

```

+-----+
| IEEE 802.1 REVISION REQUEST 0159 |
+-----+

```

DATE: 26 February 2015
NAME: Bob Noseworthy
COMPANY/AFFILIATION: University of New Hampshire's InterOperability Lab
E-MAIL: ren@iol.unh.edu

REQUESTED REVISION:
STANDARD: 802.1AS-2011
CLAUSE NUMBER: 10.2.11
CLAUSE TITLE: PortSyncSyncSend State Machine

RATIONALE FOR REVISION:

PortSyncSyncSend SM (Fig 10-8) does not behave as intended if the device was grandmaster and then has priority1 set to 255, resulting in continuous erroneous transmission of Sync messages from a port that transitions from having priority1 set to a value not equal to 255, then set via management action to 255.

=====

In the normal case, a device that is not GrandMaster capable will have priority1 set to 255. In this case gmPresent is always FALSE and the SiteSyncSync SM (Figure 10-3) will never issue a PortSyncSync structure to the PortSyncSyncSend SM.

=====

In the case of a device that can be a grandmaster, consider the following case:

If such a grandmaster-capable device has a priority1 value that is not 255 (eg: 248) then gmPresent will be TRUE and if such a device wins election as grandmaster via the BMCA (as is exceedingly likely even if just for a short time if the device is asCapable before its link partner) then the device's ClockMaster SM will issue a PortSyncSync structure through the SiteSyncSync SM to the PortSyncSyncSend SM. This results in a PortSyncSync structure being propagated to the Media Dependent layer by the entry to the SEND_MD_SYNC state and resulting action of the txMDSync function.

As the device's ClockMaster is the source of the Sync, the syncReceiptTimeoutTime is set to a maximum value of 0xFFFFFFFF per 10.2.2.3.3 (and 10.2.8.2.1(h)). [Note: localPortNumber=0 per 10.2.8.2.1, and thus the selectedRole[localPortNumber]=slavePort per 10.3.12.1.4(h)(12) (hence why the SiteSyncSync SM forward's the sync)]

Now consider such a device having its priority1 value changed to 255 via management action.

Such a management change may be considered by some to be a violation of 8.6.2.1 which states: "The value of priority1 shall be 255 for a time-aware system that is not grandmaster-capable. The value of priority1 shall be less than 255 for a time-aware system that is grandmaster-capable." However it is not clear if a specific instance of a time-aware system can change from being "not grandmaster-capable" to grandmaster-capable" through only management action.

Subclause 10.1.2 titled "Grandmaster-capable time-aware system" does strongly indicate that a device may elect to be grandmaster-capable or not "at-will", quoting 10.1.2: "A time-aware system may be grandmaster-capable. An implementation may optionally provide the ability to configure a time-aware system as grandmaster-capable via a management interface." (Note this

issue has been discussed by Geoff Garner and 802.1AS-Rev members in January 2015, who indicate that yes, priority1 may be changed to 255 via management action as it is read/write)

If such a change of priority1 to 255 is allowed, then a minor problem with the standard exists, specifically the device will continue to send Sync messages until a Sync message is received from a link partner, or the device's selectedRole[thisPort] != MasterPort (eg, it becomes a SlavePort). Furthermore, because the Sync message transmitted by the device is generated only by the PortSyncSyncSend SM, the preciseOriginTimestamp field is not updated at any time during the 'erroneous' transmission (though the followUpCorrectionField should be properly computed)

It is believed that the standard intends to allow devices to arbitrarily change priority1 values to/from 255 at management's discretion. The result of this change should prevent any Sync information from being transmitted when priority1 is 255. As noted in the NOTE in 10.1.2 quoted below: "NOTE-While a time-aware system that is not grandmaster-capable can never be the grandmaster of the gPTP domain, such a time-aware system contains a best master selection function, invokes the best master selection algorithm, and conveys synchronization information received from the current grandmaster."

There are two points here:

1- a non-grandmaster-capable system still participates in BMCA (and as a result, still sends Announce messages per existing requirements not in dispute) 2- a non-grandmaster-capable time-aware system may still act as a time-relay (a bridge that never wishes to be (or cannot be) grandmaster, but can relay time)

The following is provided by Geoff Garner:

=====

""

In the discussion at the January meeting of 802.1AS-Rev/D0.8, we discussed the proposed fix to an issue where a device that is gmCapable has its priority1 changed to 255. If there are no other gmCapable devices in the network, this device will continue to send Sync messages. The reason is that, in this situation, there is nothing that will trigger the PortRoleSelection state machine and updRolesTree(), which if triggered would result in the Announce messages sent by this node reflecting the changed priority1 information. This would eventually result in gmPresent changing to FALSE. However, since updRolesTree() is not triggered, this will not happen, and the node will keep sending Sync messages.

""

PROPOSED REVISION TEXT:

This issue was raised in 2014 to Geoff Garner and discussed through 802.1AS-Rev meetings. As the following remedy is included in 802.1AS-Rev/D0.8 and D0.9, the proposed remedy is offered to raise awareness of the issue and align 802.1AS-2011 with the behavior anticipated to be in 802.1AS-Rev.

The remaining text is provided by Geoff Garner:

=====

The proposed fix is in D0.8, in 10.3.12.1.1 and in Figure 10-14 (the updRolesTree() SM). Specifically, a new variable: systemIdentityChange is introduced. This is a Boolean that is set to TRUE whenever any attribute of the systemIdentity (see 10.3.2) changes (this includes priority1, clockClass, clockAccuracy, offsetScaledLogVariance, priority2, and clockIdentity). The variable is reset in the ROLE_SELECTION state of the PortRoleSelection SM. It is added as a condition (via an OR) to transition from this state back to itself. With this variable, if priority1 changes, systemIdentityChange will become TRUE, the transition will occur, and updRolesTree() will get invoked (and systemIdentityChange will

be reset to FALSE.

Related to this, it was decided in the meeting to remove the note below systemIdentityChange that currently reads: NOTE - In practice, the systemIdentity changes when one of the attributes priority1, clockClass, clockAccuracy, offsetScaledLogVariance, and priority2 change, e.g., due to management action, degradation or loss of the ClockSource, etc. The systemIdentity also includes the attribute clockIdentity, but this attribute does not change.

Instead, the following normative statement would be added:

"The systemIdentity shall change when one of the attributes priority1, clockClass, clockAccuracy, offsetScaledLogVariance, and priority2 change, e.g., due to management action, degradation or loss of the ClockSource, etc. The systemIdentity also includes the attribute clockIdentity, but this attribute does not change.

In addition, a PICs would be added.

IMPACT ON EXISTING NETWORKS:

If the interpretations above are correct, a remedy to eliminate the erroneous Sync transmissions by a non-grandmaster-capable system would reduce risk of timing error in a time-aware system.

```

+-----+
| Please attach supporting material, if any |
| Submit to:- Glenn Parsons, Chair IEEE 802.1 |
| and copy:- John Messenger, Vice-Chair IEEE 802.1 |
| E-Mail: stds-802-1-maint-req@ieee.org |
| |
| +----- For official 802.1 use -----+ |
| | REV REQ NUMBER: 0159 | |
| | DATE RECEIVED: 26 February 2015 | |
| | EDITORIAL/TECHNICAL | |
| | ACCEPTED/DENIED | |
| | BALLOT REQ'D YES/NO | |
| | Status: R | |
| +-----+ |
+-----+

```