System Tests For LXI Devices Implementing IEEE 1588-2008

Version 1.26

29 January 2009

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Notice of Rights All rights reserved. This document is the property of the LXI Consortium. It may be reproduced, unaltered, in whole or in part, provided the LXI copyright notice is retained on every document page.

Notice of Liability The information contained in this document is subject to change without notice. "Preliminary" releases are for specification development and proof-of-concept testing and may not reflect the final "Released" specification.

The LXI Consortium, Inc. makes no warranty of any kind with regard to this material, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The LXI Consortium, Inc. shall not be liable for errors or omissions contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

LXI Standards Documents are developed within the LXI Consortium and LXI Technical Working Groups sponsored by the LXI Consortium Board of Directors. The LXI Consortium develops its standards through a consensus development process modeled after the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Consortium and serve without compensation. While the LXI Consortium administers the process and establishes rules to promote fairness in the consensus development process, the LXI Consortium does not exhaustively evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an LXI Consortium Standard is wholly voluntary. The LXI Consortium and its members disclaim liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other LXI Consortium Standard document.

The LXI Consortium does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. LXI Consortium Standards documents are supplied "as is". The existence of an LXI Consortium Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the LXI Consortium Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every LXI Consortium Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any LXI Consortium Standard.

In publishing and making this document available, the LXI Consortium is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the LXI Consortium undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other LXI Consortium Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of LXI Consortium, the Consortium will initiate action to prepare appropriate responses. Since LXI Consortium Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, LXI Consortium and the members of its working groups are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration. Requests for interpretations of this standard must be sent to interpretations@lxistandard.org using the form “Request for Interpretation of an LXI Standard Document” found on the LXI Consortium’s Web site: http://www.lxistandard.org

Trademarks Product and company names listed are trademarks or trade names of their respective companies. No investigation has been made of common-law trademark rights in any work.

LXI is a registered trademark of the LXI Consortium, and this specification outlines the requirements needed to claim LXI compatibility. The specification strives to ensure compatibility among devices when used in a test system environment, and the authors tried to bias in favor of a simple user experience. It also strives to define interfaces rather than implementations, whenever possible, to preserve company-specific intellectual property.

This specification is the property of the LXI Consortium, a Delaware 501c3 corporation, for the use of its members.

The LXI Consortium draws attention to the fact that it is claimed that compliance with this document may involve the use of a patent or patents concerning technology described within the document. The LXI Consortium takes no position concerning the evidence, validity and scope of this patent right. The holder of this patent right has assured the LXI Consortium that he/she is willing to negotiate licenses under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with the LXI Consortium. Additional information may be obtained from the LXI Consortium’s Web site:

http://www.lxistandard.org

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. The LXI Consortium shall not be held responsible for identifying any or all such patent rights.

Comments for Revision Comments for revision of LXI Consortium Standards are welcome from any interested party, regardless of membership affiliation with LXI Consortium. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards should be addressed to:
Secretary, LXI Consortium Standards Board
www.lxistandard.org

Revision History

Date	Author	Version	Notes
5-Feb-08	Bruce Hamilton	1	Initial Tests created
6-Feb-08	Conrad Proft	1.1	Removed strike out changes
10-Mar-08	Bruce Hamilton Conrad Proft	1.11	Finalized corrections and deletions to Agilent internal document before making available to LXI Consortium
19-Mar-08	Jochen Wolle	1.12	Add PTP Monitor to recommended test tools Categorize Tests 1-9
25-Mar-08	John Eidson Jochen Wolle	1.13	Test 4 updated Categorize Tests 10-13
2-April-08	Jochen Wolle	1.14	Categorize Test 14 - 25
10-April-08	John Eidson Jochen Wolle	1.15	Merge of John Eidson's proposals (Test 13, Test 26 and Test 29) Categorize Test 26 – 28; add Test 29
21-April-08	John Eidson Jochen Wolle	1.16	Consolidated requirements for the PC Clock Minor Corrections in Test #29, #26; Finished Test #26 on management messages Reordering of Test #
21-May-08	Jochen Wolle, Kai Schmidt	1.17	Accepted all changes from John Include changes discussed during Toronto PlugFest Minor changes from 19-June-2008 telephone conference
5-August-08	John Eidson	1.18	Revise test specification format, add references to IEEE 1588, add conclusions for each test
26-Oct-08	John Eidson Michael Schultheis	1.22	Add tests from consortium meeting: reject sync from foreign master (during transiets) in test #4,
11-Nov-08	John Eidson Michael Schultheis	1.23	Added corrections to several sections to allow alternate responses to management messages.
24-Nov-08	John Eidson Michael Schultheis	1.24	Added correction to 4.D and 4.E to account for the order of regular and probe messages. Modified 9 based on 21 Nov 2008 IEEE 1588 interpretations committee action on the timeout mechanism.
15-Jan-09	John Eidson Michael Schultheis Lynn Wheelwright	1.25	Added frontmatter. Corrected test configuration in test 9. Modified 4.B.6, 4.B.12, and 4.C.5 by changing the 50ms ithreshold to 10 seconds. Changed document title.
29-Jan-09	John Eidson Michael Schultheis Lynn Wheelwright	1.26	Added sections to the introduction:1- noting existence of consortium owned operational test suite based on this document, and 2- general comment on error returns.

Table of Contents

Introduction

General:

This document provides a number of procedures to test the functionality of IEEE 1588-2008 in LXI Devices. These procedures are a fraction of the total number of tests that could be performed to test for 1588-2008 compliance. They are limited to testing LXI Devices which implement the delay request mechanism only. However, these procedures do represent the most problematic tests that should be covered.

The reader is assumed to have the following background:

- A basic understanding of the LXI specification
- A detailed understanding of IEEE 1588-2008
- A detailed understanding of the LXI IEEE 1588 Profile
- A basic understanding of software testing methods.

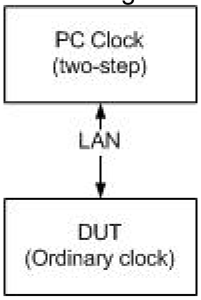
Test suite

The LXI Consortium maintains a PC based test suite that implements the test procedures specified in this document. This test suite can be used to verify the operation of IEEE 1588-2008 implementations for conformance to both IEEE 1588-2008 and the LXI IEEE-1588 Profile. This test suite is available to members of the LXI Consortium.

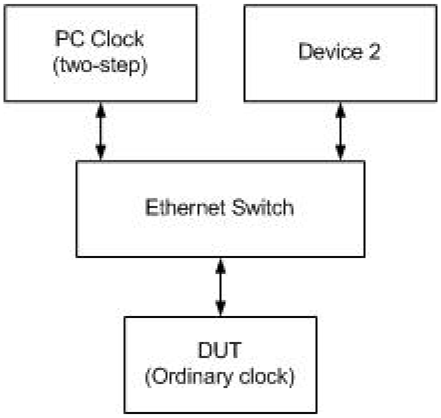
Configuration:

The following test configurations are used in this suite:

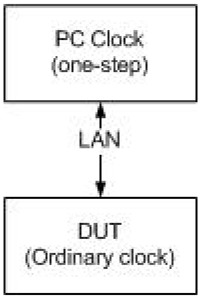
- Configuration 1: The PC Clock in this configuration must be a two-step clock, 3.1.47.



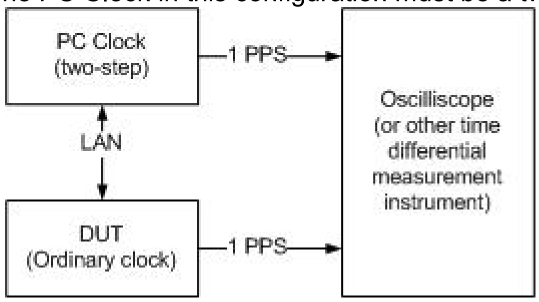
- Configuration 2: Device 2 should be a known conformant IEEE 1588 version 2 device supporting the LXI 1588 Profile (alternatively a second copy of the DUT may be used although this may complicate some tests.) The PC Clock in this configuration must be a two-step clock, 3.1.47.



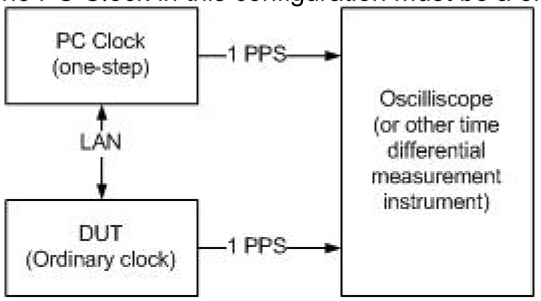
- Configuration 3: The PC Clock in this configuration must be a one-step clock, 3.1.21.



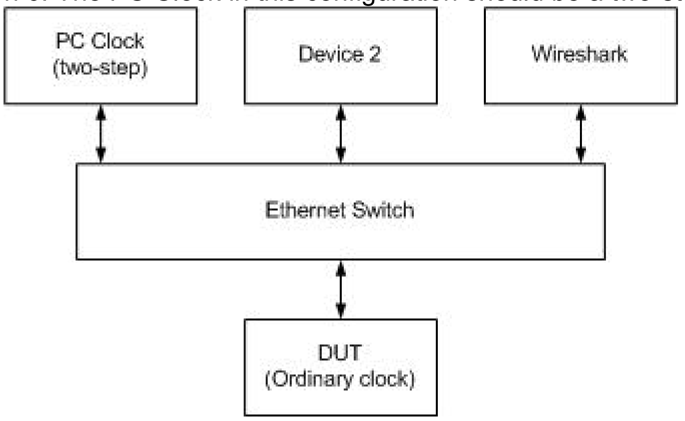
- Configuration 4: The PC Clock in this configuration must be a two-step clock, 3.1.47.



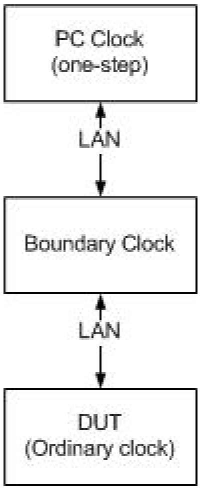
- Configuration 5: The PC Clock in this configuration must be a one-step clock, 3.1.21.



- Configuration 6: The PC Clock in this configuration should be a two-step clock, 3.1.47.



- Configuration 7:



IEEE 1588-2008 management error returns

IEEE 1588-2008 provides for error return codes to be used when a management message fails to execute properly. These error codes are specified in Table 72 of IEEE 1588-2008. However in only two failure cases are the specific error codes identified. In all other cases the choice of the most appropriate code for a given failure is left to the device designer.

Most of the procedures specified in this document make use of these management messages. The operation of this test suite is expected to generate error returns. For each test this specification lists the most probable error returns based on the semantics of the test. However since it is impossible to know the details of all device designs, any legal return as specified by Table 72 is acceptable.

System Test Specification Format:
System tests cases in this document have the following format:

- **Name**: a short human-oriented title for the test case, e.g. “Election of New Master After Better Clock Joins Domain”. This name is used as the title of the major section that describes the test.
- **Identifier**: a unique identifier for the test case, e.g. 1588-Master-Clock-06
- **Category**: defines if the test is mandatory or a vendor declaration or if it can only be tested during a plugfest. If relevant, the required states of the DUT during testing are listed.
- **Purpose**: a short description of the purpose of the test case in terms of the features of the software to be tested, and the particular behavior being verified by this test.
- **Test configuration:** Describes the test configuration used during the test.
- **Warning:** Any special requirements that must be observed during the test.

Each major test section if subdivided into:

- **Subtests:** each of these tests or prepares to test specific requirements for validation of the overall test.
- **Steps:** each subtest is divided into steps which detail the exact procedures to be followed. Each step defines:
  - **Entry state:** The state of the PC Clock(s) and the DUT prior to the actions
  - **Actions:** The specific actions such as configuration of a PC Clock or sending a message that constitute the step.
  - **Expected results**: Lists the observations that must be made and the expected results of these observations
  - **Record:** Indicates if any of the observations must be recorded for use later in the subtest or test.
  - **Conclusion:** Details the implications of the expected results being satisfied. The implications may be pass, fail, partial validation, repeat until, or proceed to next step.

Equipment/Setup Recommendations

PC Clock-

This is the pricinciple device used during these tests. It is an integrated clock and testing device with accompanying software on a PC.

The PC Clock must have the following features:

- Must be an IEEE 1588-2008 clock implementing the LXI 1588 profile (essentially: delay request-response mechanism, IPv4 transport, minor extension of range on Sync rate)
- Be a good enough clock to permit reasonably accurate and stable synchronization by the DUT, or by the PC Clock to the DUT as required
- Must have test fields or properties as follows:
    - Ability to read the datasets of the clock without the need to send management messages.
    - Ability to modify the versionPTP field in all transmitted messages.
    - Ability to change the domain of the PC Clock. Normal value of this field is 0 (the default domain) but can be modified for individual tests as specified.
    - Must be able to set the alternateMasterFlag in the PTP Announce, Sync, Follow_Up and Delay_Resp messages. Normal value of flag is FALSE but can be modified for individual tests as specified.
    - Must be able to set the value of portDS.logMinDelayReqInterval in the port data set of the PC Clock (which is logMessageInterval in a Delay_Resp message sent from the PC Clock) to any value between portDS.logSyncInterval and portDS.logSyncInterval +5. Normally portDS.logSyncInterval = 0 and portDS.logMinDelayReqInterval = +5.
    - Must be able, when the alternateMasterFlag is TRUE, to be forced into the master state so that it sends Announce, Sync, and possibly Follow_Up messages. Normal mode of operation is as defined by the 1588 state machine.
    - Must be able to append the NOT_DEFINED TLV to any PTP message. Normal operation is not to append this TLV.
    - Must be able to target a management message to all or any PTP clocks or ports in the domain. Normal mode of operation is to address to all ports in the system, e.g. targetPortIdentity.clockIdentity and targetPortIdentity.portNumber are both all 1s.
    - Must be able to send management messages with an illegal value of actionField.
    - delayReqModification: This field is always added to the correctionField in Delay_Req messages prior to transmission from the PC Clock when in the slave state. Normal value of this field is 0 but can be modified for individual tests as specified (note the master is also responsible for adding the correctionField of the received delayReqMessage into the correctionField of the DelayResp message- see 11.3.2 of IEEE 1588-2008).
    - syncModification: This field is always added to the correctionField in Sync messages prior to transmission from the PC Clock when in the master state. Normal value of this field is 0 but can be modified for individual tests as specified.
    - followUpModification: This field is always added to the correctionField in Follow_Up messages prior to transmission from the PC Clock when in the master state. Normal value of this field is 0 but can be modified for individual tests as specified.
    - delayRespModification: This field is always added to the correctionField in Delay_Resp messages prior to transmission from the PC Clock when in the master state. Normal value of this field is 0 but can be modified for individual tests as specified. This modification field must be configurable so that it can be added to

Delay_Resp messages directed at say clock A but not added to clock B. i.e. modification based on requestingPortIdentity of Delay_Resp message

- startingBoundaryHops: This field of management messages is normally implementation dependent. In the PC Clock the value is normally 15 but can be modified for individual tests as specified.
- managementId: The managementId field of management messages must be changeable.
- boundaryHops: This field of management messages is normally identical to startingBoundaryHops (normally 15 in the PC Clock) when the message is issued. In the PC Clock can be modified for individual tests as specified.
- stepsRemoved: Must be able to modify the stepsRemoved field in the announce message.
- sequenceId: The sequenceId of any message sent from the clock must be changeable.
- clockIdentity: By changing the various clock identity fields, it should be possible to "emulate" the existence of other clocks like e. g. boundary clocks.
- Ability to capture and log for delivery to test script the received timestamps for Announce, Sync, Delay_Req messages.
- Ability to set all best master clock and timeout attributes and other configurable attributes of the PC Clock: priority1, prioirty2, clockClass, clockAccuracy, clockVariance, logSyncInterval, logAnnounceInterval, Announce_Timeout, Domain Number…
- Must have the ability to suppress issuing Announce or Delay_Resp message when in the master state while retaining the master state AND being able to listen for Announce messages without changing its state. One way to do this is to allow the Domain to be changed but still have the ability to receive and parse messages in domain = 0 but without changing the state of the PC Clock (now in domain not = 0. In effect this allows operation in say domain = 1 but act like wireshark in domain = 0. This is needed for some of the timeout testing see for example Test 2 BMC subtest C and following
- Must have the ability to set the hardwareCompatibility bit in the transport specific field of transmitted Announce, Sync, and Delay_Req messages per Annex D.
- Must correctly respond to the hardwareCompatibility bit in the transport specific field of received Announce, Sync, and Delay_Req messages per Annex D.
- Must have the ability to be in the master state EVEN when receiving Announce messages from a better master

Recommended for reading LAN packet contents

- Wireshark [Released under the terms of the GNU General Public License, Wireshark is free software. Formally named Ethereal]
- Network Monitor 3.x with PTPV2 plug-in [Microsoft Netmon]

Recommended for reading/setting instrument state and IEEE 1588 Monitoring/Control

- Agilent I/O Libraries Suite 15.x which includes
    - Agilent Connection Expert and Interactive LXI, a GUI for reading/setting common user accessible parameters.
    - Agilent Connection Expert and Interactive I/O for sending SCPI commands and viewing results.
- The device's Sync Configuration Web Page [LXI spec from lxistandard.org]

Management node or means of sending arbitrary messages

- Management Messages – set/get data to/from clocks type messages

- Announce Messages – operational protocol type messages

Network switch

- Any Ethernet switch such as those made by Netgear, Linksys, etc., capable of the speeds required by the clocks, and having at least 4 ports.
- An Ethernet Hub such as the Netgear DS104 or DS108.
- A transparent clock, such as those made by Hirschmann. A transparent clock will give better synchronization performance than an ordinary switch.
- A boundary clock, such as those made by Hirschmann. A boundary clock will give better synchronization than an ordinary switch. Since the switch itself may become a 1588 master and/or slave, it should not be used for some tests, and is required for others, as noted below.

Other hardware for connecting and measuring signals

- 2/4-Channel Oscilloscope with 10nsec per division time resolution
- LAN Crossover cable for connecting device to device or PC to device without a Network Switch (necessary only for cases where Auto-MDIX is not supported by one or both devices). Note that modern PC's all have Gigabit Ethernet which operates Auto-MDIX, so no LAN Crossover cable is required.
- LAN Cables, CAT5 or better, for general connecting of devices to Network Switch

Test results

The result of a test shall be one of the following qualifiers:

- **Passed**: the device passed the test without any errors
- **Failed**: the device didn't pass the test

In case the test result failed, the error message presented to the user displays which of the IEEE 1588 subclauses of the IEEE 1588-2008 or LXI IEEE 1588 Profile specification were violated.

Test Cases

The rest of this chapter lists the 1588 test cases.

Each test is structured as a series of one or more subtests. Each subtest may contain one or more steps. For each subtest the beginning and ending states of both the PC Clock and the DUT are given. If the state is not relevant to the test it is indicated in braces, e.g. <master>. For each sequence of steps in a test or subtest, as applicable, for which a conformance conclusions can be reached it will be indicated by Pass, Fail, (and possibly Fail with warning) along with the clauses in IEEE 1588-2008 whose implementation is validated by the test.

1. DUT responds correctly to Management Messages

Identifier: xxx

Category: Mandatory

Purpose:

Management messages are used to command the device, and to set and get its configuration and state variables. These tests check that the messages are recognized, that they do the right thing, and that they detect errors properly. Management messages associated with IEEE 1588-

2008 optional clauses that are NOT required or are prohibited under the LXI 1588 profile are NOT tested.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
1.A Address testing

1.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and both members of the targetPortIdentity all 1s

Expected results: The DUT should return a management message DEFAULT_DATA_SET with actionField RESPONSE.

Record: The clockIdentity and numberPorts fields of the response from the DUT

Conclusion: Expected results verified => validates global addressing (clockIdentity and portIdentity both all 1s) of 15.3.1 Table 36

1.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity all 1s and targetPortIdentity.portNumber = numberPorts +1. (Most LXI devices will have only a single port with portNumber = 1)

Expected results: No return should be received from DUT.

Record:

Conclusion: Expected results verified for both 1.A.1and 1.A.2=> completes the validation of the broadcast port specific (clockIdentity all 1s and portIdentity correct/incorrect port number) addressing of 15.3.1 Table 36

1.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = clockIdentity of DUT recorded in 1.A.1 and targetPortIdentity.portNumber all 1s

Expected results: The DUT should return a management message DEFAULT_DATA_SET with actionField RESPONSE. The clockIdentity and numberPorts fields of the response should agree with the values recorded in 1.A.1.

Record:

Conclusion: Expected results verified => validates the clock specific all ports (clockIdentity of DUT and portIdentity all 1s) addressing of Table 36 15.3.1.

1.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = clockIdentity of DUT from 1.A.1 and targetPortIdentity.portNumber = 1.

Expected results: The DUT should return a management message DEFAULT_DATA_SET with actionField RESPONSE. The clockIdentity and numberPorts fields of the response should agree with the values recorded in 1.A.1.

Record:

Conclusion: Expected results verified =>partially validates the clock and port specific (clockIdentity of DUT and portIdentity actual port number) addressing of Table 36 15.3.1.

1.A.5

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = clockIdentity of DUT from 1.A.1 and targetPortIdentity.portNumber = numberPorts +1.

Expected results: No return should be received from DUT

Record:

Conclusion: Expected results verified for both 1.A.4 and 1.A.5 => partially validates the clock and port specific (clockIdentity of DUT and portIdentity correct/incorrect port number) addressing of Table 36 15.3.1.

1.A.6

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = all 1s and targetPortIdentity.portNumber = 1.

Expected results: The DUT should return a management message DEFAULT_DATA_SET with actionField RESPONSE. The clockIdentity and numberPorts fields of the response should agree with the values recorded in 1.A.1.

Record:

Conclusion: Expected results verified =>partially validates the clock and port specific (clockIdentity of DUT and portIdentity actual port number) addressing of Table 36 15.3.1.

1.A.7

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = NOT the clockIdentity of DUT from 1.A.1 and targetPortIdentity.portNumber = all 1s.

Expected results: No return should be received from DUT

Record:

Conclusion: Expected results verified => partially validates the clock and port specific (clockIdentity of DUT and portIdentity correct/incorrect port number) addressing of Table 36 15.3.1.

1.A.8

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = NOT the clockIdentity of DUT from 1.A.1 and targetPortIdentity.portNumber = 1.

Expected results: No return should be received from DUT

Record:

Conclusion: Expected results verified => partially validates the clock and port specific (clockIdentity of DUT and portIdentity correct/incorrect port number) addressing of Table 36 15.3.1.

1.A.9

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = NOT the clockIdentity of DUT from 1.A.1 and targetPortIdentity.portNumber = numberPorts +1.

Expected results: No return should be received from DUT

Record:

Conclusion: Expected results verified => completes the validation of the clock and port specific (clockIdentity of DUT and portIdentity correct/incorrect port number) addressing of Table 36 15.3.1.

1.B Table 37 & 38 RESPONSE tests of management messages

1.B.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that the normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.

Expected results: Setup for rest of test.

Record:

Conclusion: Move to next step

1.B.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET(the ORIGINAL MESSAGE) with actionField GET and targetPortIdentity.clockIdentity = clockIdentity of DUT from 1.A.1 and targetPortIdentity.portNumber = 1.

Expected results: DUT should return a management message DE-FAULT_DATA_SET with actionField RESPONSE.

a-The targetPortIdentity field of the Response should be identical to the portIdentity of the PC Clock, i.e. the sourcePortIdentity field of the ORIGINAL MESSAGE
b-The startingBoundaryHops of the response should be 0.
c-The sequenceId of the header should be the sequenceId of the header of the ORIGINAL MESSAGE
d-The managementId field should be the managementId field of the ORIGINAL MESSAGE

Record:

Conclusion: Expected results verified =>
a-If ‘a’ is true, the addressing requirement of 15.4.1.3 is verified for management messages.
b-If ‘b’ is true, the startingBoundaryHops requirement of 15.4.1.4 is partially verified for management messages.
c-If ‘c’ is true, the sequenceId requirement of 15.4.1.2 is verified for management messages.
d-If ‘d’ is true, the managementId requirement of 15.4.1.6 Table 38 RESPONSE is verified.

1.B.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Modify the PC Clock to send startingBoundaryHops field to 12 and the boundaryHops field to 8. Send the management message DEFAULT_DATA_SET with actionField GET and targetPortIdentity.clockIdentity = clockIdentity of DUT from 1.A.1 and targetPortIdentity.portNumber = 1.

Expected results: The DUT should return a management message DEFAULT_DATA_SET with actionField RESPONSE. The startingBoundaryHops of the response should be 4.

Record:

Conclusion: Expected results verified for 1.B.2 (b), and for 1.B.3=> completes the validation of requirement of 15.4.1.4 for management messages.

1.B.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that the normal values for startingBoundaryHops = 0 and boundaryHops = 0 are used in sending management messages.

Expected results: This is a clean up prior to other tests

Record:

Conclusion: Expected results verified => no conclusion

1.C Table 40 test of all management messages for correct error response if illegal actionField is used for a management message. This test is for messages for which SET is illegal

1.C.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a-The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b-The management messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for next steps

Record:

Conclusion: Continue to next step

1.C.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message CLOCK_DESCRIPTION with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value CLOCK_DESCRIPTION and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006, or NOT_SETTABLE 0x0005 or GENERAL_ERROR 0xFFFE . The sequenceId field of the returned management message should be identical to the sequenceId field in the originnal-CLOCK_DESCRITPION management message. (The sequenceId check could be repeated for each management message but will only be checked for each type of actionField response)

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message CLOCK_DESCRIPTION 15.5.3.1.2 and sequenceId handling of 15.4.1.2

1.C.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message SAVE_IN_NON_VOLATILE_STORAGE with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value SAVE_IN_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message SAVE_IN_NON_VOLATILE_STORAGE 15.5.3.1.4

1.C.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message RESET_NON_VOLATILE_STORAGE 15.5.3.1.5

1.C.5

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value INITIALIZE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message INITIALIZE 15.5.3.1.6

1.C.6

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message FAULT_LOG with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value FAULT_LOG and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message FAULT_LOG 15.5.3.1.7

1.C.7

Entry state: PC Clock: <unknown>, DUT: <unknown>

Send the management message FAULT_LOG_RESET with actionField SET

Action: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value FAULT_LOG_RESET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message FAULT_LOG_RESET 15.5.3.1.8

1.C.8

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value DEFAULT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETTABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message DEFAULT_DATA_SET 15.5.3.3.1

1.C.9

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message CURRENT_DATA_SET with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value CURRENT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETTABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message CURRENT_DATA_SET 15.5.3.4.1

1.C.10

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PARENT_DATA_SET with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value PARENT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETTABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message PARENT_DATA_SET 15.5.3.5.1

1.C.11

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message TIME_PROPERTIES_DATA_SET with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value TIME_PROPERTIES_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETTABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message TIME_PROPERTIES_DATA_SET 15.5.3.6.1

1.C.12

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PORT_DATA_SET with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value PORT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETTABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message PORT_DATA_SET 15.5.3.7.1

1.C.13

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message ENABLE_PORT with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value ENABLE_PORT and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message ENABLE_PORT 15.5.3.2.3

1.C.14

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DISABLE_PORT with actionField SET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value DISABLE_PORT and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 SET 15.5.4, for management message DISABLE_PORT 15.5.3.2.4

1.D Table 40 test of all management messages for correct error response if illegal actionField is used for a management message. This test is for messages for which GET is illegal

1.D.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a-The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b-The management messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for next steps

Record:

Conclusion: Continue to next step

1.D.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Snd the management message SAVE_IN_NON_VOLATILE_STORAGE with actionField GET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value SAVE_IN_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE The sequenceId field of the returned management message should be identical to the sequenceId field in the original SAVE_IN_NON_VOLATILE_STORAGE management message.

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 GET 15.5.4, for management message SAVE_IN_NON_VOLATILE_STORAGE 15.5.3.1.4. and sequenceId handling of 15.4.1.2

1.D.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField GET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 GET 15.5.4, for management message RESET_NON_VOLATILE_STORAGE 15.5.3.1.5

1.D.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField GET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value INITIALIZE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 GET 15.5.4, for management message INITIALIZE 15.5.3.1.6

1.D.5

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message FAULT_LOG_RESET with actionField GET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value FAULT_LOG_RESET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 GET 15.5.4, for management message FAULT_LOG_RESET 15.5.3.1.8

1.D.6

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message ENABLE_PORT with actionField GET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value ENABLE_PORT and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 GET 15.5.4, for management message ENABLE_PORT 15.5.3.2.3

1.D.7

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DISABLE_PORT with actionField GET

Expected results: A normal RESPONSE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value DISABLE_PORT and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 RESPONSE, Table 40 GET 15.5.4, for management message DISABLE_PORT 15.5.3.2.4

1.E Table 40 test of all management messages for correct error response if illegal actionField is used for a management message. This test is for messages for which COMMAND is illegal

1.E.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a-The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b-The management messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for next steps

Record:

Conclusion: Continue to next step

1.E.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message CLOCK_DESCRIPTION with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value CLOCK_DESCRIPTION and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE The sequenceId field of the returned management message should be identical to the sequenceId field in the original CLOCK_DESCRIPTION management message.

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message CLOCK_DESCRIPTION 15.5.3.1.2 and sequenceId handling of 15.4.1.2

1.E.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message USER_DESCRIPTION with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value USER_DESCRIPTION and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message USER_DESCRIPTION 15.5.3.1.3

1.E.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message FAULT_LOG with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value FAULT_LOG and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message FAULT_LOG 15.5.3.1.7

1.E.5

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value DEFAULT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message DEFAULT_DATA_SET 15.5.3.3.1

1.E.6

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message CURRENT_DATA_SET with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value CURRENT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message CURRENT_DATA_SET 15.5.3.4.1

1.E.7

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PARENT_DATA_SET with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value PARENT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message PARENT_DATA_SET 15.5.3.5.1

1.E.8

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message TIME_PROPERTIES_DATA_SET with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value TIME_PROPERTIES_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message TIME_PROPERTIES_DATA_SET 15.5.3.6.1

1.E.9

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PORT_DATA_SET with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value PORT_DATA_SET and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message PORT_DATA_SET 15.5.3.7.1

1.E.10

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PRIORITY1 with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value PRIORITY1 and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message PRIORITY1 15.5.3.3.2

1.E.11

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PRIORITY2 with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value PRIORITY2 and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message PRIORITY2 15.5.3.3.3

1.E.12

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DOMAIN with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value DOMAIN and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message DOMAIN 15.5.3.3.4

1.E.13

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message SLAVE_ONLY with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value SLAVE_ONLY and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message SLAVE_ONLY 15.5.3.3.5

1.E.14

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message LOG_ANNOUNCE_INTERVAL with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value LOG_ANNOUNCE_INTERVAL and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message LOG_ANNOUNCE_INTERVAL 15.5.3.7.2

1.E.15

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message ANNOUNCE_RECEIPT_TIMEOUT with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value ANNOUNCE_RECEIPT_TIMEOUT and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message ANNOUNCE_RECEIPT_TIMEOUT 15.5.3.7.3

1.E.16

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message LOG_SYNC_INTERVAL with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value LOG_SYNC_INTERVAL and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message LOG_SYNC_INTERVAL 15.5.3.7.4

1.E.17

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message VERSION_NUMBER with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value VERSION_NUMBER and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message VERSION_NUMBER 15.5.3.7.7

1.E.18

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message TIME with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value TIME and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message TIME 15.5.3.2.1

1.E.19

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message CLOCK_ACCURACY with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value CLOCK_ACCURACY and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message CLOCK_ACCURACY 15.5.3.2.2

1.E.20

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message UTC_PROPERTIES with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value UTC_PROPERTIES and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message UTC_PROPERTIES 15.5.3.6.2

1.E.21

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message TRACEABILITY_PROPERTIES with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value TRACEABILITY_PROPERTIES and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message TRACEABILITY_PROPERTIES 15.5.3.6.3

1.E.22

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message TIMESCALE_PROPERTIES with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value TIMESCALE_PROPERTIES and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message TIMESCALE_PROPERTIES 15.5.3.6.4

1.E.23

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DELAY_MECHANISM with actionField COMMAND

Expected results: A normal ACKNOWLEDGE message should be received with an MANAGEMENT_ERROR_STATUS_TLV appended. This appended TLV should have the managementId field value DELAY_MECHANISM and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates Table 38 ACKNOWLEDGE, Table 40 COMMAND 15.5.4, for management message DELAY_MECHANISM 15.5.3.7.5

1.F Protocol and semantics (Table 38, 39, and 40, 15.5.2 & sub clauses) test of messages that normally use actionField GET or GET/SET

1.F.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a-The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b-The management messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c-priority1 and priority2 both = 128

Expected results: Setup for next steps

Conclusion: Continue to next step

1.F.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message NULL_MANAGEMENT with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field NULL_MANAGEMENT should be received from DUT. No data should be in the response. The sequenceId field of the returned management message should be identical to the sequenceId field in the original NULL_MANAGEMENT management message.

Record:

Conclusion: Expected results verified => validates GET/RESPONSE Table 38 for 15.5.3.1.1 and sequenceId handling of 15.4.1.2

1.F.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message NULL_MANAGEMENT with actionField SET

Expected results: A management message with actionField = RESPONSE and a managementId field NULL_MANAGEMENT should be received from DUT. No data should be in the response.

Record:

Conclusion: Expected results verified => validates SET/RESPONSE Table 38 for 15.5.3.1.1

1.F.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message NULL_MANAGEMENT with actionField COMMAND

Expected results: A management message with actionField = ACKNOWLEDGE and a managementId field NULL_MANAGEMENT should be received from DUT-A. No data should be in the response.

Record:

Conclusion: Expected results verified => validates ACKNOWLEDGE Table 38 for 15.5.3.1.1 and completes GET, SET, COMMAND validation of 15.5.3.1.1 (NULL_MANAGEMENT)

1.F.5

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE In ignore the response if NOT_SUPPORTED. If the response is GENERAL_ERROR the remainder of this sequence of tests is open to question.

Record:

Conclusion: Expected results verified => no conclusion: semantics are tested later

1.F.6

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. In either case ignore the response.

Record:

Conclusion: Expected results verified => no conclusion: semantics are tested later. However this should cause all data sets to contain their default values for the next series of steps.

1.F.7

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message USER_DESCRIPTION with actionField SET and userDescription field of type PTPText with lengthField of UInteger8 = 13 and textField Octer[13] containing the ASCI characters one;two;three

Expected results: A management message with actionField = RESPONSE and a managementId field USER_DESCRIPTION should be received from DUT-A. The userDescription field in the RESPONSE should be a PTPText field per 15.5.3.1.3.1with lengthField of UInteger8 = 13 and textField Octet[13] containing the ASCI characters one;two;three

Record:

Conclusion: Expected results verified => validates SET/RESPONSE semantics Table 38 for USER_DESCRIPTION 15.5.3.1.3

1.F.8

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message USER_DESCRIPTION with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field USER_DESCRIPTION should be received from DUT-A. The userDescription field in the RESPONSE should be a PTPText field per 15.5.3.1.3.1with lengthField of UInteger8 = 13 and textField Octet[13] containing the ASCI characters one;two;three

Record:

Conclusion: Expected results verified => validates GET/RESPONSE semantics Table 38 for USER_DESCRIPTION 15.5.3.1.3

1.F.9

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message CLOCK_DESCRIPTION with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field CLOCK_DESCRIPTION should be received from DUT. The fields in the RESPONSE should have the following values:

- clockType should be either 0x8000 (ordinary clock), or 0x8800 (ordinary clock and management node) or rarely another legal combination from table 42
- physicalLayerProtocol should be a PTPText field probably with lengthField of UInteger8 = 8 and textField Octet[8] containing the ASCI characters Ethernet or lengthField of 10 and textField containing the ASCI characters IEEE 802.3
- physicalAddressLength should be UInteger 6
- physicalAddress should be Octet[6] containing the DUT's MAC address in hex form, e.g. 0xACDE48234567
- protocolAddress should be of type PortAddress with networkProtocol field 0x0001 (UDP/IPv4), addressLength UInteger16 = 4, and addressField Octet[4] IP address as either hex field 0xE0000181 corresponding to IP multicast address

224.0.1.129 or the IP address of the port e.g. 0x821B4001 corresponding to IP 130.27.64.1

- manufacturerIdentity should be Octet[3] containing the OUI of the manufacturer in hex form, e.g. for Agilent 0x000443 (00-04-43)
- productDescription should be a PTPText field per 15.5.3.1.2.7 e.g. a lengthField of UInteger8 = 20 for a textField Octet[20] containing ASCI characters Agilent;5818A;001050
- revisionData should be a PTPText field per 15.5.3.1.2.8 e.g. a lengthField of UInteger8 = xx for a textField Octet[xx] containing ASCI characters as specified in 15.5.3.1.2.8 e.g. 01;01;01 with xx = 8
- userDescription should be PTPText field per 15.5.3.1.2.9 with lengthField of UInteger8 = 13 and textField Octet[13] containing the ASCI characters one;two;three
- profileIdentity should be Octet[6] 0x0021D6000100 (the LXI profile identification)

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for CLOCK_DESCRIPTION 15.5.3.1.2 and correct operation of maintaining implementation specific data with respect to both USER_DESCRIPTION 15.5.3.1.3 and CLOCK_DESCRIPTION 15.5.3.1.2

1.F.10

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field DEFAULT_DATA_SET should be received from DUT. The expected values are indicated in the 'record' section of this step. Note that the values should be the default values pertaining to the device as a result of 1.F.5 and 1.F.6.

Record: Record:

- TSC (Boolean): probably TRUE (indicates that device is a two step clock)
- SO (Boolean): FALSE (per LXI profile slave only not allowed)
- numberPorts UInteger16: typically = 1 for most instruments
- priority1 (UInteger8): should be 128
- clockQuality.clockClass UInteger8: should be 248 unless the device falls under RULE 2.9.3 of the LXI Profile (few will) in which case it should be 6, 7 or 127 (most likely 127 in the test environment)
- clockQuality.clockAccuracy Enumeration8 (UInteger8): in the range 0x20 to 0x31 or else 0xFE should agree with device data sheet and Table 6
- clockQuality.offsetScaledLogVariance UInteger16: should be in the range 0x4435 to 0x72B6 or 17461 to 29366 corresponding to actual variances of 10^{-18} to 10^{-4} or deviations of 1ns to 10 ms respectively
- priority2 (UInteger8): should be 128
- clockIdentity(Octet[8]): implementation specific see 7.5.2.2.2
- domainNumber (UInteger8): should be 0

Conclusion: Expected results verified => partially validates the GET/RESPONSE semantics Table 38 for DEFAULT_DATA_SET 15.5.3.3.1 If the values are as they should be the default values are validated J.3.2 and LXI profile. If values are NOT as they should be either GET does not work for DEFAULT_DATA_SET or the default values are not correct.

1.F.11

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PRIORITY1 with actionField SET and priority1 field UInteger8 = 12

Expected results: A management message with actionField = RESPONSE and a managementId field PRIORITY1 should be received from DUT. The priority1 field in the

RESPONSE should be a UInteger8 = 12. This should also place the DUT in master state and PC Clock in slave state although this is not germane to this test.

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for PRIORITY1 15.5.3.3.2

1.F.12

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PRIORITY1 with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field PRIORITY1 should be received from DUT. The priority1 field in the RESPONSE should be a UInteger8 = 12

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for PRIORITY1 15.5.3.3.2 (database semantics 8.2.1.4.1 verification partially satisfied)

1.F.13

Entry state: Entry state: PC Clock: slave, DUT: master

Action: Send the management message PRIORITY2 with actionField SET and priority2 field UInteger8 = 45

Expected results: A management message with actionField = RESPONSE and a managementId field PRIORITY2 should be received from DUT. The priority2 field in the RESPONSE should be a UInteger8 = 45

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for PRIORITY2 15.5.3.3.3

1.F.14

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PRIORITY2 with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field PRIORITY2 should be received from DUT. The priority2 field in the RESPONSE should be a UInteger8 = 45

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for PRIORITY2 15.5.3.3.3 (database semantics 8.2.1.4.2 verification partially satisfied)

1.F.15

Entry state: Entry state: PC Clock: slave, DUT: master

Action: Send the management message CLOCK_ACCURACY with actionField SET and clockAccuracy field Enumeration8 (UInteger8) = (clockAccuracy from 1.F.10)+1

Expected results: A management message with actionField = RESPONSE and a managementId field CLOCK_ACCURACY should be received from DUT. The clockAccuracy field in the RESPONSE should be a UInteger8 = (clockAccuracy from 1.F.10) +1

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for CLOCK_ACCURACY 15.5.3.2.2

1.F.16

Entry state: PC Clock: slave, DUT: master

Action: Send the management message CLOCK_ACCURACY with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field CLOCK_ACCURACY should be received from DUT. The clockAccuracy field in the RESPONSE should be a UInteger8 = (clockAccuracy from 1.F.10) +1

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for CLOCK_ACCURACY 15.5.3.2.2 (database semantics 8.2.1.3.1.2 verification partially satisfied)

1.F.17

Entry state: PC Clock: slave, DUT: master

Action: Send the management message DOMAIN with actionField SET and domainNumber field UInteger8 = 1

Expected results: A management message with actionField = RESPONSE and a managementId field DOMAIN should be received from DUT (but it will be possibly be received in domain = 1). The domainNumber field in the RESPONSE should be a UInteger8 = 1

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for DOMAIN 15.5.3.3.4

1.F.18

Entry state: PC Clock: master domain 0, DUT: master domain 1

Action: Set the Domain of PC Clock to 1. Send the management message DOMAIN with actionField GET and domainNumber field UInteger8 = 1

Expected results: A management message with actionField = RESPONSE and a managementId field DOMAIN should be received from DUT (but it will be possibly be received in domain = 1). The domainNumber field in the RESPONSE should be a UInteger8 = 1

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for DOMAIN 15.5.3.3.4 (database semantics of 8.2.1.4.3 partially satisfied)

1.F.19

Entry state: PC Clock: slave domain 1, DUT: master domain 1

Action: Set the Domain of PC Clock to 1. Send the management message DEFAULT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field DEFAULT_DATA_SET should be received from DUT. The expected values should be those recorded in 1.F.10 except for: priority1 should now be 12, priority2 should now be 45, clockAccuracy should now be (clockAccuracy from 1.F.10) +1, and domainNumber should be 1

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for DEFAULT_DATA_SET 15.5.3.3.1 The semantics of PRIORITY1 15.5.3.3.2, PRIORITY2 15.5.3.3.3, CLOCK_ACCURACY 15.5.3.2.2, and DOMAIN 15.5.3.3.4 are validated.

1.F.20

Entry state: PC Clock: slave domain 1, DUT: master domain 1

Action: Send the management message DOMAIN with actionField SET and domainNumber field UInteger8 = 0

Expected results: A management message with actionField = RESPONSE and a managementId field DOMAIN should be received from DUT (but it will be possibly be received in domain = 1). The domainNumber field in the RESPONSE should be a UInteger8 = 0

Record:

Conclusion: Expected results verified => places DUT back in domain 0

1.F.21

Entry state: PC Clock: slave domain 1, DUT: master domain 0

Action: Set the Domain of PC Clock to 0. Send the management message CURRENT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field CURRENT_DATA_SET should be received from DUT. The expected values of the data fields cannot be tested at this point since they depend on the quality of the synchronization.

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for CURRENT_DATA_SET 15.5.3.4.1

1.F.22

Entry state: PC Clock: slave domain 0, DUT: master domain 0

Action: Send the management message PARENT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field PARENT_DATA_SET should be received from DUT. The expected values are:

- parentPortIdentity.ClockIdentity Octer[8]; should be identical to clockIdentity recorded in 1.F.10 above.
- parentPortIdentity.portNumber UInteger16; if a single port device (most LXI devices will be) this should be = 0.
- PS (Boolean): probably FALSE (indicates that device is not computing statistics) TRUE is acceptable
- observedParentOffsetScaledLogVariance UInteger16: indeterminate
- observedParentClockPhaseChangeRate (Integer32): indeterminate
- grandmasterPriority1 (UInteger8): should be 12 (same as default data set as modified in 1.F.12)
- grandmasterClockQuality.clockClass UInteger8: should be identical to the clockQuality.clockClass recorded in 1.F.10 above
- grandmasterClockQuality.clockAccuracy Enumeration8 (UInteger8): should be identical to the clockQuality.clockAccuracy recorded in 1.F.10 above
- grandmasterClockQuality.offsetScaledLogVariance UInteger16: should be identical to the clockQuality.offsetScaledLogVariance recorded in 1.F.10 above
- grandmasterPriority2 (UInteger8): should be 45 (same as default data set as modified in 1.F.13)
- grandmasterIdentity(Octet[8]): same as clockIdentity value recorded in 1.F.10 above

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for PARENT_DATA_SET 15.5.3.5.1

1.F.23

Entry state: PC Clock: slave domain 0, DUT: master domain 0

Action: Send the management message TIME_PROPERTIES_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field TIME_PROPERTIES_DATA_SET should be received from DUT. The expected values should be as indicated in the record section. Note that expected values are for a device with clockClass = 248 or less frequently a device with clockClass 187 that has not been set to the PTP timescale or in contact with a traceable UTC source.

Record: Record:

- currentUtcOffset Integer16; should be 33 before 0 hours 1 January 2009 and 34 after 0 hours 1 January 2009. This value can change every six months- the potential update window for leap second changes.
- FTRA Boolean; normally FALSE
- TTRA Boolean: normally FALSE
- PTP Boolean: normally FALSE
- UTCV Boolean: normally FALSE but TRUE is acceptable if currentUtcOffset is as indicated in 'a'
- LI-59 Boolean: normally FALSE
- LI-61 Boolean: normally FALSE
- timeSource Enumeration8 (UInteger8): normally 0xA0 INTERNAL_OSCILLATOR see 7.6.2.6

Conclusion: Expected results verified => partially validates the GET/RESPONSE semantics Table 38 for TIME_PROPERTIES_DATA_SET 15.5.3.6.1 and the default values of 8.2.4

1.F.24

Entry state: PC Clock: slave, DUT: master

Action: Send the management message UTC_PROPERTIES with actionField SET and:

a- currentUtcOffset Integer: 29
b- UTCV Boolean: TRUE
c- LI-59 Boolean: TRUE
d- LI-61 Boolean: TRUE

Expected results: A management message with actionField = RESPONSE and a managementId field UTC_PROPERTIES should be received from DUT. The fields in the RESPONSE should be identical to 'a' through 'd'

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for UTC_PROPERTIES 15.5.3.6.2

1.F.25

Entry state: PC Clock: slave, DUT: master

Action: Send the management message UTC_PROPERTIES with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field UTC_PROPERTIES should be received from DUT. The fields in the RESPONSE should be identical to 'a' through 'd' in 1.F.24

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for UTC_PROPERTIES 15.5.3.6.2 (database semantics 8.2.4 verification partially satisfied)

1.F.26

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TRACEABILITY_PROPERTIES with actionField SET and:

a- FTRA Boolean: TRUE
b- TTRA Boolean: TRUE

Expected results: A management message with actionField = RESPONSE and a managementId field TRACEABILITY_PROPERTIES should be received from DUT. The fields in the RESPONSE should be identical to ‘a’ through ‘b’

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for TRACEABILITY_PROPERTIES 15.5.3.6.3

1.F.27

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TRACEABILITY_PROPERTIES with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field TRACEABILITY_PROPERTIES should be received from DUT. The fields in the RESPONSE should be identical to ‘a’ through ‘b’ in 1.F.26

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for TRACEABILITY_PROPERTIES 15.5.3.6.3 (database semantics 8.2.4 verification partially satisfied)

1.F.28

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIMESCALE_PROPERTIES with actionField SET and:

a- PTP Boolean: TRUE
b- timeSource Enumeration8 (UInteger8): 0x60 HAND_SET

Expected results: A management message with actionField = RESPONSE and a managementId field TIMESCALE_PROPERTIES should be received from DUT. The fields in the RESPONSE should be identical to ‘a’ through ‘b’

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for TIMESCALE_PROPERTIES 15.5.3.6.4

1.F.29

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIMESCALE_PROPERTIES with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field TIMESCALE_PROPERTIES should be received from DUT. The fields in the RESPONSE should be identical to ‘a’ through ‘b’ in 1.F.28

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for TIMESCALE_PROPERTIES 15.5.3.6.4 (database semantics 8.2.4 verification partially satisfied)

1.F.30

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME_PROPERTIES_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field TIME_PROPERTIES_DATA_SET should be received from DUT. The expected values should be as set in 1.F.24 to 1.F.29 namely:

- currentUtcOffset Integer16; 29
- FTRA Boolean; TRUE
- TTRA Boolean: TRUE
- PTP Boolean: TRUE
- UTCV Boolean: TRUE
- LI-59 Boolean: TRUE
- LI-61 Boolean: TRUE
- timeSource Enumeration8 (UInteger8): 0x60

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for TIME_PROPERTIES_DATA_SET 15.5.3.6.1 and the default values of 8.2.4

1.F.31

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field PORT_DATA_SET should be received from DUT. The fields in the RESPONSE should have the values indicated in the record section.

Record: Record:

a- portIdentity.ClockIdentity Octet[6]: same as clockIdentity in 1.F.10 'I' above
b- portIdentity.portNumber UInteger16: =1 for a single port LXI device
c- portState Enumeration8 (UInteger8): = 0x06 master
d- logMinDelayReqInterval Integer8: any integer in the range logSyncInterval (see 'h' below) to logSyncInterval +5
e- peerMeanPathDelay TimeInterval (Integer64): 0 (see 8.2.5.3.3)
f- logAnnounceInterval Integer8: 1
g- announceReceiptTimeout UInteger8: 3
h- logSyncInterval Integer8: 0
i- delayMechanism Enumeration8 (UInteger8): 0x01 E2E see 8.2.5.4.4
j- logMinPdelayReqInterval Integer8; normally 0
k- versionNumber UInteger4: 2

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for PORT_DATA_SET 15.5.3.7.1

1.F.32

Entry state: PC Clock: slave, DUT: master

Action: Send the management message LOG_ANNOUNCE_INTERVAL with actionField SET and logAnnounceInterval field (Integer8) = 2

Expected results: A management message with actionField = RESPONSE and a managementId field logAnnounceInterval should be received from DUT. The logAnnounceInterval field in the RESPONSE should be an Integer8 = 2.

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for LOG_ANNOUNCE_INTERVAL 15.5.3.7.2

1.F.33

Entry state: PC Clock: slave, DUT: master

Action: Send the management message LOG_ANNOUNCE_INTERVAL with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field LOG_ANNOUNCE_INTERVAL should be received from DUT. The logAnnounceInterval field in the RESPONSE should be an Integer8 = 2

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for LOG_ANNOUNCE_INTERVAL 15.5.3.7.2 (database semantics 8.2.5 verification partially satisfied)

1.F.34

Entry state: PC Clock: slave, DUT: master

Action: Send the management message ANNOUNCE_RECEIPT_TIMEOUT with actionField SET and announceReceiptTimeout field (UInteger8) = 4

Expected results: A management message with actionField = RESPONSE and a managementId field ANNOUNCE_RECEIPT_TIMEOUT should be received from DUT. The announceReceiptTimeout field in the RESPONSE should be an UInteger8 = 4.

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for ANNOUNCE_RECEIPT_TIMEOUT 15.5.3.7.3

1.F.35

Entry state: PC Clock: slave, DUT: master

Action: Send the management message ANNOUNCE_RECEIPT_TIMEOUT with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field ANNOUNCE_RECEIPT_TIMEOUT should be received from DUT. The announceReceiptTimeout field in the RESPONSE should be a UInteger8 = 4

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for ANNOUNCE_RECEIPT_TIMEOUT 15.5.3.7.3 (database semantics 8.2.5 verification partially satisfied)

1.F.36

Entry state: PC Clock: slave, DUT: master

Action: Send the management message LOG_SYNC_INTERVAL with actionField SET and logSyncInterval field (Integer8) = -1

Expected results: A management message with actionField = RESPONSE and a managementId field logSyncInterval should be received from DUT. The logSyncInterval field in the RESPONSE should be an Integer8 = -1.

Record:

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for LOG_SYNC_INTERVAL 15.5.3.7.4

1.F.37

Entry state: PC Clock: slave, DUT: master

Action: Send the management message LOG_SYNC_INTERVAL with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field LOG_SYNC_INTERVAL should be received from DUT. The logSyncInterval field in the RESPONSE should be an Integer8 = -1

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for LOG_SYNC_INTERVAL 15.5.3.7.4 (database semantics 8.2.5 verification partially satisfied)

1.F.38

Entry state: PC Clock: slave, DUT: master

Action: Send the management message DELAY_MECHANISM with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field DELAY_MECHANISM should be received from DUT. The delayMechanism field in the RESPONSE should be an Enumeration8 (UInteger8) = 0x01 E2E

> **Note:** Since the P2P mechanism is not a required part of the LXI Profile the use of this management message with actionField SET is not needed. If implemented the RESPONSE should be returned with an MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value DELAY_MECHANISM and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for DELAY_MECHANISM 15.5.3.7.5 (database semantics 8.2.5 verification partially satisfied)

1.F.39

Entry state: PC Clock: slave, DUT: master

Action: Send the management message LOG_MIN_PDELAY_REQ_INTERVAL with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field LOG_MIN_PDELAY_INTERVAL should be received from DUT. The logMinPdelayReqInterval field in the RESPONSE should be an Integer8 = 0

> **Note:** Since the P2P mechanism is not a required part of the LXI Profile the use of this management message with actionField SET is not needed. If implemented the RESPONSE should be returned with an MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value LOG_MIN_PDELAY_REQ_INTERVAL and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for LOG_MIN_PDELAY_REQ_INTERVAL 15.5.3.7.6 (database semantics 8.2.5 verification partially satisfied)

1.F.40

Entry state: PC Clock: slave, DUT: master

Action: Send the management message VERSION_NUMBER with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field VERSION_NUMBER should be received from DUT. The versionNumber field in the RESPONSE should be a UInteger8 = 2

Note: Since the multiple version support is not a required part of the LXI Profile the use of this management message with actionField SET is not needed. If implemented the RESPONSE should be returned with an MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value VERSION_NUMBER and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for VERSION_NUMBER 15.5.3.7.7 (database semantics 8.2.5 verification partially satisfied)

1.F.41

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field PORT_DATA_SET should be received from DUT. The fields in the RESPONSE should be as follows:

a- portIdentity.ClockIdentity Octet[6]: same as clockIdentity in 1.F.10 'l' above
b- portIdentity.portNumber UInteger16: =1 for a single port LXI device
c- portState Enumeration8 (UInteger8): = 0x06 master
d- logMinDelayReqInterval Integer8: any integer in the range logSyncInterval (see 'g' below) to logSyncInterval +5
e- peerMeanPathDelay TimeInterval (Integer64): 0 (see 8.2.5.3.3)
f- logAnnounceInterval Integer8: 2
g- announceReceiptTimeout UInteger8: 4
h- logSyncInterval Integer8: -1
i- delayMechanism Enumeration8 (UInteger8): 0x01 E2E see 8.2.5.4.4
j- logMinPdelayReqInterval Integer8; normally 0
k- versionNumber UInteger4: 2

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for PORT_DATA_SET 15.5.3.7.1 and the port database semantics 8.2.5

1.F.42

Entry state: PC Clock: slave, DUT: master

Action: Send the management message SLAVE_ONLY with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field SLAVE_ONLY should be received from DUT. The SO field in the RESPONSE should be a Boolean = FALSE

Note: Since the slave only clocks are not permitted by the LXI Profile the use of this management message with actionField SET is not needed. If implemented the RESPONSE should be returned with an MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value SLAVE_ONLY and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or NOT_SETABLE 0x0005 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for DELAY_MECHANISM 15.5.3.7.5 (database semantics 8.2.5 verification partially satisfied)

1.F.43

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME with actionField SET and currentTime field Timestamp:

a- UInteger48 secondsField = 0x 0011 0100 0010 or decimal 73,031,221,264 (about the year 4286)
b- UInteger32 nanosecondsField = 5200

Expected results: Expected results: A management message with actionField = RESPONSE and a managementId field TIME should be received from DUT. The currentTime field in the RESPONSE should be as indicated in the record section. Note that some clocks, e.g. a clockClass 6 GPS linked clock, may not accept a TIME/SET message. However any clock with clockClass >127 must based on the LXI Profile. If the device does not accept a TIME message then 1.F.43 and 1.F.44 are inappropriate. Suggest a warning to alert vendor to check that this the intended design.

Record: Record:

a- UInteger48 secondsField = 0x 0011 0100 0010 or decimal 73,031,221,264 to a value +20 greater (whatever seems reasonable for a turnaround time in the device and PC Clock code)
b- UInteger32 nanosecondsField = any value less than 10^9. (testing is not precise enough to verify this value)

Conclusion: Expected results verified => validates the SET/RESPONSE semantics Table 38 for TIME 15.5.3.2.1

1.F.44

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field TIME should be received from DUT. The currentTime field in the RESPONSE should be:

a- UInteger48 secondsField = 0x 0011 0100 0010 or decimal 73,031,221,264 to a value +40 greater (whatever seems reasonable for a turnaround time in the device and PC Clock code for 1.F.41 and 1.F.42)
b- UInteger32 nanosecondsField = any value less than 10^9. (testing is not precise enough to verify this value)

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics Table 38 for TIME 15.5.3.2.1

1.F.45

Entry state: PC Clock: slave, DUT: master

Action: Send the management message FAULT_LOG with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field FAULT_LOG should be received from DUT. The data field of the RESPONSE should be as defined in Table 47. If faults have NOT occurred then the UInteger16 numberOfFaultRecords field should be 0. If faults have been logged then:

a- UInteger16 numberOfFaultRecords field should be 1 or greater, and should be follwed by an array of records of type FaultRecord, see 5.3.10
b- Each fault record should be have an appropriate severityCode from Table 46, a valid timestamp, and appropriate textual descriptions per 5.3.10.
c- It is possible that a pad field is needed.

Record:

Conclusion: Expected results verified => validates the GET/RESPONSE semantics for FAULT_LOG 15.5.3.1.7

1.G Protocol and semantics (Table 38, 39, and 40, 15.5.2 & sub clauses) test of messages that normally use actionField COMMAND

1.G.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The management messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- priority1 and priority2 both = 128

Expected results: Setup for next steps

Conclusion: Continue to next step

1.G.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE Inignore the response if NOT_SUPPORTED. If the response is GENERAL_ERROR the validity of the remainder of this test sequence is questionable. The sequenceId field of the returned management message should be identical to the sequenceId field in the original RESET_NON_VOLATILE_STORAGE management message.

Record:

Conclusion: Expected results verified => Verifies sequenceId handling of 15.4.1.2 no further conclusion: semantics are tested later. Set up for 1.G.6

1.G.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. In either case ignore the response.

Record:

Conclusion: Expected results verified => no conclusion: semantics are tested later. However this should cause all data sets to contain their default values for the next series of steps.

1.G.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PRIORITY1 with actionField SET and priority1 field UInteger8 = 15

Expected results: A management message with actionField = RESPONSE and a managementId field PRIORITY1 should be received from DUT. The priority1 field in the RESPONSE should be a UInteger8 = 15. This should also place the DUT in master state and PC Clock in slave state although this is not germane to this test.

Record:

Conclusion: Expected results verified => set up for 1.G.6

1.G.5

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PRIORITY2 with actionField SET and priority2 field UInteger8 = 48

Expected results: A management message with actionField = RESPONSE and a managementId field PRIORITY2 should be received from DUT. The priority2 field in the RESPONSE should be a UInteger8 = 48

Record:

Conclusion: Expected results verified => set up for 1.G.6

1.G.6

Entry state: PC Clock: slave, DUT: master

Action: Send the management message DEFAULT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field DEFAULT_DATA_SET should be received from DUT. The expected values are indicated in the ‘record’ section of this step.

Record: Record:

a- TSC (Boolean): probably TRUE (indicates that device is a two step clock) (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

b- SO (Boolean): FALSE (per LXI profile slave only not allowed) (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

c- numberPorts UInteger16: typically = 1 for most instruments (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

d- priority1 (UInteger8): should be 15

e- clockQuality.clockClass UInteger8: should be 248 unless the device falls under RULE 2.9.3 of the LXI Profile (few will) in which case it should be 6, 7 or 127 (most likely 127 in the test environment) (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

f- clockQuality.clockAccuracy Enumeration8 (UInteger8): in the range 0x20 to 0x31 or else 0xFE should agree with device data sheet and Table 6 (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

g- clockQuality.offsetScaledLogVariance UInteger16: should be in the range 0x4435 to 0x72B6 or 17461 to 29366 corresponding to actual variances of 10^{-18} to 10^{-4} or deviations of 1ns to 10 ms respectively (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

h- priority2 (UInteger8): should be 48

i- clockIdentity(Octet[8]): implementation specific see 7.5.2.2.2 (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

j- domainNumber (UInteger8): should be 0 (same as recorded in 1.F step 1.F.9 if 1.F was executed successfully immediately prior to 1.G)

Conclusion: Expected results verified => set up for next step. Note that the test for reasonableness and/or correctness of these values was performed earlier in 1.F step 1.F.9.

1.G.7

Entry state: PC Clock: slave, DUT: master

Action: Send the management message SAVE_IN_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: An ACKNOWLEDGE message should be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value SAVE_IN_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE.

Record: Whether or not an error TLV was received.

Conclusion: Expected results verified => no conclusion: semantics are tested later

1.G.8

Entry state: PC Clock: slave, DUT: master

Action: Send the management message INITIALIZE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message should be returned.

Record:

Conclusion: Expected results verified => no conclusion: semantics are tested later. However this should cause all data sets to contain their default values for the next series of steps. The default values depend on the results of 1.G.7, e.g. whether they are the 'initialialization set' see 8.1.3, or the results of the save to non-volatile storage in 1.G.7.

1.G.9

Entry state: PC Clock: <unknown>, DUT: <unknown

Action: Send the management message DEFAULT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field DEFAULT_DATA_SET should be received from DUT. The expected values should be the same as recorded in 1.G.6 above with the possible exceptions indicated as follows:

a- TSC (Boolean): same as in 1.G.6
b- SO (Boolean): same as in 1.G.6
c- numberPorts UInteger16: same as in 1.G.6
d- priority1 (UInteger8): If an error TLV was received in 1.G.7 the value should be 15 else it should be 128.
e- clockQuality.clockClass UInteger8: same as in 1.G.6
f- clockQuality.clockAccuracy Enumeration8 (UInteger8): same as in 1.G.6
g- clockQuality.offsetScaledLogVariance UInteger16: same as in 1.G.6
h- priority2 (UInteger8): If an error TLV was received in 1.G.7 the value should be 48 else it should be 128.
i- clockIdentity(Octet[8]): same as in 1.G.6
j- domainNumber (UInteger8): same as in 1.G.6

Record:

Conclusion: Expected results verified => if an error TLV was received in 1.G.7 this validates the semantics of SAVE_IN_NON_VOLATILE_STORAGE 15.5.3.1.4 (at least for data in the default data base). If no error TLV was received it partially validates the semantics of INITIALIZE 15.5.3.1.6 in the presence of non-volatile storage

1.G.10

Entry state: PC Clock: <unknown>, DUT: <unknown

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE.

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step for 1.G.12

1.G.11

Entry state: PC Clock: <unknown>, DUT: <unknown

Action: Send the management message INITIALIZE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message should be returned.

Record:

Conclusion: Expected results verified => no conclusion: However 1.G.10 and 1.G.11 should cause all data sets to contain their initialization set, see 8.1.3, default values for the next series of steps.

1.G.12

Entry state: PC Clock: <unknown>, DUT: <unknown

Action: Send the management message DEFAULT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field DEFAULT_DATA_SET should be received from DUT. The expected values should be the same as recorded in 1.G.6 above with the exceptions indicated as follows:

a- TSC (Boolean): same as in 1.G.6
b- SO (Boolean): same as in 1.G.6
c- numberPorts UInteger16: same as in 1.G.6
d- priority1 (UInteger8): should be 128.
e- clockQuality.clockClass UInteger8: same as in 1.G.6
f- clockQuality.clockAccuracy Enumeration8 (UInteger8): same as in 1.G.6
g- clockQuality.offsetScaledLogVariance UInteger16: same as in 1.G.6
h- priority2 (UInteger8): it should be 128.
i- clockIdentity(Octet[8]): same as in 1.G.6
j- domainNumber (UInteger8): same as in 1.G.6

Record:

Conclusion: Expected results verified => if no error TLV was received in 1.G.10 this validates the semantics of RESET_NON_VOLATILE_STORAGE 15.5.3.1.4 (at least for data in the default data base). If no error TLV was received in 1.G.10, it validates the semantics of INITIALIZE 15.5.3.1.6 in the presence of non-volatile storage. If an error was received it validates the semantics of INITIALIZE 15.5.3.1.6 in the absence of non-volatile storage.

1.G.13

Entry state: PC Clock: <unknown>, DUT: <unknown

Action: Send the management message PRIORITY1 with actionField SET and priority1 field UInteger8 = 15

Expected results: A management message with actionField = RESPONSE and a managementId field PRIORITY1 should be received from DUT. The priority1 field in the RESPONSE should be a UInteger8 = 15.

Record:

Conclusion: Expected results verified => places DUT in master state. Announce messages should be received after 3 Announce intervals.

1.G.14

Entry state: PC Clock: slave, DUT: master

Action: Send the management message DISABLE_PORT with actionField COMMAND

Expected results: An ACKNOWLEDGE message may be returned with managementId DISABLE_PORT. There is some ambiguity on whether one should receive the acknowledge from a disabled port. Announce messages should cease being received from DUT (need to watch for at least 1 Announce interval)

Record:

Conclusion: Expected results verified => cessation of Announce messages validates semantics of DISABLE_PORT 15.5.3.2.4

1.G.15

Entry state: PC Clock: master, DUT: disabled

Action: Send the management message ENABLE_PORT with actionField COMMAND

Expected results: An ACKNOWLEDGE message should be returned with managementId ENABLE_PORT. Announce messages should resume being received from DUT (need to watch for at least 3 Announce interval)

Record:

Conclusion: Expected results verified => resumption of Announce messages validates semantics of ENABLE_PORT 15.5.3.2.3

1.G.16

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message FAULT_LOG with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field FAULT_LOG should be received from DUT. The expected results are:

a- The numberOfFaultRecords should be a UInteger16. Value is probably 0 unless the DUT has recorded a fault during testing. Definition of a fault is implementation specific.

b- The value of numberOfFaultRecords should indicate the number of following fields of type FaultRecord. If there are fault records the formats should match the definitions in 15.5.3.1.7.

If the device does not support a fault log:

a- This could be indicated by a return value of numberOfFaultRecords = 0, although this could also indicate that there simply have been no faults,

b- The return of a MANAGEMENT_ERROR_STATUS_TLV appended. The managementId field should be FAULT_LOG and the managementErrorId value should be either NOT_SUPPORTED of GENERAL_ERROR.

Record:

Conclusion: Expected results verified => If there are fault records validates the operation of the log in recording and FAULT_LOG 15.5.3.1.7

1.G.17

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message FAULT_LOG_RESET with actionField COMMAND

A management message with actionField = ACKNOWLEDGE and a managementId field FAULT_LOG_RESET should be received from DUT. If the device does not support a fault log, the return should have a MANAGEMENT_ERROR_STATUS_TLV appended. The managementId field should be FAULT_LOG_RESET and the managementErrorId value should be either NOT_SUPPORTED of GENERAL_ERROR.

Expected results:

Record:

Conclusion: Expected results verified => no conclusion

1.G.18

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message FAULT_LOG with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field FAULT_LOG should be received from DUT. The expected results are:

a- The numberOfFaultRecords should be a UInteger16. Value is should be 0.
b- There should be no fault records in the return.

If the device does not support a fault log:

a- This could be indicated by a return value of numberOfFaultRecords = 0, although this could also indicate that there simply have been no faults,
b- The return of a MANAGEMENT_ERROR_STATUS_TLV appended. The managementId field should be FAULT_LOG and the managementErrorId value should be either NOT_SUPPORTED of GENERAL_ERROR.

Record:

Conclusion: Expected results verified => if both 1.G.17 and 1.G.18 results are as expected this verifies the semantics of FAULT_LOG_RESET 15.5.3.1.8

1.H Disregard management messages where the managementId is undefined in the receiving clock, 15.5.2.3

1.H.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message to the DUT with managementId field = $FFFF_{16}$ (undefined in all implementations) with actionField COMMAND

Expected results: An ACKNOWLEDGE message should be received with a managementId field of $FFFF_{16}$ and with a MANAGEMENT_ERROR_STATUS TLV appended. The tlvType of the appended TLV should be MANAGEMENT_ERROR_STATUS. The TLVmanagementErrorId value of the appended TLV should be NOT_SUPPORTED 0x0006 or NO_SUCH_ID 0x0002 or GENERAL_ERROR 0xFFFE

Record:

Conclusion: Expected results verified => partially validates 15.5.2.3

1.H.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message to the DUT with managementId field = FFFF16 (undefined in all implementations) with actionField GET

Expected results: A RESPONSE message should be received with a managementId field of $FFFF_{16}$ and with a MANAGEMENT_ERROR_STATUS TLV appended. The tlvType of the appended TLV should be MANAGEMENT_ERROR_STATUS. The TLVmanagementErrorId value of the appended TLV should be NOT_SUPPORTED 0x0006 or NO_SUCH_ID 0x0002 or GENERAL_ERROR 0xFFFE

Expected results:

Record:

Conclusion: Expected results verified => partially validates 15.5.2.3

1.H.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message to the DUT with managementId field = $FFFF_{16}$ (undefined in all implementations) with actionField SET

Expected results: A RESPONSE message should be received with a managementId field of $FFFF_{16}$ and with a MANAGEMENT_ERROR_STATUS TLV appended. The tlvType of the appended TLV should be MANAGEMENT_ERROR_STATUS. The TLVmanagementErrorId value of the appended TLV should be NOT_SUPPORTED 0x0006 or NO_SUCH_ID 0x0002 or GENERAL_ERROR 0xFFFE

Expected results:

Record:

Conclusion: Expected results verified => completes the validation of 15.5.2.3

2. Best Master Clock Test

Identifier: xxx

Category: Mandatory, master and slave

Purpose:

The best master clock algorithm, BMC, is supposed to assure that there is always exactly one grandmaster in a system and that it is the 'best' clock in the system. The BMC also establishes a master-slave hierarchy with the root being the grandmaster.

For single port clocks this procedure tests:

- That the BMC attributes: priority1, clcokClass, clockAccuracy, offsetScaledLogVariance, priority2, and clockIdentity are all used correctly in the evaluation of the BMC
- That extremum values if any for these attributes are checked
- That the Announce_Receipt_Timeout_Expires event correctly reflects the value of Announce interval and the value of portDS.announceReceiptTimeout
- That the contents of the parent dataset is correct
- That the contents of the Announce messages are correct

The general procedure is to vary the Announce messages transmitted by the PC Clock to test each of the attributes and decision paths in the BMC. The results of each such check are observed in one of two ways:

- By query of the DUT using a management message to verify the contents of the appropriate data sets,
- By observing the presence and content or absence of Announce messages from the DUT

Prior to the test, the DUT is initialized to make sure the default values of attributes are present. These values are then read with a management message or entered from a data sheet into the PC Clock to be used in the various queries.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

2.A Initialization of BMC tests

2.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- a-The normal values for startingBoundaryHops = 0 and boundaryHops = 0 are used in sending management messages.

b- b-The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

2.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

2.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

2.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET.

Expected results: DUT should return a management message DEFAULT_DATA_SET with actionField RESPONSE.

Record: From the response record as:

- P1= priority1,
- CC=clockQuality.clockClass,
- CA=clockQuality.clockAccuracy,
- CV=clockQuality.offsetScaledLogVariance,
- P2= priority2,
- CI= clockIdentity.

NOTE: As an alternative this information could be entered into the PC Clock from a data sheet.

Conclusion: Expected results verified => Information for rest of test is now in PC Clock

2.B Test each decision of BMC with extremum values

2.B.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 0
c- clockQuality.clockAccuracy = 0
d- clockQuality.offsetScaledLogVariance = 0
e- priority2 = 0
f- clockIdentity = 0 (all bits in all octets)
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)

Expected results: Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

2.B.2

Entry state: PC Clock: master, DUT: slave or passive

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record: The portState

Conclusion: Expected results verified => validates 9.3.4 Figure 27 GMpriority1 minimum extreme

2.B.3

Entry state: PC Clock: master, DUT: slave or passive

Action: Send the management message PARENT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PARENT_DATA_SET with actionField RESPONSE.

a- If the portState recorded in 2.B.2 is 09 (hex) –slave then The fields in the RESPONSE should be that of the PC Clock (see 2.B.1):

1. parentPortIdentity.clockIdentity Octet[8]: 0
2. parentPortIdentity.portNumber UInteger16: 1

3. grandmasterPriority1 UInteger8: 0
4. grandmasterClockQuality.clockClass UInteger8: 0
5. grandmasterClockQualtiy.clockAccuracy Enumeration8 (UInteger8): 0
6. grandmasterClockQuality.offsetScaledLogVariance UInteger16: 0
7. grandmasterPriority2 UInteger8: 0
8. grandmasterIdentity Octet[8]: 0

b- If the portState recorded in 2.B.2 is 07 (hex) –passive then The fields in the RESPONSE should be that of the DUT (see2.A.4):

1. parentPortIdentity.clockIdentity Octet[8]: CI
2. parentPortIdentity.portNumber UInteger16: 1
3. grandmasterPriority1 UInteger8: P1
4. grandmasterClockQuality.clockClass UInteger8: CC
5. grandmasterClockQualtiy.clockAccuracy Enumeration8 (UInteger8): CA
6. grandmasterClockQuality.offsetScaledLogVariance UInteger16: CV
7. grandmasterPriority2 UInteger8: P2
8. grandmasterIdentity Octet[8]: CI

Record:

Conclusion: Expected results verified => validates operation of parent data set for case of table 16, 9.3.5 if portState is slave and table 15, 9.3.5 if portState is passive.

2.B.4

Entry state: PC Clock: Master, DUT: slave (or passive)

Action: Modify the default dataset of the PC Clock as follows: priority1 = 255

Expected results: The PC Clock should be come slave after 3 Announce intervals

Record:

Conclusion: Proceed to next step

2.B.5

Entry state: PC Clock: Slave, DUT: Master

Action: Send the management message PARENT_DATA_SET with actionField GET

Expected results: DUT should return a management message PARENT_DATA_SET with actionField RESPONSE. DUT should issue Announce messages.

a- The content of the BMC critical Announce message fields should be as follows (values from 2.A step 2.A.4):

1. grandmasterPriority1 UInteger8: P1
2. grandmasterClockQuality.clockClass UInteger8: CC
3. grandmasterClockQuality.clockAccuracy Enumeration8 (UInteger8): CA
4. grandmasterClockQuality.offsetScaledLogVariance UInteger16: CV
5. grandmasterPriority2 UInteger8: P2
6. grandmasterIdentity Octet[8]: CI
7. stepsRemoved UInteger16: 0

b- The fields in the RESPONSE should be that of the DUT (see2.A.4):

1. parentPortIdentity.clockIdentity Octet[8]: CI
2. parentPortIdentity.portNumber UInteger16: 1
3. grandmasterPriority1 UInteger8: P1
4. grandmasterClockQuality.clockClass UInteger8: CC

5. grandmasterClockQualtiy.clockAccuracy Enumeration8 (UInteger8): CA
6. grandmasterClockQuality.offsetScaledLogVariance UInteger16: CV
7. grandmasterPriority2 UInteger8: P2
8. grandmasterIdentity Octet[8]: CI

Record:

Conclusion: Expected results verified => validates 9.3.4 Figure 27 GMpriority1 maximum extreme. Validates operation of parent data set for case of table 13, 9.3.5

2.B.6

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 06 (hex) –master

Record:

Conclusion: Expected results verified => confirms 2.B.5 validation of 9.3.4 Figure 27 GMpriority1 maximum extreme.

2.B.7

Entry state: PC Clock: Slave, DUT: Master

Action: Modify the default dataset of the PC Clock as follows: priority1 = P1

Expected results: The PC Clock should be come master after 3 Announce intervals (by virtue of test on clockClass)

Record:

Conclusion: Expected results verified => validates 9.3.4 Figure 27 GM class minimum extreme

2.B.8

Entry state: PC Clock: master, DUT: slave or passive

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => confirms 2.B.7 validation of 9.3.4 Figure 27 GM class minimum extreme

2.B.9

Entry state: PC Clock: Master, DUT: slave (or passive)

Action: Modify the default dataset of the PC Clock as follows: clockClass = 255

Expected results: The PC Clock should be come slave after 3 Announce intervals and Announce messages from DUT should resume.

Record:

Conclusion: validates 9.3.4 Figure 27 GM class maximum extreme

2.B.10

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 06 (hex) –master

Record:

Conclusion: Expected results verified => confirms 2.B.9 validation of 9.3.4 Figure 27 GM class maximum extreme.

2.B.11

Entry state: PC Clock: Slave, DUT: Master

Action: Modify the default dataset of the PC Clock as follows: clockClass = CC

Expected results: The PC Clock should be come master after 3 Announce intervals (by virtue of test on clockAccuracy)

Record:

Conclusion: Expected results verified => validates 9.3.4 Figure 27 GM accuracy minimum extreme

2.B.12

Entry state: PC Clock: master, DUT: slave or passive

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => confirms 2.B.11 validation of 9.3.4 Figure 27 GM accuracy minimum extreme

2.B.13

Entry state: PC Clock: Master, DUT: slave (or passive)

Action: Modify the default dataset of the PC Clock as follows: clockAccuracy = 255

Expected results: The PC Clock should be come slave after 3 Announce intervals and Announce messages from DUT should resume.

Record:

Conclusion: validates 9.3.4 Figure 27 GM accuracy maximum extreme

2.B.14

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 06 (hex) –master

Record:

Conclusion: Expected results verified => confirms 2.B.13 validation of 9.3.4 Figure 27 GM accuracy maximum extreme.

2.B.15

Entry state: PC Clock: Slave, DUT: Master

Action: Modify the default dataset of the PC Clock as follows: clockAccuracy = CA

Expected results: The PC Clock should be come master after 3 Announce intervals (by virtue of test on offsetScaledLogVariance)

Record:

Conclusion: Expected results verified => validates 9.3.4 Figure 27 GM offsetScaledLogVariance minimum extreme

2.B.16

Entry state: PC Clock: master, DUT: slave or passive

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => confirms 2.B.15 validation of 9.3.4 Figure 27 GM offsetScaledLogVariance minimum extreme

2.B.17

Entry state: PC Clock: Master, DUT: slave (or passive)

Action: Modify the default dataset of the PC Clock as follows: offsetScaledLogVariance = FFFF (hex)

Expected results: The PC Clock should be come slave after 3 Announce intervals and Announce messages from DUT should resume.

Record:

Conclusion: validates 9.3.4 Figure 27 GM offsetScaledLogVariance maximum extreme

2.B.18

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 06 (hex) –master

Record:

Conclusion: Expected results verified => confirms 2.B.17 validation of 9.3.4 Figure 27 GM offsetScaledLogVariance maximum extreme.

2.B.19

Entry state: PC Clock: Slave, DUT: Master

Action: Modify the default dataset of the PC Clock as follows: offsetScaledLogVariance = CV

Expected results: The PC Clock should be come master after 3 Announce intervals (by virtue of test on priority2)

Record:

Conclusion: Expected results verified => validates 9.3.4 Figure 27 GM priority2 minimum extreme

2.B.20

Entry state: PC Clock: master, DUT: slave or passive

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => confirms 2.B.19 validation of 9.3.4 Figure 27 GM priority2 minimum extreme

2.B.21

Entry state: PC Clock: Master, DUT: slave (or passive)

Action: Modify the default dataset of the PC Clock as follows: priority2 = 255

Expected results: The PC Clock should be come slave after 3 Announce intervals and Announce messages from DUT should resume.

Record:

Conclusion: validates 9.3.4 Figure 27 GM priority2 maximum extreme

2.B.22

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 06 (hex) –master

Record:

Conclusion: Expected results verified => confirms 2.B.21 validation of 9.3.4 Figure 27 GM priority2 maximum extreme.

2.B.23

Entry state: PC Clock: Slave, DUT: Master

Action: Modify the default dataset of the PC Clock as follows: priority2 = P2

Expected results: The PC Clock should be come master after 3 Announce intervals (by virtue of test on identity)

Record:

Conclusion: Expected results verified => validates 9.3.4 Figure 27 GM identity minimum extreme

2.B.24

Entry state: PC Clock: master, DUT: slave or passive

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => confirms 2.B.23 validation of 9.3.4 Figure 27 GM identity minimum extreme

2.B.25

Entry state: PC Clock: Master, DUT: slave (or passive)

Action: Modify the default dataset of the PC Clock as follows: identity (Octet[6]) = all bits 1

Expected results: The PC Clock should be come slave after 3 Announce intervals and Announce messages from DUT should resume.

Record:

Conclusion: validates 9.3.4 Figure 27 GM identity maximum extreme

2.B.26

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 06 (hex) –master

Record:

Conclusion: Expected results verified => confirms 2.B.25validation of 9.3.4 Figure 27 GM identity maximum extreme.

3. Test BMC related timeout ANNOUNCE_RECEIPT_TIMEOUT_EVENT

Identifier: xxx

Category: Mandatory, master and slave

Purpose:

The test first examines the DUT in the slave state by causing the PC Clock to cease sending Announce messages to the DUT (by moving the PC Clock to a different domain) and observing how long it takes for the DUT to start sending Announce messages.
The test then examines the DUT in the listening state by first initializing it and then observing how long it takes to start Announce messages given that there are no other clocks in the domain.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

3.A Initialization of BMC related timeout on Announce messages

3.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- a-The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.

b- b-The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

3.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

3.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

3.B

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 0
c- domainNumber = 0

Expected results: DUT should become slave or passive after at most 3 Announce intervals, PC Clock should become master.

Record:

Conclusion: Expected results verified => set up is complete for test of ANNOUNCE_RECEIPT_TIMEOUT in DUT in slave (or passive) state, see Figure 23, 9.2.5

3.B.1

Entry state: PC Clock: master, DUT: slave or passive

Action: Change the PC Clock to domainNumber 1 (or otherwise cause it to stop sending Announce messages)

Expected results: PC Clock Announce messages now in domain 1, Eventually DUT should become master

Record: The time TSTART to the value of the time in the PC Clock when the domain was changed.

Conclusion: Expected results verified => no conclusion, continue to next step

3.B.2

Entry state: PC Clock: master, DUT: slave or passive

Action: Wait until an Announce message is received from the DUT in domain 0.

Expected results: No Announce messages should be received from DUT for 3 Announce intervals. When an Announce message is received note the time as TEND

Record: TEND

Conclusion: Expected results verified => OK to continue

3.B.3

Entry state: PC Clock: master in domain 1, DUT: master in domain 0

Action: Compute TEND−TSTART.

Expected results: TEND−TSTART should be greater than 3 Announce intervals.

Record:

Conclusion: Expected results verified => validates ANNOUNCE_RECEIPT_TIMEOUT_EVENT for a DUT in slave state Figure 23 9.2.5 and 9.2.6.11.

3.B.4

Entry state: PC Clock: master in domain 1, DUT: master in domain 0

Action: Send the management message INITIALIZE with actionField CMD. NOTE THIS MUST BE SENT IN domain 0. (Alternatively the PC Clock can be placed in domain 0 PROVIDED it does NOT send out announce messages even though it is temporarily in the master state)

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record: The time TSTART that the initialize message was sent.

Conclusion: Expected results verified => OK to continue

3.B.5

Entry state: PC Clock: master in domain 1, DUT: listening in domain 0

Action: Wait until an Announce message is received from the DUT in domain 0.

Expected results: DUT should start issuing Announce messages after the timeout.

Record: The time TEND that the Announce message was received

Conclusion: Expected results verified => OK to continue

3.B.6

Entry state: PC Clock: master in domain 1, DUT: master in domain 0

Action: Compute TEND−TSTART.

Expected results: TEND−TSTART should be greater than 3 Announce intervals.

Record: The logMessageInterval field of the Announce message received from the DUT as DUT-AI

Conclusion: Expected results verified => validates ANNOUNCE_RECEIPT_TIMEOUT_EVENT for a DUT in listening state Figure 23 9.2.5 and 9.2.6.11.

3.B.7

Entry state: PC Clock: master in domain 1, DUT: master in domain 0

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 0
c- Announce interval = 2 (i.e. 4 seconds) assuming DUT-AI = 1 the default value.

Expected results: Announce interval of PC Clock should be 4 seconds in domain 1.

Record:

Conclusion: Expected results verified => set up is complete for test of foreign master qualification, 9.3.2.4.4 and portions of 9.3.2.5

3.B.8

Entry state: PC Clock: master in domain 1, DUT: master in domain 0

Action: Set the default dataset of the PC Clock attributes as follows:

- Domain = 0

Expected results: The PC Clock should stay master after moving to domain 0 and send Announce messages every 4 seconds. The DUT should remain master since it should disqualify Announce messages unless at least 2 are received within 4 of the DUT's Announce intervals, e.g. 2 Announce messages within 4 seconds. Observe the DUT for at least 12 seconds after moving the PC Clock to domain 0.

Record:

Conclusion: Expected results verified => If after 12 seconds the DUT is still sending Announce messages this verifies the qualification of foreign masters, 9.3.2.4.4 and portions of 9.3.2.5.

4. Clock ignores irrelevant messages

Identifier: xxx

Category: Mandatory, master and slave

Purpose:

There are several sections of IEEE 1588 that specify conditions under which a clock disregards all or part of a received PTP message. This test covers the following of these:

- versionPTP: Discard versons >2 18.1, translate version 1 optional 18.2(not required by LXI).
- domainNumber: Discard messages from a different domain, 9.5.1
- alternateMasterFlag: Discard messages with this flag TRUE, 9.1 and LXI Profile 2.10.1, 9.3.2.2 for Announce,
- Delay_Resp message not from current master or not associated with a Delay_Req from the slave clock: Discard, 9.5.7
- Follow_Up message not from current master or not associated with a Sync from the master: Disdard, 9.5.5.
- When NOT in MASTER state: discard Delay_Req 9.5.6
- When in DISABLED state; discard and not send Announce 9.5.3, SYNC 9.5.4, Follow_Up 9.5.5, Delay_Req 9.5.6, Delay_Resp 9.5.7
- TLV undefined in the receiving clock: disregard the TLV except for length, 14.1, 14.3.2.3

Note that several of these circumstances have been tested or partially in other parts of this suite specifically:

- Devices in the DISABLED state, clause 9.2.5, tested for not sending Announce in test 1.G.13
- Management messages addressed to a different port, fully tested, test 1 A.
- Management messages with a disallowed actionField, fully tested, 15.5.2.3 by tests 1.C, 1.D, and 1.E
- managementId undefined in the receiving clock: disregard and send error TLV, 15.5.2.3, test 1

The following are requirements of IEEE 1588 but are too difficult to test and should be vendor certified (by code inspection):

- When in INITIALIZING, or FAULTY state; discard and not send Announce 9.5.3, SYNC 9.5.4, Follow_Up 9.5.5, Delay_Req 9.5.6, Delay_Resp 9.5.7 (no way to place DUT in faulty state and initialize is a short transient state with no visibility for easy testing)

The general procedure is to:

Prior to the test, the DUT is initialized to make sure the default values of attributes are present. These values are then read with a management message or entered from a data sheet into the PC Clock to be used in the various queries.

Test configuration:

Configuration 2

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
4.A Initialization of relevant message tests

4.A.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a-The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b-The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup step

Record:

Conclusion: Continue to next step

4.A.2

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to both the DUT and Device 2

Expected results: This step is a set up step. An ACKNOWLEDGE messages may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

4.A.3

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to both the DUT and Device 2

Expected results: DUT and Device 2 should each return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue (DUT should have default initialization set parameters)

4.B Test discard of versionPTP not equal 2.

4.B.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET to the Device 2.

Expected results: Device 2 should return a management message DEFAULT_DATA_SET with actionField RESPONSE.

Record: clockIdentity field as device2 identity

Conclusion: Expected results verified => continue to next step

4.B.2

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set Device 2 to:

a- priority1 = 45: management message mangagementId PRIORITY1, actionField SET, priority1 field 45
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)
d- Time of T: management message managementId TIME, actionField SET, time field T

Set PC Clock:

a- priority1 = 0,
b- domain = 0,
c- versionNumber = 3.
d- Time of T+1hour.

Expected results: DUT should become a slave to Device 2 in domain 0. PC Clock should be master in domain 0. Expect Announce messages to be issued by both PC Clock and Device 2 and Delay_Req messages by the DUT. versionPTP fields should be for Device 2 = 2, PC Clock = 3, and DUT = 2.

Record:

Conclusion: Expected results verified => presence of messages partially verifies discard messages (Announce) with incorrect versionPTP fields18.2 Wait 1 minute for the DUT to synchronize before proceeding

4.B.3 Test for rejecting messages with versionPTP >2

Entry state: Device 2-V2: master in domain 0, PC Clock- V3: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message PORT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Announce) with incorrect versionPTP fields18.2

4.B.4

Entry state: Device 2-V2: master in domain 0, PC Clock-V3: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message PARENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PARENT_DATA_SET with actionField RESPONSE. The parentPortIdentity field should match the value of device2 identity from 4.B step 4.B.1.

Record:

Expected results: Expected results verified => partially verifies discard messages (Announce) with incorrect versionPTP fields18.2

4.B.5

Entry state: Device 2-V2: master in domain 0, PC Clock-V3: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the Device2.

Expected results: Device 2 should return a management message TIME with actionField RESPONSE.

Record: Value of currentTime as device2 time

Conclusion: Expected results verified => Execute the next step as soon as possible.

4.B.6

Entry state: Device 2-V2: master in domain 0, PC Clock-V3: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the DUT.

Expected results: DUT should return a management message TIME with actionField RESPONSE. The currentTime field should agree to the value of device2 time from 4.B step 4.B.5 to within 10 s.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Sync) with incorrect versionPTP fields18.2 (will not test discard Delay_Resp- too difficult)

4.B.7

Entry state: Device 2-V2: master in domain 0, PC Clock-V3: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the DUT- BUT from the PC Clock so that the versionPTP field is 3.

Expected results: DUT should NOT return a management message.

Record:

Conclusion: Expected results verified => partially verifies discard messages (management) with incorrect versionPTP fields18.2. This completes the discard incorrect versionPTP for versions >2 EXCEPT for Delay_Req which will be tested in 4.F.

4.B.8 Continue test with versionPTP = 1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set PC Clock: versionNumber = 1.

Expected results: DUT should remain a slave to Device 2 in domain 0. PC Clock should be master in domain 0. Expect Announce messages to be issued by both Device 2 and the PC Clock and Delay_Req messages by the DUT. versionPTP fields should be for Device 2 = 2, PC Clock = 1, and DUT = 2.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Announce) with incorrect versionPTP fields18.2 Wait 1 minute for the DUT to synchronize before going to next step.

4.B.9

Entry state: Device 2-V2: master in domain 0, PC Clock- V1: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message PORT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Announce) with incorrect versionPTP fields18.2

4.B.10

Entry state: Device 2-V2: master in domain 0, PC Clock-V1: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message PARENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PARENT_DATA_SET with actionField RESPONSE. The parentPortIdentity field should match the value of device2 identity from 4.B step 4.B.1.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Announce) with incorrect versionPTP fields18.2

4.B.11

Entry state: Device 2-V2: master in domain 0, PC Clock-V3: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the Device2.

Expected results: Device 2 should return a management message TIME with actionField RESPONSE.

Record: Value of currentTime as device2 time

Conclusion: Expected results verified => Execute the next step as soon as possible.

4.B.12

Entry state: Device 2-V2: master in domain 0, PC Clock-V1: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the DUT.

Expected results: DUT should return a management message TIME with actionField RESPONSE. The currentTime field should agree to the value of device2 time from 4.B step 4.B.11 to within 10 s.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Sync) with incorrect versionPTP fields18.2 (will not test discard Delay_Resp- too difficult)

4.B.13

Entry state: Device 2-V2: master in domain 0, PC Clock-V1: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the DUT- BUT from the PC Clock so that the versionPTP field is 1.

Expected results: DUT should NOT return a management message.

Record:

Conclusion: Expected results verified => Verifies discard messages (management) with incorrect versionPTP fields18.2. This completes the discard incorrect versionPTP for version = 1 EXCEPT for Delay_Req which will be tested in 3 E

4.C Discard messages with alternateMaster flag TRUE, 9.1 and LXI Profile 2.10.1, 9.3.2.2 for Announce

4.C.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set Device 2 to:

a- priority1 = 45: management message mangagementId PRIORITY1, actionField SET, priority1 field 45
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)
d- Time of T: management message managementId TIME, actionField SET, time field T

Set PC Clock:

a- priority1 = 0,
b- domain = 0,
c- versionNumber = 2.
d- Time of T+1hour.
e- ALTERNATE_MASTER flag TRUE

Expected results: DUT should become a slave to Device 2 in domain 0. PC Clock should be master in domain 0. Expect Announce messages to be issued by both Deive 2 and the PC Clock and Delay_Req messages by the DUT. ALTERNATE_MASTER flag fields should be for Device 2 = FALSE, PC Clock = TRUE, and DUT = FALSE.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Announce) with alternateMaster flag TRUE, 9.1 and LXI Profile 2.10.1, 9.3.2.2 for Announce Wait 1 minute for the DUT to synchronize before going to next step.

4.C.2

Entry state: Device 2: master in domain 0, PC Clock: alternate master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message PORT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be 09 (hex) –slave if CC>127 and 07 (hex) passive if CC<128. No Announce messages should be received from DUT.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Announce) with alternateMaster flag TRUE, 9.1 and LXI Profile 2.10.1, 9.3.2.2 for Announce

4.C.3

Entry state: Device 2: master in domain 0, PC Clock: alternate master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message PARENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PARENT_DATA_SET with actionField RESPONSE. The parentPortIdentity field should match the value of device2 identity from 4.B step 4.B.1.

Record:

Conclusion: Expected results verified => partially verifies discard messages (Announce) with alternateMaster flag TRUE, 9.1 and LXI Profile 2.10.1, 9.3.2.2 for Announce

4.C.4

Entry state: Device 2-V2: master in domain 0, PC Clock-V3: master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the Device2.

Expected results: Device 2 should return a management message TIME with actionField RESPONSE.

Record: Value of currentTime as device2 time

Conclusion: Expected results verified => Execute the next step as soon as possible.

4.C.5

Entry state: Device 2: master in domain 0, PC Clock: alternate master in domain 0, DUT: slave or passive in domain 0

Action: Send the management message TIME with actionField GET to the DUT.

Expected results: DUT should return a management message TIME with actionField RESPONSE. The currentTime field should agree to the value of currentTime from 4.C step 4.C.4 within 10 s.

Record:

Conclusion: Expected results verified => partially verifies discard messages with alternateMaster flag TRUE, 9.1 and LXI Profile 2.10.1, (will not test discard Delay_Resp- too difficult) This completes the discard incorrect alternate master flag.

4.D Delay_Resp message not from current master or not associated with a Delay_Req from the slave clock: Discard, 9.5.7

4.D.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set Device 2 to:

a- priority1 = 128: management message mangagementId PRIORITY1, actionField SET, priority1 field 128
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

a- priority1 = 0,
b- domain = 0,
c- versionNumber = 2.
d- ALTERNATE_MASTER flag FALSE

Expected results: DUT and Device 2 should both become a slave to PC Clock in domain 0. PC Clock should be master in domain 0. Expect Delay_Req messages to be issued by both Device 2 and the DUT. Allow 3 Announce intervals + for DUT and Device 2 to synchronize.

Record:

Conclusion: Expected results verified => OK to continue

4.D.2

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE.

Record: meanPathDelay as normalPathDelay

Conclusion: Expected results verified => repeat 4.D.2 until normalPathDelay is stable to within ± 10 ms. When stable proceed to 4.D.3

4.D.3

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Modify PC Clock Delay_Resp messages SENT TO DEVICE 2 ONLY by setting the delayRespModification field for messages to DEVICE 2 to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000. Delay_Resp messages to DUT should NOT be modified by this field. Repeat on all Delay_Resp messages through 4.D.4

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.D.3 until at least 2 Delay_Resp messages to Device 2 have been sent with the modification. Then proceed to 4.D.4.

4.D.4

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Delay_Resp messages that are a response to a clock other than the DUT are being discarded.

Record:

Conclusion: Expected results verified => Verification that DUT ignores Delay_Resp messages sent in response to a Delay_Req from a different clock (Device 2). A change in meanPathDelay indicates the DUT is NOT discarding irrelevant Delay_Resp messages, which is an error.

4.D.5

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set Device 2 to:

a- priority1 = 0: management message mangagementId PRIORITY1, actionField SET, priority1 field 0

b- domain = 0, (should be default from 4.A)

c- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

a- priority1 = 45,
b- domain = 0,
c- versionNumber = 2.
d- ALTERNATE_MASTER flag FALSE
e- Set to IGNORE Announce, Sync, Follow_Up from Device 2 and to surpress sending Announce, Sync, Follow_Up, and Delay_Resp messages (this ensures that even though Device 2 has a lower priority1 than the PC Clock, the PC Clock will be in the master state- but without sending normal master messages)

Expected results: DUT should become a slave to Device 2 in domain 0. PC Clock should be master in domain 0 but surpresses all master messages. Expect Delay_Req messages to be issued by the DUT. Allow 3 Announce intervals + for DUT to synchronize to Device 2.

Record:

Conclusion: Expected results verified => OK to continue

4.D.6

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync, Follow_Up and Delay_Resp messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. NOTE: The following steps check that the DUT in the slave state ignores Delay_Resp messages that are not from its current master, or if from the current master have the same sequenceId as the last DUT issued Delay_Req message. This is tested by having a Delay_Resp message sent from the PC Clock IN ADDITION to the one sent from the true master Device 2. It is possible that the DUT simply accepts the first Delay_Resp received after sending a Delay_Req message without actually checking the sourcePortIdentity and sequenceId fields. For implementations making this mistake the order in which the actual and the test probe Delay_Resp messages are received could influence the test results. Therefore the script is written assuming that the sending order can be controlled. Thus 4.D.7and 4.D.8 test the AFTER case and 4.D.9 and 4.D.10 test the BEFORE case.

Record: meanPathDelay as normalPathDelay

Conclusion: Expected results verified => repeat 4.D.6 until normalPathDelay is stable to within ± 10 ms. When stable proceed to 4.D.7

4.D.7

Entry state: Device 2: master PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Modify PC Clock to surpress Announce, Sync, Follow_Up but do send Delay_Resp messages in response to Delay_Req messages received from DUT.

a. Modify the delayRespModification field for messages to the DUT to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.
b. Modify the sequenceNumber field of the Delay_Resp messages so that instead of matching the sequenceNumber of the associated Delay_Req message, the sequenceNumber of the Delay_Resp message is +256 greater than the sequenceNumber of the associated Delay_Req message. The associated Delay_Req message is the last one received from the DUT.
c. Modify the sourcePortIdentity field of the Delay_Resp messages to be the identity of Device 2 rather than PC Clock (so that the message appears to be coming from Device 2)

d. Ensure that these Delay_Resp messages are sent shortly AFTER the Delay_Resp messages from Device 2 are sent.

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.D.7 until at least 2 Delay_Resp messages to DUT have been sent with the modification. Then proceed to next step.

4.D.8

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Delay_Resp messages with irrelevant sequenceId values are being discarded.

Record:

Conclusion: Expected results verified => Partial verification that DUT ignores Delay_Resp messages sent from a clock with incorrect sequnceIds. A change in meanPathDelay indicates the DUT is NOT discarding irrelevant Delay_Resp messages, which is an error.

4.D.9

Entry state: Device 2: master PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Modify PC Clock to surpress Announce, Sync, Follow_Up but do send Delay_Resp messages in response to Delay_Req messages received from DUT.

e. Modify the delayRespModification field for messages to the DUT to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.
f. Modify the sequenceNumber field of the Delay_Resp messages so that instead of matching the sequenceNumber of the associated Delay_Req message, the sequenceNumber of the Delay_Resp message is +256 greater than the sequenceNumber of the associated Delay_Req message. The associated Delay_Req message is the last one received from the DUT.
g. Modify the sourcePortIdentity field of the Delay_Resp messages to be the identity of Device 2 rather than PC Clock (so that the message appears to be coming from Device 2)
h. Ensure that these Delay_Resp messages are sent shortly BEFORE the Delay_Resp messages from Device 2 are sent.

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.D.7 until at least 2 Delay_Resp messages to DUT have been sent with the modification. Then proceed to next step.

4.D.10

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Delay_Resp messages with irrelevant sequenceId values are being discarded.

Record:

Conclusion: Expected results verified => Verification that DUT ignores Delay_Resp messages sent from a clock with incorrect sequnceIds. A change in meanPathDelay indicates the DUT is NOT discarding irrelevant Delay_Resp messages, which is an error.

4.D.11

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Modify PC Clock to surpress Announce, Sync, Follow_Up but do send Delay_Resp messages in response to Delay_Req messages received from DUT.

a. Modify the delayRespModification field for messages to the DUT to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.
b. The sequenceNumber field of the Delay_Resp messages should be that of the last Delay_Req message event though this message was in fact sent to Device 2 by the DUT.
c. The sourcePortIdentity field of the Delay_Resp messages should be that of the PC Clock.
d. Ensure that these Delay_Resp messages are sent shortly AFTER the Delay_Resp messages from Device 2 are sent.

Action:

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.D.11 until at least 2 Delay_Resp messages to DUT have been sent with the modification. Then proceed to next step.

4.D.12

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Delay_Resp messages from a clock other than its master are being discarded.

Record:

Conclusion: Expected results verified => Partial verification that DUT ignores Delay_Resp messages sent from a clock with correct sequnceIds but not from the current master. A change in meanPathDelay indicates the DUT is NOT discarding irrelevant Delay_Resp messages, which is an error.

4.D.13

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Modify PC Clock to surpress Announce, Sync, Follow_Up but do send Delay_Resp messages in response to Delay_Req messages received from DUT.

a. Modify the delayRespModification field for messages to the DUT to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.
b. The sequenceNumber field of the Delay_Resp messages should be that of the last Delay_Req message event though this message was in fact sent to Device 2 by the DUT.
c. The sourcePortIdentity field of the Delay_Resp messages should be that of the PC Clock.
d. Ensure that these Delay_Resp messages are sent shortly BEFORE the Delay_Resp messages from Device 2 are sent.

Action:

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.D.13 until at least 2 Delay_Resp messages to DUT have been sent with the modification. Then proceed to next step.

4.D.14

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Delay_Resp messages from a clock other than its master are being discarded.

Record:

Conclusion: Expected results verified => Verification that DUT ignores Delay_Resp messages sent from a clock with correct sequnceIds but not from the current master. A change in meanPathDelay indicates the DUT is NOT discarding irrelevant Delay_Resp messages, which is an error.

4.E Follow_Up message not from current master or not associated with a Sync message from the current master: Discard, 9.5.5

4.E.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Device 2 to:

a- priority1 = 0: management message mangagementId PRIORITY1, actionField SET, priority1 field 0
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

a- priority1 = 0,
b- domain = 45,
c- versionNumber = 2.
d- ALTERNATE_MASTER flag FALSE
e- Set to IGNORE Announce, Sync, Follow_Up from Device 2 and to surpress sending Announce, Sync, Follow_Up, and Delay_Resp messages. (this ensures that even though Device 2 has a lower priority1 than the PC Clock, the PC Clock will be in the master state- but without sending normal master messages)

Expected results: DUT should become a slave to Device 2 in domain 0. PC Clock should be master in domain 0 but surpresses all master messages. Expect Delay_Req messages to be issued by the DUT. Allow 3 Announce intervals + for DUT to synchronize.

Record:

Conclusion: Expected results verified => OK to continue

4.E.2

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE.

Record: meanPathDelay as normalPathDelay

Conclusion: Expected results verified => repeat 4.E.2 until normalPathDelay is stable to within ± 10 ms. When stable proceed to 4.E.3

4.E.3

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Modify PC Clock to send ONLY Follow_Up messages.

a- The sourcePortIdentity field should be that of the PC Clock as expected.

b- The sequenceId field value should be that of the last Sync message sent by Device 2.

c- Modify the followUpModification of these Follow_Up messages to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.

d- These Follow_Up messages shall be sent AFTER the normal Follow_Up message is sent from Device 2. Repeat on all Follow_Up messages through 4.E.4

e-

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.E.3 until at least 2 Follow_Up messages to Device 2 have been sent with the modification. Then proceed to 4.E.4. NOTE: The following steps check that the DUT in the slave state ignores Follow_Up messages that are not from its current master, or if from the current master have the same sequenceId as the last master issued Sync message. This is tested by having a Follow_Up message sent from the PC Clock IN ADDITION to the one sent from the true master Device 2. It is possible that the DUT simply accepts the first Follow_Up received after receiving a Sync message without actually checking the sourcePortIdentity and sequenceId fields. For implementations making this mistake the order in which the actual and the test probe Follow_Up messages are received could influence the test results. Therefore the script is written assuming that the sending order can be controlled. Thus 4.E.3 and 4.E.4 test the AFTER case and 4.E.5 and 4.E.6 test the BEFORE case.

4.E.4

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Follow_Up messages from a clock other than its master are being discarded based on sourcePortIdentity.

Record:

Conclusion: Expected results verified => Partial verification that DUT ignores Follow_Up messages from a clock other than its master (Device 2)

4.E.5

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Modify PC Clock to send ONLY Follow_Up messages.

a- The sourcePortIdentity field should be that of the PC Clock as expected.
b- The sequenceId field value should be that of the last Sync message sent by Device 2.
c- Modify the followUpModification of these Follow_Up messages to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.
d- These Follow_Up messages shall be sent BEFORE the normal Follow_Up message is sent from Device 2. Repeat on all Follow_Up messages through 4.E.6

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.E.5 until at least 2 Follow_Up messages to Device 2 have been sent with the modification. Then proceed to 4.E.6.

4.E.6

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Follow_Up messages from a clock other than its master are being discarded based on sourcePortIdentity.

Record:

Conclusion: Expected results verified => Verification that DUT ignores Follow_Up messages from a clock other than its master (Device 2)

4.E.7

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Device 2 to:

d- priority1 = 0: management message mangagementId PRIORITY1, actionField SET, priority1 field 0
e- domain = 0, (should be default from 4.A)
f- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

f- priority1 = 45,
g- domain = 0,
h- versionNumber = 2.
i- ALTERNATE_MASTER flag FALSE
j- Set to IGNORE Announce, Sync, Follow_Up from Device 2 and to surpress sending Announce, Sync, Follow_Up, and Delay_Resp messages. (this ensures that even though Device 2 has a lower pri-

ority1 than the PC Clock, the PC Clock will be in the master state- but without sending normal master messages)

Expected results: DUT should become a slave to Device 2 in domain 0. PC Clock should be master in domain 0 but surpresses all master messages. Expect Delay_Req messages to be issued by the DUT. Allow 3 Announce intervals + for DUT to synchronize.

Record:

Conclusion: Expected results verified => OK to continue
Conclusion:

4.E.8

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed),, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE.

Record: meanPathDelay as normalPathDelay

Conclusion: Expected results verified => repeat 4.E.8 until normalPathDelay is stable to within ± 10 ms. When stable proceed to 4.E.9

4.E.9

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed),, DUT: slave

Action: Modify PC Clock to:

a) Send only Follow_Up messages to the DUT.
b) Modify the followUpModification field for Follow_Up messages to the DUT to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.
c) The sourcePortIdentity field should be that of Device 2.
d) Modify the sequenceNumber field of the Follow_Up messages so that instead of matching the sequenceNumber of the last Sync message sent by Device 2, the sequenceNumber of the Follow_Up message is +256 greater than the sequenceNumber of the last Sync message sent by Device 2.
e) These Follow_Up messages shall be sent AFTER the normal Follow_Up message is sent from Device 2. Repeat on all Follow_Up messages through 4.E.10

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.E.9 until at least 2 Delay_Resp messages to DUT have been sent with the modification. Then proceed to next step.

4.E.10

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Follow_Up messages from its master are being discarded based on sequenceId values not matching an associated Sync message.

Record:

Conclusion: Expected results verified => Partial verification that DUT ignores Follow_Up messages not associated with Sync messages based on sequenceId.

4.E.11

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed),, DUT: slave

Action: Modify PC Clock to:

a) Send only Follow_Up messages to the DUT.
b) Modify the followUpModification field for Follow_Up messages to the DUT to a value of 2^{+25} ns, (~33.6 ms) i.e. 0x 0000 0200 0000 0000.
c) The sourcePortIdentity field should be that of Device 2.
d) Modify the sequenceNumber field of the Follow_Up messages so that instead of matching the sequenceNumber of the last Sync message sent by Device 2, the sequenceNumber of the Follow_Up message is +256 greater than the sequenceNumber of the last Sync message sent by Device 2.
e) These Follow_Up messages shall be sent BEFORE the normal Follow_Up message is sent from Device 2. Repeat on all Follow_Up messages through 4.E.12

Expected results:

Record:

Conclusion: Expected results verified => repeat 4.E.11 until at least 2 Delay_Resp messages to DUT have been sent with the modification. Then proceed to next step.

4.E.12

Entry state: Device 2: master, PC Clock: master (but with Announce, Sync and Follow_Up messages surpressed), DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The meanPathDelay field should be within ± 10 ms of the value normalPathDelay indicating that Follow_Up messages from its master are being discarded based on sequenceId values not matching an associated Sync message.

Record:

Conclusion: Expected results verified => Verification that DUT ignores Follow_Up messages not associated with Sync messages based on sequenceId.

4.F When NOT in MASTER state: discard Delay_Req 9.5.6

4.F.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set Device 2 to:

a- priority1 = 128: management message mangagementId PRIORITY1, actionField SET, priority1 field 128
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

a- priority1 = 0,
b- domain = 0,
c- versionNumber = 2.

Expected results: DUT and Device 2 should both become a slave to PC Clock in domain 0. PC Clock should be master in domain 0. Wait 3 Announce intervals

Record:

Conclusion: Expected results verified => OK to continue

4.F.2

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Observe messages from DUT for a minimum of 32 Sync intervals

Expected results: DUT should never issue a Delay_Resp message. Note that this is an indirect test for discarding Delay_Req messages sent from Device 2 to the master PC Clock.

Record:

Conclusion: Expected results verified => validates discarding of Delay_Req, 9.5.6

4.G When in DISABLED state; discard and not send Announce 9.5.3, SYNC 9.5.4, Follow_Up 9.5.5, Delay_Req 9.5.6, Delay_Resp 9.5.7

4.G.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set Device 2 to:

a- priority1 = 128: management message mangagementId PRIORITY1, actionField SET, priority1 field 128
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

a- priority1 = 0,
b- domain = 0,
c- versionNumber = 2.

Expected results: DUT and Device 2 should both become a slave to PC Clock in domain 0. PC Clock should be master in domain 0. Wait 3 Announce intervals

Record:

Conclusion: Expected results verified => OK to continue

4.G.2

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Send the management message PORT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE.

Record: The value of the portIdentity field as DUT port identity

Conclusion: Expected results verified => Continue to 4.G.3

4.G.3

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Send the management message DISABLE_PORT with actionField COMMAND to the DUT

Expected results: An ACKNOWLEDGE message may be returned with managementId DISABLE_PORT. There is some ambiguity on whether one should receive the acknowledge from a disabled port.

Record:

Conclusion: Expected results verified => go on to next step

4.G.4

Entry state: Device 2: slave, PC Clock: master, DUT: disabled

Action: Observe messages from DUT for a minimum of 32 Sync intervals

Expected results: DUT should never issue any message. In particular observe the DUT for Announce, Sync, Follow_Up, Delay_Req, and Delay_Resp messages. Messages FROM the DUT will have the value of sourcePortIdentity in the message header equal to the value DUT port identity from 4.G.2

Record:

Conclusion: Expected results verified => validates discard and not send of Announce 9.5.3, SYNC 9.5.4, Follow_Up 9.5.5, Delay_Req 9.5.6, Delay_Resp 9.5.7

4.G.5

Entry state: PC Clock: master, DUT: disabled

Action: Send the management message ENABLE_PORT with actionField COMMAND

Expected results: An ACKNOWLEDGE message should be returned with managementId ENABLE_PORT.

Record:

Conclusion: Expected results verified => this is a cleanup step.

4.H Disregard TLVs undefined in the receiving clock except for the length, 14.1, 14.3.2.3

4.H.1

Entry state: Device 2: <unknown>, PC Clock:<unknown>, DUT: <unknown>

Action: Set Device 2 to:

a- priority1 = 255: management message mangagementId PRIORITY1, actionFieId SET, priority1 field 255
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

a- priority1 = 0,
b- domain = 0,
c- versionNumber = 2.

Expected results: DUT and Device 2 should both become a slave to PC Clock in domain 0. PC Clock should be master in domain 0. Allow 3 Announce intervals + for DUT to synchronize.

Record:

Conclusion: Expected results verified => OK to continue

4.H.2

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE.

Record: meanPathDelay as normalPathDelay and offsetFromMaster as normalOffsetFromMaster

Conclusion: Expected results verified => repeat this step until normalPathDelay and normalOffsetFromMaster are both stable to within ± 10 ms. When stable proceed to next step

4.H.3

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Set the PC Clock so that it appends the NOT_DEFINED TLV with no data (see LXI IEEE 1588 Profile) to all transmitted messages

Expected results: Allow 1 minute for any changes in DUT synchronization or state

Record:

Conclusion: Expected results verified => OK to continue

4.H.4

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The value of the meanPathDelay field is within ± 10 ms of the value of normalPathDelay and the value of the offsetFromMaster field is within ± 10 ms of the normalOffsetFromMaster from 4.H.2

Record:

Conclusion: Expected results verified => partially validates ignoring undefined TLVs (appended to Announce, Sync, and Delay_Resp messages from PC Clcok)

4.H.5

Entry state: Device 2: slave, PC Clock: master, DUT: slave

Action: Set Device 2 to:

a- priority1 = 255: management message mangagementId PRIORITY1, actionField SET, priority1 field 255
b- domain = 0, (should be default from 4.A)
c- versionNumber = 2. (should be default from 4.A)

Set PC Clock:

a- priority1 = 255,
b- domain = 0,
c- versionNumber = 2.

Set PC Clock so that it appends the NOT_DEFINED TLV with no data (see LXI IEEE 1588 Profile) to all transmitted messages.

Expected results: Allow 1 minute for any changes in synchronization and state

Record:

Conclusion: Expected results verified => OK to continue

4.H.6

Entry state: Device 2: slave, PC Clock: slave, DUT: master

Action: Observe the meanPathDelay and offsetFromMaster values in the PC Clock

Expected results: DUT should be master and should send Announce, Sync, and Delay_Resp messages. The value of the meanPathDelay of PC Clock is within ± 10 ms of the value of normalPathDelay and the value of the offsetFromMaster from PC Clock is within ± 10 ms of the normalOffsetFromMaster from 4.H.2

Record:

Conclusion: Expected results verified => validates ignoring (except for length) undefined TLVs, 14.1, 14.3.2.3 appended to Delay_Req messages from PC Clock

5. Slaves honor the Delay_Req inter-message interval

Identifier: xxx

Category: Mandatory slave

Purpose:

In its Delay_Resp messages, the master tells slaves how frequently slaves are allowed to send Delay_Req messages. The master sends the minimum average inter-message time (strictly, the $\log_2$ of this time) in the Delay_Resp messages. Slaves may send slower than that, but not faster. Slaves are required to make a fresh random choice of time for each message, while honoring this minimum. If they choose to set their average to the minimum allowed, the interval over which they choose random inter-message times is [0, 2*t*] seconds, where *t* is the time published by the master. 9.5.11.2

This test is conducted for two values of the expected intervals to verify that the slave correctly responds to different values.

Note that a slave can use any start time and interval that it wants before it hears its first Delay_Resp message.

Note: This test is not applicable to devices with clockClass <128 (since these will never be in the slave state but rather the passive state under similar circumstances).

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

5.A Initialization of test of Delay_Req inter-message interval

5.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.

b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

5.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

5.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

5.B Test of Delay_Req inter-message interval

5.B.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- logSyncInterval = 0 (interval of 1 s) (default value in LXI 1588 profile)
j- portDS.logMinDelayReqInterval = logSyncInterval +1 = 1 (interval of 2 s)

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

5.B.2

Entry state: PC Clock: master, DUT: slave

Action: Examine the logInterMessageInterval of Delay_Resp messages issued by the PC Clock.

Expected results: logInterMessageIntervalof Delay_Resp should = 1 under test conditions. Wait until at least 2 Delay_Req messages have been transmitted by DUT before proceeding to next step.

Record: $2^{\text{logInterMessageInterval}}$ as meanDRQInterval

Conclusion: Expected results verified => continue to next step

5.B.3

Entry state: PC Clock: master, DUT: slave or passive

Action: Record the time of receipt at the PC Clock of Delay_Req messages from DUT. Record for 51 messages (about 100 s). Compute the following:

a. Compute the inter-message intervals from this data.
b. Compute the autocorrelation of the inter-message intervals
c. Using R do a Ljung-Box test to verify that the autocorrelation is nearly 0 (indicating that the intervals are random)
d. Compute the mean value of the inter-message intervals. This should NOT be significantly less than meanDRQInterval from previous step.
e. Compute the number of intervals > 2 x meanDRQInterval. There should be very few that exceed this value, say <10%
f. Run a Kolmogorov-Smirnov test against a uniform distribution to see if the distribution of inter-message intervals is uniform. Test should fail to reject the hypothesis.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => partially validates the correct intervals between Delay_Req messages from a slave, 9.5.11.2

5.B.4

Entry state: PC Clock: master, DUT: slave

Action: Send the management message LOG_SYNC_INTERVAL with actionField SET and a value of logSyncInterval = -1.

Expected results: The DUT should now expect Sync messages at the new interval

Record:

Conclusion: Expected results verified => continue to next step

5.B.5

Entry state: PC Clock: master, DUT: slave

Action: Modify the following default dataset of the PC Clock attributes as indicated:

a- logSyncInterval = -1 (interval of 0.5 s)
b- portDS.logMinDelayReqInterval = logSyncInterval +1 = 0 (interval of 1 s)

Expected results: The PC Clock should send Sync messages every 0.5 s.

Record:

Conclusion: Expected results verified => continue to next step

5.B.6

Entry state: PC Clock: master, DUT: slave

Action: Examine the logInterMessageInterval of Delay_Resp messages issued by the PC Clock.

Expected results: logInterMessageIntervalof Delay_Resp should = 0 under test conditions. Wait until at least 2 Delay_Req messages have been transmitted by DUT before proceeding to next step.

Record: $2^{\text{logInterMessageInterval}}$ as meanDRQInterval

Conclusion: Expected results verified => continue to next step

5.B.7

Entry state: PC Clock: master, DUT: slave or passive

Action: Record the time of receipt at the PC Clock of Delay_Req messages from DUT. Record for 51 messages (about 50 s). Compute the following:

a. Compute the inter-message intervals from this data.
b. Compute the autocorrelation of the inter-message intervals

c. Using R do a Ljung-Box test to verify that the autocorrelation is nearly 0 (indicating that the intervals are random)
d. Compute the mean value of the inter-message intervals. This should NOT be significantly less than meanDRQInterval from previous step.
e. Compute the number of intervals > 2 x meanDRQInterval. There should be very few that exceed this value, say <10%
f. Run a Kolmogorov-Smirnov test against a uniform distribution to see if the distribution of inter-message intervals is uniform. Test should fail to reject the hypothesis.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the correct intervals between Delay_Req messages from a slave, 9.5.11.2

6. Clock meets timing constraints

Identifier: xxx

Category: Mandatory master

Purpose:

This test checks timing requirements on Announce and Sync messages from a clock in the master state outlined in section 7.7 and 9.5 of IEEE 1588.

The tests will be conducted in the following order:

1. Test Announce intervals. 9.5.8 A node shall, with 90% confidence, issue messages with intervals within +/- 30% of the value of the interval computed from portDS.logAnnounceInterval.
2. Test Sync intervals. 9.5.9.2 A node shall, with 90% confidence, issue messages with intervals within +/- 30% of the value of the interval computed from portDS.logSyncInterval.
3. Test accuracy of originTimestamp in Sync messages. 9.5.9.4 0 or no worse than 1 second from actual time of master.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
6.A Initialization of test of timing tests

6.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

6.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

6.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

6.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 255
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- logSyncInterval = 0 (interval of 1 s) (default value in LXI 1588 profile)
j- portDS.logMinDelayReqInterval = logSyncInterval = 0 (interval of 1 s)

Expected results: The DUT should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

6.A.5

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE.

Record: Record as:

a- LA = logAnnounceInterval field (should be the default = 1, i.e. 2 s)
b- LS = logSyncInterval (should be default normally 0, i.e. 1s, but permitted any integer from -4 to +1)

Conclusion: Expected results verified => verifies default values from LXI 1588 Profile (Sync) and J.3.2 (Announce)

6.B Test of Announce and Sync inter-message interval

6.B.1

Entry state: PC Clock: slave, DUT: master

Action: Record the time of receipt at the PC Clock of Announce messages from DUT. Record for 51 messages (about 100 s). Compute the following:

a. Compute the inter-message intervals from this data.
b. Compute the number of inter-message intervals within ± 30 % of the value of 2^{+LA}. This number should be >90% of the computed intervals.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the correct intervals between Announce messages from a master, 9.5.8

6.B.2

Entry state: PC Clock: slave, DUT: master

Action: Record the time of receipt at the PC Clock of Sync messages from DUT. Record for 51 messages (about 100 s). Compute the following:

a. Compute the inter-message intervals from this data.
b. Compute the number of inter-message intervals within ± 30 % of the value of 2^{+LS}. This number should be >90% of the computed intervals.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the correct intervals between Sync messages from a master, 9.5.9.2

6.C Test of Sync originTimestamp field for one-step DUTs

6.C.1

Entry state: PC Clock: slave, DUT: master

Action: Observe the value of offsetFromMaster in the PC Clock for at least 10 sync intervals.

Expected results: The observed offsetFromMaster values should be stable and within the claimed synchronization accuracy of the DUT.

Record:

Conclusion: Expected results verified => The clocks are sufficiently well synchronized to continue this test.

6.C.2

Entry state: PC Clock: slave, DUT: master

Action: The PC Clock records the originTimestamp of the Sync messages and the time of receipt at the PC Clock of Sync messages from DUT. Record for 10 messages (about 20 s). Compute the following:

a. Compute the differences between the received timestamp and the originTimestamp field of each Sync message.
b. Verify that all these differences are less than ± 1 s.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the originTimestamp field of Sync messages from a master, 9.5.9.3.

6.D Test of Sync originTimestamp field for two-step DUTs

6.D.1

Entry state: PC Clock: slave, DUT: master

Action: Observe the value of offsetFromMaster in the PC Clock for at least 10 sync intervals.

Expected results: The observed offsetFromMaster values should be stable and within the claimed synchronization accuracy of the DUT.

Record:

Conclusion: Expected results verified => The clocks are sufficiently well synchronized to continue this test.

6.D.2

Entry state: PC Clock: slave, DUT: master

Action: The PC Clock record the originTimestamp of Sync messages and the time of receipt at the PC Clock of Sync messages from DUT. It also records the preciseOriginTimestamp of the corresponding Follow_Up messages from the DUT. Record for 10 messages (about 20 s). Compute the following:

a. Verify that either:
   i. The originTimestamps are all 0, or
   ii. All differences between the received timestamp and the originTimestamp for each message are less than ± 1 s.
b. Verify that the differences between the received timestamps for each Sync message and the preciseOriginTimestamp in the associated Follow_Up message are less than ± 1 s.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the originTimestamp field of Sync messages from a master, 9.5.9.4

7. DUT uses correctionField correctly

Identifier: xxx

Category: Mandatory master and slave

Purpose:

The timing messages in IEEE 1588 version 2 contain a field named correctionField. These fields contain modifications of the timestamp fields and must be correctly used by a slave clock in order to properly synchronize to its parent clock. Likewise a master clock must ensure that the combination of master transmitted timestamps and correctionFields correctly indicates the time intended.

The correctionField in any of the applicable messages enters into the computation of both the offsetFromMaster and the meanPathDelay at the slave. The tests use the computation of the value of meanPathDelay as the basis for verification since this does not require the use of external measurments of offset, e.g. a 1 PPS signal. The basic technique is to first determine the nominal value of meanPathDelay and then to perturb the system by modification of various correctionFields and observing the effect on the meanPathDelay. The test values used are on the order of 60 ms. The effects these test values produce in the value of meanPathDelay computed by the device in the slave state should be greater- and therefore easily visible- than the normal value of meanPathDelay and the jitter in this value for any reasonable implementation of PTP.

Vendor certification:

Sections 9.5.9.3 and 9.5.10 require that a device in the master state ensure that the sum of a correctionField and either the originTimestamp or preciseOriginTimestamp actually reflect the exact value of the time a Sync message was transmitted. The only way to verify this is by a very precise measurement of the offsets between the master and slave, correcting for all other problems. This is beyond the scope of these conformance tests and must be verified by the vendor.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

7.A Initialization of test of timing tests

7.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- a-The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.

b- b-The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

7.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

7.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

7.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

7.A.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The value of the meanPathDelay should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks

Note: If problems are observed visual examination of timestamp and correctionFields in Sync and Follow_Up messages is warranted.

Note the meanPathDelay field in this management message is of type Integer64 and is in ns times 2^{+16}

Record: Record as:

a- normalDelay = meanPathDelay field of RESPONSE

Conclusion: Expected results verified => repeat this step until the value of meanPathDelay is stable, then continue to next step.

7.B Test of correctionField in Sync messages

7.B.1

Entry state: PC Clock: master, DUT: slave

Action: Set the value of the syncModification field in the PC Clock to a value of 2^{+26} ns, (~67.1 ms), i.e. 0x 0000 0400 0000 0000

Expected results: This modifies the correctionField in Sync messages and should result in an offset of ~33.6 ms in the normalDelay computed by the slave.

Record:

Conclusion: Expected results verified => allow 10 Sync messages before proceeding to next step.

7.B.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The compute the value of :

normalDelay − meanPathDelay = delayPerturbation

The value of delayPerturbation should be 2^{+25} ns (~33.6 ms), i.e. 0x 0000 0200 0000 0000 to within the worst case stability error of the system, (approximately 2^{+23} ns (~8 ms), i.e. 0x 0000 0080 0000 0000 for a software implementation)

Record:

Conclusion: Expected results verified => verifies Sync correctionField semantics in slave, 11.3.2

7.C Test of correctionField in Follow_Up messages

7.C.1

Entry state: PC Clock: master, DUT: slave

Action: Set the value of the followUpModification field in the PC Clock to a value of 2^{+26} ns, (~67.1 ms), i.e. 0x 0000 0400 0000 0000. Set the value of syncModification to 0.

Expected results: This modifies the correctionField in Follow_Up messages and should result in an offset of ~33.6 ms in the normalDelay computed by the slave.

Record:

Conclusion: Expected results verified => allow 10 Follow_Up messages before proceeding to next step.

7.C.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The compute the value of :

normalDelay − meanPathDelay = delayPerturbation

The value of delayPerturbation should be 2^{+25} ns (~33.6 ms), i.e. 0x 0000 0200 0000 0000 to within the worst case stability error of the system, (approximately 2^{+23} ns (~8 ms), i.e. 0x 0000 0080 0000 0000 for a software implementation)

Record:

Conclusion: Expected results verified => verifies Follow_Up correctionField semantics in slave, 11.3.2

7.D Test of correctionField in Delay_Resp messages

7.D.1

Entry state: PC Clock: master, DUT: slave

Action: Set the value of the delayRespModification field in the PC Clock to a value of 2^{+26} ns, (~67.1 ms), i.e. 0x 0000 0400 0000 0000. Set the value of followUpModification to 0.

Expected results: This modifies the correctionField in Delay_Resp messages and should result in an offset of ~33.6 ms in the normalDelay computed by the slave.

Record:

Conclusion: Expected results verified => allow 10 Delay_Resp messages before proceeding to next step. Note that depending on the slave this may take up to 300 Sync messages or around 5 minutes.

7.D.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The compute the value of :

normalDelay − meanPathDelay = delayPerturbation

The value of delayPerturbation should be 2^{+25} ns (~33.6 ms), i.e. 0x 0000 0200 0000 0000 to within the worst case stability error of the system, (approximately 2^{+23} ns (~8 ms), i.e. 0x 0000 0080 0000 0000 for a software implementation)

Record:

Conclusion: Expected results verified => verifies Delay_Resp correctionField semantics in slave, 11.3.2

7.E Initialization of correctionField test in Delay_Req messages

7.E.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 255
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0

Expected results: The PC Clock should be come slave after 3 Announce intervals

Record:

Conclusion: Proceed to next step

7.E.2

Entry state: PC Clock: slave, DUT: master

Action: Set to 0 all modification fields in the PC Clock, i.e. syncModification, followUpModification, delayRespModification, and delayReqModification.

Expected results: Observe the computed value of meanPathDelay in the PC Clock. The value of the meanPathDelay should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks

Record: Record as:

a- normalDelay = computed meanPathDelay of PC Clock

Conclusion: Expected results verified => repeat this step until the value of meanPathDelay is stable, then continue to next step.

7.F Test of correctionField in DelayReq messages

7.F.1

Entry state: PC Clock: slave, DUT: master

Action: Set the value of the delayReqModification field in the PC Clock to a value of 2^{+26} ns, (~67.1 ms), i.e. 0x 0000 0400 0000 0000

Expected results: This modifies the correctionField in delayReq messages. The DUT adds this field from the correctionField in the returned Delay_Resp message to the slave, i.e. the PC Clock. This should result in an offset of ~33.6 ms in the normalDelay computed by the PC Clock.

Record:

Conclusion: Expected results verified => allow 10 Delay_Req messages before proceeding to next step. (this may take up to 5 minutes)

7.F.2

Entry state: PC Clock: master, DUT: slave

Action: Observe the computed value of meanPathDelay in the PC Clock.

Expected results: The compute the value of :

normalDelay − meanPathDelay = delayPerturbation

The value of delayPerturbation should be 2^{+25} ns (~33.6 ms), i.e. 0x 0000 0200 0000 0000 to within the worst case stability error of the system, (approximately 2^{+23} ns (~8 ms), i.e. 0x 0000 0080 0000 0000 for a software implementation)

Record:

Conclusion: Expected results verified => verifies Delay_Req correctionField semantics in master, 11.3.2

8. Clock synchronizes to both one-step and two-step masters

Identifier: xxx

Category: Mandatory

Purpose:

One-step clocks do not send Follow_Up messages but any clock, one- or two-step, must be able to synchronize to it. Two-step clocks do send Follow_Up messages and any clock, one- or two-step, must correctly use the Follow_Up message as part of synchronization, see 11.2.

The test is to verify that the DUT correctly synchronizes to both types of clocks. Specifically this requires:

- A DUT in the slave state correctly synchronize to a one-step master clock, and
- A DUT in the slave state correctly synchronize to a two-step master clock.

The first test has effectively been accomplished by the execution of test 7 in which a two-step clock is used and the measure of synchronization is the meanPathDelay having a reasonable value.

This test is specified in two ways:

- Using the meanPathDelay as a measure of synchronization, albeit indirect, and
- Using an external measurement of 1 PPS signals from the DUT and PC Clock. The disadvantage of this is the requirement for high quality 1 PPS signals and the use of exter-

nal devices such as scopes. The LXI standard requires the 1 PPS signal so the issue is the presence of the 1 PPS on the PC Clock- effectively prohibiting the use of a software PC based solution for the PC Clock for this test.

Test configuration:

Configuration 1, 3, 4 and 5

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
8.A Two-step test, path delay method

Test configuration:

Configuration 1

8.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- All modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

8.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

8.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

8.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

8.A.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Observe the value of the meanPathDelay field of the RESPONSE. The value of the meanPathDelay should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks

Record:

Conclusion: Expected results verified => indirectly verifies that the DUT synchronizes to a two-step clock.

8.B One-step test, path delay method

Test configuration:

Configuration 3

8.B.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- All modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

8.B.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

8.B.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

8.B.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

8.B.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Observe the value of the meanPathDelay field of the RESPONSE. The value of the meanPathDelay should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks

Record:

Conclusion: Expected results verified => indirectly verifies that the DUT synchronizes to a one-step clock.

8.C Two-step test, external 1 PPS measurement method

Test configuration:

Configuration 4

8.C.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- All modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

8.C.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

8.C.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

8.C.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

8.C.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Observe the value of the offsetFromMaster field of the RESPONSE. The value of the offsetFromMaster should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks. Observe the difference between the 1 PPS signals from the PC Clock and the DUT. This difference should be less than the offsetFromMaster when stable and averaged over several minutes.

Record:

Conclusion: Expected results verified => verifies that the DUT synchronizes to a two-step clock.

8.D One-step test, external 1 PPS measurement method

Test configuration:

Configuration 5

8.D.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.

b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

c- all modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

8.D.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

8.D.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

8.D.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

8.D.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Observe the value of the offsetFromMaster field of the RESPONSE. The value of the offsetFromMaster should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks. Observe the difference between the 1 PPS signals from the PC Clock and the DUT. This difference should be less than the offsetFromMaster when stable and averaged over several minutes.

Record:

Conclusion: Expected results verified => verifies that the DUT synchronizes to a one-step clock.

9. Clock honors V1 HW compatibility bit

Identifier: xxx

Category: Mandatory

Purpose:

Some devices designed under IEEE 1588-2002 but still usable with IEEE 1588-2008 use the packet length in deciding whether to timestamp packets. Annex D of IEEE 1588-2008 requires that clocks interacting with such a device respond correctly to the hardwareCompatibility bit in the transport specific fields of Announce, Sync, and Delay_Req messages.

The test first checks that a slave clock correctly responds to Announce and Sync messages with this bit set. The second part of the test checks that a master correctly responds to a Delay_Req with this bit set.

The verification is by:

- Ensuring that the devices synchronize as measured by the meanPathDelay metric, and
- By examining the formats of the affected messages, Sync and Delay_Req.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
9.A Initialization for test with DUT in slave state

9.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

9.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

9.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

9.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20

d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- hardwareCompatibility bit = FALSE

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

Conclusion: Proceed to next step

9.A.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Observe the value of the meanPathDelay field of the RESPONSE. The value of the meanPathDelay should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks

Record: Record the value of meanPathDelay as normalDelay

Conclusion: Expected results verified => Repeat this step until the value of the meanPathDelay is stable, then proceed to next steps.

9.B hardwareCompatibilty bit set test, DUT is slave

9.B.1

Entry state: PC Clock: master, DUT: slave

Action: Set PC Clock so that hardwareCompatibilty bit is TRUE

Expected results: Announce and Sync messages transmitted from the PC Clock should have the hardwareCompatibilty bit TRUE and the Sync message should be padded as specified in Annex D.

Record:

Conclusion: Wait for 3 Announce intervals then continue to next step

9.B.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Observe the value of the meanPathDelay field of the RESPONSE. Compute the value of normalDelay – meanPathDealy. This value should be 0 within the measurement accuracy of the devices, e.g. <50 ns if hardware assist, <1 ms if software clocks.

Record:

Conclusion: Expected results verified => partially verifies that the DUT correctly interprets the hardwareCompatibility bit.

9.B.3

Entry state: PC Clock: master, DUT: slave

Action: Observe the Delay_Req messages from the DUT using Wireshark.

Expected results: The Delay_Req messages should be padded as specified in Annex D. The hardwareCompatibility bit should be FALSE.

Record:

Conclusion: Expected results verified => partially verifies that the DUT correctly interprets the hardwareCompatibility bit.

9.B.4

Entry state: PC Clock: master, DUT: slave

Action: Set PC Clock so that hardwareCompatibiltiy bit is FALSE

Expected results: The PC Clock will set the hardwareCompatibilty bit in Announce and Sync messages to FALSE

Record: Start Wireshark capturing Announce, Sync, and Delay_Req messages before changing this bit.

Conclusion: Continue to next step.

9.B.5

Entry state: PC Clock: master, DUT: slave

Action: The DUT should continue to pad Delay_Req messages for 1 Announce interval after observing that the hardwareCompatibility bit in Announce or Sync messages received from the PC Clock is now set to FALSE.

Expected results: The DUT should revert to non-padded Delay_Req messages after 1 Announce interval following the last Announce or Sync message received from the PC Clock with the bit set to TRUE. Note that per a 21 November 2008 message from the IEEE 1588 interpretations committee, the timeout mechanism for the hardware compatibility bit of Annex D is flawed. The recommended workaround is to continue to pad messages even if the hardware compatibility bit is changed to FALSE until the next initialization of the node. Consequently if the results of the action in 9.B.4 do not result in the DUT ceasing to pad Delay_Req messages this is NOT a failure provided the results of 9.B.6 and 9.B.7 are satisfied.

Record:

Conclusion: Expected results verified => possibly completes the validation that the DUT correctly handles the hardwareCompatibilty bit as specified in Annex D when in the slave state.

9.B.6

Entry state: PC Clock: master, DUT: slave

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record: Start Wireshark capturing Announce, Sync, and Delay_Req messages to allow observation of the Delay_Req messages from the DUT before and after the initialization.

Conclusion: Expected results verified => OK to continue

9.B.7

Entry state: PC Clock: master, DUT: slave

Action: Observe the padding of Delay_Req messages issued by the DUT.

Expected results: The DUT should revert to non-padded Delay_Req messages after the initialization.

Record:

Conclusion: Expected results verified => completes the validation that the DUT correctly handles the hardwareCompatibilty bit as specified in Annex D when in the slave state.

9.C Initialization for test with DUT in master state

9.C.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

9.C.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

9.C.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

9.C.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 255
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- hardwareCompatibility bit = FALSE

Expected results: The PC Clock should be come slave after 3 Announce intervals

Record:

Conclusion: Proceed to next step

9.C.5

Entry state: PC Clock: slave, DUT: master

Action: Observe the value of meanPathDelay computed by the PC Clock

Expected results: The value of the meanPathDelay should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks

Record: Record the value of meanPathDelay as normalDelay

Conclusion: Expected results verified => Repeat this step until the value of the meanPathDelay is stable, then proceed to next steps.

9.D hardwareCompatibilty bit set test, DUT is master

9.D.1

Entry state: PC Clock: slave, DUT: master

Action: Set PC Clock so that hardwareCompatibilty bit is TRUE

Expected results: Delay_Req messages transmitted from the PC Clock should have the hardwareCompatibilty bit TRUE and should be padded as specified in Annex D.

Record:

Conclusion: Wait for 2 such Delay_Req messages to be transmitted by the PC Clock then continue to next step

9.D.2

Entry state: PC Clock: slave, DUT: master

Action: Observe the value of meanPathDelay computed by the PC Clock

Expected results: Compute the value of normalDelay – meanPathDealy. This value should be 0 within the measurement accuracy of the devices, e.g. <50 ns if hardware assist, <1 ms if software clocks.

Record:

Conclusion: Expected results verified => partially verifies that the DUT correctly interprets the hardwareCompatibility bit.

9.D.3

Entry state: PC Clock: slave, DUT: master

Action: Observe the Sync and Announce messages from the DUT using Wireshark.

Expected results: Both the Announce and Sync messages from the DUT should have the hardwareCompatibility bit FALSE. The Sync messages should be padded as specified in Annex D.

Record:

Conclusion: Expected results verified => partially verifies that the DUT correctly interprets the hardwareCompatibility bit.

9.D.4

Entry state: PC Clock: slave, DUT: master

Action: Set PC Clock so that hardwareCompatibiltiy bit is FALSE

Expected results: The PC Clock will set the hardwareCompatibilty bit in Delay_Req messages to FALSE

Record: Start Wireshark capturing Announce, Sync, and Delay_Req messages before changing this bit.

Conclusion: Continue to next step.

9.D.5

Entry state: PC Clock: slave, DUT: master

Action: The DUT should continue to pad Sync messages for 1 Announce interval after observing that the hardwareCompatibility bit in Delay_Req messages received from the PC Clock is now set to FALSE.

Expected results: The DUT should revert to non-padded Sync messages after 1 Announce interval following the last Delay_Req message received from the PC Clock with the bit set to TRUE. Note that per a 21 November 2008 message from the IEEE 1588 interpretations committee, the timeout mechanism for the hardware compatibility bit of Annex D is flawed. The recommended workaround is to continue to pad messages even if the hardware compatibility bit is changed to FALSE until the next initialization of the node. Consequently if the results of the action in 9.D.4 do not result in the DUT ceasing to pad Sync messages this is NOT a failure provided the results of 9.D.6 and 9.D.7 are satisfied.

Record:

Conclusion: Expected results verified => possibly completes the validation that the DUT correctly handles the hardwareCompatibilty bit as specified in Annex D when in the master state.

9.D.6

Entry state: PC Clock: slave, DUT: master

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record: Start Wireshark capturing Announce, Sync, and Delay_Req messages before issuing this command to allow observation of the Sync messages issued by the DUT before and after the initialization.

Conclusion: Expected results verified => OK to continue

9.D.7

Entry state: PC Clock: slave, DUT: master

Action: Observe the Sync messages issued by the DUT.

Expected results: The DUT should revert to non-padded Sync messages after the initialization of 9.D.6

Record:

Conclusion: Expected results verified => completes the validation that the DUT correctly handles the hardwareCompatibilty bit as specified in Annex D when in the master state.

10. Reject Rogue Frames

Identifier: xxx

Category: Mandatory

Purpose:

Subsection 9.3.2.5 of IEEE 1588-2008 says to ignore Announce messages which have traversed more than 255 boundary clocks — a preposterous depth for a tree. That requirement is intended to break loops.

The test checks that the DUT ignores such Announce messages.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

10.A Test for excess stepsRemoved

10.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

10.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

10.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

10.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0

b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- hardwareCompatibility bit = FALSE
j- stepsRemoved = 255

Expected results: The DUT should be come master after 3 Announce intervals since the Announce messages from the PC Clock should be ignored. The DUT should issue Announce and Sync messages.

Record:

Conclusion: Expected results verified => verifys that the DUT correctly handles stepsRemoved = 255 per 9.3.2.5

11. Protocol not affected by sequence number roll-over

Identifier: xxx

Category:

- Mandatory: testing the response of the DUT to sequence rollover in messages from the PC Clock, i.e. Announce, Sync, Follow_Up
- Vendor Declaration:
    - That the sequence numbers of messages originating from the DUT when in the master state, i.e. Announce, Sync, and Management, rollover at 2^{16}, not 2^{8} or 2^{32}.
    - That the sequence numbers of messages originating from the DUT when in the slave state, i.e. Delay_Req, rollover at 2^{16}, not 2^{8} or 2^{32}.

Purpose:

Subsection 7.3.7 of IEEE 1588-2008 requires separate sequenceId values for certain messages with rollover properties defined by the datatype UInteger16.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

11.A Setup

11.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.

b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

c- all modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

11.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

11.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

11.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- hardwareCompatibility bit = FALSE
j- stepsRemoved = 0

Expected results: The DUT should be come slave after 3 Announce intervals

Record:

Conclusion: Expected results verified => continue to next step

11.A.5

Entry state: PC Clock: master, DUT: slave, or passive

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The value of the meanPathDelay should be stable and reasonable, e.g. <50 ns if hardware assist, <1 ms if software clocks

Note the meanPathDelay field in this management message is of type Integer64 and is in ns times 2^{+16}

Record: Record as:

a- normalDelay = meanPathDelay field of RESPONSE

Conclusion: Expected results verified => repeat this step until the value of meanPathDelay is stable, then continue to next subtest.

11.B Test of sequenceId rollover handling in DUT for Announce messages

11.B.1

Entry state: PC Clock: master, DUT: slave

Action: Set the PC Clock for Announce message sequenceId to a value of 0xFFF0

Expected results: Successive Announce messages should have sequenceId values (hex): FFF0, FFF1, FFF2…FFFE, FFFF, 0000, 0001…

Record:

Conclusion: Expected results verified => continue to next step

11.B.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The compute the value of :

normalDelay − meanPathDelay = delayPerturbation

The value of delayPerturbation should be stable and less than ~10ms when observed over at least 20 Announce intervals (to ensure that the rollover has occurred and any nonconformant side effects observed.

Record:

Conclusion: Expected results verified => verifies correct rollover handling of Announce messages by the DUT

11.C Test of sequenceId rollover handling in DUT for Sync and Follow_Up messages

11.C.1

Entry state: PC Clock: master, DUT: slave

Action: Set the PC Clock for Sync (Follow_Up should happen automatically) message sequenceId to a value of 0xFFF0

Expected results: Successive Sync and Follow_Up messages should have sequenceId values (hex): FFF0, FFF1, FFF2…FFFE, FFFF, 0000, 0001…

Record:

Conclusion: Expected results verified => continue to next step

11.C.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. The compute the value of :

normalDelay − meanPathDelay = delayPerturbation

The value of delayPerturbation should be stable and less than ~10ms when observed over at least 20 Sync intervals (to ensure that the rollover has occurred and any nonconformant side effects observed.

Record:

Conclusion: Expected results verified => verifies correct rollover handling of Sync and Follow_Up messages by the DUT

12. Separate sequence number spaces maintained

Identifier: xxx

Category: Mandatory

Purpose:

Subsection 7.3.7 of IEEE 1588-2008 requires separate sequenceId number spaces for Sync, Delay_Req, Announce, Signaling and Management messages.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

12.A Setup

12.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.

b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s

c- all modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

12.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

12.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

12.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 255
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- hardwareCompatibility bit = FALSE
j- stepsRemoved = 0

Expected results: The DUT should be come master after 3 Announce intervals

Record:

Conclusion: Expected results verified => continue to next step

Conclusion:
Conclusion:

12.B Test of sequenceId handling in DUT for Announce, Sync, Follow_Up, Delay_Resp messages

12.B.1

Entry state: PC Clock: slave, DUT: master

Action: Observe sequenceId values for Announce, Sync, and Follow_Up messages from the DUT and Delay_Resp messages from the DUT as a result of the Delay_Req messages from the PC Clock. Observe at least 3 such pairs of Delay_Req Delay_Resp, at least 3 pairs of Sync and Follow_Up, and at least 3 Announce messages (this could take several minutes)

Expected results: Compare the sequenceId fields of these messages and test for the following properties:

- The sequenceId values for the Delay_Req messages from the PC Clock should match the resulting Delay_Resp messages from the DUT for each pair (information only-not part of this test)
- The sequenceId values for each Follow_Up should match the sequenceId value from the previous Sync message from the DUT. Furthermore the values should differ from those of the Delay_Resp messages from the DUT.
- The sequenceId field values of the Announce messages from the DUT should differ from those of the Sync and Delay_Resp messages from the DUT

Record: The value of the sequenceId from the last Announce, Sync, and Delay_Resp message issued while the DUT is master before being placed in the slave state in

Conclusion: Expected results verified => verifies 7.3.7 for the DUT in master state.

12.C Test of sequenceId handling in DUT for Delay_Req Delay_Resp message combination

12.C.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- hardwareCompatibility bit = FALSE
j- stepsRemoved = 0

Expected results: The PC Clock should be come master after 3 Announce intervals

Record:

12.C.2 Expected results verified => continue to next step

Entry state: PC Clock: master, DUT: slave

Action: Observe sequenceId values for Delay_Req messages from the DUT and the resulting Delay_Resp messages from the PC Clock. Observe at least 3 such pairs (this could take several minutes)

Expected results: The sequenceId values for the Delay_Resp messages from the PC Clock should match the resulting Delay_Req messages from the DUT. Further the sequenceId in the Delay_Req message should be significantly different from the values of sequenceIds for messages sent while the DUT was master as recorded in 12.B.1

Record:

Conclusion: Expected results verified => This shows that the PC Clock is correctly handling Delay_Req messages when in slave state

13. Max and min Sync message rate

Identifier: xxx
Category: Mandatory

Purpose:

This test verifies that the DUT as a slave can synchronize to a master over the range of Sync intervals required by the LXI 1588 Profile and that as a master it can service slaves over the required range of Sync intervals. The test exercises the required minimum and maximum Sync intervals and the recommended minimum interval supported by the DUT.

This test assumes that the PC Clock itself supports synchronization to sub microsecond levels.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

13.A Setup for maximum logSyncInterval test

13.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

13.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

13.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => OK to continue

13.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the default dataset of the PC Clock attributes as follows:

a- priority1= 0
b- clockQuality.clockClass = 248
c- clockQuality.clockAccuracy = 0x20
d- clockQuality.offsetScaledLogVariance = 0x4435
e- priority2 = 128
f- clockIdentity = 0 implementation specific see 7.5.2.2.2
g- portNumber = 1 (this should be the case anyway for a single port PC Clock)
h- domainNumber = 0
i- hardwareCompatibility bit = FALSE
j- stepsRemoved = 0

Set the logSyncInterval value of the port dataset of the PC Clock to +1 (the maximum required permitted value)

Expected results: The DUT should be come slave after 3 Announce intervals

Record:

Conclusion: Expected results verified => continue to next step

13.A.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message LOG_SYNC_INTERVAL with actionField SET to the DUT. The logSyncInterval value is set to +1.

Expected results: DUT should return a management message LOG_SYNC_INTERVAL with actionField RESPONSE and the logSyncInterval value should be +1.

Record:

Conclusion: Expected results verified => completes setup, continue to next step

13.B Test of maximum logSyncInterval for DUT

13.B.1

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Examine the offsetFromMaster field in the response. This value should be consistent with the vendors synchronization specification. Typically this value should definitely be less than 1 us for hardware implementations and 1 ms for software implementations.

Record: The value of offsetFromMaster field as nominalOffset.

Conclusion: Expected results verified => verifies synchronization for maximum logSyncInterval (minmum Sync interval) for the DUT as slave.

13.B.2

Entry state: PC Clock: master, DUT: slave

Action: Set the value of priority1 in the default dataset of the PC Clock to 255.

Expected results: The DUT should become master within 3 Announce intervals.

Record:

Conclusion: Expected results verified => continue to next step

13.B.3

Entry state: PC Clock: slave, DUT: master

Action: Examine the offsetFromMaster of the current dataset of the PC Clock.

Expected results: This value should be consistent with the vendors synchronization specification and the recorded value nominalOffset. Typically this value should definitely be less than 1 us for hardware implementations and 1 ms for software implementations.

Record:

Conclusion: Expected results verified => completes the verification of synchronization for maximum logSyncInterval for the DUT

13.B.4

Entry state: PC Clock: slave, DUT: master

Action: Record the time of receipt at the PC Clock of Sync messages from DUT. Record for 51 messages (about 100 s). Compute the following:

c. Compute the inter-message intervals from this data.
d. Compute the number of inter-message intervals within ± 30 % of the value of 2 seconds. This number should be >90% of the computed intervals.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the correct intervals between Sync messages from a master, 9.5.9.2

13.C Setup for minimum logSyncInterval test

13.C.1

Entry state: PC Clock: slave, DUT: master

Action: Set the logSyncInterval value of the port dataset of the PC Clock to -1 (the minimum required permitted value). Send the management message LOG_SYNC_INTERVAL with actionField SET to the DUT. The logSyncInterval value is set to -1.

Expected results: DUT should return a management message LOG_SYNC_INTERVAL with actionField RESPONSE and the logSyncInterval value should be -1.

Record:

Conclusion: Expected results verified => continue to next step

13.D Test for minimum logSyncInterval

13.D.1

Entry state: PC Clock: slave, DUT: master

Action: Examine the offsetFromMaster of the current dataset of the PC Clock. If necessary repeat this step until this value is stable (to allow for any transient resulting from change in Sync interval)

Expected results: This value should be consistent with the vendors synchronization specification and the recorded value nominalOffset from 13.B.1. Typically this value should definitely be less than 1 us for hardware implementations and 1 ms for software implementations.

Record:

Conclusion: Expected results verified => verification of synchronization for minimum logSyncInterval for the DUT as master

13.D.2

Entry state: PC Clock: slave, DUT: master

Action: Record the time of receipt at the PC Clock of Sync messages from DUT. Record for 51 messages (about 100 s). Compute the following:

e. Compute the inter-message intervals from this data.
f. Compute the number of inter-message intervals within ± 30 % of the value of 0.5 seconds. This number should be >90% of the computed intervals.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the correct intervals between Sync messages from a master, 9.5.9.2

13.D.3

Entry state: PC Clock: slave, DUT: master

Action: Set the value of priority1 in the default dataset of the PC Clock to 0.

Expected results: The DUT should become slave within 3 Announce intervals.

Record:

Conclusion: Expected results verified => continue to next step

13.D.4

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Examine the offsetFomMaster field in the response. This value should be consistent with the vendors synchronization specification. Typically this value should definitely be less than 1 us for hardware implementations and 1 ms for software implementations. The value should be consistent with the normalOffset from 13.B.1

Record:

Conclusion: Expected results verified => verifies synchronization for maximum logSyncInterval (minmum Sync interval) for the DUT as slave.

13.E Setup for recommended minimum logSyncInterval

13.E.1

Entry state: PC Clock: master, DUT: slave

Action: Set the logSyncInterval value of the port dataset of the PC Clock to -2 (we check in turn the recommended values -2, -3, -4 until we determine which, if any the DUT supports). Send the management message LOG_SYNC_INTERVAL with actionField SET to the DUT. The logSyncInterval value is set to -2.

Expected results: DUT should return a management message LOG_SYNC_INTERVAL with actionField RESPONSE and the logSyncInterval value should be -2 IF the DUT supports this value. If it does not support it (which it is not required to do) then the RESPONSE will have a 1588 ERROR TLV appended with a managementErrorId of either WRONG_VALUE or NOT_SUPPORTED or WRONG_VALUE 0x0004 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Value of logSyncInterval as testInterval

Conclusion: Repeat this step with the value of logSyncInterval decremented by 1, e.g. after -2 try -3, until an error TLV is returned indicating that the current test value is 1 too negative. Then proceed to next step

13.E.2

Entry state: PC Clock: master, DUT: slave

Action: Set the logSyncInterval value of the port dataset of the PC Clock to testInterval +1. Send the management message LOG_SYNC_INTERVAL with actionField SET to the DUT. The logSyncInterval value is set to testInterval +1.

Expected results: DUT should return a management message LOG_SYNC_INTERVAL with actionField RESPONSE and the logSyncInterval value should be testInterval +1. Check to see if testInterval +1 = -1 (indicating that the DUT only supports the required range of logSyncInterval values)

Record:

Conclusion: Expected results verified => IF the check showed that testInterval +1 = -1 then this test is complete. Otherwise continue to next steps.

13.F Test for recommended minimum logSyncInterval

13.F.1

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message CURRENT_DATA_SET with actionField RESPONSE. Examine the offsetFomMaster field in the response. This value should be consistent with the vendors synchronization specification. Typically this value should definitely be less than 1 us for hardware implementations and 1 ms for software implementations. The value should be consistent with the normalOffset from 13.B.1

Record:

Conclusion: Expected results verified => verifies synchronization for maximum logSyncInterval (minmum Sync interval) for the DUT as slave.

13.F.2

Entry state: PC Clock: slave, DUT: master

Action: Set the value of priority1 in the default dataset of the PC Clock to 255.

Expected results: The DUT should become master within 3 Announce intervals.

Record:

Conclusion: Expected results verified => continue to next step

13.F.3

Entry state: PC Clock: slave, DUT: master

Action: Examine the offsetFromMaster of the current dataset of the PC Clock. If necessary repeat this step until this value is stable (to allow for any transient resulting from change in Sync interval)

Expected results: This value should be consistent with the vendors synchronization specification and the recorded value nominalOffset from 13.B.1. Typically this value should definitely be less than 1 us for hardware implementations and 1 ms for software implementations.

Record:

Conclusion: Expected results verified => verification of synchronization for recommendded minimum logSyncInterval for the DUT as master

13.F.4

Entry state: PC Clock: slave, DUT: master

Action: Record the time of receipt at the PC Clock of Sync messages from DUT. Record for 51 messages (about 100 s). Compute the following:

g. Compute the inter-message intervals from this data.
h. Compute the number of inter-message intervals within ± 30 % of the value of $2^{+(\text{testInterval} +1)}$ seconds. This number should be >90% of the computed intervals.

Expected results: Computational results as specified in the action.

Record:

Conclusion: Expected results verified => validates the correct intervals between Sync messages from a master, 9.5.9.2

14. Must Be Able to Set UTC Time Manually

Identifier: xxx
Category: Mandatory

Purpose:

This test verifies that the DUT as a master can have its time adjusted to be traceable to UTC. UTC as required by 3.2.5 and 3.2.6 of the LXI Standard. LXI Devices can use either the web page or IEEE 1588 management messages to set the time and the related time properties.

Note that the 1588 clock actually distributes seconds from a defined epoch. The interpretation of this value as UTC depends on the value of the currentUtcOffset distributed by a master clock and the use of the POSIX algorithm for conversion into the print form of UTC.

The basic mechanism for updating the time and various properties is verified in 1.F. In this test we verify the correct semantics for these properties. In this test the PC Clock is only used to send management messages. Correct operation is observed using Wireshark and the web page of the LXI Device.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

14.A Setup

14.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

- a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
- b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
- c- all modification fields are 0
- d- priority1 = 255

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

14.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

14.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. OK to continue

14.B Test using management messages

14.B.1

Entry state: PC Clock: slave, DUT: master

Action: Send the management message UTC_PROPERTIES with actionField SET and the following field values:

- a- currentUtcOffset Integer16; 240 (Note- in step 14.B.6 the actual time is set to 0, the PTP epoch. The actual value of currentUtcOffset at the epoch is 8 s. In this test 240 is used so that the effect of this additive constant can be observed even in the presence of uncertainty due to test execution times of several seconds)
- b- UTCV Boolean: TRUE
- c- LI-59 Boolean: TRUE

d- LI-61 Boolean: FALSE

Expected results: A management message with actionField = RESPONSE and a managementId field UTC_PROPERTIES should be received from DUT. The expected values should be as specified in the action of this step.

Record:

Conclusion: Expected results verified => continue to next step

14.B.2

Entry state: PC Clock: slave, DUT: master

Action: Observe the leap59 and leap61 flagsField values in the next Announce message from the DUT.

Expected results: The value of leap59 should be TRUE and leap61 should be FALSE.

Record:

Conclusion: Expected results verified => partially verifies IEEE 1588 13.3.2.6. continue to next step

14.B.3

Entry state: PC Clock: slave, DUT: master

Action: Send the management message UTC_PROPERTIES with actionField SET and the following field values:

a- currentUtcOffset Integer16; 240
b- UTCV Boolean: TRUE
c- LI-59 Boolean: FALSE
d- LI-61 Boolean: TRUE

Expected results: A management message with actionField = RESPONSE and a managementId field UTC_PROPERTIES should be received from DUT. The expected values should be as specified in the action of this step.

Record:

Conclusion: Expected results verified => continue to next step

14.B.4

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TRACEABILITY_PROPERTIES with actionField SET and the following field values:

a- FTRA Boolean: TRUE
b- TTRA Boolean: TRUE

Expected results: A management message with actionField = RESPONSE and a managementId field TRACEABILITY _PROPERTIES should be received from DUT. The expected values should be as specified in the action of this step.

Record:

Conclusion: Expected results verified => continue to next step

14.B.5

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIMESCALE_PROPERTIES with actionField SET and the following field values:

a- PTP Boolean: TRUE
b- timeSource Enumeration8 (UInteger8): 0x60 HAND_SET

Expected results: A management message with actionField = RESPONSE and a managementId field TIMESCALE _PROPERTIES should be received from DUT. The expected values should be as specified in the action of this step.

Record:

Conclusion: Expected results verified => continue to next step

14.B.6

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME with actionField SET and the following field values:

- currentTime type Timestamp: secondsField UInteger48 = 0, nanosecondsField UInteger32 = 0.

Expected results: A management message with actionField = RESPONSE and a managementId field TIME should be received from DUT. The expected values should be as specified in the action of this step.

Record:

Conclusion: Expected results verified => continue to next step

14.B.7

Entry state: PC Clock: slave, DUT: master

Action: Observe the contents of the next Sync and Announce messages after step 14.B.6.

Expected results: Depending on how rapidly this step is executed the expected values in the Sync and Announce messages are as follows:

Announce:

a- flagField:

i. leap61 = TRUE
ii. leap59 = FALSE
iii. currentUtcOffsetValid = TRUE
iv. ptpTimescale = TRUE
v. timeTraceable = TRUE
vi. frequencyTraceable = TRUE

b- currentUtcOffset = 240

c- timeSource = 0x60

Sync:

a- originTimestamp.secondsField = 0 (plus whatever time has elapsed since 14.B.6, this should be at most a few seconds)

b- originTimestamp.nanosecondsField = any value < 10^{+9} is legal.

Record:

Conclusion: Expected results verified => verifies LXI rules 3.2.5 and 3.2.6 when set by management messages. Also verifies IEEE 1588 13.3.2.6, 13.5.1, and 13.6.1 for the fields listed above.

14.B.8

Entry state: PC Clock: slave, DUT: master

Action: Observe the contents of the next 30 Sync, and if the DUT is a 2-step clock the Follow_Up messages after step 14.B.7- this should take about 30 seconds with the default Sync interval of 1 s.

Expected results: For a 1-step DUT, the value of the originTimestamp.secondsField in the Sync message should on average increase by 1 for each Sync message. For a 2-step DUT, the value of the preciseOriginTimestamp.secondsField in the Follow_Up message should on average increase by 1 for each Follow_Up message.

Record:

Conclusion: Expected results verified => verifies IEEE 1588 7.2.4

14.B.9

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME with actionField SET and the following field values:

- currentTime type Timestamp: secondsField UInteger48 = 0, nanosecondsField UInteger32 = 0.

Expected results: A management message with actionField = RESPONSE and a managementId field TIME should be received from DUT. The expected values should be as specified in the action of this step.

Record:

Conclusion: Expected results verified => continue to next step

14.B.10

Entry state: PC Clock: slave, DUT: master

Action: Observe the DUT Sync web page (and any other display showing the current and possibly local time).

Expected results: The value of Current PTP time should be within a few seconds of 0. The current local time (if available) should be within a few seconds of 00:04:00 on 1 January 1970 UTC (00:04:00 is 4 minutes past 0 h) (actual local time will depend on the timezone)

Record:

Conclusion: Expected results verified => verifies LXI 9.6

14.C Test using LXI Sync web page (assuming implementation is read/write on time- standard is vague on this)

14.C.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. Default values of data sets in particular the currentUctOffset value of the Time Properties Dataset should be the current number of leap seconds. See past leap seconds for the current number of leap seconds = TAI - UTC. OK to continue after 3 Announce intervals

14.C.2 Setup for test using LXI Sync web page

Entry state: PC Clock: slave, DUT: master

Action: Using the LXI Sync web page set the DUT to the current time. Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. Default values of data sets in particular the currentUctOffset value of the Time Properties Dataset should be the current number of leap seconds. See past leap seconds for the current number of leap seconds = TAI - UTC. Wait for PC clock to synchronize then OK to continue

14.C.3

Entry state: PC Clock: slave, DUT: master

Action: Observe the contents of the next Announce message after step 14.C.2.

Expected results: Depending on how rapidly this step is executed the expected values in the Announce message is as follows:

Announce:

a- flagField:

i. leap61 = FALSE (unless users knows this to be true and the web page allows him to enter this value)

ii. leap59 = FALSE(unless users knows this to be true and the web page allows him to enter this value)

iii. currentUtcOffsetValid = FALSE or TRUE (depending on whether user is intentionally setting this value and whether the web page allows him to enter this information)

iv. ptpTimescale = TRUE or FALSE (depending on whether user is intentionally setting time to PTP or to an arbitrary timescale and whether the web page allows him to enter this information)

v. timeTraceable = TRUE or FALSE (since the web page is being used the value of this depends on whether the user obtained the time from a traceable source AND whether the web page allowed him to enter this information)

vi. frequencyTraceable = TRUE or FALSE (since the web page is being used the value of this depends on whether the user obtained the time from a traceable source AND whether the web page allowed him to enter this information)

b- currentUtcOffset = either the number of leap seconds valid at the time the device was designed (1588 default) or the actual value in which case currentUtcOffsetValid value above should be TRUE indicating it was set by the user.

c- timeSource = 0x60

Record:

Conclusion: partially validates LXI 9.6

14.C.4

Entry state: PC Clock: slave, DUT: master

Action: Observe the Current PTP time (and local time if available) in the PC Clock.

Expected results: The value of Current PTP time should be within a few seconds of the value entered in 14.C.2 (plus the wait between step 14.C.2 and 14.C.4) This value is computed by adding the value of currentUtcOffset to the UTC value and then running this into the reverse POSIX algorithm for conversion of number of seconds since the epoch to UTC. The current local time (if available) should be within a few seconds plus the wait between step 14.C.2 and 14.C.4 of the UTC time set in 14.C.2 (actual local time will depend on the timezone).

Record:

Conclusion: Expected results verified => verifies LXI 9.6

15. Clock describes itself accurately

Identifier: xxx

Category: Mandatory

Purpose:

This test verifies that the DUT as a master correctly describes itself in the Announce and Sync messages.

The attributes related to the timescale have been tested in test 14. The contents of the datasets have been examined in test 1 as part of the testing of management messages. This test checks that the contents of the datasets for the fields listed appear correctly in the Announce and Sync messages. Tested fields:

a- versionPTP
b- twoStepFlag
c- grandmasterPriority1
d- grandmasterClockQuality
e- grandmasterPriority2
f- grandmasterIdentity

Correct operation is observed using Wireshark and the web page of the LXI Device.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
15.A Setup

15.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 255
e- domainNumber = 0
f- sourcePortIdentity = implementation specific value for the PC Clock per 1588 7.5.2

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

15.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

15.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. OK to continue

15.B

15.B.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message DEFAULT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field DEFAULT_DATA_SET should be received from DUT. The expected values are indicated in the 'record' section of this step. Note that the values should be the default values pertaining to the device as a result of 15.A.2 and 15.A.3.

Record: Record:

a- TSC (Boolean): probably TRUE (indicates that device is a two step clock)
b- priority1 (UInteger8): should be 128
c- clockQuality.clockClass UInteger8: should be 248 unless the device falls under RULE 2.9.3 of the LXI Profile (few will) in which case it should be 6, 7 or 127 (most likely 127 in the test environment)
d- clockQuality.clockAccuracy Enumeration8 (UInteger8): in the range 0x20 to 0x31 or else 0xFE should agree with device data sheet and Table 6
e- clockQuality.offsetScaledLogVariance UInteger16: should be in the range 0x4435 to 0x72B6 or 17461 to 29366 corresponding to actual variances of 10^{-18} to 10^{-4} or deviations of 1ns to 10 ms respectively
f- priority2 (UInteger8): should be 128
g- clockIdentity(Octet[8]): implementation specific see 7.5.2.2.2
h- domainNumber (UInteger8): should be 0

Conclusion: Expected results verified => continue to next step.

15.B.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message PORT_DATA_SET with actionField GET

Expected results: A management message with actionField = RESPONSE and a managementId field PORT_DATA_SET should be received from DUT. The expected values are indicated in the 'record' section of this step. Note that the values should be the default values pertaining to the device as a result of 15.A.2 and 15.A.3.

Record: Record:

a- versionNumber: should be 2
b- portIdentity.ClockIdentity Octet[6]: same as clockIdentity in 15.B.1 (g)
c- portIdentity.portNumber UInteger16: =1 for a single port LXI device

Conclusion: Expected results verified => continue to next step.

15.B.3

Entry state: PC Clock: slave, DUT: master

Action: Observe the contents of the next Sync and Announce messages after step 15.B.2.

Expected results: The expected values in the Sync and Announce messages are as follows:

Announce:

a- versionPTP (in header): same as value of versionNumber from 15.B.2 (a)
b- sourcePortIdentity.ClockIdentity (of header): same as value of portIdentity.ClockIdentity from 15.B.2 (b)
c- sourcePortIdentity.portNumber (of header): same as value of portIdentity.portNumber from 15.B.2 (c)
d- grandmasterPriority1: same as value of priority1 from 15.B.1 (b)
e- grandmasterClockQuality.clockClass: same as value of clockQuality.clockClass from 15.B.1 (c)
f- grandmasterClockQuality.clockAccuracy: same as value of clockQuality.clockAccuracy from 15.B.1 (d)
g- grandmasterClockQuality.offsetScaledLogVariance: same as value of clockQuality.offsetScaledLogVariance from 15.B.1 (e)
h- grandmasterPriority2: same as value of priority2 from 15.B.1 (f)
i- grandmasterIdentity: same as value of clockIdentity from 15.B.1 (g)

Sync:

a- versionPTP (in header): same as value of versionNumber from 15.B.2 (a)
b- sourcePortIdentity.ClockIdentity (of header): same as value of portIdentity.ClockIdentity from 15.B.2 (b)
c- sourcePortIdentity.portNumber (of header): same as value of portIdentity.portNumber from 15.B.2 (c)
d- twoStepFlag of header: same as value of TSC from 15.B.1 (a)

Record:

Conclusion: Expected results verified => Cited fields in 13.3.1, 13.5 and 13.6

16. Time span

Identifier: xxx
Category: Mandatory

Purpose:

This test verifies that the clock in the DUT can advance from 2^{32}-1 seconds to 2^{32} seconds, both as master and as slave. This is best done by looking at the timestamps on the wire, since operating systems may not format the date correctly. The date is approximately 6:28:16 am Feb 7, 2106, TAI. (It's impossible to know what the UTC time is because the leap second offset for that date is unknown).

Correct operation when the DUT is master is observed using Wireshark.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
16.A Setup

16.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 255

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

16.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

16.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. OK to continue

16.B Test when DUT is master

16.B.1

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME with actionField SET and the following field values:

a- currentTime type Timestamp:
   i. secondsField UInteger48 = 0x 0000 FFFF FFF0 (2^{32}-16),
   ii. nanosecondsField UInteger32 = 0.

Expected results: A management message with actionField = RESPONSE and a managementId field TIME should be received from DUT. The expected values should be as specified in the action of this step.

Record:

Conclusion: Expected results verified => continue to next step

16.B.2

Entry state: PC Clock: slave, DUT: master

Action: Observe the contents of the next 20 Sync messages after step 16.B.1.

Expected results: Depending on how rapidly this step is executed the expected values of the originTimestamp.secondsField in the Sync messages are as follows (in order):

1. 0x 0000 FFFF FFF0 (2^{32}-16)
2. 0x 0000 FFFF FFF1 (2^{32}-15)
3. 0x 0000 FFFF FFF2 (2^{32}-14)
4. 0x 0000 FFFF FFF3 (2^{32}-13)
5. 0x 0000 FFFF FFF4 (2^{32}-12)
6. 0x 0000 FFFF FFF5 (2^{32}-11)
7. 0x 0000 FFFF FFF6 (2^{32}-10)
8. 0x 0000 FFFF FFF7 (2^{32}-9)
9. 0x 0000 FFFF FFF8 (2^{32}-8)
10. 0x 0000 FFFF FFF9 (2^{32}-7)
11. 0x 0000 FFFF FFFA (2^{32}-6)
12. 0x 0000 FFFF FFFA (2^{32}-5)
13. 0x 0000 FFFF FFFC (2^{32}-4)
14. 0x 0000 FFFF FFFD (2^{32}-3)
15. 0x 0000 FFFF FFFE (2^{32}-2)
16. 0x 0000 FFFF FFFF (2^{32}-1)
17. 0x 0001 0000 0000 (2^{32})
18. 0x 0001 0000 0001 (2^{32}+1)
19. 0x 0001 0000 0002 (2^{32}+2)
20. 0x 0001 0000 0003 (2^{32}+3)

Record:

Conclusion: Expected results verified => verifies rollover of Timestamp datatype 5.3.3 as master

16.C Test when DUT is slave

Entry state: PC Clock: slave, DUT: master

Action: Set priority1 of PC Clock to 0

Expected results: The DUT should become slave after 3 Announce intervals

Record:

Conclusion: Expected results verified => go to next step

16.C.1

Entry state: PC Clock: master, DUT: slave

Action: Send the time in the PC Clock to:

a- currentTime type Timestamp:

   i. secondsField UInteger48 = 0x 0000 FFFF FFF0 (2^{32}-16),

   ii. nanosecondsField UInteger32 = 0.

Expected results: This may be a big offset for the slave to sync to, so allow 3 minutes for the DUT to synchronize.

Record:

Conclusion: Expected results verified => continue to next step

16.C.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message TIME with actionField GET and observe the following field values in the RESPONSE:

a- currentTime type Timestamp:

i. secondsField UInteger48,

ii. nanosecondsField UInteger32 = ignore but should be any value less than 10^{+9}.

Expected results: Depending on how rapidly this step is executed the expected values of the currentTimestamp.secondsField in the RESPONSE messages will go through the following values in the order listed (depending on how rapidly this step is repeated relative to the Sync interval there may be missing values or repeated values but the monotonic progression should be visible):

1. 0x 0000 FFFF FFF0 (2^{32}-16)
2. 0x 0000 FFFF FFF1 (2^{32}-15)
3. 0x 0000 FFFF FFF2 (2^{32}-14)
4. 0x 0000 FFFF FFF3 (2^{32}-13)
5. 0x 0000 FFFF FFF4 (2^{32}-12)
6. 0x 0000 FFFF FFF5 (2^{32}-11)
7. 0x 0000 FFFF FFF6 (2^{32}-10)
8. 0x 0000 FFFF FFF7 (2^{32}-9)
9. 0x 0000 FFFF FFF8 (2^{32}-8)
10. 0x 0000 FFFF FFF9 (2^{32}-7)
11. 0x 0000 FFFF FFFA (2^{32}-6)
12. 0x 0000 FFFF FFFA (2^{32}-5)
13. 0x 0000 FFFF FFFC (2^{32}-4)
14. 0x 0000 FFFF FFFD (2^{32}-3)
15. 0x 0000 FFFF FFFE (2^{32}-2)
16. 0x 0000 FFFF FFFF (2^{32}-1)
17. 0x 0001 0000 0000 (2^{32})
18. 0x 0001 0000 0001 (2^{32}+1)
19. 0x 0001 0000 0002 (2^{32}+2)
20. 0x 0001 0000 0003 (2^{32}+3)

Record:

Conclusion: Expected results verified => verifies rollover of Timestamp datatype 5.3.3 as slave

17. Leap second info passed from master to slave

Identifier: xxx

Category: Mandatory

Purpose:

This test verifies that the DUT correctly updates the time properties data set from the Announce message from the grandmaster.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
17.A Setup

17.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 0
e- currentUtcOffset Integer16; should be 33 before 0 hours 1 January 2009 and 34 after 0 hours 1 January 2009. This value can change every six months- the potential update window for leap second changes.
f- FTRA Boolean; FALSE
g- TTRA Boolean: FALSE
h- PTP Boolean: FALSE
i- UTCV Boolean: FALSE
j- LI-59 Boolean: FALSE
k- LI-61 Boolean: FALSE
l- timeSource Enumeration8 (UInteger8): 0xA0 INTERNAL_OSCILLATOR see 7.6.2.6

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

17.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

17.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become slave after 3 Announce intervals. OK to continue

17.B

17.B.1

Entry state: PC Clock: master, DUT: slave

Action: Send the management message TIME_PROPERTIES_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message TIME_PROPERTIES_DATA_SET with actionField RESPONSE. Observe and record the fields indicated.

Record: Record:

- o currentUtcOffset Integer16; should be 33 before 0 hours 1 January 2009 and 34 after 0 hours 1 January 2009. This value can change every six months- the potential update window for leap second changes.
- o FTRA Boolean; normally FALSE
- o TTRA Boolean: normally FALSE
- o PTP Boolean: normally FALSE
- o UTCV Boolean: normally FALSE but TRUE is acceptable if currentUtcOffset is as indicated in ‘a’
- o LI-59 Boolean: normally FALSE
- o LI-61 Boolean: normally FALSE
- o timeSource Enumeration8 (UInteger8): normally 0xA0 INTERNAL_OSCILLATOR see 7.6.2.6

Conclusion: Expected results verified => continue to next step

17.B.2

Entry state: PC Clock: master, DUT: slave

Action: Set PC Clock so that:

a- currentUtcOffset Integer16; 28
b- FTRA Boolean; TRUE
c- TTRA Boolean: TRUE
d- PTP Boolean: TRUE
e- UTCV Boolean: TRUE
f- LI-59 Boolean: TRUE
g- LI-61 Boolean: FALSE
h- timeSource Enumeration8 (UInteger8): 0x10 ATOMIC_CLOCK see 7.6.2.6

Expected results: The DUT should copy these values into the TIME_PROPERTIES_DATA_SET as indicated in next step

Record:

Conclusion: Continue to next step

17.B.3

Entry state: PC Clock: master, DUT: slave

Action: Send the management message TIME_PROPERTIES_DATA_SET with actionField GET to the DUT.

Expected results: DUT should return a management message TIME_PROPERTIES_DATA_SET with actionField RESPONSE. Observe the fields as indicated.

- o currentUtcOffset Integer16; should be 28
- o FTRA Boolean; should be TRUE
- o TTRA Boolean: should be TRUE

- PTP Boolean: should be TRUE
- UTCV Boolean: should be TRUE
- LI-59 Boolean: should be TRUE
- LI-61 Boolean: should be FALSE
- timeSource Enumeration8 (UInteger8): should be 0x10 ATOMIC_CLOCK see 7.6.2.6

Record:

Conclusion: Expected results verified => validates Table 16, 9.3.5 for time properties.

18. Settling time

Identifier: xxx
Category: Recommended

Purpose:

This is a performance measurement, and at present does not correspond to any specification or requirement.

This test measures the time it takes a slave to adjust its clock to that of the master. The time is called the settling time and is measured from the time the DUT enters that slave state until the currentOffset value (the slave measured offset from the master) is and remains within some threshold from zero. This threshold is implementation dependent but a reasonable starting value is to be within ± 100 ns for 3 consecutive Sync intervals.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

18.A Setup

18.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 255

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

18.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

18.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. OK to continue

18.B

18.B.1

Entry state: PC Clock: slave, DUT: master

Action: Set the time in the PC Clock to the time of the slave + 10 seconds. At the same time change priority1 of the PC Clock to 0.

Expected results: After 3 Announce intervals the DUT should become slave and start to adust its clock to that of the PC Clock. Since the time observed by the DUT will appear to have jumped by 10 seconds, the DUT clock will have to settle to the new value.

Record: The time the first Announce message is sent from the PC Clock with priority1 = 0.

Conclusion: Expected results verified => continue to next step immediately after sending the first Announce message.

18.B.2

Entry state: PC Clock: Master, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: A management message with actionField = RESPONSE and a managementId field PORT_DATA_SET should be received from DUT. The value of portState in the RESPONSE should initially be MASTER, 0x06, but should change to slave, 0x09, after approximately 3 Announce intervals.

Record: The time, initialTime, at which the portState value returned from the DUT changes to slave.

Conclusion: Expected results verified => Repeat this step until the portState of the DUT changes to slave, then continue to next step

18.B.3

Entry state: PC Clock: master, DUT: slave

Action: Send the management message CURRENT_DATA_SET with actionField GET.

Expected results: A management message with actionField = RESPONSE and a managementId field CURRENT_DATA_SET should be received from DUT. Observe the value of offsetFromMaster in the RESPONSE.

Record: The value of offsetFromMaster as measuredOffset. If the value of measuredOffset changes from being outside of a window ± threshold (suggest 100 ns for hardware clocks) to inside this window, record as endTime the time the RESPONSE was received with offsetFromMaster changed from outside to inside the window

Conclusion: Expected results verified => Repeat this step until the the value of measuredOffset has remained within the window for 3 Sync intervals (or whatever time is appropriate given the damping characteristics of the servo)

18.B.4

Entry state: PC Clock: master, DUT: slave

Action: Compute the settling time as endTime − initialTime from 18.B.3 and 18.B.2.

Expected results: The settling time is within vendor specification.

Record:

Conclusion: Expected results verified => Verifies vendor’s settling time specification.

19. Correct nominal clock speed

Identifier: xxx
Category: Vendor Declaration

Purpose:

Clause J.3.4.1 requires that the frequency of a master clock be within 0.01% of the SI second. This test verifies this requirement.

Test configuration:

Configuration 4 or 5

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

19.A Setup

19.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 255

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

19.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

19.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. OK to continue

19.B

19.B.1

Entry state: PC Clock: slave, DUT: master

Action: Measure the frequency of the 1 PPS output of the DUT. This measurement must be made with an instrument that can measure the frequency of this signal with sufficient accuracy and precision and traceable to international standards to be able to verify that the test result is within 0.01% of the SI second. Note-most conventional instruments are not this accurate or precise. Standards laboratory quality and traceable instruments will be required.

Expected results: The measurement of the frequency of the 1 PPS should be within 0.01% of the SI second..

Record:

Conclusion: Expected results verified => validates J.3.4.1.

20. Clock subsystem survives time jump

Identifier: xxx

Category: Vendor Declaration

Purpose:

IEEE 1588 specifies certain changes of state and issuing of messages based on timeouts such as the ANNOUNCE_RECEIPT_TIMEOUT and the Sync and Delay_Req intervals. This test checks that a jump in the local clock does not disturb the accuracy of these timeouts and intervals for some common cases.

Test configuration:

Configuration 1

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

20.A Setup

20.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 255

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

20.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

20.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become master after 3 Announce intervals. OK to continue

20.B Test robustness of Announce and Sync message transmission

20.B.1

Entry state: PC Clock: slave, DUT: master

Action: Send the management message PORT_DATA_SET with actionField GET.

Expected results: A management message with actionField = RESPONSE and a managementId field PORT_DATA_SET should be received from DUT. Observe the values of logAnnounceInterval and log SyncInterval fields in the RESPONSE.

Record: The values:

a- Sync interval computed from $2^{logAnnounceInterval}$
b- Announce interval computed from $2^{logAnnounceInterval}$

Conclusion: Expected results verified => continue to next step

20.B.2

Entry state: PC Clock: slave, DUT: master

Action: Observe the message intervals for both Announce and Sync messages from the DUT.

Expected results: The measured values of these intervals should nominally agree with the intervals computed in 20.B.1.

Record:

Conclusion: Expected results verified => continue to next step

20.B.3

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME with actionField SET and a time field value one hour earlier than the current time of the PC Clock (which is synchronized to the DUT.

Expected results:

Record:

Conclusion: Expected results verified => continue to next step

20.B.4

Entry state: PC Clock: slave, DUT: master

Action: Observe the message intervals for both Announce and Sync messages from the DUT.

Expected results: The measured values of these intervals should nominally agree with the intervals computed in 20.B.1.

Record:

Conclusion: Expected results verified => verifies than the issuing of Announce and Sync messages from the DUT was not corrupted by the backward change in time.

20.B.5

Entry state: PC Clock: slave, DUT: master

Action: Send the management message TIME with actionField SET and a time field value one hour later than the current time of the PC Clock (which is synchronized to the DUT.

Expected results:

Record:

Conclusion: Expected results verified => continue to next step

20.B.6

Entry state: PC Clock: slave, DUT: master

Action: Observe the message intervals for both Announce and Sync messages from the DUT.

Expected results: The measured values of these intervals should nominally agree with the intervals computed in 20.B.1.

Record:

Conclusion: Expected results verified => verifies than the issuing of Announce and Sync messages from the DUT was not corrupted by the forward change in time.

20.C Test robustness of Delay_Req message transmission

20.C.1

Entry state: PC Clock: slave, DUT: master

Action: Set the value of priority1 of the PC Clock to 0.

Expected results: DUT should become slave within 3 Announce intervals.

Record:

Conclusion: Expected results verified => continue to next step

20.C.2

Entry state: PC Clock: master, DUT: slave

Action: Observe the message intervals for Delay_Req messages from the DUT.

Expected results: The measured values of this interval should be no more than 32 times the Sync interval computed in 20.B.1.

Record:

Conclusion: Expected results verified => continue to next step

20.C.3

Entry state: PC Clock: master, DUT: slave

Action: Set the time in the PC Clock to a value one hour earlier than the current time of the PC Clock

Expected results:

Record:

Conclusion: Expected results verified => continue to next step

20.C.4

Entry state: PC Clock: master, DUT: slave

Action: Observe the message intervals for Delay_Req messages from the DUT.

Expected results: The measured values of this interval should be no more than 32 times the Sync interval computed in 20.B.1.

Record:

Conclusion: Expected results verified => verifies than the issuing of Delay_Req messages from the DUT was not corrupted by the backward change in time.

20.C.5

Entry state: PC Clock: master, DUT: slave

Action: Set the time in the PC Clock to a value one hour later than the current time of the PC Clock

Expected results:

Record:

Conclusion: Expected results verified => continue to next step

20.C.6

Entry state: PC Clock: master, DUT: slave

Action: Observe the message intervals for Delay_Req messages from the DUT.

Expected results: The measured values of this interval should be no more than 32 times the Sync interval computed in 20.B.1.

Record:

Conclusion: Expected results verified => verifies than the issuing of Delay_Req messages from the DUT was not corrupted by the forward change in time.

20.D Test robustness of ANNOUNCE_RECEIPT_TIMEOUT

20.D.1

Entry state: PC Clock: master, DUT: slave

Action: Send the management message ANNOUNCE_RECEIPT_TIMEOUT with actionField GET.

Expected results: A management message with actionField = RESPONSE and a managementId field ANNOUNCE_RECEIPT_TIMEOUT should be received from DUT. Observe the values of the announceReceiptTimeout field in the RESPONSE. This should be the default value 3 (J.3.2)

Record: This value as initialTimeout.

Conclusion: Expected results verified => continue to next step

20.D.2

Entry state: PC Clock: master, DUT: slave

Action: Send the management message ANNOUNCE_RECEIPT_TIMEOUT with actionField SET with a value of announceReceiptTimeout of 10 (see J.3.2).

Expected results: A management message with actionField = RESPONSE and a managementId field ANNOUNCE_RECEIPT_TIMEOUT should be received from DUT. Observe the values of the announceReceiptTimeout field in the RESPONSE which should be 10

Record:

Conclusion: Expected results verified => continue to next step. THE NEXT STEPS SHOULD BE PERFORMED AS QUICKLY AS POSSIBLE AND MUST BE COMPLETED WITHIN 10 ANNOUNCE INTERVALS TO BE VALID

20.D.3

Entry state: PC Clock: master, DUT: slave

Action: Set domain = 1 in the PC Clock.

Expected results: The DUT should ignore future Announce messages from the PC Clock and should transisition to master. But before this happens the next step will disturb the clock in the DUT (assuming the implementation does not disregard the TIME management message SET since it is still in the slave state- not an unreasonable implementation, probably within reasonable interpretations of the intent of the standard, and is definitely reasonable behavior since a slave clock is supposed to follow the grandmaster)

Record: The time at which the domain was changed as initialTime.

Conclusion: Expected results verified => continue to next step

20.D.4

Entry state: PC Clock: master, DUT: slave

Action: Send the management message TIME with actionField SET and a time field value 25 of the new Announce interval (i.e. 10x25 x Sync interval) later than the current time of the PC Clock

Expected results:

Record:

Conclusion: Expected results verified => continue to next step

20.D.5

Entry state: PC Clock: master, DUT: slave

Action: Send the management message PORT_DATA_SET with actionField GET to the DUT.

Expected results: Observe the value of the portState field of the RESPONSE. It should initially be slave, 0x09, but should eventually change to master, 0x06.

Record: The time on the PC Clock when the DUT state changes to master and endTime.

Conclusion: Expected results verified => continue to next step

20.D.6

Entry state: PC Clock: master (domain 1), DUT: master (domain 0)

Action: Compute the switchover time as endTime − initialTime (from 20.D.5 and 20.D.3)

Expected results: This switchover time should be within approxiamately ± 1 second of the Announce interval = 10x25xSync interval =250 x Sync interval = 250 seconds.

Record:

Conclusion: Expected results verified => If switchover time is not as expected the timeout mechanism is not robust under time jumps.

21. Sync quality

Identifier: xxx

Category: Vendor Declaration

Purpose:

This is a performance measurement. Note that the set of conditions under which to measure this parameter is both important and controversial. We recommend measuring under ideal conditions (e.g. crossover cable, rock-solid master, stable temperature) and for a long time. These are not realistic conditions, but they are the only ones that allow comparison between products.

For this test the PC Clock must be a high quality clock with a good oscillator and the test must be conducted in a good thermal environment.

Test configuration:

Configuration 4 or 5

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
21.A Setup

21.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

21.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

21.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become slave after 3 Announce intervals. OK to continue

21.B

21.B.1

Entry state: PC Clock: master, DUT: slave

Action: Send the management message TIME with actionField GET.

Expected results: A management message with actionField = RESPONSE and a managementId field TIME should be received from DUT. Observe the value of the currentTime field in the RESPONSE. It should be within ± 0.25 seconds of each other.

Record:

Conclusion: Expected results verified => The slave time is close enough that observation of the 1 PPS signals is unambiguous in determining the quality of synchronization.

21.B.2

Entry state: PC Clock: master, DUT: slave

Action: Measure a large number of time intervals between the 1 PPS signals from the PC Clock and the DUT. Compute the standard deviation of these measurements.

Expected results: The measured statistics should agree with the vendor specifications.

Record:

Conclusion: Expected results verified => validates the quality of synchronization.

Notes:

- The PPS measurement is precise, but is modulo 1 second. That's the reason we test that the times are within 1 second.
- If the measurements are made under typical (rather than ideal) conditions, it is advisable to make the measurement contain the worst-case temperature conditions. This usually means measuring over at least one weekday and at least one weekend day.

22. Application of asymmetry correction

Identifier: xxx
Category: Vendor Declaration

Purpose:

The protocol assumes the packet travel time between master and slave is equal in both directions. If it is not, an uncorrected system will exhibit a systematic offset between slave and master equal to half the asymmetry. PTP can't measure this asymmetry, but if it is measured the implementation can be told its magnitude and sign, and PTP will correct for it.

This test is only useful if the implementation has a mechanism for entering the value of the asymmetry of the link to its master. The appropriate corrections for this asymmetry are specified in clause 11 of IEEE 1588.

Test configuration:

Configuration 4 or 5

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:
22.A Setup

22.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

22.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

22.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become slave after 3 Announce intervals. OK to continue

22.B

22.B.1

Entry state: PC Clock: master, DUT: slave

Action: Observe the mean value of the time interval between the 1 PPS signals of the PC Clock and the DUT for sufficient time to get a stable mean and a good measure of the deviation.

Expected results: The mean should be within ±100 ns for a hardware clock.

Record:

Conclusion: Expected results verified => continue to next step

22.B.2

Entry state: PC Clock: master, DUT: slave

Action: Enter into the DUT a value of asymmetry of +10 us

Expected results: The slave clock will re-synchronize to a new mean.

Record:

Conclusion: Expected results verified => continue to next step

22.B.3

Entry state: PC Clock: master, DUT: slave

Action: Observe the mean value of the time interval between the 1 PPS signals of the PC Clock and the DUT for sufficient time to get a stable mean and a good measure of the deviation.

Expected results: The mean should now lead the master to within ±100 ns for a hardware clock.

Record:

Conclusion: Expected results verified => verifies asymmetry correction of clause 11

23. Proper simultaneous startup of many clocks

Identifier: xxx

Category: Vendor Declaration and/or plug-fest

Purpose:

The protocol is supposed to converge to 1 master and *n*-1 slaves under almost any conditions. This checks one of the challenging conditions: many clocks waking up at the same time. The protocol should settle, and all clocks should agree on who the master is.

Test configuration:

A large number (greater than 20) of DUTs and a network switch or transparent clock to allow all DUTs and PC Clock to be connected.

Warning:

Subtests:

23.A Setup

23.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 255

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

23.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGE-

MENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

23.B

23.B.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record: The time at which this message was sent as startTime.

Conclusion: Expected results verified => DUT should become slave after 3 Announce intervals. OK to continue

23.B.2

Entry state: PC Clock: slave, some DUT: master, all other DUTs: slave

Action: Using Wireshark observe the Announce messages from the DUTS.

Expected results: Eventually the number of Announce messages observed should decrease until only one clock is sending Announce messages.

Record: The time as endTime at which the number of DUTs sending Announce messages is reduced to 1. The IP address of the DUT sending the Announce message. Compute settlingTime as endTime − startTime from 23.B.1 and 23.B.2

Conclusion: Expected results verified => continue to next step

23.B.3

Entry state: PC Clock: slave, some DUT: master, all other DUTs: slave

Action: Using Wireshark observe the Delay_Req messages from the DUTS.

Expected results: Delay_Req messages should be observed from all DUTs except the one identified in 23.B.2. The settlingTime should be about T+ 2a×logb(n) seconds, where T is the Announce receipt timeout, b is the depth of the Foreign Master Buffer and a is the Announce interval.

Record:

Conclusion: Expected results verified => Repeat steps 23.B.1 through 23.B.3 several times to be sure the results, including the selection of the same master, are consistent.

24. DUT uses grandmaster not parent data in BMC

Identifier: xxx

Category: Vendor Declaration and/or plug-fest

Purpose:

This test verifies that the DUT uses the grandmaster fields in Announce messages rather than parent fields as inputs to the best master clock algorithm.

Test configuration:

Configuration 7

Warning:

If an error message is returned as a result of sending a management message in any of the tests, the tlvType of the returned TLV should be MANAGEMENT_ERROR_STATUS see IEEE 1588-2008 15.5.4 for interpretation.

Subtests:

24.A Setup

24.A.1

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set PC Clock so that:

a- The normal values for startingBoundaryHops = 0 and boundary-Hops = 0 are used in sending management messages.
b- The manageement messages are transmitted with targetPortIdentity.clockIdentity and targetPortIdentity.portNumber both all 1s
c- all modification fields are 0
d- priority1 = 0

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

24.A.2

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Set the boundary clock priority1 = 128 using whatever interface provided.

Expected results: Setup for later steps

Record:

Conclusion: Continue to next step

24.A.3

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message RESET_NON_VOLATILE_STORAGE with actionField COMMAND to the DUT

Expected results: This step is a set up step. An ACKNOWLEDGE message may be returned with or without an error TLV attached. If non-volatile storage is supported there should be no error. If not there should be a MANAGEMENT_ERROR_STATUS_TLV appended. The appended TLV should have the managementId field value RESET_NON_VOLATILE_STORAGE and the TLVmanagementErrorId value should be NOT_SUPPORTED 0x0006 or GENERAL_ERROR 0xFFFE

Expected results: .

Record: Whether or not an error TLV was appended

Conclusion: This is a set up step

24.A.4

Entry state: PC Clock: <unknown>, DUT: <unknown>

Action: Send the management message INITIALIZE with actionField CMD to the DUT

Expected results: DUT should return a management message INITIALIZE with actionField ACKNOWLEDGE

Record:

Conclusion: Expected results verified => DUT should become slave after 3 Announce intervals. OK to continue

24.B

24.B.1

Entry state: PC Clock: master, DUT: <subject of test>, Boundary Clock: slave to PC Clock

Action: Send the management message PRIORITY1 with actionField SET to the DUT. The value of the priority1 field should be 5.

Expected results: DUT should return a management message PRIORITY1 with actionField RESPONSE. The priority1 field should be 5.

Record:

Conclusion: Expected results verified => continue to next step

24.B.2

Entry state: PC Clock: master, DUT: <subject of test>, Boundary Clock: slave to PC Clock

Action: Send the management message PORT_DATA_SET with actionField GET to the DUT

Expected results: DUT should return a management message PORT_DATA_SET with actionField RESPONSE. The portState field should be slave, 0x009.

Record:

Conclusion: Expected results verified => validates that DUT uses correct fields.