next Sync message (sync2) nominally 1 Sync interval after sync1, or approximately ½ Sync interval after the current system sent the Sync message after receiving sync1
- The current system will send the next Sync message ½ Sync interval after the previous one (i.e., the one sent after receiving sync1)
- In this manner, the current system has send several Sync messages at twice the Sync rate

Maintenance Request on PortSyncSyncSend SM - 2

- The maintenance request [3] suggests two possible solutions:

a) Remove 0.5 from the condition (currentTime – lastSyncSentTime >= 0.5*syncInterval); this would simply the condition, reducing it to

- (currentTime – lastSyncSentTime >= syncInterval) && ((rcvdPSSync && rcvdPSSyncPtr->localPortNumber != thisPort) ||

  ((lastRcvdPortNum != thisPort)))

  && portEnabled && pttPortEnabled && asCapable &&

  selectedRole[thisPort] == MasterPort

- This will result in a longer residence time, and increase the error in the transported time

b) Follow IEEE 1588-2008, which requires the actual sync interval to be within ±30% of the specified mean with 90% confidence; the solution actually suggests keeping the all the sync interval instances in this range (i.e., not only 90% of them)

Maintenance Request on PortSyncSyncSend SM - 3

- With solution (b) above, the condition becomes [3]
  - (((rcvdPSSync && (currentTime – lastSyncSentTime >= 0.7*syncInterval) && rcvdPSSyncPtr->localPortNumber != thisPort))

    || ((currentTime – lastSyncSentTime >= 1.3*syncInterval) &&

    (lastRcvdPortNum != thisPort)))

    && portEnabled && pttPortEnabled && asCapable &&

    selectedRole[thisPort] == MasterPort

    Solution (b) seems preferable, because:

    - It does not increase the residence time as much as solution (a)
    - It allows more margin for the sync interval (but still complies with IEEE 1588 – 2008)

Change of 802.1AS Title to Eliminate "Bridge" - 1

- It has been pointed out that, since the 802.1AS (i.e., gPTP) layers are above the MAC, 802.1AS need not be limited to bridged networks
  - Note that the same is true of IEEE 1588, for which transports other than those specified in IEEE 802 standards (i.e., bridged LANs) are allowed
- It therefore has been suggested that
  - The title of IEEE 802.1AS be changed to eliminate the word "bridged" (and make any other appropriate changes)
  - Each instance of the words "bridge" or "bridged" in 802.1AS (and 802.1AS-Cor-1 and 802.1ASbt) be examined, and changed unless bridging really was meant
- Note that changing the title requires a revision of 802.1AS; the title cannot be changed in an amendment (i.e., in 802.1ASbt)
  - This will lead to the next item

Change of 802.1AS Title to Eliminate “Bridge” - 2

- Note that there are approximately 150 instances of “bridge”, “bridged”, etc., aside from page headers.
  - These could be changed in 802.1ASbt, but this would be tedious, as an editing instruction would be needed for each one.
  - These changes would be easier in a revision.

Conversion of 802.1ASbt to a Revision - 1

- It has been suggested that, since 802.1AS does not have multiple amendment projects ongoing simultaneously (as has often been the case with IEEE 802.1Q), a revision would be simpler to prepare and review than an amendment
- If this is done, the editor would need to create a draft now that folds in all the changes of the corrigendum and 802.1ASbt/D0.4
  - This would be some amount of work
  - It is the understanding of the editor that, after the amendment is finished, the IEEE editor(s) could be requested to create an edition that combines the base document, corrigendum, and amendment (and thus in that case the IEEE editor(s) would do the work (though likely the editor and committee would have to review it)
  - But, with a revision, the drafts going forward would be simpler to prepare, because editing instructions would not be needed

Conversion of 802.1ASbt to a Revision - 2

- Presumably, with a revision the entire document is open for comments in sponsor ballot (in an amendment, only the clauses/subclauses specifically referred to are open for comment) (this is the understanding of the editor)
- To convert to a revision, a new or modified PAR would be needed
- If it is desired to do a revision instead of an amendment, the decision should be made sooner rather than later (and preferably before the next draft is prepared)

References - 1

[1] Bob Noseworthy, Draft presentation on the Pdelay_Req state machine issue sent in email to the author, January, 2014.

[2] Bob Noseworthy and Jeff Laird, *MDPdelayReq state machine*, 802. 1ASbt Maintanance Request submitted May 7, 2014.

[3] Bob Noseworthy and Jeff Laird, *PortSyncSyncSend state machine*, 802. 1ASbt Maintanance Request submitted May 7, 2014.