NISTIR 8002

1588 Power Profile Test Plan

Carol Perkins
Jeff Laird
Ryan McEachern
Bob Noseworthy
Julien Amelot
Ya-Shian Li-Baboud
Kevin Brady

http://dx.doi.org/10.6028/NIST.IR.8002

National Institute of Standards and Technology
U.S. Department of Commerce

This publication is available free of charge from http://dx.doi.org/10.6028/NIST.IR.8002

NISTIR 8002

1588 Power Profile Test Plan

Carol Perkins
Jeff Laird
Ryan McEachern
Bob Noseworthy
University of New Hampshire Interoperability Laboratory
Durham, NH

Julien Amelot
Ya-Shian Li-Baboud
Kevin Brady
Software Systems Division
Information Technology Laboratory

This publication is available free of charge from:
http://dx.doi.org/10.6028/NIST.IR.8002

June 2014

U.S. Department of Commerce
Penny Pritzker, Secretary

National Institute of Standards and Technology
Willie May, Acting Under Secretary of Commerce for Standards and Technology and Director

1588 Power Profile Conformance Test Plan

for IEEE C37.238: IEEE Standard Profile for Use of IEEE 1588 Precision Time Protocol in Power System Applications

Version 0.1.23

Technical Document

Funded and developed with support from

NIST

NOTICE: This is a living document. All contents are subject to change.
Individual tests and/or test groups may be added/deleted/renumbered in forthcoming revisions.
General feedback and comments are welcome through the
Software Systems Division Information Technology Laboratory
powerProfileTest@nist.gov.

Last Updated June 17, 2014

TABLE OF CONTENTS

This publication is available free of charge from http://dx.doi.org/10.6028/NIST.IR.8002

ACKNOWLEDGMENTS

We would like to acknowledge the efforts of the following individuals in the review of this test plan:

Dhananjay Anand	National Institute of Standards and Technology
Judah Levine	National Institute of Standards and Technology

INTRODUCTION

The National Institute of Standards and Technology (NIST) is an agency of the U.S. Department of Commerce, facilitating the industry adoption of IEEE Standard C37.238 for the use of IEEE 1588 in Power Systems Applications in support of the Smart Grid. The University of New Hampshire InterOperability Laboratory (UNH-IOL) is a non-profit institution designed to promote the industry adoption of standards through conformance and interoperability testing. This particular test plan has been developed to help implementers evaluate the **1588 Power Profile** functionality of their products. This test plan is aimed at validating conformance of IEEE 1588 products. The conformance tests have been identified and supported by NIST and developed at the UNH-IOL.

These tests are designed to determine if a product conforms to specifications defined in *IEEE C37.238 IEEE Standard Profile for Use of IEEE 1588 Precision Time Protocol in Power System Applications* (hereafter referred to as the “1588 Power Profile”) standard from the IEEE Power & Energy Society’s Power System Relaying Committee and Substations Committee (PSRC). Passing all tests contained in this suite does not guarantee that the tested device will successfully operate with other 1588 Power Profile products. However, when combined with a satisfactory level of interoperability testing, these tests provide a reasonable level of confidence that the 1588 capabilities of the Device Under Test (DUT) will function properly in many power application environments.

The tests contained in this document are organized in order to simplify the identification of information related to a test, and to facilitate the actual testing process. Tests are separated into groups, primarily in order to reduce setup time in the lab environment, however the different groups typically also tend to focus on specific aspects of device functionality.

This test plan format is borrowed, with explicit permission, from UNH-IOL.

The test definitions themselves are intended to provide a high-level description of the motivation, resources, procedures, and methodologies specific to each test. Formally, each test description contains the following sections:

Test Label: The test label and title constitute the first line of the test block. The test label is the concatenation of the short test suite name, group number, and the test number within the group, separated by periods.

Purpose: The purpose is a brief statement outlining what the test attempts to achieve. It is usually phrased as a simple assertion of the feature or capability to be tested.

Device Type & Prerequisites: The Device Type & Prerequisites section notes for each part of the test what the prerequisite conditions are for the given Device Type.

Device Types	Prerequisite Condition
To be determined (TBD)	To be determined (TBD)
All	All
Boundary Clock (BC)	None
Ordinary Clock (OC)	Simple Network Management Protocol (SNMP)
Transparent Clock (TC)	Grandmaster-Capable (GMC)
	Slave-Only (SO)
	Preferred Grandmaster (PrefGM)
	One-step Clock
	Two-step Clock
	Syntonized or Not Syntonized
	Multiples Priorities Allowed
	not applicable to slave-only endpoints that do not implement delay measurement
	SNMP or means of observing a value of the DUT's data set
	1PPS input

References: The References section specifies all reference material external to the test plan, including the specific references for the test in question and any other references that might be helpful in understanding the test methodology or test results. External sources are always referenced by a bracketed number (e.g. [1]) when mentioned in the test description. Any other references in the test description that are not indicated in this manner refer to elements within the test suite document itself (e.g. "Appendix 5.A" or "Table 5.1.1-1").

Resource Requirements: The Resource Requirements section specifies the test hardware and software needed to perform the test. This is generally expressed in terms of minimum requirements for abstract test gear. In some cases precise equipment requirements may be provided with examples of specific manufacturer/model information provided.

Modification History: The Modification History logs the changes for this test since its introduction.

Discussion: The Discussion is a general discussion of the test and relevant section of the specification, including any assumptions made in the design or implementation of the test as well as known limitations.

Test Setup: The setup section describes the initial configuration of the test environment. Elements of the test procedure may change the test environment as the test progresses.

Procedure: The procedure section contains the systematic instructions for carrying out the test. It provides a cookbook approach to testing, and may be interspersed with requirements to record observable results. These procedures should be the ideal test methodology, independent of specific tool limitations or restrictions. This section is separated into parts (e.g. "Part A") with corresponding observable results for each.

Observable Results: This section lists the specific observable items that can be examined by the tester in order to verify that the DUT is operating properly. When multiple values for an observable item are possible, this section provides a short discussion on how to interpret them. The determination of a pass or fail outcome for a particular test is based on the successful (or unsuccessful) detection of a specific observable item. All test-part outcomes are presumed to initially be FAIL, and remain so if any single failure condition is met. Only if no fail conditions are met, and the explicitly stated pass conditions observed, will the test part outcome be deemed a PASS.

With the exception of N/A, WARN, and INFO, if a test part results in neither a PASS nor a FAIL outcome then that test part outcome is deemed a FAIL.

A strong preference is to have any part of a test err on the side of falsely failing a device rather than falsely passing the device. Whether through automation or manual execution, tests can have only one of five outcomes:

Out-come	Meaning
PASS	Test part meets all PASS criteria, with no FAIL or WARN conditions met.
FAIL	Test part meets at least one FAIL criterion, or fails to meet any criteria.
N/A	Test part is Not Applicable to the device.
WARN	Test part does not meet a failing criterion, but behavior is not recommended and warned against.
INFO	Test part has no pass/fail criteria, but the observation may have value to the device manufacturer or industry at large.

Possible Problems: This section contains a description of known issues with the test procedure, which may affect test results in certain situations.

Summary of Tests by Device Type

Test	Part	Ordinary Clock			Boundary Clock		Transparent Clock
		PrefGM	GMC	SO	PrefGM	GMC	
Test PWR.c.1.1 – logAnnounceInterval	A	X	X		X	X	
Test PWR.c.1.2 – logSyncInterval	A	X	X		X	X	
Test PWR.c.1.3 – announceReceiptTimeout	A	X			X		
	B		X			X	
Test PWR.c.1.4 – logMinPdelayReqInterval	A	X	X	X	X	X	X
Test PWR.c.1.5 – priority1 and priority2	A	X	X		X	X	
Test PWR.c.1.6 – domainNumber	A	X	X		X	X	
	B			X			
	C	X	X	X	X	X	
Test PWR.c.2.1 – Peer Delay Mechanism	A	X	X	X	X	X	X
	B	X	X	X	X	X	X
	C	X	X		X	X	
Test PWR.c.2.2 – Pdelay_Req Message Field Values	A	X	X	X	X	X	
	B						X
	C						X
Test PWR.c.2.3 – Pdelay_Resp Message Field Values, One-Step Clock	A	X	X	X	X	X	X
Test PWR.c.2.4 – Peer Delay Turnaround Timestamps, One-Step Clock	A	X	X	X	X	X	X
Test PWR.c.2.5 – Peer Delay Message Field Values, Two-Step Clock	A	X	X	X	X	X	X
Test PWR.c.2.6 – Peer Delay Turnaround Timestamps, Two-Step Clock	A	X	X	X	X	X	X
Test PWR.c.2.7 – Restriction on Peer Delay Mechanism	A	X	X	X	X	X	X
	B	X	X	X	X	X	
	C						X
Test PWR.c.2.8 – Mean Path Delay	A	X	X	X	X	X	
	B	X	X	X	X	X	
	C						X
	D						X
Test PWR.c.2.9 – Independent Ports for Boundary Clocks	A				X	X	
	B				X	X	
Test PWR.c.2.10 – Independent Ports for Transparent Clocks	A						X
	B						X
	C						X
	D						X
Test PWR.c.3.1 – Disqualified Announce Messages, by clockIdentity	A	X	X	X	X	X	
Test PWR.c.3.2 – Disqualified Announce Messages, by Most Recent	A	X	X	X	X	X	
Test PWR.c.3.3 – Disqualified Announce Messages, by Foreign Master Window	A	X	X	X	X	X	
	B	X	X	X	X	X	
Test PWR.c.3.4 – Disqualified Announce Messages, by stepsRemoved	A	X	X	X	X	X	
	B	X	X	X	X	X	
	C	X	X	X	X	X	
	D	X	X	X	X	X	

Test	Part	Ordinary Clock			Boundary Clock		Transparent Clock
		PrefGM	GMC	SO	PrefGM	GMC	
Test PWR.c.3.5 – Disqualified Announce Messages, by alternateMasterFlag	A	X	X	X	X	X	
Test PWR.c.3.6 – Data Set Comparison on a Single Port	A	X	X	X	X	X	
	B	X	X	X	X	X	
	C	X	X	X	X	X	
Test PWR.c.3.7 – Data Set Comparison on Multiple Ports	A				X	X	
Test PWR.c.3.8 – State Decision Algorithm	A	X	X	X	X	X	
	B	X	X	X	X	X	
	C	X	X	X	X	X	
Test PWR.c.3.9 – Steps Removed	A	X	X		X	X	
	B	X	X		X	X	
	C	X	X		X	X	
	D	X	X		X	X	
	E			X			
	F			X			
	G			X			
	H			X			
Test PWR.c.3.10 – Source Port Identity	A	X	X		X	X	
	B	X	X		X	X	
	C	X	X		X	X	
	D	X	X		X	X	
	E			X			
	F			X			
	G			X			
	H			X			
Test PWR.c.3.11 – Default Slave-only	A			X			
	B			X			
Test PWR.c.4.1 – TimeInaccuracy, Traceability and Offset	A	X	X		X	X	
	B	X	X		X	X	
	C			X			
	D			X			
	E			X			
	F			X			
Test PWR.c.4.2 – SNMP MIB Default Data Set for Ordinary and Boundary Clocks	A, B	X	X	X	X	X	
	C				X	X	
	D	X	X	X			
	E-J	X	X	X	X	X	
	K-L			X			
Test PWR.c.4.3 – SNMP MIB Parent Data Set for Ordinary and Boundary Clocks	A, B	X	X	X	X	X	
Test PWR.c.4.4 – SNMP MIB Time Properties Data Set for Ordinary and Boundary Clocks	A, B	X	X	X	X	X	

Test	Part	Ordinary Clock			Boundary Clock		Transparent Clock
		PrefGM	GMC	SO	PrefGM	GMC	
Test PWR.c.4.5 – SNMP MIB Port Data Set for Ordinary and Boundary Clocks	A-O	X	X	X	X	X	
Test PWR.c.4.6 – SNMP MIB Default Data Set for Transparent Clocks	A-J						X
Test PWR.c.4.7 – SNMP MIB Port Data Set for Transparent Clocks	A-D						X
Test PWR.c.4.8 – SNMP MIB Notifications	A-D	X	X	X	X	X	X
Test PWR.c.5.1 – IEEE 802.3 Transport Mapping for Announce, Sync and Follow_Up Messages	A	X	X		X	X	
Test PWR.c.5.2 – IEEE 802.3 Transport Mapping for Forwarded Announce, Sync and Follow_Up Messages	A						X
Test PWR.c.5.3 – IEEE 802.3 Transport Mapping for Peer Delay Messages	A	X	X	X	X	X	X
Test PWR.c.5.4 – Multiple Priorities	A	X	X		X	X	
	B	X	X	X	X	X	X
Test PWR.c.5.5 – IEEE Std 802.1Q Tags	A	X	X	X	X	X	X
	B	X	X	X	X	X	X
Test PWR.c.5.6 – TransportSpecific field checking upon receipt	A	X	X	X	X	X	X
	B	X	X		X	X	
Test PWR.c.6.1 – PTP Timescale	A	X	X		X	X	
Test PWR.c.6.2 – Current Utc Offset	A	X	X	X	X	X	X
Test PWR.c.6.3 – Grandmaster Clock Class	A	X			X		
Test PWR.c.6.4 – Grandmaster Degradation of Clock Class	A	X			X		
Test PWR.c.6.5 – Slave-Only Clock Class	A			X			
Test PWR.c.6.6 – Clock Accuracy	A	X			X		
Test PWR.c.6.7 – Holdover Drift for Grandmasters	A	X	X		X	X	
Test PWR.c.6.8 – GrandmasterID	A	X	X		X	X	
	B	X	X		X	X	
Test PWR.c.6.9 – Re-synchronization Behavior	A	X	X		X	X	
	B	X	X		X	X	
	C	X	X		X	X	
	D	X	X		X	X	
Test PWR.c.7.1 – Order of TLVs	A	X	X		X	X	
	B						X
Test PWR.c.7.2 – Profile-Specific TLV Default Field Values	A, B	X	X		X	X	X
Test PWR.c.7.3 – OrganizationId and OrganizationSubType Recognition	A	X	X	X	X	X	X

Test	Part	Ordinary Clock			Boundary Clock		Transparent Clock
		PrefGM	GMC	SO	PrefGM	GMC	
Test PWR.c.7.4 – Announce Messages without TLVs	A	X	X		X	X	
	B	X	X		X	X	
	C	X	X		X	X	
	D	X	X		X	X	
	E	X	X	X	X	X	
	F	X	X	X	X	X	
	G	X	X	X	X	X	
	H	X	X	X	X	X	
Test PWR.c.7.5 – ALTERNATE_TIME_OFFSET_INDICATOR TLV with Discontinuity	A	X	X	X	X	X	
	B	X	X	X	X	X	
Test PWR.c.7.6 – Sequence of Announce Messages before Discontinuity	A	X			X		
	B		X			X	
Test PWR.c.7.7 – ALTERNATE_TIME_OFFSET_INDICATOR TLV is not UTC	A	X	X		X	X	
	B	X	X		X	X	
Test PWR.c.7.8 – Boundary Clocks Forwarding ALTERNATE_TIME_OFFSET_INDICATOR	A				X	X	
Test PWR.c.8.1 – LocalTimeInaccuracy for Grandmaster Clocks	A	X	X		X	X	
Test PWR.c.8.2 – TimeInaccuracy for Grandmaster Clocks	A	X	X		X	X	
Test PWR.c.8.3 – LocalTimeInaccuracy for Transparent Clocks	A						X
	B						X
	C						X
Test PWR.c.8.4 – TimeInaccuracy for Transparent Clocks	A						X
Test PWR.c.8.5 – GrandmasterTimeInaccuracy and NetworkTimeInaccuracy for Grandmaster Clocks	A	X	X		X	X	
Test PWR.c.8.6 – GrandmasterTimeInaccuracy and NetworkTimeInaccuracy	A	X	X	X	X	X	X
	B	X	X	X	X	X	X
	C	X	X	X	X	X	X
Test PWR.c.9.1 – Clock Identity	A	X	X		X	X	
	B	X	X	X	X	X	X
Test PWR.c.9.2 – Peer Delay One-Step and Two-Step Ingress Ports	A	X	X	X	X	X	X
	B	X	X	X	X	X	X
Test PWR.c.9.3 – Sync One-Step and Two-Step Ingress Ports	A	X	X	X	X	X	X
	B	X	X	X	X	X	X
Test PWR.c.9.4 – One-Step or Two-Step Mode Egress Ports	A	X	X	X	X	X	X
	B						X
	C						X
Test PWR.c.9.5 – One-Step or Two-Step Flags	A, B						X
	C, D						X

Summary of Tests Prerequisites and Certification Classifier

Test PWR.c.1.1 – logAnnounceInterval

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC

Test PWR.c.1.2 – logSyncInterval

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC

Test PWR.c.1.3 – announceReceiptTimeout

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	PrefGM
B	BC, OC	GMC, not PrefGM

Test PWR.c.1.4 – logMinPdelayReqInterval

Part	Applies To Device Type	Prerequisite Conditions
A	All	

Test PWR.c.1.5 – priority1 and priority2

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC

Test PWR.c.1.6 – domainNumber

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC
B	OC	Slave-Only
C	BC, OC	none

Test PWR.c.2.1 – Peer Delay Mechanism

Part	Applies To Device Type	Prerequisite Conditions
A,B	All	Not applicable to slave-only endpoints that do not implement delay measurement.
C, D	OC, BC	GMC

Test PWR.c.2.2 – Pdelay_Req Message Field Values

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	None
B	TC	Syntonized to a domain
C	TC	Not syntonized

Test PWR.c.2.3 – Pdelay_Resp Message Field Values, One-Step Clock

Part	Applies To Device Type	Prerequisite Conditions
A	All	One-step Clock

Test PWR.c.2.4 – Peer Delay Turnaround Timestamps, One-Step Clock

Part	Applies To Device Type	Prerequisite Conditions
A	All	One-step Clock

Test PWR.c.2.5 – Peer Delay Message Field Values, Two-Step Clock

Part	Applies To Device Type	Prerequisite Conditions
A	All	Two-step Clock

Test PWR.c.2.6 – Peer Delay Turnaround Timestamps, Two-Step Clock

Part	Applies To Device Type	Prerequisite Conditions
A	All	Two-step Clock

Test PWR.c.2.7 – Restriction on Peer Delay Mechanism

Part	Applies To Device Type	Prerequisite Conditions
A	All	None
B	OC, BC	None
C	TC	None

Test PWR.c.2.8 – Mean Path Delay

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	One-step Clock
B	BC, OC	Two-step Clock
C	TC	One-step Clock
D	TC	Two-step Clock

Test PWR.c.2.9 – Independent Ports for Boundary Clocks

Part	Applies To Device Type	Prerequisite Conditions
A	BC	One-step Clock
B	BC	Two-step Clock

Test PWR.c.2.10 – Independent Ports for Transparent Clocks

Part	Applies To Device Type	Prerequisite Conditions
A	TC	SNMP, One-step Clock
B	TC	SNMP, Two-step Clock
C	TC	One-step Clock
D	TC	Two-step Clock

Test PWR.c.3.1 – Disqualified Announce Messages, by clockIdentity

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	Means of observing the DUT’s grandmaster

Test PWR.c.3.2 – Disqualified Announce Messages, by Most Recent

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	SNMP or means of observing the DUT’s grandmaster

Test PWR.c.3.3 – Disqualified Announce Messages, by Foreign Master Window

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	SNMP or means of observing the DUT’s grandmaster

Test PWR.c.3.4 – Disqualified Announce Messages, by stepsRemoved

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	No SNMP, Means of observing the DUT’s grandmaster
C, D	OC, BC	SNMP

Test PWR.c.3.5 – Disqualified Announce Messages, by alternateMasterFlag

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	SNMP or means of observing the DUT's grandmaster

Test PWR.c.3.6 – Data Set Comparison on a Single Port

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	Means of observing the DUT's grandmaster
C	OC, BC	SNMP

Test PWR.c.3.7 – Data Set Comparison on Multiple Ports

Part	Applies To Device Type	Prerequisite Conditions
A	BC	SNMP or means of observing the DUT's grandmaster

Test PWR.c.3.8 – State Decision Algorithm

Part	Applies To Device Type	Prerequisite Conditions
A-C	BC, OC	SNMP

Test PWR.c.3.9 – Steps Removed

Part	Applies To Device Type	Prerequisite Conditions
A - D	OC, BC	GMC
E - H	OC	Not GMC, SNMP or means of observing the DUT's grandmaster

Test PWR.c.3.10 – Source Port Identity

Part	Applies To Device Type	Prerequisite Conditions
A - D	OC, BC	GMC
E - H	OC	Not GMC, SNMP or means of observing the DUT's grandmaster

Test PWR.c.3.11 – Default Slave-only

Part	Applies To Device Type	Prerequisite Conditions
A	OC	Slave-Only, SNMP
B	OC	Slave-Only

Test PWR.c.4.1 – TimeInaccuracy, Traceability and Offset

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	GMC
C - F	OC	Slave-Only

Test PWR.c.4.2 – SNMP MIB Default Data Set for Ordinary and Boundary Clocks

Part	Applies To Device Type	Prerequisite Conditions
A,B	OC, BC	GMC or SNMP
C	BC	GMC or SNMP
D	OC	GMC or SNMP
E - H	OC, BC	GMC or SNMP
I, J	OC, BC	GMC
K, L	OC, BC	Slave-only, SNMP

Test PWR.c.4.3 – SNMP MIB Parent Data Set for Ordinary and Boundary Clocks

Part	Applies To Device Type	Prerequisite Conditions
A,B	OC, BC	GMC or SNMP

Test PWR.c.4.4 – SNMP MIB Time Properties Data Set for Ordinary and Boundary Clocks

Part	Applies To Device Type	Prerequisite Conditions
A -D	OC, BC	GMC or SNMP

Test PWR.c.4.5 – SNMP MIB Port Data Set for Ordinary and Boundary Clocks

Part	Applies To Device Type	Prerequisite Conditions
A-O	OC, BC	GMC or SNMP

Test PWR.c.4.6 – SNMP MIB Default Data Set for Transparent Clocks

Part	Applies To Device Type	Prerequisite Conditions
A-K	TC	SNMP

Test PWR.c.4.7 – SNMP MIB Port Data Set for Transparent Clocks

Part	Applies To Device Type	Prerequisite Conditions
A-D	TC	SNMP

Test PWR.c.4.8 – SNMP MIB Notifications

Part	Applies To Device Type	Prerequisite Conditions
A-D	OC, BC, TC	GMC or SNMP

Test PWR.c.5.1 – IEEE 802.3 Transport Mapping for Announce, Sync and Follow_Up Messages

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

Test PWR.c.5.2 – IEEE 802.3 Transport Mapping for Forwarded Announce, Sync and Follow_Up Messages

Part	Applies To Device Type	Prerequisite Conditions
A	TC	None

Test PWR.c.5.3 – IEEE 802.3 Transport Mapping for Peer Delay Messages

Part	Applies To Device Type	Prerequisite Conditions
A	All	None

Test PWR.c.5.4 – Multiple Priorities

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC, Multiple Priorities Allowed
B	All	Multiple Priorities Allowed

Test PWR.c.5.5 – IEEE Std 802.1Q Tags

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	None

Test PWR.c.5.6 – TransportSpecific field checking upon receipt

Part	Applies To Device Type	Prerequisite Conditions
A	All	IEEE 802.1 AS (gPTP) support must be reported if present.
B	GMC	IEEE 802.1 AS (gPTP) support must be reported if present.

Test PWR.c.6.1 – PTP Timescale

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

Test PWR.c.6.2 – Current Utc Offset

Part	Applies To Device Type	Prerequisite Conditions
A	All	SNMP or means of observing timePropertiesDS.currentUtcOffset

Test PWR.c.6.3 – Grandmaster Clock Class

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGM

Test PWR.c.6.4 – Grandmaster Degradation of Clock Class

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGM

Test PWR.c.6.5 – Slave-Only Clock Class

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	Slave-Only, SNMP or means of observing the DUT’s clockClass

Test PWR.c.6.6 – Clock Accuracy

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGM, Capable of disconnecting primary reference

Test PWR.c.6.7 – Holdover Drift for Grandmasters

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

Test PWR.c.6.8 – GrandmasterID

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	GMC

Test PWR.c.6.9 – Re-synchronization Behavior

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	GMC
C, D	OC, BC	GMC, SNMP or means of observing the DUT’s offset from grandmaster

Test PWR.c.7.1 – Order of TLVs

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC
B	TC	None

Test PWR.c.7.2 – Profile-Specific TLV Default Field Values

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	Not Slave-Only

Test PWR.c.7.3 – OrganizationId and OrganizationSubType Recognition

Part	Applies To Device Type	Prerequisite Conditions
A	All	SNMP or means of observing the DUT’s grandmasterTimeInaccuracy

Test PWR.c.7.4 – Announce Messages without TLVs

Part	Applies To Device Type	Prerequisite Conditions
A-D	OC, BC	GMC
E-H	OC, BC	SNMP or means of observing the DUT’s grandmaster

Test PWR.c.7.5 – ALTERNATE_TIME_OFFSET_INDICATOR TLV with Discontinuity

Part	Applies To Device Type	Prerequisite Conditions
A-B	OC, BC	SNMP or means of observing the DUT's alternate time

Test PWR.c.7.6 – Sequence of Announce Messages before Discontinuity

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGMC
B	OC, BC	GMC

Test PWR.c.7.7 – ALTERNATE_TIME_OFFSET_INDICATOR TLV is not UTC

Part	Applies To Device Type	Prerequisite Conditions
A-B	OC, BC	GMC

Test PWR.c.7.8 – Boundary Clocks Forwarding ALTERNATE_TIME_OFFSET_INDICATOR

Part	Applies To Device Type	Prerequisite Conditions
A	BC	None

Test PWR.c.8.1 – LocalTimeInaccuracy for Grandmaster Clocks

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

Test PWR.c.8.2 – TimeInaccuracy for Grandmaster Clocks

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	1PPS input, GMC

Test PWR.c.8.3 – LocalTimeInaccuracy for Transparent Clocks

Part	Applies To Device Type	Prerequisite Conditions
A	TC	No SNMP, Means of observing the DUT's LocalTimeInaccuracy
B	TC	SNMP
C	TC	Means of setting the DUT's LocalTimeInaccuracy

Test PWR.c.8.4 – TimeInaccuracy for Transparent Clocks

Part	Applies To Device Type	Prerequisite Conditions
A	TC	None

Test PWR.c.8.5 – GrandmasterTimeInaccuracy and NetworkTimeInaccuracy for Grandmaster Clocks

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

Test PWR.c.8.6 – GrandmasterTimeInaccuracy and NetworkTimeInaccuracy

Part	Applies To Device Type	Prerequisite Conditions
A	BC, TC	Means of observing the DUT's LocalTimeInaccuracy
B	BC, TC	SNMP
C	BC, TC	SNMP or means of observing the DUT's LocalTimeInaccuracy

Test PWR.c.9.1 – Clock Identity

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC
B	All	None

Test PWR.c.9.2 – Peer Delay One-Step and Two-Step Ingress Ports

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	SNMP or means of observing the DUT’s offset from master

Test PWR.c.9.3 – Sync One-Step and Two-Step Ingress Ports

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	SNMP or means of observing the DUT’s offset from master

Test PWR.c.9.4 – One-Step or Two-Step Mode Egress Ports

Part	Applies To Device Type	Prerequisite Conditions
A	All	None
B	TC	One-step Clock
C	TC	Two-step Clock

Test PWR.c.9.5 – One-Step or Two-Step Flags

Part	Applies To Device Type	Prerequisite Conditions
A	TC	One-step Clock
B	TC	One-step Clock
C	TC	Two-step Clock
D	TC	Two-step Clock

This publication is available free of charge from http://dx.doi.org/10.6028/NIST.IR.8002

SECTION PWR.c: 1588 Power Profile Conformance

Overview:

This selection of tests verifies the various requirements for 1588 Power Profile products defined in the IEEE C37.238 standard.

Comments and questions regarding the documentation or implementation of these tests are welcome and may be sent to powerProfileTest@nist.gov.

Notes:

Successful completion of all tests contained in this suite does not guarantee that the tested device will successfully operate with other 1588 Power Profile products. However, when combined with a satisfactory level of interoperability testing, these tests provide a reasonable level of confidence that the Device Under Test (DUT) will function properly in many 1588 Power Profile environments.

GROUP 1: PTP Attribute Values

Overview:

Ordinary and boundary clocks have two types of data sets, referred to as clock data sets and port data sets. The clock data sets include defaultDS, currentDS, and parentDS. The defaultDS attributes describe the clock. The currentDS attributes relate to synchronization. The parentDS attributes describe the parent (the clock to which this clock synchronizes) and the grandmaster (the clock at the root of the master-slave hierarchy). There is only one port data set, namely portDS. The default attribute values are the configuration of a Precision Time Protocol (PTP) device as it is delivered from the manufacturer. In IEEE Std C37.238-2011 all nodes support specific values for attributes.

The tests defined in this group validate specific attribute values from the clock data sets and port data sets shown in the table below. The values verified from the port data sets include the logAnnounceInterval, logSyncInterval, announceReceiptTimeout and logMinPdelayReqInterval. The values verified from the clock data sets include the priority1, priority2, slaveOnly, and domainNumber.

PTP Attribute Values

Attribute	Value	Range
portDS.logAnnounceInterval	0	None
portDS.logSyncInterval	0	None
portDS.announceReceiptTimeout	2 for all preferred grandmaster clocks, 3 for all other grandmaster-capable devices	None
portDS.logMinPdelayReqInterval	0	None
defaultDS.priority1	128 for grandmaster-capable devices 255 for slave-only devices	None
defaultDS.priority2	128 for grandmaster-capable devices 255 for slave-only devices	None
defaultDS.slaveOnly	FALSE for grandmaster-capable devices TRUE for slave-only devices	None
defaultDS.domainNumber	Default initialization value is 0	As specified by 7.1 and Table 2 of IEEE Std 1588-2008
transparentClockdefaultDS.primaryDomain	Default initialization value is 0	As specified by 7.1 and Table 2 of IEEE Std 1588-2008

Notes:

Test PWR.c.1.1 – logAnnounceInterval

Purpose: To validate the DUT's logAnnounceInterval

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.2
[2] IEEE Std 1588-2008: sub-clause 9.5.8

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-10-25 Preview release

Discussion: This test will validate the DUT's logAnnounceInterval value by observing the logMessageInterval field and frequency of Announce messages emitted from the DUT. The time between successive Announce messages is represented as $2^{\text{portDS.logAnnounceInterval}}$ seconds. Reference [1] states that the default value for portDS.logAnnounceInterval must be 0, hence Announce messages must be transmitted every 1 second. Reference [2] states that a node shall space Announce messages at ±30 % of $2^{\text{portDS.logAnnounceInterval}}$ seconds, with 90 % confidence. This translates to a minimum value of 0.7 s and a maximum value of 1.3 s. Refer to Appendix D: Calculations to calculate the mean, variance, standard deviation and with 90 % confidence. Common message headers are 34 octets long with the last octet being the logMessageInterval.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP with the following modifications: Connect all DUT ports other than DUT.TS1 (if any) to a device supporting Power Profile (a copy of the DUT is acceptable). This has the effect of maximizing the processor load on the DUT's CPU for PTP packet processing. If a Bridge is used, ensure the added Bridge has RSTP enabled to break network loops.

Test Procedure:

Part A: Default Initialization Value

A:1. Ensure all DUT ports are linked as described in the Test Setup.
A:2. Capture traffic received by Test Station 1 (TS1) for the duration of this test.
A:3. Observe 60 consecutive Announce intervals.
A:4. Calculate the 90 % confidence interval of the mean Announce interval from the observed samples using the calculations provided in Appendix D: Calculations.
 a. If the interval calculated is not fully within the 0.7 s and 1.3 s allowed range, repeat step A:3 and A:4, this time observe 600 Announce intervals.
A:5. Note any outliers. An outlier is considered to be any Announce intervals found to be greater than 2.5 s for preferred grandmasters, and 3.5 s for other grandmaster capable devices.
A:6. Observe the logMessageInterval value in 60 consecutive Announce messages.
A:7. If the device has more than one port, repeat steps A:1-2 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	No Announce messages received.
A:4	FAIL	The 90 % confidence interval for the mean of the observed Announce intervals is not fully within the 0.7 s and 1.3 s allowed range.
A:5	FAIL	Any outliers are found to be greater than 2.5 s for preferred grandmasters, or greater than 3.5 s for other grandmaster capable devices.
A:6	FAIL	The logMessageInterval value (1 octet at offset 33) in any Announce message is anything other than 0.
A:7	PASS	The 90 % confidence interval for the mean of the observed Announce intervals is fully within 0.7 s and 1.3 s; there are not outliers greater than 2.5 s (PrefGM) or 3.5 s (GMC); and, the value of the logMessageInterval observed in each Announce messages was 0.

Possible Problems: None

Test PWR.c.1.2 – logSyncInterval

Purpose: To validate the DUT's logSyncInterval

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.2
[2] IEEE Std 1588-2008: sub-clause 9.5.9

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-10-25 Preview release

Discussion: This test will validate the DUT's logSyncInterval value by observing the logSyncInterval field and frequency of Sync messages emitted from the DUT. The time between successive Sync messages is represented as $2^{portDS.logSyncInterval}$ seconds. Reference [1] states that the default value for portDS.logSyncInterval must be 0; hence Sync messages must be transmitted every 1 second. Reference [2] states that a node shall space Sync messages at ±30 % of $2^{portDS.logSyncInterval}$ seconds, with 90 % confidence. This translates to a minimum value of 0.7 s and a maximum value of 1.3 s. Refer to Appendix D: Calculations to calculate the mean, variance, standard deviation and with 90 % confidence. Common message headers are 34 octets long with the last octet being the logMessageInterval.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP with the following modifications: Connect all DUT ports other than DUT.TS1 (if any) to a device supporting Power Profile (a copy of the DUT is acceptable). This has the effect of maximizing the processor load on the DUT's CPU for PTP packet processing. If a Bridge is used, ensure the added Bridge has RSTP enabled to break network loops.

Test Procedure:

Part A: Default Initialization Value

A:1. Ensure all DUT ports are linked as described in the Test Setup.
A:2. Capture traffic received by TS1 for the duration of this test.
A:3. Observe 60 consecutive Sync intervals.
A:4. Calculate the 90 % confidence interval of the mean Sync interval from the observed samples using the calculations provided in Appendix D: Calculations.
    a. If the interval calculated is not fully within the 0.7 s and 1.3 s allowed range, repeat step A:3 and A:4, this time observe 600 Sync intervals.
A:5. Observe the logMessageInterval value in 60 Sync messages.
A:6. If the device has more than one port, repeat steps A:1-2 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	The 90 % confidence interval for the mean of the observed Sync intervals is not fully within the 0.7 s and 1.3 s allowed range.
A:6	FAIL	The logMessageInterval value (1 octet at offset 33) in any Sync message is anything other than 0.
A:7	PASS	The 90 % confidence interval for the mean of the observed Sync intervals is fully within 0.7 s and 1.3 s; and, the value of the logMessageInterval observed in each Sync messages was 0.

Possible Problems: None

Test PWR.c.1.3 – announceReceiptTimeout

Purpose: To validate the DUT's announceReceiptTimeout.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	PrefGM
B	BC, OC	GMC, not PrefGM

References: [1] IEEE Std 1588-2008: sub-clause 8.2.5.4.2
[2] IEEE Std C37.238-2011: sub-clause 5.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-1-18 Preview release

Discussion: The announceReceiptTimeout specifies the number of announceIntervals that have to pass without receipt of an Announce message before the occurrence of the event ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES. The value of portDS.announceReceiptTimeout shall be an integral multiple of announceInterval [1]. It must be 2 for all preferred grandmaster clocks, 3 for all other grandmaster-capable devices [2].

This test will also detect the DUT's portDS.announceReceiptTimeoutInterval by sending a constant stream of "better" Announce messages to the DUT then increasing the gap between the Announce messages until the DUT is seen to resume sending Announce messages. The Power Profile Announce interval is 1 s, thus for this test the announceReceiptTimeoutInterval must be 2 s for preferred grandmaster clocks and 3 s for all other grandmaster-capable devices.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Slow Rate of Announce Messages for Preferred Grandmaster Clocks

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Wait up to 10 s or for 3 Announce messages to be received from the DUT, and observe its priority1 value.

A:3. Send an Announce message every N seconds with a lower (better) priority1 value so that the DUT becomes a slave to TS1. N is initially 1.

A:4. Wait 10 s.

A:5. Observe whether any Announce messages come from DUT.TS1.

A:6. Repeat steps C:3 through C:5, increasing N by 0.5 until step C:5 observes Announce messages coming from the DUT, or when N is five (5).

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Announce messages are received.
A:5	FAIL	When N is 1-1.5 the DUT sends Announce messages.
A:5	FAIL	N reaches 2.5 and the DUT still does not send Announce messages.
A:5	PASS	When N is 2 or 2.5 the DUT sends Announce messages.

Part B: Slow Rate of Announce Messages for Other Grandmaster-Capable Devices

B:1. Capture traffic received by TS1 for the duration of this test.

B:2. Wait up to 10 s or for 3 Announce messages to be received from the DUT, and observe its priority1 value.

B:3. Send an Announce message every N seconds with a lower (better) priority1 value so that the DUT becomes a slave to TS1. N is initially 1.

B:4. Wait 10 s.

B:5. Observe whether any Announce messages come from DUT.TS1.

B:6. Repeat steps D:3 through D:5, increasing N by 0.5 s until step D:5 observes Announce messages coming from the DUT, or when N is five (5).

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Announce messages are received.
B:5	FAIL	When N is 1-2.5 s the DUT sends Announce messages.
B:5	FAIL	N reaches 3.5 s and the DUT still does not send Announce messages.
B:5	PASS	When N is 3 s or 3.5 s the DUT sends Announce messages.

Possible Problems: None

Test PWR.c.1.4 – logMinPdelayReqInterval

Purpose: To the DUT's logMinPdelayReqInterval

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	

References: [1] IEEE Std C37.238-2011: sub-clause 5.2
[2] IEEE Std 1588-2008: sub-clause 9.5.13

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-10-25 Preview release

Discussion: This test will validate the DUT's logMinPdelayReqInterval value by observing the frequency of Pdelay_Req messages. The time between successive Pdelay_Req messages must be no less than $2^{\text{portDS.logMinPdelayReqInterval}}$ seconds. Reference [1] states that the default value for portDS.logMinPdelayReqInterval must be 0. Reference [2] states that a node shall space Pdelay_Req messages at no less than $2^{\text{portDS.logMinPdelayReqInterval}}$ seconds on average. Refer to Appendix D: Calculations to calculate the mean, variance, standard deviation and with 90 % confidence.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP with the following modifications: Connect all DUT ports other than DUT.TS1 (if any) to a device supporting Power Profile (a copy of the DUT is acceptable). This has the effect of maximizing the processor load on the DUT's CPU for PTP packet processing. If a Bridge is used, ensure the added Bridge has RSTP enabled to break network loops.

Test Procedure:

Part A: Default Initialization Value

A:1. Ensure all DUT ports are linked as described in the Test Setup.
A:2. Capture traffic received by TS1 for the duration of this test.
A:3. Observe 10 consecutive Pdelay_Req intervals.
A:4. Calculate the 90 % confidence interval of the mean Pdelay_Req interval from the observed samples using the calculations provided in Appendix D: Calculations.
  a. If the interval calculated is not fully greater than 0.7 s, repeat step A:3 and A:4, this time observe 600 Pdelay_Req intervals.
A:5. If the device has more than one port, repeat steps A:1-2 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	The 90 % confidence interval for the mean of the observed Pdelay_Req intervals is not fully greater than 0.7 s.
A:5	PASS	The 90 % confidence interval for the mean of the observed Pdelay_Req intervals is fully greater than 0.7 s.

Possible Problems: None

Test PWR.c.1.5 – priority1 and priority2

Purpose: To validate the DUT's priority1 and priority2

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2012-10-25	Preview release

Discussion: This test will validate the DUT's priority1 and priority2 values by observing the priority1 and priority2 fields of Announce messages emitted from the DUT. At offset 47 and 52 of Announce messages, the grandmasterPriority1 and grandmasterPriority2 fields are each one octet long. The default value of priority1 and priority2 shall be 128 for grandmaster-capable devices and 255 for slave-only devices [1].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Grandmaster-Capable Clocks

A:1. Ensure that the DUT is in default setup.
A:2. Capture traffic received by TS1 for the duration of this test.
A:3. Wait up to 10 s or for 3 Announce messages to be received from the DUT.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	Three Announce messages are not received within 10 s.
A:3	FAIL	The grandmasterPriority1 field is not 128.
A:3	FAIL	The grandmasterPriority2 field is not 128.
A:3	PASS	The grandmasterPriority1 and the grandmasterPriority2 fields in each Announce message were 128.

Possible Problems: None

Test PWR.c.1.6 – domainNumber

Purpose: To validate the DUT's domainNumber

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	GMC
B	OC	Slave-Only
C	BC, OC	none

References: [1] IEEE Std 1588-2008: sub-clause 8.2.1.4.3
[2] IEEE Std C37.238-2011: sub-clause 5.2
[3] IEEE Std 1588-2008: sub-clause 7.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-11-04	Preview release

Discussion: This test will validate the DUT's domainNumber value by observing the domainNumber field of Announce messages and Pdelay_Resp messages emitted from the DUT. Common message headers are 34 octets long with the 5th octet stating the domainNumber [1]. The domainNumber can be configured to an integer in the range of 0 to 127, but by default the value must be initialized to 0 [2, 3].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: Domain Number in Sync and Follow_Up messages
A:1. Ensure that the DUT is in default setup.
A:2. Capture all PTP messages from the DUT.
A:3. From TS1, send one Pdelay_Req message to the DUT every second.
A:4. Wait until TS1 receives an Announce message.
A:5. Wait at least 10 s. Verify the domainNumber of all received PTP messages from the DUT.

Observable Results:

Part:Step	Status	Description
A	N/A	The DUT is not GMC.
A:5	FAIL	No Sync or Follow_Up (if two-step) messages are received.
A:5	FAIL	Any of the received PTP messages have a domainNumber other than 0.
A:5	PASS	All observed PTP messages have a domainNumber of 0.

Part B: Domain Number in Pdelay_Req, Announce, Pdelay_Resp, and Pdelay_Resp_Follow_Up messages

B:1. Ensure that the DUT is in default setup.
B:2. Capture all PTP messages from the DUT.
B:3. From TS1, send one Pdelay_Req message to the DUT every second.
B:4. Wait for the arrival of at least one of each of the following message types: Pdelay_Req, Announce, Pdelay_Resp, and if the DUT is two-step, Pdelay_Resp_Follow_Up. Observe the domainNumber of each.

Observable Results:

Part:Step	Status	Description
B:4	FAIL	TS1 does not receive each of the message types.
B:4	FAIL	Any of the received PTP messages has a domainNumber other than 0.
B:4	PASS	All observed PTP messages have a domainNumber of 0.

Part C: Not Accepting messages from other Domains

C:1. Ensure that the DUT is in default setup.
C:2. Capture traffic received and sent by TS1 for the duration of this test.
C:3. Have TS1 send Pdelay_Req messages with the value of the Domain Number 0.
C:4. Wait up to 10 s or for 3 Pdelay_Resp Messages to be received from the DUT.
C:5. Have TS1 send Pdelay_Req messages with the value of the Domain Number 2.
C:6. Wait up to 10 s or for 3 Pdelay_Resp Messages to be received from the DUT.

Observable Results:

Part:Step	Sta-tus	Description
C:4	FAIL	A Pdelay_Resp message is not received from the DUT.
C:6	FAIL	A Pdelay_Resp message is received from the DUT.
C:6	PASS	The DUT only accepts and responds to messages with a Domain Number 0.

Possible Problems: None

GROUP 2: Path Delay Mechanism

Overview:

This group covers requirements defined in IEEE 1588-2008 sub-clause 11.4, "Peer delay mechanism". In the IEEE Std C37.238-2011, this is the only path delay mechanism in operation. The peer delay mechanism involves Node-A sending Pdelay_Req messages and Node-B responding with Pdelay_Resp and Pdelay_Resp_Follow_Up messages. This messaging process allows Node-A to calculate the meanPathDelay.

Test PWR.c.2.1 – Peer Delay Mechanism

Purpose: To verify that the peer delay mechanism is the only path delay measurement mechanism in operation.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A,B	All	Not applicable to slave-only endpoints that do not implement delay measurement.
C, D	OC, BC	GMC

References: [1] IEEE Std 1588-2008: sub-clause 11.4
[2] IEEE Std C37.238-2011: sub-clause 5.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-01-24 Preview release

Discussion: This test will verify that the peer delay mechanism [1] is the only path delay measurement mechanism used by the DUT. The peer delay mechanism measures the link delay between two communicating ports with Pdelay_Req, Pdelay_Resp and possibly Pdelay_Resp_Follow_Up messages.

This test will also verify that in ordinary and boundary clocks the peer delay mechanism operates independently of whether the port is in the master or slave state. This will be tested by observing the peer delay messages emitted from the DUT while it is master (lower priority1) and then while it is slave (higher priority1).

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Does the DUT respond to Sync messages with Delay_Req messages?

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait for TS1 to receive Announce messages. Note the *priority1* value in the received messages.
A:3. From TS1, send Announce messages every second with a *priorty1* value less (better) than that transmitted by the DUT.
A:4. Send Sync messages from TS1.
A:5. Wait 10 s. Notice any Delay_Req messages received from the DUT.

Observable Results:

Part:Step	Status	Description
A:5	FAIL	Any Delay_Req messages are received.
A:5	PASS	No Delay_Req messages are received.

Part B: Does the DUT respond to Pdelay_Req messages with Delay_Resp messages?

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Send Pdelay_Req messages from TS1.
B:3. Wait 10 s. Notice any Delay_Resp messages received from the DUT.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	Any Delay_Resp messages are received.
B:3	PASS	No Delay_Resp messages are received.

Part C: Does the DUT use the peer delay mechanism regardless of master port state?

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. Wait for TS1 to receive Announce messages. Note the *priority1* value in the received messages.
C:3. From TS1, send Announce messages every second with a *priorty1* value less (better) than that transmitted by the DUT.
C:4. Send Pdelay_Req messages from TS1.
C:5. Wait 10 s. Note any Pdelay messages exchanged.

Observable Results:

Part:Step	Status	Description
C:5	FAIL	Pdelay_Req and Pdelay_Resp messages are not received.
C:5	PASS	The peer delay mechanism works in master state.

Part D: Does the DUT use the peer delay mechanism regardless of slave port state?

D:1. Wait for TS1 to receive Announce messages. Note the *priority1* value in the received messages.
D:2. From TS1, send 3 Announce messages every second with a *priorty1* value greater (worse) than that transmitted by the DUT.
D:3. Send Pdelay_Req messages from TS1.
D:4. Wait 10 s. Note any Pdelay messages exchanged.

Observable Results:

Part:Step	Status	Description
D:5	FAIL	Pdelay_Req and Pdelay_Resp messages are not received.
D:5	PASS	The peer delay mechanism works in slave state.

Possible Problems: None

Test PWR.c.2.2 – Pdelay_Req Message Field Values

Purpose: To validate the DUT's Pdelay_Req message field values.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	None
B	TC	Syntonized to a domain
C	TC	Not syntonized

References: [1] IEEE Std 1588-2008: sub-clause 11.4.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2012-02-19	Preview release

Discussion: This test will verify that Pdelay_Req messages are prepared and sent correctly by observing the domainNumber, correctionField and originTimestamp of Pdelay_Req messages emitted from the DUT. The first step of the peer delay mechanism is for the delay requester, Node-A, to prepare and send Pdelay_Req messages [1]. The correctionField shall be set to zero. The originTimestamp shall be set to zero or an estimate no worse than ± 1 s of the egress timestamp, t_1, of the Pdelay_Req message.

The domainNumber field default initialization value was tested in Test Pwr.c.7 – defaultDS.domainNumber. This test generalizes to other domainNumber values. If Node-A is an ordinary or boundary clock, the domainNumber field shall be set to the domain of Node-A. If Node-A is a syntonized peer-to-peer transparent clock, the domainNumber field shall be set to the domain being measured. The domain being measured is either the primary syntonization domain or one of the alternate domains if syntonization to multiple domains is implemented. If Node-A is not a syntonized peer-to-peer transparent clock, the domainNumber field shall be set to zero.

Refer to Appendix C: Table 1: Pdelay_Req Message Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: BC, OC

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Wait up to 10 s for 3 Pdelay_Req messages to be received from the DUT.
  a. Observe and print the value of the domainNumber.
  b. Observe and print the value of the correctionField.
  c. Observe and print the value of the originTimestamp.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	Three Pdelay_Req messages are not received.
A:2	FAIL	The domainNumber is not the domain of the DUT node.
A:2	FAIL	The correctionField is not zero.
A:2	FAIL	The originTimestamp is not zero.
A:2	PASS	In all Pdelay_Req messages the domainNumber is that of the DUT's node, the correctionField is zero and the originTimestamp is zero.

Part B: Syntonized TC

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Through vendor specific means, ensure the DUT is syntonized to the TS on domain 1.
B:3. Wait up to 10 s for 3 Pdelay_Req messages to be received from the DUT.
  a. Observe and print the value of the domainNumber.
  b. Observe and print the value of the correctionField.
  c. Observe and print the value of the originTimestamp.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	Three Pdelay_Req messages are not received.
B:3	FAIL	The domainNumber is not domain 1.
B:3	FAIL	The correctionField is not zero.
B:3	FAIL	The originTimestamp is not zero.
B:3	PASS	In all Pdelay_Req messages the domainNumber is 1, the correctionField is zero and the originTimestamp is zero.

Part C: Non-Syntonized TC

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. Wait up to 10 s for 3 Pdelay_Req messages to be received from the DUT.
  a. Observe and print the value of the domainNumber.
  b. Observe and print the value of the correctionField.
  c. Observe and print the value of the originTimestamp.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	Three Pdelay_Req messages are not received.
C:2	FAIL	The domainNumber is not zero
C:2	FAIL	The correctionField is not zero.
C:2	FAIL	The originTimestamp is not zero.
C:2	PASS	In all Pdelay_Req messages the domainNumber is zero, the correctionField is zero and the originTimestamp is zero.

Possible Problems: The values of the correctionField may vary if asymmetry corrections are required.

Test PWR.c.2.3 – Pdelay_Resp Message Field Values, One-Step Clock

Purpose: To validate Pdelay_Resp message field values in one-step clocks.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	One-step Clock

References: [1] IEEE Std 1588-2008: sub-clause 11.4.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-02-19 Preview release

Discussion: This test will verify that Pdelay_Resp messages are prepared and sent correctly by observing the domainNumber, correctionField, sequenceId, requestReceiptTimestamp and requestingPortIdentity of Pdelay_Resp messages emitted from the DUT. For one-step clocks, the second step of the peer delay mechanism is for the delay responder, Node-B, to prepare and send a Pdelay_Resp message according to [1]. Four fields of the Pdelay_Resp message are copied from corresponding fields in the received Pdelay_Req message, as indicated in Table 2: Pdelay_Resp Message Fields. The correctionField should be first copied from the correctionField of the Pdelay_Req message and then increased by the turnaround time. The requestReceiptTimestamp field of the Pdelay_Resp message shall be set to 0.
Refer to Appendix C Table 2: Pdelay_Resp Message Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Pdelay_Resp Field Values

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Send a Pdelay_Req message every second from TS1. Alternate correctionField values between 0 and 0x0000 4000 0000 0000 (approximately 1 s).
  a. Print the value of the domainNumber in each Pdelay_Req sent.
  b. Print the value of the correctionField in each Pdelay_Req sent.
  c. Print the value of the sequenceId in each Pdelay_Req sent
  d. Print the value of the sourcePortIdentity in each Pdelay_Req sent.

A:3. Wait up to 10 s for 3 Pdelay_Resp messages to be received from the DUT.
  a. Observe and print the value of the domainNumber.
  b. Observe and print the value of the correctionField.
  c. Observe and print the value of the sequenceId.
  d. Observe and print the value of the requestReceiptTimestamp.
  e. Observe and print the value of the requestingPortIdentity.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	No Pdelay_Resp message is received.
A:3	FAIL	The domainNumber field of the Pdelay_Resp message is not the same as the domain-Number of the Pdelay_Req message.
A:3	FAIL	The correctionField of each Pdelay_Resp message is not greater than the correctionField of the corresponding Pdelay_Req message.
A:3	WARN	The correctionFields of alternate Pdelay_Resp message do not oscillate in value as the cor-rectionFields of the corresponding Pdelay_Req messages do.
A:3	FAIL	The sequenceId field of the Pdelay_Resp message is not the same as the sequenceId field of the immediately preceding Pdelay_Req message.
A:3	FAIL	The requestReceiptTimestamp field of the Pdelay_Resp message is not zero.
A:3	FAIL	The requestingPortIdentity field of the Pdelay_Resp message is not the same as the sourcePortIdentity field of the Pdelay_Req message.
A:3	PASS	All fields of the Pdelay_Resp messages are correct.

Possible Problems: The values of the correctionField may vary if asymmetry corrections are required.

Test PWR.c.2.4 – Peer Delay Turnaround Timestamps, One-Step Clock

Purpose: To verify that the correctionField of the Pdelay_Resp message is reasonable in one-step clocks.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	One-step Clock

References: [1] IEEE Std 1588-2008: sub-clause 11.4.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-02-19 Preview release

Discussion: This test will validate the correctionField of the DUT's Pdelay_Resp messages by comparing it to a known upper bound. For informational purposes the test also calculates the mean and variance of the correctionField values; refer to Appendix D: Calculations

For one-step clocks, the second step of the peer delay mechanism is for the delay responder, Node-B, to prepare and send a Pdelay_Resp message according to [1]. The correctionField must be copied from the correctionField of the Pdelay_Req message and then increased by the turnaround time. If there is no asymmetry correction then the correctionField of a Pdelay_Req message shall be 0[1]. In this test the correctionField in Pdelay_Req messages sent by the test station will be 0, so the correctionField observed in the DUT's Pdelay_Resp messages will be the DUT's indication of its Pdelay turnaround time. This turnaround time must not be greater than the time between the test station's sending of the Pdelay_Req and the test station's receiving of the corresponding Pdelay_Resp, commonly designated as $t_4 – t_1$.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: correctionField

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Send a Pdelay_Req message every second from TS1.
A:3. Wait 10 s for Pdelay_Resp messages to be received from the DUT.
A:4. For one minute record the test station's send-to-receive time difference $t_4 – t_1$ and the correctionField of the corresponding Pdelay_Resp message received from DUT.
A:5. Calculate the mean and the variance of the correctionField values.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	Fewer than 55 Pdelay_Resp messages are received.
A:4	FAIL	The correctionField value in each of the Pdelay_Resp messages is not greater than the correctionField value from its corresponding Pdelay_Req message, i.e., 0.
A:4	FAIL	The correctionField value in any Pdelay_Resp message is greater than $t_4 – t_1$ where t_1 is the departure time of the Pdelay_Req from TS1 and t_4 is the arrival time of the DUT's Pdelay_Resp at TS1.
A:5	INFO	The mean of the correctionField values is reported.
A:5	INFO	The variance of the correctionField values is reported.
A:5	PASS	The correctionFields of the Pdelay_Resp messages are all greater than 0 and less than $t_4 – t_1$.

Possible Problems: The values of the correctionField may vary if asymmetry corrections are required.

Test PWR.c.2.5 – Peer Delay Message Field Values, Two-Step Clock

Purpose: To validate Pdelay_Resp and Pdelay_Resp_Follow_Up message field values in two-step clocks.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	Two-step Clock

References: [1] IEEE Std 1588-2008: sub-clause 11.4.3
[2] IEEE Std C37.238-2011: sub-clause 5.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-02-19 Preview release

Discussion: This test will verify that Pdelay_Resp and Pdelay_Resp_Follow_Up messages are prepared and sent correctly by observing fields in such messages emitted from the DUT. For two-step clocks, the second step of the peer delay mechanism is for the delay responder, Node-B, to prepare and send Pdelay_Resp and Pdelay_Resp_Follow_Up messages according to [1] and [2]. The domainNumber, sequenceId, and requestingPortIdentity fields of the Pdelay_Resp messages should be copied from corresponding Pdelay_Req message fields as specified in Table 2. The correctionField of the Pdelay_Resp should be set to zero and then reduced by any fractional nanosecond portion of Pdelay_Req arrival time t_2. The requestReceiptTimestamp field should be set to the seconds and nanoseconds portion of t_2.

The domainNumber, correctionField, sequenceId, and requestingPortIdentity fields of the Pdelay_Resp_Follow_Up messages should be copied from corresponding Pdelay_Req message fields as specified in Table 3. The responseOriginTimestamp field should be set to the seconds and nanoseconds portion of Pdelay_Resp departure time t_3. Then any fractional nanosecond portion of t_3 should be added to correctionField.

Refer to Appendix C Table 1: Pdelay_Req Message Fields
Refer to Appendix C Table 2: Pdelay_Resp Message Fields
Refer to Appendix C Table 3: Pdelay_Resp_Follow_Up Message Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Field Values

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Send a Pdelay_Req message every second from TS1.
A:3. Wait up to 10 s for Pdelay_Resp and Pdelay_Resp_Follow_Up messages to be received from the DUT.
A:4. For one minute record the correctionField and requestReceiptTimestamp of each received Pdelay_Resp message and the correctionField and responseOriginTimestamp of each received Pdelay_Resp_Follow_Up message.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	No Pdelay_Resp message is received.
A:3	FAIL	The domainNumber field of a Pdelay_Resp message is not the same as the domainNumber from the Pdelay_Req message.
A:3	FAIL	The correctionField of a Pdelay_Resp message is larger than a nanosecond.
A:3	FAIL	The sequenceId field of a Pdelay_Resp message is not the same as the sequenceId field from the Pdelay_Req message.
A:3	FAIL	In a Pdelay_Resp message, the requestReceiptTimestamp.nanoseconds > 999,999.999.
A:3	FAIL	The requestingPortIdentity field of a Pdelay_Resp message is not the same as the sourcePortIdentity field from the Pdelay_Req message.
A:3	FAIL	No Pdelay_Resp_Follow_Up message is received.
A:3	FAIL	The domainNumber field of a Pdelay_Resp_Follow_Up message is not the same as the domainNumber from the Pdelay_Req message.
A:3	FAIL	The correctionField of a Pdelay_Resp_Follow_Up message is greater than a nanosecond.
A:3	FAIL	The sequenceId field of a Pdelay_Resp_Follow_Up message is not the same as the sequenceId field from the Pdelay_Req message.
A:3	FAIL	In a Pdelay_Resp_Follow_Up message, the responseOriginTimestamp.nanoseconds > 999,999.999.
A:3	FAIL	The requestingPortIdentity field of a Pdelay_Resp_Follow_Up message is not the same as the sourcePortIdentity field from the Pdelay_Req message.
A:3	INFO	The Pdelay_Resp message should be transmitted as soon as possible after the receipt of the associated Pdelay_Req message.
A:3	INFO	The Pdelay_Resp_Follow_Up message should be transmitted as soon as possible after the transmission of the associated Pdelay_Resp message.
A:3	PASS	All fields of the Pdelay_Resp and Pdelay_Resp_Follow_Up messages are correct.

Possible Problems: The values of the correctionField may vary if asymmetry corrections are required.

Test PWR.c.2.6 – Peer Delay Turnaround Timestamps, Two-Step Clock

Purpose: To verify that the timestamp fields and correctionField values in Pdelay_Resp and Pdelay_Resp_Follow_Up messages are reasonable in two-step clocks.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	Two-step Clock

References: [1] IEEE Std 1588-2008: sub-clause 11.4.3
[2] IEEE Std C37.238-2011: sub-clause 5.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-02-19 Preview release

Discussion: This test will validate the fields in the DUT's Pdelay_Resp and Pdelay_Resp_Follow_Up messages that are used to indicate the DUT's peer delay turnaround time, typically represented as $t_3 - t_2$. The fields being checked are correctionField and requestReceiptTimestamp in Pdelay_Resp messages and correctionField and response-OriginTimestamp in Pdelay_Resp_Follow_Up messages. The turnaround time is validated by comparing it to a known upper bound. For informational purposes the test also calculates the mean and variance of the turnaround time; refer to Appendix D: Calculations
For two-step clocks, the second step of the peer delay mechanism is for the delay responder, Node-B, to prepare and send Pdelay_Resp and Pdelay_Resp_Follow_Up messages according to [1]. The requestReceiptTimestamp field should be set to the seconds and nanoseconds portion of t_2.

Two-step devices *should* use 11.4.3 option (c) 8 of IEEE Std 1588-2008 to populate the timestamp and correction fields of Pdelay_Resp and Pdelay_Resp_Follow_Up messages [2]. (This '*should*' is under review by the Working Group H24/SubC7 in efforts to change it to a 'shall'.) After the test station has received both response messages from the DUT, the test station can deduce the DUT's Pdelay turnaround time $t_3 - t_2$ from the following.

$$t_3 = responseOriginTimestamp + cF\ (both\ in\ Pdelay\_Resp\_Follow\_Up)$$
$$t_2 = requestReceiptTimestamp - cF\ (both\ in\ Pdelay\_Resp)$$

The test station will not add any asymmetry correction to its Pdelay_Req correctionField, so this will not need to be considered in the Pdelay_Resp_Follow_Up correctionField (shown as 0 in table 1). The DUT's Pdelay turnaround time, $t_3 - t_2$, must not be greater than the time between when the test station sends the Pdelay_Req and when the test station receives the corresponding Pdelay_Resp, commonly designated as $t_4 - t_1$.

Refer to Appendix C Table 1: Pdelay_Req Message Fields
Refer to Appendix C Table 2: Pdelay_Resp Message Fields
Refer to Appendix C Table 3: Pdelay_Resp_Follow_Up Message Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Turnaround Time

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Send a Pdelay_Req message every second from TS1. Alternate correctionField values between 0 and 0x0000 4000 0000 0000 (approximately 1 s).

A:3. Wait up to 10 s for Pdelay_Resp messages to be received from the DUT.

A:4. For one minute record the test station's send-to-receive time difference $t_4 – t_1$ and the turnaround time of the corresponding pair of Pdelay_Resp and Pdelay_Resp_Follow_Up messages received from DUT.TS1.

A:5. Calculate the mean and the variance of the correctionField values.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	Fewer than 55 Pdelay_Resp/Pdelay_Resp_Follow_Up message pairs are received.
A:3	WARN	The correctionFields of alternate Pdelay_Resp message do not oscillate in value as the correctionFields of the corresponding Pdelay_Req messages do.
A:4	FAIL	The turnaround time for each of the Pdelay_Resp and Pdelay_Resp_Follow_Up message pairs is not greater than 0.
A:4	FAIL	The turnaround time for any Pdelay_Resp/Pdelay_Resp_Follow_Up message pair is greater than $t_4 – t_1$ where t_1 is the departure time of the Pdelay_Req from TS1 and t_4 is the arrival time of the DUT's Pdelay_Resp at TS1.
A:5	INFO	The mean of the turnaround times is reported.
A:5	INFO	The variance of the turnaround times is reported.
A:5	PASS	The turnaround times of the DUT's Pdelay_Resp/Pdelay_Resp_Follow_Up messages are all greater than 0 and less than $t_4 – t_1$.

Possible Problems: The values of the correctionField may vary if asymmetry corrections are required.

Test PWR.c.2.7 – Restriction on Peer Delay Mechanism

Purpose: To verify that the proper action is taken after the receipt of zero or multiple Pdelay_Resp messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	None
B	OC	None
C	BC	None
D	TC	None

References: [1] IEEE Std 1588-2008: sub-clause 11.4.4

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-02-19 Preview release

Discussion: This test will verify that the proper action is taken after the DUT receives a varying number of Pdelay_Resp messages by observing whether the DUT port retransmits a Pdelay_Req message or enters the FAULTY state. Specific actions should be taken after the delay requester, Node-A, receives 0, 1 or multiple Pdelay_Resp messages for a transmitted Pdelay_Req [1]. The receipt of multiple responses can be detected by observing that the sourcePortIdentity fields of the Pdelay_Resp messages differ. When no Pdelay_Resp message is received, Node-A should periodically retransmit a Pdelay_Req message to check for the appearance of Node-B. The standard does not specify a retransmission rate, so this test produces a result of WARN if no retransmitted Pdelay_Req is received within 10 s. When a single Pdelay_Resp message is received, the protocol of 11.4 should be executed. When multiple Pdelay_Resp messages are received, ordinary and boundary clock ports should enter the FAULTY state, and peer-to-peer transparent clocks should enter a fault condition. For this case, the device may periodically retransmit a Pdelay_Req message and the port must discard received Sync and Follow_Up messages.

Refer to Appendix C Table 4: Action after Receipt of Pdelay_Resp Message

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: 0 Pdelay_Resp Received

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s for 3 Pdelay_Req messages to be received from the DUT.
A:3. Respond to five consecutive Pdelay_Req messages from the DUT with Pdelay_Resp and Pdelay_Resp_Follow_Up messages, observing whether the DUT continues to send Pdelay_Req messages.
A:4. Stop responding to the DUT's Pdelay_Resp messages.
A:5. Wait up to 10 s for a Pdelay_Req message to be received from the DUT.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Pdelay_Req message is received.
A:3	FAIL	The DUT stops sending Pdelay_Req messages.
A:5	WARN	The DUT does not transmit another Pdelay_Req within 10 s.
A:5	PASS	The DUT continues to transmit Pdelay_Req messages.

Part B: Multiple Pdelay_Resp Received on OC

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Wait up to 10 s or for a Pdelay_Req message to be received from the DUT.
B:3. Respond to each Pdelay_Req message with one valid Pdelay_Resp message from TS1 for the duration of this test.
B:4. Send Pdelay_Req messages from TS1 to the DUT for the duration of this test.
B:5. With TS1 as grandmaster, send Sync and Follow_Up messages from TS1 for the duration of this test.
B:6. Create a jump in TS1's time by sending Sync and Follow_Up messages from TS1 with timestamps increased by 10.5 s.
B:7. Observe the DUT's behavior after this jump has occurred until the DUT's time is synced to TS1 or 120 s goes by.
B:8. Respond to the next incoming Pdelay_Req message from the DUT with two Pdelay_Resp messages, each with differing sourcePortIdentity fields.
B:9. Using SNMP or a vendor-provided method, observe whether DUT.TS1 enters the FAULTY state.
B:10. Record the time it takes the DUT to send the next Pdelay_Req message.
B:11. Go back to responding to each incoming Pdelay_Req message from the DUT with one Pdelay_Resp message.
B:12. Stop sending Sync and Follow_Up messages to the DUT for 30 s and observe the DUT's behavior.
B:13. Resume sending Sync and Follow_Up messages to the DUT.
B:14. Respond to the next incoming Pdelay_Req message from the DUT with two Pdelay_Resp messages, each with differing sourcePortIdentity fields.
B:15. At the same time, create a jump in TS1's time by sending Sync and Follow_Up messages from TS1 with timestamps increased by 10.5 s.
B:16. Observe the DUT's behavior after this jump has occurred and using SNMP or a vendor-provided method, observe whether DUT.TS1 enters the FAULTY state.
B:17. When the next Pdelay_Req message is received, resume normal response with just one Pdelay_Resp message.
B:18. At the same time, send Sync and Follow_Up messages as if the jump never occurred (decrement the timestamps by 10.5 s).
B:19. Observe the DUT's behavior for the same amount of time it took to retransmit another Pdelay_Req message in step B:10.
B:20. Using SNMP or a vendor-provided method, observe whether DUT.TS1 enters the FAULTY state.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req message is received.
B:7	WARN	The DUT's behavior does not show a jump in the time.
B:7	INFO	Note the DUT's behavior after the jump.
B:9	FAIL	The device does not enter the FAULTY state.
B:9	FAIL	No FaultyState notification is received indicating the device has entered FAULTY state.
B:10	FAIL	No Pdelay_Req message is received.
B:10	INFO	Note the time it took the DUT to retransmit another Pdelay_Req message after the faulty response.
B:12	FAIL	The DUT fails to operate.
B:12	INFO	Note the DUT's behavior when it doesn't receive Sync and Follow_Up messages.
B:16	FAIL	The device does not enter the FAULTY state.
B:16	FAIL	No FaultyState notification is received indicating the device has entered FAULTY state.
B:16	FAIL	The DUT behaves as if there is a jump in time, similar to the behavior noted in step B:7.
B:19	FAIL	No Pdelay_Req message is received.
B:20	FAIL	The device remains in FAULTY state.
B:21	PASS	The device enters the FAULTY state when two Pdelay_Resp messages are received in response to two Pdelay_Req messages and discards Sync and Follow_Up messages when in this state.

Part C: Multiple Pdelay_Resp Received on BC

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. With TS2 as grandmaster, send valid Sync and Follow_Up messages from TS2.
C:3. Wait up to 10 s for the DUT to forward the Sync and Follow_Up messages received at DUT.TS2 out DUT.TS1.
C:4. Wait up to 10 s for 3 Pdelay_Req messages to be received from the DUT.
C:5. From TS1 and within a single Pdelay_Req interval respond to a Pdelay_Req message with two Pdelay_Resp messages, each with differing sourcePortIdentity fields.
C:6. Using SNMP or a vendor-provided method, observe whether DUT.TS1 enters the FAULTY state.
C:7. Observe whether DUT.TS1 discontinues forwarding Sync and Follow_Up messages received from TS2.

Observable Results:

Part:Step	Status	Description
C:3	FAIL	Sync and Follow_Up messages are not forwarded.
C:4	FAIL	No Pdelay_Req message is received.
C:6	FAIL	The device does not enter the FAULTY state.
C:6	FAIL	No FaultyState notification is received indicating the device has entered FAULTY state.
C:7	FAIL	The device continues forwarding Sync and Follow_Up messages.
C:7	PASS	The device enters the FAULTY state and stops forwarding Sync and Follow_Up messages.

Part D: Multiple Pdelay_Resp Received on TC

D:1. Capture traffic received by TS1 for the duration of this test.
D:2. With TS2 as grandmaster, send valid Sync and Follow_Up messages from TS2.
D:3. Wait up to 10 s for the DUT to forward the Sync and Follow_Up messages received at DUT.TS2 out DUT.TS1.
D:4. Wait up to 10 s for 3 Pdelay_Req messages to be received from the DUT.
D:5. From TS1 and within a single Pdelay_Req interval respond to a Pdelay_Req message with two Pdelay_Resp messages, each with differing sourcePortIdentity fields.
D:6. Using SNMP or a vendor-provided method, observe whether DUT.TS1 enters the FAULTY state.
D:7. Observe whether DUT.TS1 discontinues forwarding Sync and Follow_Up messages received from TS2.

Observable Results:

Part:Step	Status	Description
D:2	FAIL	Sync and Follow_Up messages are not forwarded.
D:3	FAIL	No Pdelay_Req message is received.
D:3	FAIL	The device does not enter a fault condition (not to be confused with FAULTY state).
D:3	FAIL	The device continues forwarding Sync and Follow_Up messages.
D:3	PASS	The device enters the fault condition and stops forwarding Sync and Follow_Up messages.

Possible Problems: None

Test PWR.c.2.8 – Mean Path Delay

Purpose: To verify that the meanPathDelay is computed correctly.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	One-step Clock
B	BC, OC	Two-step Clock
C	TC	One-step Clock
D	TC	Two-step Clock

References: [1] IEEE Std 1588-2008: sub-clause 11.4.3
[2] IEEE Std 1588-2008: sub-clause 11.5.2.2
[3] IEEE Std C37.238-2011: sub-clause 5.3

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames on a common time base.

Modification History: 2013-03-06 Preview release

Discussion: This test will validate the DUT's meanPathDelay value by comparing it with the known mean path delay. Since neither the 1588 standard nor the Power Profile establish any accuracy requirements for meanPathDelay, this test just gives a warning if the DUT's meanPathDelay is not within 100 ns of the actual mean path delay. The known mean path delay is derived by measuring the cable length. The speed of light in Ethernet cable is 66 % of *c*, or 5.0 ns/m, so every 1.0 m of cable contributes 5.0 ns to the one-way path delay. The test uses the mean of several path delay measurements. In either case, one-step or two-step, the mean path delay value should not vary by much for the requesting and responding nodes when the test setup remains the same. To validate this, the mean and variance of the meanPathDelay is calculated;
refer to Appendix D: Calculations
To validate the value of the meanPathDelay in transparent clocks the correctionField of the forwarded Sync messages, for one-step clocks, and Follow_Up messages, for two-step clocks, will be observed [2].

Devices shall measure and calculate the meanPathDelay for each instance of a peer delay measurement. For one-step clocks the calculation for the meanPathDelay is shown below [1].

$$mPD = \frac{(t_4 - t_1) - correctionField\ of\ Pdelay\_Resp}{2}$$

For two-step clocks the calculation for the meanPathDelay is shown below.

$$mPD = \frac{(t_4 - t_1) - (responseOriginTimestamp - requestReceiptTimestamp) - correctionFields}{2}$$

The Working Group H24/SubC7 is reviewing the exception, currently stated in IEEE C37.238 [3], that slave-only devices that are the last connection are not required to perform path delay measurement. The concern is that even with a short cable length a high inaccuracy could be introduced into the system if the last slave clock is path delay optional.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP. The cable between TS1 and DUT.TS1 can be any length, but its propagation delay must be known.

Test Procedure:

Part A: One-Step DUT MeanPathDelay

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s or for a Pdelay_Req message to be received from the DUT.
A:3. Respond to a Pdelay_Req message with a Pdelay_Resp message.
A:4. Observe the DUT's meanPathDelay,
  a. by requesting ieeeC37238portDS.MPathDly, if SNMP is supported, or
  b. by means provided, if observable.
A:5. Repeat steps A:2-A:4 60 times.
A:6. Calculate the mean and variance of the observed meanPathDelay values.
A:7. Compare this observed meanPathDelay with the known mean path delay derived from the cable.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Pdelay_Req message is received.
A:4	N/A	meanPathDelay is not observable.
A:6	INFO	The variance of the meanPathDelay value is reported.
A:7	WARN	The average observed meanPathDelay is not within 100 ns of the actual mean path delay.
A:7	PASS	The average observed meanPathDelay is within 100 ns of the actual mean path delay.

Part B: Two-Step DUT MeanPathDelay

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Wait up to 10 s or for a Pdelay_Req message to be received from the DUT.
B:3. Respond to a Pdelay_Req message with a Pdelay_Resp and a Pdelay_Resp_Follow_Up message.
B:4. Observe the DUT's meanPathDelay,
  a. by requesting ieeeC37238portDS.MPathDly, if SNMP is supported, or
  b. by means provided, if observable.
B:5. Repeat steps B:2-B:4 60 times.
B:6. Calculate the mean and variance of the observed meanPathDelay values.
B:7. Compare this observed meanPathDelay with the known mean path delay derived from the cable.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req message is received.
B:4	N/A	meanPathDelay is not observable.
B:6	INFO	The variance of the meanPathDelay value is reported.
B:7	WARN	The average observed meanPathDelay is not within 100 ns of the actual mean path delay.
B:7	PASS	The average observed meanPathDelay is within 100 ns of the actual mean path delay.

Part C: One-Step Transparent Clock DUT MeanPathDelay

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. For the duration of this test, generate and send Sync messages from TS1 to the DUT with the correctionField empty.
C:3. Wait up to 10 s or for a Pdelay_Req message to be received by TS1 from the DUT.
C:4. Respond to a Pdelay_Req message with a Pdelay_Resp message.
C:5. Wait up to 10 s or for a Sync message to be received by TS2 from the DUT.
C:6. Observe the correctionField of the Sync message.
C:7. Observe the DUT's meanPathDelay, by requesting ieeeC37238TCportDS.MPathDly, if SNMP is supported.
C:8. Repeat steps C:2-C:7 60 times.
C:9. Calculate the mean and variance of the observed correctionField values.
C:10. Compare this observed correctionField with the known mean path delay derived from the cable.

Observable Results:

Part:Step	Status	Description
C:3	FAIL	No Pdelay_Req message is received by TS1.
C:5	FAIL	No Sync message is received by TS2.
C:7	N/A	correctionField is not observable.
C:9	INFO	The variance of the meanPathDelay (contents of the correctionField) value is reported.
C:10	WARN	The average observed and requested meanPathDelay is not within 100 ns of the actual mean path delay.
C:10	PASS	The average observed and requested meanPathDelay is within 100 ns of the actual mean path delay.

Part D: Two-Step Transparent Clock DUT MeanPathDelay

D:1. Capture traffic received by TS1 and TS2 for the duration of this test.
D:2. For the duration of this test generate and send Sync and Follow_Up messages from TS1 to the DUT with the correctionField empty.
D:3. Wait up to 10 s or for a Pdelay_Req message to be received by TS1 from the DUT.
D:4. Respond to a Pdelay_Req message with a Pdelay_Resp and a Pdelay_Resp_Follow_Up message.
D:5. Wait up to 10 s or for a Sync and Follow_Up messages to be received by TS2 from the DUT.
D:6. Observe the correctionField of the Follow_Up message.
D:7. Observe the DUT's meanPathDelay, by requesting ieeeC37238TCportDS.MPathDly, if SNMP is supported.
D:8. Repeat steps D:2-D:7 60 times.
D:9. Calculate the mean and variance of the observed correctionField values.
D:10. Compare this observed correctionField with the known mean path delay derived from the cable.

Observable Results:

Part:Step	Status	Description
D:3	FAIL	No Pdelay_Req message is received by TS1.
D:5	FAIL	Neither a Sync nor Follow_Up message is received by TS2.
D:6	N/A	correctionField is not observable.
D:9	INFO	The variance of the meanPathDelay (contents of the correctionField) value is reported.
D:10	WARN	The average observed and requested meanPathDelay is not within 100 ns of the actual mean path delay.
D:10	PASS	The average observed and requested meanPathDelay is within 100 ns of the actual mean path delay.

Possible Problems: Means of observing the meanPathDelay for ordinary and boundary clocks may not be available.

Test PWR.c.2.9 – Independent Ports for Boundary Clocks

Purpose: To verify that link delay measurement is made independently by each port on a boundary clock implementing the peer delay mechanism.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC	One-step Clock
B	BC	Two-step Clock

References: [1] IEEE Std 1588-2008: sub-clause 11.4.1
[2] IEEE Std 1588-2008: sub-clause 8.2.1.2.3

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames.

Modification History:	2013-04-15	Preview release

Discussion: This test will validate the DUT's portDS.MeanPathDelay value on two ports. This test will require two cables of different length with known delay. The ports will be connected to cables that differ in length and therefore meanPathDelay. If the link delay measurement is made independently by each port then the meanPathDelay should differ between ports [1].

The speed of light in Ethernet cables is approximately 2/3 *c*, or 5.0 ns/m, so every 10 m of cable length difference contributes a difference in meanPathDelay of approximately 50 ns. This test uses a difference of roughly 100 m, for a meanPathDelay difference of roughly 500 ns.

This test does not apply to ordinary clocks since they have only one port [2].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP. Use a <1 m cable between TS1 and DUT.TS1. Use a 100 m cable between TS2 and DUT.TS2.

Note that another way to achieve the same effect is to use equal-length cables (of any length) and simulate a 100-meter-longer cable between TS2 and DUT.TS2. To do this, for each Pdelay_Resp_Follow_Up message transmitted from TS2 have TS2 subtract 1000 ns (i.e., one round-trip link delay) from the responseOriginTimestamp before transmitting the message. This makes the turnaround time appear 1000 ns shorter than it is which makes the round-trip propagation delay appear 1000 ns longer than it is. In this case also subtract 500 ns from the preciseOrigin-Timestamp field of any Follow_Up messages transmitted from TS2. This makes the one-way propagation delay of Sync messages appear 500 ns longer than it is, matching the illusion created by the Pdelay_Resp_Follow_Up messages.

Test Procedure:

Part A: One-Step DUT Port peerMeanPathDelay

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.
A:2. For each of the two test stations, wait up to 10 s for a Pdelay_Req message to be received from the DUT.
A:3. On both links, respond to each Pdelay_Req message with a Pdelay_Resp message.
A:4. After 5 s, observe the DUT's meanPathDelay of DUT.TS1 and DUT.TS2, by requesting ieeeC37238portDS.MPathDly.
Make both observations within five seconds of each other.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Pdelay_Req message is received.
A:4	FAIL	meanPathDelay is not observable.
A:4	FAIL	The observed meanPathDelays are not at least 400 ns apart.
A:4	PASS	The two meanPathDelays observed are at least 400 ns apart.

Part B: Two-Step DUT Port peerMeanPathDelay

B:1. Capture traffic received by TS1 and TS2 for the duration of this test.
B:2. For each of the two test stations, wait up to 10 s for a Pdelay_Req message to be received from the DUT.
B:3. On both links, respond to each Pdelay_Req message with a Pdelay_Resp and a Pdelay_Resp_Follow_Up message.
B:4. After 5 s, observe the DUT's meanPathDelay of DUT.TS1 and DUT.TS2, by requesting ieeeC37238portDS.MPathDly.
Make both observations within five seconds of each other.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req message is received.
B:4	FAIL	meanPathDelay is not observable.
B:4	FAIL	The observed meanPathDelays are not at least 400 ns apart.
B:4	PASS	The two meanPathDelays observed are at least 400 ns apart.

Possible Problems: None

Test PWR.c.2.10 – Independent Ports for Transparent Clocks

Purpose: To verify that link delay measurement is made independently by each port of a transparent clock implementing the peer delay mechanism.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	TC	SNMP, One-step Clock
B	TC	SNMP, Two-step Clock
C	TC	One-step Clock
D	TC	Two-step Clock

References: [1] IEEE Std 1588-2008: sub-clause 11.4.1

Resource Requirements: Three test stations capable of transmitting and receiving arbitrary MAC frames. Two cables of different length with known delay.

Modification History:	2013-04-09	Preview release

Discussion: This test will validate the DUT's portDS.peerMeanPathDelay value on two ports. This test will require two cables of different length with known delay. The ports will be connected to cables that differ in length and therefore meanPathDelay. If the link delay measurement is made independently by each port then the meanPathDelays should differ between ports [1].

The speed of light in Ethernet cables is approximately 2/3 *c*, or 5.0 ns/m, so every 10 m of cable length difference contributes a difference in meanPathDelay of approximately 50 ns. This test uses a difference of roughly 100 m, for a meanPathDelay difference of roughly 500 ns.

A transparent clock adds the value of the incoming meanPathDelay to the correctionField of the outgoing Sync message if it is a one-step clock, or the correctionField of the outgoing Follow_Up message if it is a two-step clock.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP. Use a <1 m cable between TS1 and DUT.TS1. Use a 100 m cable between TS2 and DUT.TS2.

Note that another way to achieve the same effect is to use equal-length cables (of any length) and simulate a 100-meter-longer cable between TS2 and DUT.TS2. To do this, for each Pdelay_Resp_Follow_Up message transmitted from TS2 have TS2 subtract 1000 ns (i.e., one round-trip link delay) from the responseOriginTimestamp before transmitting the message. This makes the turnaround time appear 1000 ns shorter than it is, which makes the round-trip propagation delay appear 1000 ns longer than it is. In this case also subtract 500 ns from the preciseOrigin-Timestamp field of any Follow_Up messages transmitted from TS2. This makes the one-way propagation delay of Sync messages appear 500 ns longer than it is, matching the illusion created by the Pdelay_Resp_Follow_Up messages.

Test Procedure:

Part A: One-Step DUT Port peerMeanPathDelay

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.
A:2. For each of the two test stations, wait up to 10 s for a Pdelay_Req message to be received from the DUT.
A:3. On both links, respond to each Pdelay_Req message with a Pdelay_Resp message.
A:4. After 5 s, observe the peerMeanPathDelay of DUT.TS1 and DUT.TS2,
   a. by means provided, if observable, or
   b. by requesting ieeeC37238TCportDS.MPathDly, if SNMP is supported.
   Make both observations within five seconds of each other.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Pdelay_Req message is received.
A:4	FAIL	meanPathDelay is not observable.
A:4	FAIL	The observed meanPathDelays are not at least 400 ns apart.
A:4	PASS	The two meanPathDelays observed are at least 400 ns apart.

Part B: Two-Step DUT Port peerMeanPathDelay

B:1. Capture traffic received by TS1 and TS2 for the duration of this test.
B:2. For each of the two test stations, wait up to 10 s for a Pdelay_Req message to be received from the DUT.
B:3. On both links, respond to each Pdelay_Req message with a Pdelay_Resp and a Pdelay_Resp_Follow_Up message.
B:4. After 5 s, observe the peerMeanPathDelay of DUT.TS1 and DUT.TS2,
   a. by means provided, if observable, or
   b. by requesting ieeeC37238TCportDS.MPathDly, if SNMP is supported.
   Make both observations within five seconds of each other.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req message is received.
B:4	FAIL	meanPathDelay is not observable.
B:4	FAIL	The observed meanPathDelays are not at least 400 ns apart.
B:4	PASS	The two meanPathDelays observed are at least 400 ns apart.

Part C: One-Step correctionField of Sync Message

C:1. Capture traffic received by TS1, TS2 and TS3 for the duration of this test.
C:2. For TS1, wait up to 10 s or for a Pdelay_Req message to be received from the DUT.
C:3. Respond to each Pdelay_Req message with a Pdelay_Resp message along with a Sync message.
C:4. For TS3, wait up to 10 s or for a Sync message to be received from the DUT.
C:5. Observe the correctionField of the Sync message.
C:6. For TS2, wait up to 10 s or for a Pdelay_Req message to be received from the DUT.
C:7. Respond to each Pdelay_Req message with a Pdelay_Resp message along with a Sync message.
C:8. For TS3, wait up to 10 s or for a Sync message to be received from the DUT.
C:9. Observe the correctionField of the Sync message.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	No Pdelay_Req message is received.
C:4	FAIL	No Sync message is received.
C:9	FAIL	The observed correctionFields are not at least 400 ns apart from the correctionField observed in step C:5.
C:9	PASS	The observed correctionFields are at least 400 ns apart from the correctionField observed in step C:5.

Part D: Two-Step correctionField of Follow Up Message

D:1. Capture traffic received by TS1, TS2 and TS3 for the duration of this test.
D:2. For TS1, wait up to 10 s or for a Pdelay_Req message to be received from the DUT.
D:3. Respond to each Pdelay_Req message with a Pdelay_Resp message along with Sync and Follow Up messages.
D:4. For TS3, wait up to 10 s or for a Follow Up message to be received from the DUT.
D:5. Observe the correctionField of the Follow Up message.
D:6. For TS2, wait up to 10 s or for a Pdelay_Req message to be received from the DUT.
D:7. Respond to each Pdelay_Req message with a Pdelay_Resp message along with Sync and Follow Up messages.
D:8. For TS3, wait up to 10 s or for a Follow Up message to be received from the DUT.
D:9. Observe the correctionField of the Follow Up message.

Observable Results:

Part:Step	Status	Description
D:2	FAIL	No Pdelay_Req message is received.
D:4	FAIL	No Sync and Follow Up message is received.
D:9	FAIL	The observed correctionFields are not at least 400 ns apart from the correctionField observed in step D:5.
D:9	PASS	The observed correctionFields are at least 400 ns apart from the correctionField observed in step D:5.

Possible Problems: None

GROUP 3: Best Master Clock Algorithm

Overview:

This group covers requirements defined in IEEE 1588-2008 sub-clause 9.3 "Best master clock algorithm", especially sections 9.3.2 (BMCA), 9.3.3 (State decision algorithm) and 9.3.4 (Data set comparison algorithm). The best master clock algorithm comprises two parts: a data set comparison algorithm followed by a state decision algorithm.

The tests defined in this group validate the data set comparison algorithm, defined in sub-clause 9.3.4. These tests change various fields in the Announce messages originating from the test station (TS) to verify that the device under test (DUT) selects the proper grandmaster clock. Verification of best master clock selection is determined through observation of the DUT's Announce message behavior and, if accessible, the DUT's parentDS data set.

Notes:

Test PWR.c.3.1 – Disqualified Announce Messages, by clockIdentity

Purpose: To verify that the DUT selects the correct grandmaster clock based on the validity of the clockIdentity field in incoming Announce messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	Means of observing the DUT's grandmaster

References: [1] IEEE Std 1588-2008: sub-clause 9.3.2.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-02-26	Preview release

Discussion: The test will verify that the DUT disqualifies Announce messages sent and received by the same port [1]. To observe whether the DUT has disqualified an incoming Announce message we establish a connection, and then we send it both valid and invalid Announce messages. The valid Announce messages will use sourcePortIdentity.clockIdentity values that differ from the DUT's sourcePortIdentity while the unqualified Announce messages will use sourcePortIdentity.clockIdentity values that are the same as the DUT's sourcePortIdentity.clockIdentity values. If the DUT selects to the grandmaster that sent the invalid Announce messages then it did not disqualify the appropriate messages.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Disqualified Announce messages, by clockIdentity

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. For the duration of this test, have TS1 send Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity, grandmasterPriority1 and sourcePortIdentity.clockIdentity fields, identical to the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**6**. For the grandmasterPriority1 field use a value one less than the DUT's grandmasterPriority1. For the sourcePortIdentity.clockIdentity field, use a value differing from the DUT's clockIdentity.

A:3. For the duration of this test, have TS1 also send, intermingled with the above Announce messages, a distinct stream of different Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity, grandmasterPriority1 and sourcePortIdentity.clockIdentity fields, identical to the corresponding fields in DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**7**. For the grandmasterPriority1 field use a value two less than the DUT's grandmasterPriority1. For the sourcePortIdentity.clockIdentity field, use the DUT's clockIdentity.

A:4. After 5 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided.

A:5. If the device has more than one port, repeat steps A:1-4 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
A:4	PASS	The DUT disqualified (and ignored) Announce messages that used its clockIdentity, and the DUT qualified (and reacted to) Announce messages that did not use its clockIdentity.

Possible Problems: None

Test PWR.c.3.2 – Disqualified Announce Messages, by Most Recent

Purpose: To verify that the DUT disqualifies Announce messages that are not the most recently received from a given clock.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	SNMP or means of observing the DUT's grandmaster

References: [1] IEEE Std 1588-2008: sub-clause 9.3.2.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-02-26 Preview release

Discussion: The test will verify that the DUT disqualifies Announce messages that are not the most recently received from a given clock [1]. The DUT will receive a stream of Announce messages with higher grandmasterPriority1 values than that of the DUT's, indicating that TS1 should not be grandmaster. However, the last Announce message in the stream will have a grandmasterPriority1 lower than the DUT's, indicating that TS1 should be made grandmaster. If the DUT only considers the most recent Announce message received it will make TS1 its grandmaster.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Disqualified Announce messages, by most recent

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. For the duration of this test, after each message is sent observe the DUT's grandmaster,

  a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
  b. by other means, if provided

A:3. Have TS1 send a stream of 10 Announce messages, one each second for 10 s, with all fields that influence the BMCA, other than the grandmasterIdentity and grandmasterPriority1 fields, identical to the corresponding fields in the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe445566.

A:4. For the grandmasterPriority1 field in the first 9 messages use a value one greater than the DUT's grandmasterPriority1. Observe the DUT's grandmasterIdentity between one and two seconds after each Announce message is received.

A:5. For the grandmasterPriority1 field in the last Announce message use a value one less than the DUT's grandmasterPriority1. Observe the DUT's grandmasterIdentity between one and two seconds after the Announce message is received.

A:6. If the device has more than one port, repeat steps A:1-5 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	The DUT's grandmasterIdentity is ever 0x102233fffe445566.
A:5	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe445566.
A:5	PASS	The DUT disqualified Announce messages that were not the most recently received, and the DUT qualified the Announce message that was the most recently received.

Possible Problems: None

Test PWR.c.3.3 – Disqualified Announce Messages, by Foreign Master Window

Purpose: To verify that the DUT disqualifies Announce messages that were not preceded by at least one recent Announce message from the same clock.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	SNMP or means of observing the DUT's grandmaster

References: [1] IEEE Std 1588-2008: sub-clause 9.3.2.5
[2] IEEE Std 1588-2008: sub-clause 9.3.2.4.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-02-27 Preview release

Discussion: The test will verify that the DUT disqualifies an Announce message that is the only non-identical Announce message received in a single foreign master time window [1].

The central requirement is that if at least FOREIGN_MASTER_THRESHOLD (2) Announce messages have not been received within FOREIGN_MASTER_TIME_WINDOW (4 announceIntervals) then the Announce message is disqualified. Once qualification occurs the clock shall be considered in the BMCA.

Reference [2] states that the size of foreignMasterDS shall be at least five records. If five new clocks arrive on the network and all begin sending Announce messages sufficiently frequently then the Announce messages from all five clocks must be considered in the BMCA.

To observe whether the DUT has disqualified an incoming Announce message we vary how often Announce messages are emitted from TS1. This test ensures that when Announce messages arrive more frequently than every four seconds the DUT qualifies them and that when Announce messages arrive less frequently than every four seconds the DUT disqualifies them.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

This publication is available free of charge from http://dx.doi.org/10.6028/NIST.IR.8002

Test Procedure:

Part A: Disqualified Announce messages, by Foreign Master Window

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. For the duration of this test have TS1 send Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and grandmasterPriority1 fields, identical to the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe445566. For the grandmasterPriority1 field use a value one less than the DUT's grandmasterPriority1.

A:3. Send the Announce messages once every four announceIntervals.

A:4. After 10 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided.

A:5. Increase the rate at which the Announce messages are sent to once every two announceIntervals.

A:6. After 10 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided.

A:7. Increase the rate at which the Announce messages are sent to once each announceInterval.

A:8. After 10 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided.

A:9. If the device has more than one port, repeat steps A:1-8 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	The DUT's grandmasterIdentity is 0x102233fffe445566.
A:6	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe445566.
A:8	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe445566.
A:8	PASS	The DUT disqualified Announce messages that are the only non-identical Announce messages received in a single foreign master time window, and the DUT qualified Announce messages that are not the only non-identical Announce messages received in a single foreign master time window.

Part B: Five Foreign Masters

B:1. For the duration of this test have TS1 send five simultaneous yet distinct streams of Announce messages with all fields that influence the BMCA other than the grandmasterIdentity and the grandmasterPriority1 fields, identical to the DUT's data sets.

B:2. For the first stream, send Announce messages **five** times every four announce intervals (i.e., once every 800 ms) with the following values:
   a. grandmasterIdentity = 0x102233fffe44556**5**
   b. grandmasterPriority1 = **one** less that the DUT's grandmasterPriority1 field

B:3. For the second stream, send Announce messages **four** times every four announce intervals (i.e., once every 1000 ms) with the following values:
   a. grandmasterIdentity = 0x102233fffe44556**6**
   b. grandmasterPriority1 = **two** less that the DUT's grandmasterPriority1 field

B:4. For the third stream, send Announce messages **three** times every four announce intervals (i.e., once every 1300 ms) with the following values:
   a. grandmasterIdentity = 0x102233fffe44556**7**
   b. grandmasterPriority1 = **three** less that the DUT's grandmasterPriority1 field

B:5. For the fourth stream, send Announce messages **twice** every four announce intervals (i.e., once every 2000 ms) with the following values:
   a. grandmasterIdentity = 0x102233fffe44556**8**
   b. grandmasterPriority1 = **four** less that the DUT's grandmasterPriority1 field

B:6. For the fifth stream, send Announce messages **once** every four announce intervals (i.e., once every 4000 ms) with the following values:
   a. grandmasterIdentity = 0x102233fffe44556**9**
   b. grandmasterPriority1 = **five** less that the DUT's grandmasterPriority1 field

B:7. After 10 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided

B:8. If the device has more than one port, repeat steps B:1-7 for one other port on the device.

Observable Results:

Part:Step	Status	Description
B:7	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe445568.
B:7	PASS	The DUT has a foreignMasterDS data set with a minimum capacity of five foreign master records.

Possible Problems: None

Test PWR.c.3.4 – Disqualified Announce Messages, by stepsRemoved

Purpose: To verify that the DUT selects the correct grandmaster clock based on the validity of the stepsRemoved in incoming Announce messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	No SNMP, Means of observing the DUT's grandmaster
C, D	OC, BC	SNMP

References: [1] IEEE Std 1588-2008: sub-clause 9.3.2.5

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-02-26	Preview release

Discussion: The test will verify the DUT disqualifies Announce messages with the value 255 or greater in the stepsRemoved field for the BMCA [1]. The stepsRemoved field is a 16-bit field, therefore the largest possible value is 65535 ($2^{16} - 1$). To observe whether the DUT has disqualified an incoming Announce message we establish a connection in which the DUT is sending Announce messages, and then we send it Announce messages that could change its state and cause it to stop sending Announce messages.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Disqualification by stepsRemoved equal to 255

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Send to the DUT Announce messages that use a stepsRemoved value of **255** (0x00FF), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.

A:3. After 5 s, observe the DUT's grandmaster,
  a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
  b. by other means, if provided

A:4. Send to the DUT Announce messages that use a stepsRemoved value of **254** (0x00FE), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.

A:5. After 5 s, observe the DUT's grandmaster,
  a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
  b. by other means, if provided

A:6. If the device has more than one port, repeat steps A:1-6 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	The DUT's grandmasterIdentity is 0x102233fffe44556**6**.
A:5	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
A:5	PASS	The DUT disqualified (and ignored) Announce messages whose stepsRemoved value was too high, and the DUT qualified (and reacted to) Announce messages whose stepsRemoved value was valid.

Part B: Disqualification by stepsRemoved greater than 255

B:1. Capture traffic received by TS1 for the duration of this test.

B:2. Send to the DUT Announce messages that use a stepsRemoved value of **65535** (0xFFFF), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.

B:3. After 5 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided

B:4. Send to the DUT Announce messages that use a stepsRemoved value of **10** (0x000A), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.

B:5. After 5 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided

B:6. If the device has more than one port, repeat steps B:1-6 for one other port on the device.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	The DUT's grandmasterIdentity is 0x102233fffe44556**6**.
B:5	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
B:5	PASS	The DUT disqualified (and ignored) Announce messages whose stepsRemoved value was too high, and the DUT qualified (and reacted to) Announce messages whose stepsRemoved value was valid.

Part C: Disqualification by stepsRemoved equal to 255, SNMP

C:1. Capture traffic received by TS1 for the duration of this test.

C:2. Send to the DUT Announce messages that use a stepsRemoved value of **255** (0x00FF), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.

C:3. After 5 s, observe the DUT's stepsRemoved by requesting ieeeC37238currentDS.StepsRemoved.

C:4. Observe the DUT's grandmaster by requesting ieeeC37238parentDS.GMClkIdentity.

C:5. Send to the DUT Announce messages that use a stepsRemoved value of **254** (0x00FE), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.

C:6. After 5 s, observe the DUT's stepsRemoved by requesting ieeeC37238currentDS.StepsRemoved.

C:7. Observe the DUT's grandmaster by requesting ieeeC37238parentDS.GMClkIdentity.

C:8. If the device has more than one port, repeat steps C:1-6 for one other port on the device.

Observable Results:

Part:Step	Status	Description
C:3	FAIL	The DUT's stepsRemoved is not 0x00FF.
C:4	FAIL	The DUT's grandmasterIdentity is 0x102233fffe44556**6**.
C:6	FAIL	The DUT's stepsRemoved is not 0x00FE.
C:7	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
C:7	PASS	The DUT disqualified (and ignored) Announce messages whose stepsRemoved value was too high, and the DUT qualified (and reacted to) Announce messages whose stepsRemoved value was valid.

Part D: Disqualification by stepsRemoved greater than 255, SNMP

D:1. Capture traffic received by TS1 for the duration of this test.
D:2. Send to the DUT Announce messages that use a stepsRemoved value of **65535** (0xFFFF), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.
D:3. After 5 s, observe the DUT's stepsRemoved by requesting ieeeC37238currentDS.StepsRemoved.
D:4. Observe the DUT's grandmaster by requesting ieeeC37238parentDS.GMClkIdentity.
D:5. Send to the DUT Announce messages that use a stepsRemoved value of **10** (0x000A), grandmasterIdentity value of 0x102233fffe44556**6** and that use a lower (better) priority1 value than the corresponding value of the DUT's data set.
D:6. After 5 s, observe the DUT's stepsRemoved by requesting ieeeC37238currentDS.StepsRemoved.
D:7. Observe the DUT's grandmaster by requesting ieeeC37238parentDS.GMClkIdentity.
D:8. If the device has more than one port, repeat steps D:1-6 for one other port on the device.

Observable Results:

Part:Step	Status	Description
D:3	FAIL	The DUT's stepsRemoved is not 0xFFFF.
D:4	FAIL	The DUT's grandmasterIdentity is 0x102233fffe44556**6**.
D:6	FAIL	The DUT's stepsRemoved is not 0x000A.
D:7	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
D:7	PASS	The DUT disqualified (and ignored) Announce messages whose stepsRemoved value was too high, and the DUT qualified (and reacted to) Announce messages whose stepsRemoved value was valid.

Possible Problems: None

Test PWR.c.3.5 – Disqualified Announce Messages, by alternateMasterFlag

Purpose: To verify that the DUT selects the correct grandmaster clock based on the alternateMasterFlag of incoming Announce messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	SNMP or means of observing the DUT's grandmaster

References: [1] IEEE Std 1588-2008: sub-clause 9.3.2.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-02-26	Preview release

Discussion: The test will verify the DUT discards Announce messages with alternateMasterFlag TRUE except for the provisions of the master cluster option for the BMCA [1]. To observe whether the DUT has discarded an incoming Announce message we vary the alternateMasterFlag field and observe the DUT's grandmaster.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Discarded Announce messages, by alternateMasterFlag

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. For the duration of this test, have TS1 send Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and grandmasterPriority1 fields, identical to the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**6**. For the grandmasterPriority1 field use a value one less than the DUT's grandmasterPriority1. Use FALSE for the alternateMasterFlag.

A:3. For the duration of this test, have TS1 also send, intermingled with the above Announce messages, a distinct stream of different Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and grandmasterPriority1 fields, identical to the corresponding fields in the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**7**. For the grandmasterPriority1 field use a value two less than the DUT's grandmasterPriority1. Use TRUE for the alternateMasterFlag..

A:4. After 5 s, observe the DUT's grandmaster,
  a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
  b. by other means, if provided

A:5. If the device has more than one port, repeat steps A:1-4 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
A:4	PASS	The DUT discarded Announce messages whose alternateMasterFlag was TRUE, and the DUT accepted Announce messages whose alternateMasterFlag was FALSE.

Possible Problems: None

Test PWR.c.3.6 – Data Set Comparison on a Single Port

Purpose: To verify that the DUT selects the correct grandmaster clock based on the value of the grandmasterPriority1, grandmasterClockQuality, grandmasterPriority2, and grandmasterIdentity fields in Announce messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	Means of observing the DUT's grandmaster
C	OC, BC	SNMP

References:
[1] IEEE Std 1588-2008: sub-clause 9.3.4
[2] IEEE Std 1588-2008: sub-clause 7.6.2
[3] IEEE Std 1588-2008: Figure 27
[4] IEEE Std 1588-2008: Table 12

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-02-20	Preview release

Discussion: This test will verify that the DUT selects the correct grandmaster when receiving Announce messages with various values in the grandmasterPriority1, grandmasterClockQuality, grandmasterPriority2 and grandmasterIdentity fields. Reference [1] describes the data set comparison algorithm that is employed by IEEE 1588 Power Profile devices during selection of a grandmaster clock. When a new or existing clock in a PTP domain transmits an Announce message the device under test must determine whether the announcing clock is better than the current grandmaster, whose information is stored in the DUT's parentDS data set. When two new or existing clocks in a PTP domain transmit Announce messages the device under test must determine which announcing clock is better. After receiving and qualifying an Announce message, the DUT generates a STATE_DECISION_EVENT, which initiates the best master clock algorithm.

The best master clock algorithm compares one clock to another by comparing data sets that represent those clocks. This test ensures that the DUT performs the proper action based on the value of the grandmasterPriority1, clockClass, clockAccuracy, offsetScaledLogVariance, grandmasterPriority2, and grandmasterIdentity fields. The grandmasterClockQuality field is of type ClockQuality. A ClockQuality structure comprises three fields: clockClass, clockAccuracy, and offsetScaledLogVariance. Reference [2] states that lower values take precedence over higher ones.

In reference [3] the first step in comparing data sets describing different grandmasters is to compare the grandmasterPriority1 values. The value of the priority1 field can be anywhere from 0 to 255. If the priority1 values are equal then the second step is to compare the clockClass values. The clockClass field is the first octet of the grandmasterClockQuality field. If the clockClass values are equal then the third step is to compare the clockAccuracy values. The clockAccuracy field is the second octet of the grandmasterClockQuality field in Announce messages. Similarly, the fourth step is to compare the offsetScaledLogVariance values which occupy the third and fourth octets of the grandmasterClockQuality. The fifth step is to compare the grandmasterPriority2 values. The value of the priority2 field can be anywhere from 0 to 255. Finally, if all of the previous values match the sixth and tie-breaking step is to compare the grandmasterIdentity values.

Refer to Appendix C Table 5: Announce Message Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Data Set Comparison between self and one foreign master

A:1. Have TS1 send Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and the **grandmasterPriority1** field, identical to the corresponding fields of the DUT data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**6**. For the **grandmasterPriority1** field use a value *greater than* the DUT's **grandmasterPriority1** value.

A:2. After 5 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided

A:3. Decrease the value of the **grandmasterPriority1** field of the Announce messages emitted to a value *less than* the DUT's **grandmasterPriority1** value.

A:4. After 5 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided

A:5. Repeat steps A:1-4 four times. Each time instead of altering the **grandmasterPriority1** field, alter one of the following fields.
   a. clockClass
   b. clockAccuracy
   c. offsetScaledLogVariance
   d. priority2

A:6. If the device has more than one port, repeat steps A:1-5 for one other port on the device.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	The DUT's grandmasterIdentity is 0x102233fffe44556**6**.
A:4	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
A:5	PASS	The DUT selected the best grandmaster.

Part B: Data Set Comparison between two foreign masters

B:1. Capture traffic received by TS1 for the duration of this test.

B:2. For the duration of this test, have TS1 send Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and **grandmasterPriority1** fields, identical to the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**6**. For the **grandmasterPriority1** field use a value *one less* than the DUT's **grandmasterPriority1**.

B:3. For the duration of this test, have TS1 also send a distinct stream of different Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and **grandmasterPriority1** fields, identical to the corresponding fields in DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**7**. For the **grandmasterPriority1** field use a value *two less* than the DUT's **grandmasterPriority1**.

B:4. After 5 s, observe the DUT's grandmaster,
   a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
   b. by other means, if provided

B:5. Repeat steps B:1-4 four times. Each time instead of altering the **grandmasterPriority1** field, alter one of the following fields.
   a. clockClass
   b. clockAccuracy
   c. offsetScaledLogVariance
   d. priority2

B:6. If the device has more than one port, repeat steps B:1-5 for one other port on the device.

Observable Results:

Part:Step	Status	Description
B:4	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**7**.
B:5	PASS	The DUT selected the best grandmaster.

Part C: Data Set Comparison between self and one foreign master with SNMP

C:1. Have TS1 send Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and the **grandmasterPriority1** field, identical to the corresponding fields of the DUT data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**6**. For the **grandmasterPriority1** field use a value *greater than* the DUT's **grandmasterPriority1** value.

C:2. After 5 s, observe the DUT's parent **grandmasterPriority1** by requesting ieeeC37238parentDS.**GMPriority1**.

C:3. Observe the DUT's grandmaster by requesting ieeeC37238parentDS.GMClkIdentity.

C:4. Decrease the value of the **grandmasterPriority1** field of the Announce messages emitted to a value *less than* the DUT's **grandmasterPriority1** value.

C:5. After 5 s, observe the DUT's parent **grandmasterPriority1** by requesting ieeeC37238parentDS.**GMPriority1**.

C:6. Observe the DUT's grandmaster by requesting ieeeC37238parentDS.GMClkIdentity.

C:7. Repeat steps C:1-6 four times. Each time instead of altering and observing the **grandmasterPriority1**, alter and observe one of the following fields.
   a. clockClass, ieeeC37238parentDS.GMClkClass
   b. clockAccuracy, ieeeC37238parentDS.GMClkAccuracy
   c. offsetScaledLogVariance, ieeeC37238parentDS.GMOfstScdLVar
   d. priority2, ieeeC37238parentDS.GMPriority2

C:8. If the device has more than one port, repeat steps C:1-7 for one other port on the device.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	The DUT's parent **grandmasterPriority1** is *less than* the DUT's **grandmasterPriority1** value.
C:3	FAIL	The DUT's grandmasterIdentity is 0x102233fffe44556**6**.
C:5	FAIL	The DUT's parent **grandmasterPriority1** is *greater than* the DUT's **grandmasterPriority1** value.
C:6	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**6**.
C:7	PASS	The DUT selected the best grandmaster.

Possible Problems: None

Test PWR.c.3.7 – Data Set Comparison on Multiple Ports

Purpose: To verify that the DUT selects the correct grandmaster clock based on the value of the grandmasterPriority1, grandmasterClockQuality, grandmasterPriority2, and grandmasterIdentity fields in Announce messages received on multiple ports.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC	SNMP or means of observing the DUT's grandmaster

References: [1] IEEE Std 1588-2008: sub-clause 9.3.4

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-03-04	Preview release

Discussion: This test will verify that the DUT selects the correct grandmaster when receiving Announce messages on multiple ports with various values in the grandmasterPriority1, grandmasterClockQuality, grandmasterPriority2 and grandmasterIdentity fields. The best of all Announce messages received on multiple ports is determined using the data set comparison algorithm [1].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Data Set Comparison between four foreign masters

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.

A:2. For the duration of this test, have TS1 send to the *first* port on the DUT Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and **grandmasterPriority1** fields, identical to the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**6**. For the **grandmasterPriority1** field use a value *one less* than the DUT's **grandmasterPriority1**.

A:3. For the duration of this test, have TS1 also send to the *first* port on the DUT, a distinct stream of different Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and **grandmasterPriority1** fields, identical to the corresponding fields in the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**7**. For the **grandmasterPriority1** field use a value *two less* than the DUT's **grandmasterPriority1**.

A:4. For the duration of this test, have TS2 send to a *second* port on the DUT Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and **grandmasterPriority1** fields, identical to the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**8**. For the **grandmasterPriority1** field use a value *three less* than the DUT's **grandmasterPriority1**.

A:5. For the duration of this test, have TS2 also send to the *second* port on the DUT, a distinct stream of different Announce messages with all fields that influence the BMCA, other than the grandmasterIdentity and **grandmasterPriority1** fields, identical to the corresponding fields in the DUT's data sets. For the grandmasterIdentity field use the value 0x102233fffe44556**9**. For the **grandmasterPriority1** field use a value *four less* than the DUT's **grandmasterPriority1**.

A:6. After 5 s, observe the DUT's grandmaster,
  a. by requesting ieeeC37238parentDS.GMClkIdentity, if SNMP is supported, or
  b. by other means, if provided

A:7. Repeat steps A:1-6 four times. Each time instead of altering the **grandmasterPriority1** field, alter one of the following fields.
  a. clockClass
  b. clockAccuracy
  c. offsetScaledLogVariance
  d. priority2

A:8. If the device has more than two ports, repeat steps A:1-7 for a different pair of ports on the device.

Observable Results:

Part:Step	Status	Description
A:6	FAIL	The DUT's grandmasterIdentity is 0x102233fffe44556**6**, 0x102233fffe44556**7** or 0x102233fffe44556**8**.
A:6	FAIL	The DUT's grandmasterIdentity is not 0x102233fffe44556**9**.
A:6	PASS	The DUT selected the best grandmaster.

Possible Problems: None

Test PWR.c.3.8 – State Decision Algorithm

Purpose: To verify that the DUT properly uses the state decision algorithm to determine the state of each of its ports.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-C	BC, OC	SNMP

References: [1] IEEE Std 1588-2008: sub-clause 9.3.4
[2] IEEE Std 1588-2008: sub-clause 9.3.2.2
[3] IEEE Std 1588-2008: Figure 26
[4] IEEE Std 1588-2008: sub-clause 9.2.5

Resource Requirements: Four test stations capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-03-05	Preview release

Discussion: This test will verify that the DUT uses the state decision algorithm to determine the BMC event applicable to the state machine of each port. An ordinary or boundary clock uses the data set comparison algorithm [1] to determine the best of all Announce messages received on each of its ports, E_{rbest}. Then it will use the data set comparison algorithm to determine the best of all of those, E_{best}; i.e. the best Announce message received by the clock. Then the clock will use E_{rbest} and E_{best} and its own defaultDS data set, D_0, with the state decision algorithm to determine the BMC event applicable to the state machine [2]. This test will verify that the DUT enters the BMC_PASSIVE and BMC_MASTER states as depicted in the state decision algorithm [3]. To validate the DUT enters these states, the behavior of the device is checked with the behavior of each state specified by [4]. The states M2, M3, and S1 from the state machine are not covered in this test.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: State Decision Algorithm Output P1

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.

A:2. Make the defaultDS.clockQuality.clockClass between 1 through 127,
   a. by means provided, if observable, or
   b. by write, if SNMP is supported

A:3. Have the TS1 and TS2 send Announce messages to the DUT, with all fields that influence the BMCA, other than the sourcePortIdentity field, identical to the DUT message fields. Set the sourcePortIdentity field of the TS1 to a value one less than the DUT's sourcePortIdentity value. Set the sourcePortIdentity field of the TS2 to a value two less than the DUT's sourcePortIdentity value.

A:4. Observe the messages emitted from the DUT.

A:5. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
A:4	FAIL	The DUT sends any messages other than Pdelay_Req, Pdelay_Resp, Pdelay_Resp_Fol-low_Up, or signaling messages, or management messages that are a required response to another management message.
A:4	FAIL	The DUT is not in the BMC_PASSIVE state.
A:5	PASS	The DUT's grandmaster is TS2.

Part B: State Decision Algorithm Output M1

B:1. Capture traffic received by TS1 and TS2 for the duration of this test.

B:2. Make the defaultDS.clockQuality.clockClass between 1 through 127,
   a. by means provided, if observable, or
   b. by write, if SNMP is supported

B:3. Have the TS1 and TS2 send Announce messages to the DUT, with all fields that influence the BMCA, other than the sourcePortIdentity field, identical to the DUT message fields. Set the sourcePortIdentity field of the TS1 to a value one more than the DUT's sourcePortIdentity value. Set the sourcePortIdentity field of the TS2 to a value two more than the DUT's sourcePortIdentity value.

B:4. Observe the messages emitted from the DUT.

B:5. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
B:4	FAIL	The DUT is not behaving as a master port.
B:5	FAIL	The DUT is not grandmaster.
B:5	PASS	The DUT is in the BMC_MASTER state.

Part C: State Decision Algorithm Output P2

C:1. Capture traffic received by TS1, TS2, TS3 and TS4 for the duration of this test.

C:2. Make the defaultDS.clockQuality.clockClass greater than 127,
   a. by means provided, if observable, or
   b. by write, if SNMP is supported

C:3. Have the TS1 and TS2 send Announce messages to one port of the DUT, with all fields that influence the BMCA, other than the sourcePortIdentity field, identical to the DUT message fields. Set the sourcePortIdentity field of the TS1 to a value four less than the DUT's sourcePortIdentity value. Set the sourcePortIdentity field of the TS2 to a value three less than the DUT's sourcePortIdentity value.

C:4. Have the TS3 and TS4 send Announce messages to a different port of the DUT, with all fields that influence the BMCA, other than the sourcePortIdentity field, identical to the DUT message fields. Set the sourcePortIdentity field of the TS3 to a value two less than the DUT's sourcePortIdentity value. Set the sourcePortIdentity field of the TS4 to a value one less than the DUT's sourcePortIdentity value.

C:5. Observe the messages emitted from the DUT.

C:6. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
C:4	FAIL	The DUT sends any messages other than Pdelay_Req, Pdelay_Resp, Pdelay_Resp_Follow_Up, or signaling messages, or management messages that are a required response to another management message.
C:4	FAIL	The DUT is not in the BMC_PASSIVE state.
C:5	PASS	The DUT's grandmaster is TS1.

Possible Problems: None

Test PWR.c.3.9 – Steps Removed

Purpose: To verify that the DUT selects the correct grandmaster clock based on the value of the stepsRemoved field in Announce messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A - D	OC, BC	GMC
E - H	OC	Not GMC, SNMP or means of observing the DUT's grandmaster

References: [1] IEEE Std 1588-2008: sub-clause 9.3.4
[2] IEEE Std 1588-2008: figure 28
[3] IEEE Std 1588-2008: table 12

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2012-12-17	Preview release

Discussion: This test will verify that the DUT selects the correct grandmaster when receiving Announce messages with various values in the stepsRemoved field. Reference [1] describes the data set comparison algorithm that is employed by IEEE 1588 Power Profile devices during selection of a grandmaster clock. Reference [2] describes how to choose between two data sets when their grandmaster fields are all the same, including even the grandmasterIdentity. When the grandmaster fields are all the same the grandmaster selection process is more specific than choosing a grandmaster clock. Rather, in this case the decision regards, first, which port *on a communication path* is on the shortest path to the common grandmaster (i.e., E_{rbest}), and, second, which port *on the DUT* has the shortest path to the common grandmaster (i.e., E_{best}).

This test ensures that the DUT performs the proper action based on the value of the stepsRemoved field. The stepsRemoved field is at offset 61 in Announce messages.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: stepsRemoved, TS < DUT - 1

A:1. Power on the DUT acting as grandmaster.

A:2. Have the TS send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT message fields. Set the stepsRemoved field to a value *two or more less than* the DUT's stepsRemoved value.

A:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	The DUT continues sending Announce messages.
A:3	PASS	The DUT stops sending Announce messages.

Part B: stepsRemoved, TS > DUT + 1

B:1. Power on the DUT acting as grandmaster.

B:2. Have the TS send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT message fields. Set the stepsRemoved field to a value *two or more greater than* the DUT's stepsRemoved value.

B:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	The DUT stops sending Announce messages.
B:3	PASS	The DUT continues sending Announce messages.

Part C: stepsRemoved, TS = DUT - 1

C:1. Power on the DUT acting as grandmaster.

C:2. Have the TS send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT message fields. Set the stepsRemoved field to a value *one less than* the DUT's stepsRemoved value.

C:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
C:3	FAIL	The DUT stops sending Announce messages.
C:3	PASS	The DUT continues sending Announce messages.

Part D: stepsRemoved, TS = DUT + 1

D:1. Power on the DUT acting as grandmaster.

D:2. Have the TS send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT message fields. Set the stepsRemoved field to a value *one greater than* the DUT's stepsRemoved value.

D:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
D:3	FAIL	The DUT continues sending Announce messages.
D:3	PASS	The DUT stops sending Announce messages.

Part E: stepsRemoved, TS1 < TS2 – 1

E:1. Have the TS1 and TS2 send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT's values. Set the stepsRemoved field of TS1 to a value *two or more less than* the TS2's stepsRemoved value.

E:2. Observe the DUT's grandmaster,

a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
E:2	FAIL	The DUT's grandmaster is not TS1.
E:2	PASS	The DUT's grandmaster is TS1.

Part F: stepsRemoved, TS1 > TS2 + 1

F:1. Have the TS1 and TS2 send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT's values. Set the stepsRemoved field of TS1 to a value *two or more greater than* the TS2's stepsRemoved value.

F:2. Observe the DUT's grandmaster,

a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
F:2	FAIL	The DUT's grandmaster is not TS2.
F:2	PASS	The DUT's grandmaster is TS2.

Part G: stepsRemoved, TS1 = TS2 – 1

G:1. Have the TS1 and TS2 send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT's values. Set the stepsRemoved field of TS1 to a value *one less than* the TS2's stepsRemoved value.

G:2. Observe the DUT's grandmaster,

a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
G:2	FAIL	The DUT's grandmaster is not TS1.
G:2	PASS	The DUT's grandmaster is TS1.

Part H: stepsRemoved, TS1 = TS2 + 1

H:1. Have the TS1 and TS2 send Announce messages with all fields, other than the stepsRemoved field, that influence the BMCA identical to the DUT's values. Set the stepsRemoved field of TS1 to a value *one greater than* the TS2's stepsRemoved value.

H:2. Observe the DUT's grandmaster,

a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
H:2	FAIL	The DUT's grandmaster is not TS2.
H:2	PASS	The DUT's grandmaster is TS2.

Possible Problems: Parts C and D may lead to error-1 indicating that one of the messages was transmitted and received on the same port.

Test PWR.c.3.10 – Source Port Identity

Purpose: To verify that the DUT selects the correct grandmaster clock based on the value of the sourcePortIdentity field in Announce messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A - D	OC, BC	GMC
E - H	OC	Not GMC, SNMP or means of observing the DUT's grandmaster

References: [1] IEEE Std 1588-2008: sub-clause 9.3.4
[2] IEEE Std 1588-2008: figure 28
[3] IEEE Std 1588-2008: table 12

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2012-12-17	Preview release

Discussion: This test will verify that the DUT selects the correct grandmaster when receiving Announce messages with various values in the sourcePortIdentity field. Reference [1] describes the data set comparison algorithm that is employed by IEEE 1588 Power Profile devices during selection of a grandmaster clock. Reference [2] describes how to choose between two data sets when their grandmaster fields are all the same, including even the grandmasterIdentity. When the grandmaster fields are all the same the grandmaster selection process is more specific than choosing a grandmaster clock. Rather, in this case the decision regards, first, which port *on a communication path* is on the shortest path to the common grandmaster (i.e., E_{rbest}), and, second, which port *on the DUT* has the shortest path to the common grandmaster (i.e., E_{best}). This test naturally follows the previous test (stepsRemoved). If two incoming Announce messages share the same grandmaster fields and also the same stepsRemoved fields then the final tie-breaking distinction is based on the clockIdentity of the two Announce message sources.

This test ensures that the DUT performs the proper action based on the value of the sourcePortIdentity field. The clockIdentity field is the first eight octets of the sourcePortIdentity at offset 20 in message headers. The portNumber field is the last two octets of the sourcePortIdentity at offset 28 in message headers.

Refer to Appendix C Table 6: IEEE C37.238 Message Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: sourcePortIdentity.clockIdentity, TS < DUT

A:1. Configure the DUT to act as grandmaster.

A:2. Have the TS send Announce messages with all fields that influence the BMCA, other than the clockIdentity field, identical to the DUT message fields. Set the clockIdentity field to a value *less than* the DUT's clock-Identity value.

A:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	The DUT continues sending Announce messages.
A:3	PASS	The DUT stops sending Announce messages.

Part B: sourcePortIdentity.clockIdentity, TS > DUT

B:1. Power on the DUT acting as grandmaster.

B:2. Have the TS send Announce messages with all fields that influence the BMCA, other than the clockIdentity field, identical to the DUT message fields. Set the clockIdentity field to a value *greater than* the DUT's clockIdentity value.

B:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	The DUT stops sending Announce messages.
B:3	PASS	The DUT continues sending Announce messages.

Part C: sourcePortIdentity.portNumber, TS < DUT

C:1. Power on the DUT acting as grandmaster.

C:2. Have the TS send Announce messages with all fields that influence the BMCA, other than the portNumber field, identical to the DUT message fields. Set the portNumber field to a value *less than* the DUT's portNumber value.

C:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
C:3	FAIL	The DUT continues sending Announce messages.
C:3	PASS	The DUT stops sending Announce messages.

Part D: sourcePortIdentity.portNumber, TS > DUT

D:1. Power on the DUT acting as grandmaster.

D:2. Have the TS send Announce messages with all fields that influence the BMCA, other than the portNumber field, identical to the DUT message fields. Set the portNumber field to a value *greater than* the DUT's portNumber value.

D:3. Observe whether the DUT continues sending Announce messages.

Observable Results:

Part:Step	Status	Description
D:3	FAIL	The DUT stops sending Announce messages.
D:3	PASS	The DUT continues sending Announce messages.

Part E: sourcePortIdentity.clockIdentity, TS1 < TS2 < DUT

E:1. Have the TS1 and TS2 send Announce messages with all fields that influence the BMCA, other than the clockIdentity field, identical to the DUT's values. Set the clockIdentity field of TS2 to a value *less than* the DUT's clockIdentity value. Set the clockIdentity field of TS1 to a value *less than* the TS2's clockIdentity value.

E:2. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
E:2	FAIL	The DUT's grandmaster is not TS1.
E:2	PASS	The DUT's grandmaster is TS1.

Part F: sourcePortIdentity.clockIdentity, TS2 < TS1 < DUT

F:1. Have the TS1 and TS2 send Announce messages with all fields that influence the BMCA, other than the clockIdentity field, identical to the DUT's values. Set the clockIdentity field of TS1 to a value *less than* the DUT's clockIdentity value. *Set the* clockIdentity field of TS2 to a value *less than* the TS1's clockIdentity value.

F:2. *Observe* the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
F:2	FAIL	The DUT's grandmaster is not TS2.
F:2	PASS	The DUT's grandmaster is TS2.

Part G: sourcePortIdentity.portNumber, TS1 < TS2 < DUT

G:1. Have the TS1 and TS2 send Announce messages with all fields that influence the BMCA, other than the portNumber field, identical to the DUT's values. Set the portNumber field of TS2 to a value *less than* the DUT's portNumber value. Set the portNumber field of TS1 to a value *less than* the TS2's portNumber value.

G:2. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
G:2	FAIL	The DUT's grandmaster is not TS1.
G:2	PASS	The DUT's grandmaster is TS1.

Part H: sourcePortIdentity.portNumber, TS2 < TS1 < DUT

H:1. Have the TS1 and TS2 send Announce messages with all fields that influence the BMCA, other than the portNumber field, identical to the DUT's values. Set the portNumber field of TS1 to a value *less than* the DUT's portNumber value. Set the portNumber field of TS2 to a value *less than* the TS1's portNumber value.

H:2. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
H:2	FAIL	The DUT's grandmaster is not TS2.
H:2	PASS	The DUT's grandmaster is TS2.

Possible Problems: None

Test PWR.c.3.11 – Default Slave-only

Purpose: To verify that the default configuration is for a slave-only clock if it is not the DUT's primary function to be a grandmaster clock.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC	Slave-Only, SNMP
B	OC	Slave-Only

References: [1] IEEE Std C37.238-2011: sub-clause 5.4.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-07 Preview release

Discussion: This test will verify that the default configuration of the DUT is slave-only by observing all messages emitted from the DUT. Unless a device's primary function is a grandmaster clock, its default configuration shall be slave-only clock. Slave-only clocks shall not transmit Announce messages and shall have priority1 and clockClass values set to 255. [1]

This requirement is already stated in IEEE 1588 Default Profile. This requirement is under review by the Working Group H24/SubC7 and this test may be subject to removal.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: priority1 and clockClass
A:1. Ensure that the DUT is in default setup.
A:2. Observe the DUT's priority1 value.
A:3. Observe the DUT's clockClass value.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	priority1 is not 255.
A:3	FAIL	clockClass is not 255.
A:3	PASS	priority1 and clockClass are 255.

Part B: No Announce Messages
B:1. Capture traffic received by TS1 for the duration of this test part.
B:2. Ensure that the DUT is in default setup.
B:3. Wait one minute or for an Announce message to be received from the DUT.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	An Announce message is received.
B:3	PASS	No Announce message is received.

Possible Problems: None

GROUP 4: Management Mechanism

Overview:

This group covers requirements defined in IEEE Std C37.238-2011 sub-clause 5.5, "Management mechanism". Management messages are used to access specific attributes. The IEEE C37.238 MIB is to be supported by grandmaster clocks and all other devices that choose to support SNMP. When an SNMP MIB is not supported, at least the four values TimeInaccuracy, Traceability to a standard, offset from the grandmaster and whether the offset from grandmaster exceeds a limit must be provided by all devices except transparent clocks.

The Working Group H24/SubC7 is reviewing the need for grandmaster-capable devices to support a SNMP MIB. The concern is that it is unfeasible to equip all master-capable devices with a MIB. Pending a decision from the working group this test may need to be revised or removed.

The Working Group H24/SubC7 is also reviewing the LeapEvLatest, UTCOfstNext and LeapEvExpiry MIB objects. These objects indirectly define a mechanism that is not mentioned in the rest of the C37.238. Tests on these objects are pending more explanation on the three MIB objects.

Notes:

Test PWR.c.4.1 – TimeInaccuracy, Traceability and Offset

Purpose: To verify that the DUT reports TimeInaccuracy, traceability to a recognized standard time source, offset from the grandmaster, and whether that offset is too high.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	GMC
C - F	OC	Slave-Only

References: [1] IEEE Std C37.238-2011: sub-clause 5.5
[2] IEEE Std C37.238-2011: sub-clause D.3.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-08	Preview release

Discussion: This test will verify that the DUT reports TimeInaccuracy, traceability to a recognized standard time source, offset from the grandmaster, and whether the offset exceeds a preconfigured limit. The test accomplishes this in one of two ways. If the DUT supports the IEEE C37.238 MIB, the test requests the DUT's corresponding MIB objects. If the DUT does not support the IEEE C37.238 MIB, the test uses the method specified by the vendor.

All grandmaster-capable devices shall report TimeInaccuracy and traceability to a recognized standard time source [1]. Grandmaster-capable devices are required to support an SNMP MIB, and when any device supports SNMP the IEEE C37.238 MIB must be used. Therefore grandmaster-capable devices must support the C37.238 SNMP MIB. For devices that support the C37.238 MIB this test references the parentDS.GMTimeInacc and timePropDS.TmeTraceable MIB objects. All devices that do not support the SNMP MIB except transparent clocks shall specify whether and how the following information is made available:

- TimeInaccuracy
- Traceability to a standard recognized time source
- Offset from the grandmaster
- If the offset from a grandmaster exceeds a configurable limit

The Working Group H24/SubC7 is reviewing the requirement that all grandmaster-capable devices shall report TimeInaccuracy and traceability to a recognized standard time source. The concern is that these values are already reported in Announce messages and the purpose of the requirement is unclear.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: TimeInaccuracy GMC

A:1. Observe the DUT's grandmaster TimeInaccuracy by requesting ieeeC37238parentDS.GMTimeInacc.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The TimeInaccuracy cannot be observed.
A:1	PASS	The TimeInaccuracy is observable.

Part B: Traceability GMC

B:1. Observe the DUT's time traceable by requesting ieeeC37238timePropDS.TmeTraceable.

Observable Results:

Part:Step	Status	Description
B:1	FAIL	The traceability cannot be observed.
B:1	PASS	The traceability is observable.

Part C: TimeInaccuracy

C:1. Observe the DUT's grandmaster TimeInaccuracy by requesting ieeeC37238parentDS.GMTimeInacc.

Observable Results:

Part:Step	Status	Description
C:1	FAIL	The device does not specify the availability of the TimeInaccuracy.
C:1	PASS	The TimeInaccuracy is observable as specified.

Part D: Traceability

D:1. Query the DUT's TmeTraceable.

Observable Results:

Part:Step	Status	Description
D:1	FAIL	The device does not specify the availability of the traceability.
D:1	PASS	The traceability is observable as specified.

Part E: Offset

E:1. Query the DUT's currentDS.OfstFrMaster.

Observable Results:

Part:Step	Status	Description
E:1	FAIL	The device does not specify the availability of the offset.
E:1	PASS	The offset is observable as specified.

Part F: Offset Limit

F:1. Query the DUT's defaultDS.OfstFrMLimit.

Observable Results:

Part:Step	Status	Description
F:1	FAIL	The device does not specify the availability of whether the offset exceeds the limit.
F:1	PASS	The limit breach of the offset is observable as specified.

Possible Problems:

Test PWR.c.4.2 – SNMP MIB Default Data Set for Ordinary and Boundary Clocks

Purpose: To verify the correct use of an SNMP MIB Default Data Set.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A,B	OC, BC	GMC or SNMP
C	BC	GMC or SNMP
D	OC	GMC or SNMP
E - H	OC, BC	GMC or SNMP
I, J	OC, BC	GMC
K, L	OC, BC	Slave-only, SNMP

References: [1] IEEE Std C37.238-2011: sub-clause 5.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-08 Preview release

Discussion: This test will verify that a grandmaster-capable device supports the default data set of an SNMP MIB by requesting an IEEE C37.238 MIB object from the DUT [1]. By the same process, this test will also validate the use of the default data set of the IEEE C37.238 MIB by other devices that choose to implement SNMP. The table referenced below includes links to tests in this document that cover the SNMP MIB specifications.

Refer to Appendix C Table 15: SNMP MIB Test Coverage

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Default TwoStepFlag for Boundary and Ordinary Clocks

A:1. Observe the DUT's TwoStepFlag by requesting ieeeC37238defaultDS.TwoStepFlag.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The TwoStepFlag object is not observed.
A:1	PASS	The TwoStepFlag is TRUE if the clock is a two-step clock, otherwise the value is FALSE.

Part B: Default ClkIdentity for Boundary and Ordinary Clocks

B:1. Observe the DUT's clock identity by requesting ieeeC37238defaultDS.ClkIdentity.

Observable Results:

Part:Step	Status	Description
B:1	FAIL	The clock identity object is not observed.
B:1	PASS	The clock identity observed is the clock identity of the local clock.

Part C: Default NumberPorts on Boundary Clock

C:1. Observe the DUT's number ports by requesting ieeeC37238defaultDS.NumberPorts.

Observable Results:

Part:Step	Status	Description
C:1	FAIL	The number ports object is not observed.
C:1	PASS	The number ports observed is the number of PTP ports on the DUT.

Part D: Default NumberPorts on Ordinary Clock

D:1. Observe the DUT's number ports by requesting ieeeC37238defaultDS.NumberPorts.

Observable Results:

Part:Step	Status	Description
D:1	FAIL	The number ports object is not observed to be 1.
D:1	PASS	The number ports observed is 1.

Part E: Default OfsScdLogVar on Boundary and Ordinary Clocks

E:1. Observe the DUT's offset scaled log variance by requesting ieeeC37238defaultDS.OfsScdLogVar.

Observable Results:

Part:Step	Status	Description
E:1	FAIL	The offset scaled log variance object is not observed.
E:1	PASS	The value is a scaled, offset representation of an estimate of the PTP variance. The PTP variance characterizes the precision and frequency stability of the grandmaster clock.

Part F: Domain Number in Boundary and Ordinary Clocks

F:1. Ensure that the DUT is in default setup.
F:2. Capture traffic received by TS1 for the duration of this test.
F:3. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
F:4. Observe the DUT's domain number by requesting ieeeC37238defaultDS.DomainNumber.
F:5. Change the DUT's domain number by writing ieeeC37238defaultDS.DomainNumber to be 100.
F:6. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
F:7. Observe the DUT's domain number by requesting ieeeC37238defaultDS.DomainNumber.

Observable Results:

Part:Step	Status	Description
F:3	FAIL	Three Announce messages are not received.
F:3	FAIL	The domainNumber is not 0.
F:4	FAIL	The domainNumber is not 0.
F:6	FAIL	The domainNumber is not 100.
F:7	FAIL	The domainNumber is not 100.
F:4	PASS	The value of the domain number requested and observed in Announce messages changed from 0 to 100.

Part G: Default EngTimeInacc on Boundary and Ordinary Clocks

G:1. Observe the DUT's engineering networkTimeInaccuracy by requesting ieeeC37238defaultDS.EngTimeInacc.

Observable Results:

Part:Step	Status	Description
G:1	FAIL	The engineering networkTimeInaccuracy object is not observed.
G:1	PASS	The value represents the worst networkTimeInaccuracy from this device to all preferred grandmasters.

Part H: Default Offset Limit on Boundary and Ordinary Clocks

H:1. Observe the DUT's offset from master limit by requesting ieeeC37238defaultDS.OfstFrMLimit.

Observable Results:

Part:Step	Status	Description
H:1	FAIL	The device does not specify the availability of whether the offset exceeds the limit.
H:1	PASS	The limit breech of the offset is observable as specified.

Part I: Priority1and Priority2 on GMC Boundary and Ordinary Clocks

I:1. Ensure that the DUT is in default setup.

I:2. Observe the DUT's priority1 and priority2 by requesting ieeeC37238defaultDS.Priority1 and ieeeC37238defaultDS.Priority2.

I:3. Change the DUT's priority1 and priority2 by writing ieeeC37238defaultDS.Priority1 and ieeeC37238defaultDS.Priority2 to be 130.

I:4. Wait 5 s; observe the DUT's priority1 and priority2 by requesting ieeeC37238defaultDS.Priority1 and ieeeC37238defaultDS.Priority2.

Observable Results:

Part:Step	Status	Description
I:2	FAIL	The value of the priority1 field is not 128.
I:2	FAIL	The value of the priority2 field is not 128.
I:4	FAIL	The value of the priority1 field is not 130.
I:4	FAIL	The value of the priority2 field is not 130.
I:4	PASS	The GMC DUT's priority1 and priority2 started as 128 and changed to 130.

Part J: SlaveOnly value on GMC Boundary and Ordinary Clocks

J:1. Ensure that the DUT is in default setup.

J:2. Observe the DUT's slaveOnly by requesting ieeeC37238defaultDS.SlaveOnly.

J:3. Change the DUT's slaveOnly by writing ieeeC37238defaultDS.SlaveOnly to be TRUE.

J:4. Wait 5 s; observe the DUT's slaveOnly by requesting ieeeC37238defaultDS.SlaveOnly.

Observable Results:

Part:Step	Status	Description
J:2	FAIL	The slaveOnly field is TRUE.
J:4	FAIL	The slaveOnly field is FALSE.
J:4	PASS	The slaveOnly field changed from FALSE to TRUE.

Part K: Priority1 and Priority2 on Slave-Only Boundary and Ordinary Clocks

K:1. Ensure that the DUT is in default setup.

K:2. Observe the DUT's priority1 and priority2 by requesting ieeeC37238defaultDS.Priority1 and ieeeC37238defaultDS.Priority2.

K:3. Change the DUT's priority1 and priority2 by writing ieeeC37238defaultDS.Priority1 and ieeeC37238defaultDS.Priority2 to be 250.

K:4. Wait 5 s; observe the DUT's priority1 and priority2 by requesting ieeeC37238defaultDS.Priority1 and ieeeC37238defaultDS.Priority2.

Observable Results:

Part:Step	Status	Description
K:2	FAIL	The grandmasterPriority1 field is not 255.
K:2	FAIL	The grandmasterPriority2 field is not 255.
K:4	FAIL	The grandmasterPriority1 field is not 250.
K:4	FAIL	The grandmasterPriority2 field is not 250.
K:4	PASS	The DUT's priority1 and priority2 started as 255 and changed to 250.

Part L: SlaveOnly value for SlaveOnly Boundary and Ordinary Clocks

L:1. Ensure that the DUT is in default setup.

L:2. Observe the DUT's slaveOnly by requesting ieeeC37238defaultDS.SlaveOnly.

L:3. Change the DUT's slaveOnly by writing ieeeC37238defaultDS.SlaveOnly to be FALSE.

L:4. Wait 5 s; observe the DUT's slaveOnly by requesting ieeeC37238defaultDS.SlaveOnly.

Observable Results:

Part:Step	Status	Description
L:2	FAIL	The slaveOnly field is FALSE.
L:4	FAIL	The slaveOnly field is TRUE.
L:4	PASS	The slaveOnly field changed from TRUE to FALSE.

Possible Problems: The EngTimeInacc value may not be set if the DUT is not the end device.

Test PWR.c.4.3 – SNMP MIB Parent Data Set for Ordinary and Boundary Clocks

Purpose: To verify the correct use of an SNMP MIB Parent Data Set.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A,B	OC, BC	GMC or SNMP

References: [1] IEEE Std C37.238-2011: sub-clause 5.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-08 Preview release

Discussion: This test will verify that a grandmaster-capable device supports the parent data set of an SNMP MIB by requesting an IEEE C37.238 MIB object from the DUT [1]. By the same process, this test will also validate the use of the parent data set of the IEEE C37.238 MIB by other devices that choose to implement SNMP. The table referenced below includes links to tests in this document that cover the SNMP MIB specifications.

Refer to Appendix C Table 15: SNMP MIB Test Coverage

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Parent PortNumber for Ordinary and Boundary Clocks

A:1. Capture traffic received and sent by TS1 for the duration of this test.
A:2. Observe the DUT's port number by requesting ieeeC37238parentDS.PortNumber.
A:3. Send Announce, Sync and Follow_Up messages from TS1 to the DUT.
A:4. Wait up to 10 s for at least one message of each type to be received from the DUT by TS1.
A:5. Observe the DUT's port number by requesting ieeeC37238parentDS.PortNumber.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	The port number object is not observed.
A:2	FAIL	The initialization value of the port number is not 0.
A:5	FAIL	The port number object is not observed.
A:5	PASS	The port number is the portIdentity of the port on the master that issues the Sync messages used in synchronizing the DUT.

Part B: Parent Stats for Ordinary and Boundary Clocks

- B:1. Capture traffic received by TS1 for the duration of this test.
- B:2. Observe the DUT's observedParentOffsetScaledLog Variance by requesting ieeeC37238parentDS.ObsOfstScdLVar.
- B:3. Observe the DUT's observedParentClockPhaseChangeRate members by requesting ieeeC37238parentDS.ObsPhChgRate.
- B:4. Observe the DUT's stats by requesting ieeeC37238parentDS.Stats.
- B:5. Send Announce, Sync and Follow_Up messages from TS1 to the DUT with timestamps that do not vary.
- B:6. Wait up to 10 s for at least one message of each type to be received from the DUT by TS1.
- B:7. Observe the DUT's observedParentOffsetScaledLog Variance by requesting ieeeC37238parentDS.ObsOfstScdLVar.
- B:8. Observe the DUT's observedParentClockPhaseChangeRate members by requesting ieeeC37238parentDS.ObsPhChgRate.
- B:9. Repeat steps B:4 to B:8 with timestamps that vary by ±30 s.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	The observedParentOffsetScaledLog Variance object is not observed.
B:2	FAIL	The initialization value of the observedParentOffsetScaledLog Variance object is not 0xFFFF.
B:3	FAIL	The observedParentClockPhaseChangeRate member's object is not observed.
B:3	FAIL	The initialization value of the observedParentClockPhaseChangeRate member's object is not 0x7FFFF FFFF.
B:4	FAIL	The stats object is not observed.
B:4	FAIL	The initialization value of the stats is not FALSE.
B:7	FAIL	The observedParentOffsetScaledLog Variance object is not observed.
B:7	INFO	Note the value of the observedParentOffsetScaledLog Variance object.
B:8	FAIL	The observedParentClockPhaseChangeRate member's object is not observed.
B:8	INFO	Note the value of the observedParentClockPhaseChangeRate member's object.
B:9	FAIL	The stats object is not observed.
B:9	INFO	The stats object is FALSE.
B:9	FAIL	The observedParentOffsetScaledLog Variance object is not observed.
B:9	FAIL	The stats object is TRUE and the value of the observedParentOffsetScaledLog Variance object (an estimate of the parent clock's PTP variance as observed by the slave clock) is not greater than what was reported in step B:7.
B:9	FAIL	The observedParentClockPhaseChangeRate member's object is not observed.
B:9	FAIL	The stats object is TRUE and the value of the observedParentClockPhaseChangeRate member's object does not appear to be an estimate of the parent clock's phase change rate as observed by the slave clock.
B:9	PASS	The value of the stats object is TRUE if the clock has a port in the SLAVE state and the clock has computed statistically valid estimates of the parentDS.observedParentOffsetScaledLog Variance and the parentDS.observedParentClockPhaseChangeRate members.

Possible Problems: None.

Test PWR.c.4.4 – SNMP MIB Time Properties Data Set for Ordinary and Boundary Clocks

Purpose: To verify the correct use of an SNMP MIB Time Properties Data Set.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A -D	OC, BC	GMC or SNMP

References: [1] IEEE Std C37.238-2011: sub-clause 5.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-08	Preview release

Discussion: This test will verify that a grandmaster-capable device supports the time properties data set of an SNMP MIB by requesting an IEEE C37.238 MIB object from the DUT [1]. By the same process, this test will also validate the use of the time properties data set of the IEEE C37.238 MIB by other devices that choose to implement SNMP. The table referenced below includes links to tests in this document that cover the SNMP MIB specifications.

Refer to Appendix C Table 15: SNMP MIB Test Coverage

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: TimeProp CurUTCOfstVd for Ordinary and Boundary Clocks

A:1. Observe the DUT's CurUTCOfst by requesting ieeeC37238timePropDS.CurUTCOfst.
A:2. Observe the DUT's CurUTCOfstValid by requesting ieeeC37238timePropDS.CurUTCOfstVd.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The currentUtcOffset is not 35+n, where n is the number of leap seconds after July 1, 2012.
A:1	FAIL	The currentUtcOffset is not TAI – UTC.
A:2	PASS	The currentUtcOffsetValid is TRUE if the timePropertiesDS.currentUtcOffset is TAI - UTC.

Part B: TimeProp Leap59 for Ordinary and Boundary Clocks

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Observe the DUT's leap59 by requesting ieeeC37238timePropDS.Leap59.
B:3. Send an announcement that indicates the last minute of the current UTC day contains 59 s.
B:4. Wait up to 10 s for a LeapSecAnnounced notification.
B:5. Observe the DUT's leap59 by requesting ieeeC37238timePropDS.Leap59.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	The leap59 object is not observed.
B:2	FAIL	The leap59 object is TRUE indicating that the last minute of the current UTC day contains 59 s.
B:4	FAIL	No LeapSecAnnounced notification is received indicating that a leap second has been announced.
B:5	FAIL	The leap59 object is not observed.
B:5	PASS	The leap59 object is TRUE indicating that the last minute of the current UTC day contains 59 s.

Part C: TimeProp Leap61 for Ordinary and Boundary Clocks

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. Observe the DUT's leap61 by requesting ieeeC37238timePropDS.Leap61.
C:3. Send an announcement that indicates the last minute of the current UTC day contains 61 s.
C:4. Wait up to 10 s for a LeapSecAnnounced notification.
C:5. Observe the DUT's leap61 by requesting ieeeC37238timePropDS.Leap61.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	The leap61 object is not observed.
C:2	FAIL	The leap61 object is TRUE indicating that the last minute of the current UTC day contains 61 s.
C:4	FAIL	No LeapSecAnnounced notification is received indicating that a leap second has been announced.
C:5	FAIL	The leap61 object is not observed.
C:5	PASS	The leap61 object is TRUE indicating that the last minute of the current UTC day contains 61 s.

Part D: TimeProp Local Time Name on Boundary and Ordinary Clocks

D:1. Observe the DUT's local time name by requesting ieeeC37238timePropDS.LocalTName.

D:2. Change the local time name by writing ieeeC37238timePropDS.LocalTName.

Observable Results:

Part:Step	Status	Description
D:1	FAIL	The local time name object is not observed.
D:1	FAIL	The value of the local time name is not the text name of the alternate timescale (i.e. NTP).
D:2	PASS	The value of the local time name has changed to the new local time name.

Possible Problems: None.

Test PWR.c.4.5 – SNMP MIB Port Data Set for Ordinary and Boundary Clocks

Purpose: To verify the correct use of an SNMP MIB Port Data Set.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-O	OC, BC	GMC or SNMP

References: [1] IEEE Std C37.238-2011: sub-clause 5.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-08	Preview release

Discussion: This test will verify that a grandmaster-capable device supports the port data set of an SNMP MIB by requesting an IEEE C37.238 MIB object from the DUT [1]. By the same process, this test will also validate the use of the port data set of the IEEE C37.238 MIB by other devices that choose to implement SNMP. The table referenced below includes links to tests in this document that cover the SNMP MIB specifications.

Refer to Appendix C Table 15: SNMP MIB Test Coverage

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Port Clock Identity on Boundary and Ordinary Clocks

A:1. Observe the DUT's clock identity by requesting ieeeC37238portDS.ClkIdentity.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The clock identity object is not observed.
A:1	FAIL	The first three octets are not the first half of the local port's MAC address.
A:1	FAIL	The middle two octets are not 0xFFFE.
A:1	FAIL	The last three octets are not the second half of the local port's MAC address.
A:1	PASS	The clockIdentity is constructed with the first three octets are the first half of the local port's MAC address, the middle two octets are 0xFFFE, and the last three octets are the second half of the local port's MAC address.

Part B: Port Number on Boundary and Ordinary Clocks

B:1. Observe the DUT's port number by requesting ieeeC37238portDS.PortNumber.

Observable Results:

Part:Step	Status	Description
B:1	FAIL	The port number object is not observed.
B:1	FAIL	The port number object is not 1 for a port on a PTP node supporting a single PTP port.
B:1	FAIL	The value of the port number for a PTP node supporting N PTP ports is not 1, 2, …N, respectively.
B:1	PASS	The value of the port number is 1 for single port devices or 1, 2, ...N for a device with N ports.

Part C: Port State on Boundary and Ordinary Clocks

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. Observe the DUT's port state by requesting ieeeC37238portDS.PortState.
C:3. Have TS1 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS1 to a value **greater** than the DUT's grandmasterPriority1 value.
C:4. Wait up to 10 s for at least three Announce messages to be received from the DUT by TS1.
C:5. Observe the DUT's port state by requesting ieeeC37238portDS.PortState.
C:6. Repeat steps C:3-5 with the grandmasterPriority1 field of TS1 to a value **less** than that of the DUT's grandmasterPriority1 value.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	The port state object is not observed.
C:2	FAIL	The value of the port state object is not 01 while initializing.
C:4	FAIL	No Announce messages are received by the DUT.
C:5	FAIL	The value of the port state object is not 06 while in the master state.
C:6	FAIL	No Announce messages are received by the DUT.
C:6	FAIL	The value of the port state object is not 09 while in the slave state.
C:6	PASS	The value of the port state is the current state of the protocol engineer associated with this port.

Part D: LogAnnounceInt on Boundary and Ordinary Clocks

D:1. Capture traffic received by TS1 for the duration of this test.
D:2. Observe the DUT's logAnnounceInterval by requesting ieeeC37238portDS.LogAnnounceInt.
D:3. Wait 5 s, observe 60 consecutive Announce intervals.
D:4. Change the DUT's logAnnounceInterval by writing ieeeC37238portDS.LogAnnounceInt to be 1.
D:5. Wait 5 s, observe 60 consecutive Announce intervals.
D:6. Observe the DUT's logAnnounceInterval by requesting ieeeC37238portDS.LogAnnounceInt.

Observable Results:

Part:Step	Status	Description
D:2	FAIL	The value of the logAnnounceInterval requested was not 0.
D:3	FAIL	The average time between Announce messages is not between 0.7 and 1.3 s.
D:5	FAIL	The average time between Announce messages is not between 1.7 and 2.3 s.
D:5	FAIL	The logMessageInterval value (1 octet at offset 33) in any Announce message is anything other than 1.
D:6	PASS	The value of the logAnnounceInterval requested and observed in Announce messages was 1.

Part E: AnnounceReceiptTimeout for Boundary and Ordinary Clocks

E:1. Observe the DUT's AnnounceReceiptTimeout by requesting ieeeC37238portDS.AnnounceRctTout.
E:2. Change the DUT's AnnounceReceiptTimeout by writing ieeeC37238portDS.AnnounceRctTout to be 4.
E:3. Wait 5 s, observe the DUT's AnnounceReceiptTimeout by requesting ieeeC37238portDS.AnnounceRctTout.

Observable Results:

Part:Step	Status	Description
E:1	FAIL	If the DUT is a preferred grandmaster clock, FAIL if the AnnounceReceiptTimeout value is not 2. Otherwise, if the DUT is another grandmaster, FAIL if the AnnounceReceiptTimeout value is not 3.
E:3	PASS	The AnnounceReceiptTimeout value is 4.

Part F: LogSyncInt on Boundary and Ordinary Clocks

F:1. Capture traffic received by TS1 for the duration of this test.
F:2. Observe the DUT's logSyncInterval by requesting ieeeC37238portDS.LogSyncInt.
F:3. Wait 5 s, observe 60 consecutive Sync intervals.
F:4. Change the DUT's logSyncInterval by writing ieeeC37238portDS.LogSyncInt to be 1.
F:5. Wait 5 s, observe 60 consecutive Sync intervals.
F:6. Observe the DUT's logSyncInterval by requesting ieeeC37238portDS.LogSyncInt.

Observable Results:

Part:Step	Status	Description
F:2	FAIL	The value of the logSyncInterval requested and observed in Sync messages was not 0.
F:3	FAIL	The average time between Sync messages is not between 0.7 and 1.3 s.
F:5	FAIL	The average time between Sync messages is not between 1.7 and 2.3 s.
F:5	FAIL	The logMessageInterval value (1 octet at offset 33) in any Sync message is anything other than 1.
F:6	PASS	The value of the logSyncInterval requested and observed in Sync messages was 1.

Part G: Port Delay Mechanism on Boundary and Ordinary Clocks

G:1. Observe the DUT's delay mechanism by requesting ieeeC37238portDS.DelayMech.

Observable Results:

Part:Step	Status	Description
G:1	FAIL	The delay mechanism object is not observed.
G:1	PASS	The value of the delay mechanism is 02 for peer to peer.

Part H: logMinPdelayReqInterval on Boundary and Ordinary Clocks

H:1. Capture traffic received by TS1 for the duration of this test.
H:2. Observe 10 consecutive Pdelay_Req intervals.
H:3. If the device has more than one port, repeat steps B:1-2 for one other port on the device.
H:4. Observe the DUT's logMinPdelayReqInterval by requesting ieeeC37238portDS.logMinPdelayReqInterval.
H:5. Change the DUT's logMinPdelayReqInterval by writing ieeeC37238portDS.logMinPdelayReqInterval to be 1.
H:6. Wait 5 s, observe 10 consecutive Pdelay_Req intervals.
H:7. Observe the DUT's logMinPdelayReqInterval by requesting ieeeC37238portDS.logMinPdelayReqInterval.

Observable Results:

Part:Step	Status	Description
H:2	FAIL	The average time between Pdelay_Req messages is less than 0.7 s.
H:4	FAIL	The value of the logMinPdelayReqInterval requested and observed was not 0.
H:6	FAIL	The average time between Pdelay_Req messages is less than 1.7 s.
H:7	PASS	The value of the logMinPdelayReqInterval requested and observed was not 1.

Part I: Port Version Number on Boundary and Ordinary Clocks

I:1. Observe the DUT's version number by requesting ieeeC37238portDS.VersionNumber.

Observable Results:

Part:Step	Status	Description
I:1	FAIL	The version number object is not observed.
I:1	PASS	The value of the version number is the PTP version in use on the port.

Part J: Port Enabled on Boundary and Ordinary Clocks

J:1. Capture traffic received by TS1 for the duration of this test.
J:2. Send Pdelay_Req messages from TS1.
J:3. Wait up to 10 s for messages to be received from the DUT.
J:4. Observe the DUT's port enabled by requesting ieeeC37238portDS.PortEnabled.
J:5. Disable the port by writing ieeeC37238timePropDS.PortEnabled.
J:6. Send Pdelay_Req messages from TS1.
J:7. Wait up to 10 s for messages to be received from the DUT.
J:8. Observe the DUT's port enabled by requesting ieeeC37238portDS.PortEnabled.

Observable Results:

Part:Step	Status	Description
J:3	FAIL	No Pdelay_Resp messages are received from the DUT.
J:4	FAIL	The port enabled object is not observed.
J:4	FAIL	The value of the port enabled is not TRUE if the port is enabled.
J:7	FAIL	Pdelay_Resp messages are received from the DUT.
J:8	PASS	The value of the port enabled is FALSE meaning the port has been disabled.

Part K: Port Delay Asymmetry on Boundary and Ordinary Clocks

K:1. Capture traffic received by TS1 for the duration of this test.
K:2. Send Pdelay_Req messages from TS1.
K:3. Wait up to 10 s for messages to be received from the DUT.
K:4. Observe the DUT's delay asymmetry by requesting ieeeC37238portDS.DlyAsymmetry.
K:5. Change the delay asymmetry by writing ieeeC37238portDS.DlyAsymmetry.
K:6. Send Pdelay_Req messages from TS1.
K:7. Wait up to 10 s for messages to be received from the DUT.
K:8. Observe the DUT's delay asymmetry by requesting ieeeC37238portDS.DlyAsymmetry.

Observable Results:

Part:Step	Status	Description
K:3	FAIL	No Pdelay_Resp messages are received from the DUT.
K:4	FAIL	The delay asymmetry object is not observed.
K:7	FAIL	No Pdelay_Resp messages are received from the DUT.
K:8	PASS	The observed delay asymmetry has changed since step G:4.

Part L: Port Profile Identification on Boundary and Ordinary Clocks

L:1. Observe the DUT's profile identification by requesting ieeeC37238portDS.ProfileId.
L:2. Change the profile identification to the default profile by writing ieeeC37238portDS.ProfileId to be '0'.
L:3. Wait up to 10 s, and observe the DUT's profile identification by requesting ieeeC37238portDS. ProfileId.

Observable Results:

Part:Step	Status	Description
L:1	FAIL	The profile identification object is not observed.
L:1	FAIL	The profile identification object is not '1' for power.
L:3	FAIL	The profile identification object is not '0' for default.
L:3	PASS	The profile identification indicates the PTP profile in use.

Part M: Port Network Protocol on Boundary and Ordinary Clocks

M:1. Observe the DUT's network protocol by requesting ieeeC37238portDS.NetProtocol.
M:2. Change the network protocol to '2,' indicating the udplpv4, by writing ieeeC37238portDS.NetProtocol.
M:3. Wait up to 10 s, and observe the DUT's network protocol by requesting ieeeC37238portDS.NetProtocol.

Observable Results:

Part:Step	Status	Description
M:1	FAIL	The network protocol object is not observed.
M:1	FAIL	The network protocol object is not '1' for ieee8023.
M:3	FAIL	The network protocol object is not '2' for udplpv4.
M:3	PASS	The network protocol indicates the network protocol in use.

Part N: Port VlanId on Boundary and Ordinary Clocks

N:1. Observe the DUT's VlanId by requesting ieeeC37238portDS.VlanId.
N:2. Change the VlanId to '1' by writing ieeeC37238portDS.VlanId.
N:3. Wait up to 10 s, and observe the DUT's VlanId by requesting ieeeC37238portDS.VlanId.

Observable Results:

Part:Step	Status	Description
N:1	FAIL	The VlanId object is not observed.
N:1	FAIL	The VlanId object is not '0'.
N:3	FAIL	The VlanId object is not '1'.
N:3	PASS	The VlanId indicates the port VLAN ID in use.

Part O: Port Priority on Boundary and Ordinary Clocks

O:1. Observe the DUT's priority by requesting ieeeC37238portDS.Priority.
O:2. Change the priority to '1' by writing ieeeC37238portDS.Priority.
O:3. Wait up to 10 s, and observe the DUT's priority by requesting ieeeC37238portDS.Priority.

Observable Results:

Part:Step	Status	Description
O:1	FAIL	The priority object is not observed.
O:1	FAIL	The priority object is not '4'.
O:3	FAIL	The priority object is not '1'.
O:3	PASS	The priority indicates the port priority in use.

Possible Problems: None.

Test PWR.c.4.6 – SNMP MIB Default Data Set for Transparent Clocks

Purpose: To verify the correct use of an SNMP MIB Default Data Set.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-K	TC	SNMP

References: [1] IEEE Std C37.238-2011: sub-clause 5.5

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-08 Preview release

Discussion: This test will also validate the use of the default data set of the IEEE C37.238 MIB by devices that choose to implement SNMP. The table referenced below includes links to tests in this document that cover the SNMP MIB specifications.

Refer to Appendix C Table 15: SNMP MIB Test Coverage

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Default Clock Identity on Transparent Clocks

A:1. Observe the DUT's clock identity by requesting ieeeC37238TCDefaultDS.ClkIdentity.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The clock identity object is not observed.
A:1	FAIL	The first three octets are not the first half of the local clock's MAC address.
A:1	FAIL	The middle two octets are not 0xFFFE.
A:1	FAIL	The last three octets are not the second half of the local clock's MAC address.
A:1	PASS	The value is the clockIdentity attribute of the local clock.

Part B: Default NumberPorts on Transparent Clocks

B:1. Observe the DUT's number ports by requesting ieeeC37238TCDefaultDS.NumberPorts.

Observable Results:

Part:Step	Sta-tus	Description
B:1	FAIL	The number ports object is not observed.
B:1	PASS	The number ports observed is the number of PTP ports on the DUT.

Part C: Default Delay Mechanism on Transparent Clocks

C:1. Observe the DUT's delay mechanism by requesting ieeeC37238TCDefaultDS.DelayMech.

Observable Results:

Part:Step	Status	Description
C:1	FAIL	The delay mechanism object is not observed.
C:1	PASS	The value of the delay mechanism is 02 for peer to peer.

Part D: Default Primary Domain on Transparent Clocks

D:1. Ensure that the DUT is in default setup.
D:2. Observe the DUT's primary domain number by requesting ieeeC37238TCdefaultDS.PriDomain.
D:3. Change the DUT's domain number by writing ieeeC37238TCdefaultDS.PriDomain to be 100.
D:4. Observe the DUT's domain number by requesting ieeeC37238TCdefaultDS.PriDomain.

Observable Results:

Part:Step	Status	Description
D:2	FAIL	The primary domain number is not 0.
D:4	FAIL	The primary domain number is not 100.
D:4	PASS	The primary domain number did not change from 0 to 100.

Part E: Default Syntonize on Transparent Clocks

E:1. Observe the DUT's syntonized object by requesting ieeeC37238TCDefaultDS.Syntonize.
E:2. Change the DUT's syntonized object by writing ieeeC37238TCDefaultDS.Syntonize.
E:3. Wait up to 10 s, and observe the DUT's syntonized object by requesting ieeeC37238TCDefaultDS.Syntonize.

Observable Results:

Part:Step	Status	Description
E:1	FAIL	The Syntonize object is not observed.
E:1	FAIL	The value of the Syntonize is not TRUE if syntonization is enabled.
E:3	PASS	The value of the Syntonize object changed.

Part F: Default CurGMaster on Transparent Clocks

F:1. Observe the DUT's current grandmaster by requesting ieeeC37238TCDefaultDS.CurGMaster.

Observable Results:

Part:Step	Status	Description
F:1	FAIL	The current grandmaster object is not observed.
F:1	PASS	The value comprises the current grandmaster identity.

Part G: Default TwoStepFlag for Transparent Clocks

G:1. Observe the DUT's TwoStepFlag by requesting ieeeC37238TCDefaultDS.TwoStepFlag.
G:2. Change the DUT's TwoStepFlag by writing ieeeC37238TCDefaultDS.TwoStepFlag.
G:3. Wait up to 10 s, and observe the DUT's TwoStepFlag object by requesting ieeeC37238TCDefaultDS.TwoStepFlag.

Observable Results:

Part:Step	Status	Description
G:1	FAIL	The TwoStepFlag object is not observed.
G:1	FAIL	The TwoStepFlag is not TRUE if the clock is a two-step clock, otherwise the value is FALSE.
G:3	PASS	The TwoStepFlag changed.

Part H: Default GMIdentity on Transparent Clocks

H:1. Capture traffic received by TS1 and TS2 for the duration of this test.

H:2. Have TS1 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS1 to a value **less** than the DUT's grandmasterPriority1 value. Set the grandmasterIdentity value to 0x102233fffe445567.

H:3. Wait up to 10 s for at least three Announce messages to be forwarded from the DUT to TS2.

H:4. Observe the DUT's GMIdentity by requesting ieeeC37238TCDefaultDS.GMIdentity.

Observable Results:

Part:Step	Status	Description
H:3	FAIL	No Announce messages are received by TS2.
H:4	FAIL	The GMIdentity object is not observed.
H:4	PASS	The value of the GMIdentity is 0x102233fffe445567.

Part I: Default Network Protocol on Transparent Clocks

I:1. Observe the DUT's network protocol by requesting ieeeC37238TCDefaultDS.NetProtocol.

I:2. Change the network protocol to '2,' indicating the udplpv4, by writing ieeeC37238TCDefaultDS.NetProtocol.

I:3. Wait up to 10 s, and observe the DUT's network protocol by requesting ieeeC37238TCDefaultDS.NetProtocol.

Observable Results:

Part:Step	Status	Description
I:1	FAIL	The network protocol object is not observed.
I:1	FAIL	The network protocol object is not '1' for ieee8023.
I:3	FAIL	The network protocol object is not '2' for udplpv4.
I:3	PASS	The network protocol indicates the network protocol in use.

Part J: Default VlanId on Transparent Clocks

J:1. Observe the DUT's VlanId by requesting ieeeC37238TCDefaultDS.VlanId.

J:2. Change the VlanId to '1' by writing ieeeC37238TCDefaultDS.VlanId.

J:3. Wait up to 10 s, and observe the DUT's VlanId by requesting ieeeC37238TCDefaultDS.VlanId.

Observable Results:

Part:Step	Status	Description
J:1	FAIL	The VlanId object is not observed.
J:1	FAIL	The VlanId object is not '0'.
J:3	FAIL	The VlanId object is not '1'.
J:3	PASS	The VlanId indicates the port VLAN ID in use.

Part K: Default Priority on Transparent Clocks

K:1. Observe the DUT's priority by requesting ieeeC37238TCDefaultDS.Priority.

K:2. Change the priority to '1' by writing ieeeC37238TCDefaultDS.Priority.

K:3. Wait up to 10 s, and observe the DUT's priority by requesting ieeeC37238TCDefaultDS.Priority.

Observable Results:

Part:Step	Status	Description
K:1	FAIL	The priority object is not observed.
K:1	FAIL	The priority object is not '4'.
K:3	FAIL	The priority object is not '1'.
K:3	PASS	The priority indicates the port priority in use.

Possible Problems: None.

Test PWR.c.4.7 – SNMP MIB Port Data Set for Transparent Clocks

Purpose: To verify the correct use of an SNMP MIB Port Data Set.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-D	TC	SNMP

References: [1] IEEE Std C37.238-2011: sub-clause 5.5

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-08 Preview release

Discussion: This test will also validate the use of the port data set of the IEEE C37.238 MIB by devices that choose to implement SNMP. The table referenced below includes links to tests in this document that cover the SNMP MIB specifications.

Refer to Appendix C Table 15: SNMP MIB Test Coverage

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Port Number on Transparent Clocks

A:1. Observe the DUT's port number by requesting ieeeC37238TCPortDS.PortNumber.

A:2. Repeat step A:1 for each port on the DUT.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The port number object is not observed.
A:2	FAIL	The value of the port number is all-zeros or all-ones indicating "all-ports" in Management messages and in Signaling messages or a NULL portNumber value.
A:2	PASS	The values of the port numbers for the N ports on the DUT is 1, 2, ...N, respectively.

Part B: logMinPdelayReqInterval for Transparent Clocks

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Observe 10 consecutive Pdelay_Req intervals.
B:3. If the device has more than one port, repeat steps B:1-2 for one other port on the device.
B:4. Observe the DUT's logMinPdelayReqInterval by requesting ieeeC37238portTCDS.logMinPdelayReqInterval.
B:5. Change the DUT's logMinPdelayReqInterval by writing ieeeC37238portTCDS.logMinPdelayReqInterval to be 1.
B:6. Wait 5 s, observe 10 consecutive Pdelay_Req intervals.
B:7. Observe the DUT's logMinPdelayReqInterval by requesting ieeeC37238portTCDS.logMinPdelayReqInterval.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	The average time between Pdelay_Req messages is less than 1 s.
B:4	FAIL	The value of the logMinPdelayReqInterval requested and observed was not 0.
B:6	FAIL	The average time between Pdelay_Req messages is less than 2 s.
B:7	PASS	The value of the logMinPdelayReqInterval requested and observed was not 1.

Part C: Port Faulty on Transparent Clocks

C:1. Observe the DUT's faulty object by requesting ieeeC37238TCPortDS.Faulty.

Observable Results:

Part:Step	Status	Description
C:1	FAIL	The faulty object is FALSE when initializing.
C:1	PASS	The value of the faulty object is TRUE if the port is faulty and FALSE if the port is operating normally.

Part D: Port Delay Asymmetry on Transparent Clocks

D:1. Capture traffic received by TS1 and TS2 for the duration of this test.
D:2. Send Pdelay_Req messages from TS1 to the DUT.
D:3. Wait up to 10 s for messages to be forwarded from the DUT to TS2.
D:4. Observe the DUT's delay asymmetry by requesting ieeeC37238TCPortDS.DlyAsymm.
D:5. Change the delay asymmetry by writing ieeeC37238TCPortDS.DlyAsymm.
D:6. Send Pdelay_Req messages from TS1 to the DUT.
D:7. Wait up to 10 s for messages to be forwarded from the DUT to TS2.
D:8. Observe the DUT's delay asymmetry by requesting ieeeC37238TCPortDS.DlyAsymm.

Observable Results:

Part:Step	Status	Description
D:3	FAIL	No Pdelay_Resp messages are received from the DUT.
D:4	FAIL	The delay asymmetry object is not observed.
D:7	FAIL	No Pdelay_Resp messages are received from the DUT.
D:8	PASS	The observed delay asymmetry has changed since step C:4.

Possible Problems: None.

Test PWR.c.4.8 – SNMP MIB Notifications

Purpose: To verify the correct use of an SNMP MIB Time Properties Data Set.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-D	OC, BC, TC	GMC or SNMP

References: [1] IEEE Std C37.238-2011: sub-clause 5.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-08 Preview release

Discussion: This test will verify that a grandmaster-capable device supports notifications of an SNMP MIB by making events happen and observing the DUT's response [1]. By the same process, this test will also validate the use of notifications of the IEEE C37.238 MIB by other devices that choose to implement SNMP. The table referenced below includes links to tests in this document that cover the SNMP MIB specifications.

Refer to Appendix C Table 15: SNMP MIB Test Coverage

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Events Change of Master

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.
A:2. Look for a ChangeOfMaster notification for the duration of this test.
A:3. Have TS1 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS1 to a value **less** than the DUT's grandmasterPriority1 value. Set the grandmasterIdentity value to 0x102233fffe44556**6**.
A:4. Wait up to 10 s for at least three Announce messages to be received by the DUT.
A:5. Observe the DUT's grandmaster identity by requesting ieeeC37238defaultDS.GMIdentity.
A:6. Have TS2 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS2 to a value **less** than theTS1's grandmasterPriority1 value. Set the grandmasterIdentity value to 0x102233fffe44556**7**.
A:7. Wait up to 10 s for at least three Announce messages to be received by the DUT.
A:8. Observe the DUT's grandmaster identity by requesting ieeeC37238defaultDS.GMIdentity.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	No Announce messages are received by the DUT.
A:5	FAIL	The value of the GMIdentity is not 0x102233fffe44556**6**.
A:8	FAIL	The value of the GMIdentity is not 0x102233fffe44556**7**.
A:8	PASS	A ChangeOfMaster notification received from the DUT indicates that a new grandmaster has been selected.

Part B: Event Port State Change

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Look for a PortStateChange notification for the duration of this test.
B:3. Observe the DUT's port state by requesting ieeeC37238portDS.PortState.
B:4. Have TS1 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS1 to a value **greater** than the DUT's grandmasterPriority1 value.
B:5. Wait up to 10 s for at least three Announce messages to be received from the DUT by TS1.
B:6. Observe the DUT's port state by requesting ieeeC37238portDS.PortState.
B:7. Repeat steps B:3-5 with the grandmasterPriority1 field of TS1 to a value **less** than that of the DUT's grandmasterPriority1 value.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	The port state object is not observed.
B:3	FAIL	The value of the port state object is not 01 while initializing.
B:5	FAIL	No Announce messages are received by the DUT.
B:6	FAIL	The value of the port state object is not 06 while in the master state.
B:6	FAIL	No PortStateChange notification is received to indicate that port state has changed.
B:7	FAIL	No Announce messages are received by the DUT.
B:7	FAIL	The value of the port state object is not 09 while in the slave state.
B:7	PASS	PortStateChange notifications are received from the DUT indicating when the port state has changed.

Part C: Event Other Profile Detect

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. Look for an OtherProfileDetect notification for the duration of this test.
C:3. Ensure TS1 is using the default profile.
C:4. Have TS1 send Announce messages to the DUT. Set the grandmasterPriority1 field of TS1 to a value **greater** than the DUT's grandmasterPriority1 value.
C:5. Wait up to 10 s for at least three Announce messages to be received from the DUT by TS1.
C:6. Repeat steps C:3-4 with the grandmasterPriority1 field of TS1 to a value **less** than that of the DUT's grandmasterPriority1 value.

Observable Results:

Part:Step	Status	Description
C:5	FAIL	No Announce messages are received by the DUT.
C:6	FAIL	No Announce messages are received by the DUT.
C:6	PASS	OtherProfileDetect notifications are received indicating another PTP profile has been detected.

Part D: Event PTP Service Stopped and Started

D:1. Capture traffic received by TS1 for the duration of this test.
D:2. Look for a PTPService Started notification for the duration of this test.
D:3. Wait up to 10 s for at least one Pdelay_Req message to be received from the DUT.
D:4. Disable the DUT's PTP service.
D:5. Wait up to 10 s for a PTPServiceStopped notification.
D:6. Enable the DUT's PTP service.
D:7. Wait up to 10 s for a PTPServiceStarted notification.

Observable Results:

Part:Step	Status	Description
D:3	FAIL	No Pdelay_Req messages are received from the DUT.
D:5	FAIL	No PTPService Stopped notification is received indicating the PTP service has stopped.
D:7	PASS	A PTPService Started notification is received indicating the PTP service has started.

Possible Problems: None.

GROUP 5: Transport Mechanism

Overview:

This group covers requirements defined in Annex F of the IEEE Std 1588-2008, identifying the PTP over IEEE Std 802.3 transport mapping. In the IEEE Std C37.238-2011, this is the only transport mechanism used in a substation system.

Notes:

Test PWR.c.5.1 – IEEE 802.3 Transport Mapping for Announce, Sync and Follow_Up Messages

Purpose: To verify all devices generate and send Announce, Sync and Follow_Up messages over the IEEE 802.3 transport mapping.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.8
[2] IEEE Std 1588-2008: sub-clause 7.3.8.1

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-15	Preview release

Discussion: This test will validate the only transport mechanism in use by the DUT is PTP over IEEE 802.3 transport mapping by observing six fields in Announce, Sync and Follow_Up messages emitted from the DUT.

This test will validate that the multicast communication model is in use by observing that the **unicastFlag** bit is FALSE in messages emitted from the DUT [1]. A FALSE value for the unicastFlag indicates that the message was transmitted as a multicast message.

This test will validate that the PTP message starts at the first octet of the payload by observing the **transportSpecific** field of messages emitted from the DUT [1]. The transportSpecific field should be at offset 0 for each type of message.

This test will validate the values of the t**ransportSpecific** field by observing messages emitted from the DUT. The transportSpecific field should be of value '0' for all PTP layer 2 Ethernet transmissions that are not covered by any other enumeration values [1].

This test will validate the value of the **EtherType** field by observing messages emitted from the DUT. The EtherType field is located in the Ethernet specific layer at offset -2. The value is specified to be $88F7_{16}$ for the EtherType communication service [1].

This test will validate the MAC addresses by observing the **destination field** of messages emitted from the DUT. By default the destination address for peer delay mechanism messages shall be 01-80-C2-00-00-0E. For all other IEEE C37.238 messages the default destination address shall be 01-1B-19-00-00-00 [1], [2].

This test will validate the values of the **PCP** and **VID** fields by observing messages emitted from the DUT. [1] adds a 32-bit field between MAC addresses and the EtherType field. The first 16 bits are the Tag Protocol Identifier (TPID) set to a value of 0x8100. The next 16 bits are for the Tag Control Information (TCI) which includes the 3-bit Priority Code Point (PCP) field, the 1-bit Drop Eligible Indicator (DEI) field, and the 12-bit VLAN Identifier (VID) field. The PCP field should have a default value of 4 and the VID field should have a default value of 0 [2].

Refer to Appendix C Table 6: IEEE C37.238 Message Fields
Refer to Appendix C Table 7: Multicast MAC Addresses
Refer to Appendix C Table 8: Ethernet Transport Specific Field
Refer to Appendix C Table 9: IEEE 802.1 Q Header Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Announce, Sync, Follow_Up

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s for at least one message of each type to be received from the DUT.
A:3. Observe the value of the unicastFlag field in each type of message.
A:4. Observe the location of the transportSpecific field in each type of message.
A:5. Observe the value of the transportSpecific field in each type of message.
A:6. Observe the EtherType field in each type of message.
A:7. Observe the destination field in each type of message.
A:8. Observe the PCP field in each type of message.
A:9. Observe the VID field in each type of message.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	In any Announce messages, the unicastFlag field **is TRUE**.
A:3	FAIL	In any Sync messages, the unicastFlag field **is TRUE**.
A:3	FAIL	In any Follow_Up messages, the unicastFlag field **is TRUE**.
A:4	FAIL	In any Sync messages, the transportSpecific field **is not** at offset zero.
A:5	FAIL	In any Announce messages, the transportSpecific field **is not** ‘0’.
A:5	FAIL	In any Sync messages, the transportSpecific field **is not** ‘0’.
A:5	FAIL	In any Follow_Up messages, the transportSpecific field **is not** ‘0’.
A:6	FAIL	In any Announce messages, the EtherType field **is not** 0x88F7.
A:6	FAIL	In any Sync messages, the EtherType field **is not** 0x88F7.
A:6	FAIL	In any Follow_Up messages, the EtherType field **is not** 0x88F7.
A:7	FAIL	In any Announce messages, the destination field **is not** 01-1B-19-00-00-00.
A:7	FAIL	In any Sync messages, the destination field **is not** 01-1B-19-00-00-00.
A:7	FAIL	In any Follow_Up messages, the destination field **is not** 01-1B-19-00-00-00.
A:8	FAIL	In any Announce messages, the PCP field **is not** 4.
A:8	FAIL	In any Sync messages, the PCP field **is not** 4.
A:8	FAIL	In any Follow_Up messages, the PCP field **is not** 4.
A:9	FAIL	In any Announce messages, the VID field **is not** 0.
A:9	FAIL	In any Sync messages, the VID field **is not** 0.
A:9	FAIL	In any Follow_Up messages, the VID field **is not** 0.
A:9	PASS	In all messages, the unicastFlag field is FALSE, the transportSpecific field is '0' at offset zero, the EtherType field is 0x88F7, the destination field is 01-1B-19-00-00-00, the PCP field is 4 and the VID field is 0.

Possible Problems: None

Test PWR.c.5.2 – IEEE 802.3 Transport Mapping for Forwarded Announce, Sync and Follow_Up Messages

Purpose: To verify all transparent clocks forward Announce, Sync and Follow_Up messages over the IEEE 802.3 transport mapping.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	TC	None

References: [1] IEEE Std C37.238-2011: sub-clause 5.8
[2] IEEE Std 1588-2008: sub-clause 7.3.8.1

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-15 Preview release

Discussion: This test will validate the only transport mechanism in use by the DUT is PTP over IEEE 802.3 transport mapping by observing six fields in Announce, Sync and Follow_Up messages emitted from the DUT.

This test will validate that the multicast communication model is in use by observing that the **unicastFlag** bit is FALSE in messages emitted from the DUT [1]. A FALSE value for the unicastFlag indicates that the message was transmitted as a multicast message.

This test will validate that the PTP message starts at the first octet of the payload by observing the transportSpecific field of messages emitted from the DUT [1]. The **transportSpecific** field should be at offset 0 for each type of message.

This test will validate the values of the t**ransportSpecific** field by observing messages emitted from the DUT. The transportSpecific field should be of value ‘0’ for all PTP layer 2 Ethernet transmissions that are not covered by any other enumeration values [1].

This test will validate the value of the **EtherType** field by observing messages emitted from the DUT. The EtherType field is located in the Ethernet specific layer at offset -2. The value is specified to be $88F7_{16}$ for the EtherType communication service [1].

This test will validate the MAC addresses by observing the destination field of messages emitted from the DUT. By default the **destination address** for peer delay mechanism messages shall be 01-80-C2-00-00-0E. For all other IEEE C37.238 messages the default destination address shall be 01-1B-19-00-00-00 [1], [2].

This test will validate the values of the **PCP** and **VID** fields by observing messages emitted from the DUT. [1] adds a 32-bit field between MAC addresses and the EtherType field. The first 16 bits are the Tag Protocol Identifier (TPID) set to a value of 0x8100. The next 16 bits are for the Tag Control Information (TCI) which includes the 3-bit Priority Code Point (PCP) field, the 1-bit Drop Eligible Indicator (DEI) field, and the 12-bit VLAN Identifier (VID) field. The PCP field should have a default value of 4 and the VID field should have a default value of 0 [2].

Refer to Appendix C Table 6: IEEE C37.238 Message Fields
Refer to Appendix C Table 7: Multicast MAC Addresses
Refer to Appendix C Table 8: Ethernet Transport Specific Field
Refer to Appendix C Table 9: IEEE 802.1 Q Header Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Forwarded Announce, Sync, Follow_Up

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.
A:2. Send Announce, Sync and Follow_Up messages from TS1 to the DUT.
A:3. Wait up to 10 s for at least one message of each type to be received from the DUT by TS2.
A:4. Observe the value of the unicastFlag field in each type of message.
A:5. Observe the location of the transportSpecific field in each type of message.
A:6. Observe the value of the transportSpecific field in each type of message.
A:7. Observe the EtherType field in each type of message.
A:8. Observe the destination field in each type of message.
A:9. Observe the PCP field in each type of message.
A:10. Observe the VID field in each type of message.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	In any Announce messages, the unicastFlag field **is TRUE**.
A:4	FAIL	In any Sync messages, the unicastFlag field **is TRUE**.
A:4	FAIL	In any Follow_Up messages, the unicastFlag field **is TRUE**.
A:5	FAIL	In any Sync messages, the transportSpecific field **is not** at offset zero.
A:6	FAIL	In any Announce messages, the transportSpecific field **is not** ‘0’.
A:6	FAIL	In any Sync messages, the transportSpecific field **is not** ‘0’.
A:6	FAIL	In any Follow_Up messages, the transportSpecific field **is not** ‘0’.
A:7	FAIL	In any Announce messages, the EtherType field **is not** 0x88F7.
A:7	FAIL	In any Sync messages, the EtherType field **is not** 0x88F7.
A:7	FAIL	In any Follow_Up messages, the EtherType field **is not** 0x88F7.
A:8	FAIL	In any Announce messages, the destination field **is not** 01-1B-19-00-00-00.
A:8	FAIL	In any Sync messages, the destination field **is not** 01-1B-19-00-00-00.
A:8	FAIL	In any Follow_Up messages, the destination field **is not** 01-1B-19-00-00-00.
A:9	FAIL	In any Announce messages, the PCP field **is not** 4.
A:9	FAIL	In any Sync messages, the PCP field **is not** 4.
A:9	FAIL	In any Follow_Up messages, the PCP field **is not** 4.
A:10	FAIL	In any Announce messages, the VID field **is not** 0.
A:10	FAIL	In any Sync messages, the VID field **is not** 0.
A:10	FAIL	In any Follow_Up messages, the VID field **is not** 0.
A:10	PASS	In all messages, the unicastFlag field is FALSE, the transportSpecific field is '0' at offset zero, the EtherType field is 0x88F7, the destination field is 01-1B-19-00-00-00, the PCP field is 4 and the VID field is 0.

Possible Problems: None

Test PWR.c.5.3 – IEEE 802.3 Transport Mapping for Peer Delay Messages

Purpose: To verify all devices generate and send Pdelay_Req, Pdelay_Resp and Pdelay_Resp_Follow_Up messages over the IEEE 802.3 transport mapping.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	None

References: [1] IEEE Std C37.238-2011: sub-clause 5.8
[2] IEEE Std 1588-2008: sub-clause 7.3.8.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-15 Preview release

Discussion: This test will validate the only transport mechanism in use by the DUT is PTP over IEEE 802.3 transport mapping by observing six fields in Pdelay_Req, Pdelay_Resp and Pdelay_Resp_Follow_Up messages emitted from the DUT.

This test will validate that the multicast communication model is in use by observing that the **unicastFlag** bit is FALSE in messages emitted from the DUT [1]. A FALSE value for the unicastFlag indicates that the message was transmitted as a multicast message.

This test will validate that the PTP message starts at the first octet of the payload by observing the transportSpecific field of messages emitted from the DUT [1]. The **transportSpecific** field should be at offset 0 for each type of message.

This test will validate the values of the t**ransportSpecific** field by observing messages emitted from the DUT. The transportSpecific field should be of value '0' for all PTP layer 2 Ethernet transmissions that are not covered by any other enumeration values [1].

This test will validate the value of the **EtherType** field by observing messages emitted from the DUT. The EtherType field is located in the Ethernet specific layer at offset -2. The value is specified to be $88F7_{16}$ for the EtherType communication service [1].

This test will validate the MAC addresses by observing the destination field of messages emitted from the DUT. By default the **destination address** for peer delay mechanism messages shall be 01-80-C2-00-00-0E. For all other IEEE C37.238 messages the default destination address shall be 01-1B-19-00-00-00 [1], [2].

This test will validate the values of the **PCP** and **VID** fields by observing messages emitted from the DUT. [1] adds a 32-bit field between MAC addresses and the EtherType field. The first 16 bits are the Tag Protocol Identifier (TPID) set to a value of 0x8100. The next 16 bits are for the Tag Control Information (TCI) which includes the 3-bit Priority Code Point (PCP) field, the 1-bit Drop Eligible Indicator (DEI) field, and the 12-bit VLAN Identifier (VID) field. The PCP field should have a default value of 4 and the VID field should have a default value of 0 [2].

Refer to Appendix C Table 6: IEEE C37.238 Message Fields
Refer to Appendix C Table 7: Multicast MAC Addresses
Refer to Appendix C Table 8: Ethernet Transport Specific Field
Refer to Appendix C Table 9: IEEE 802.1 Q Header Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Pdelay_Resp & Pdelay_Resp_Follow_Up Messages

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s for at least one Pdelay_Req message to be received from the DUT.
A:3. Send Pdelay_Req messages from TS1.
A:4. Wait up to 10 s for Pdelay_Resp and Pdelay_Resp_Follow_Up messages to be received from the DUT.
A:5. Observe the value of the unicastFlag field.
A:6. Observe the value of the transportSpecific field.
A:7. Observe the EtherType field.
A:8. Observe the destination field.
A:9. Observe the PCP field.
A:10. Observe the VID field.

Observable Results:

Part:Step	Status	Description
A:5	FAIL	In any Pdelay_Req messages, the unicastFlag field **is TRUE**.
A:5	FAIL	In any Pdelay_Resp messages, the unicastFlag field **is TRUE**.
A:5	FAIL	In any Pdelay_Resp_Follow_Up messages, the unicastFlag field **is TRUE**.
A:7	FAIL	In any Pdelay_Req messages, the transportSpecific field **is not** ‘0’.
A:7	FAIL	In any Pdelay_Resp messages, the transportSpecific field **is not** ‘0’.
A:7	FAIL	In any Pdelay_Resp_Follow_Up messages, the transportSpecific field **is not** ‘0’.
A:8	FAIL	In any Pdelay_Req messages, the EtherType field **is not** 0x88F7.
A:8	FAIL	In any Pdelay_Resp messages, the EtherType field **is not** 0x88F7.
A:8	FAIL	In any Pdelay_Resp_Follow_Up messages, the EtherType field **is not** 0x88F7.
A:9	FAIL	In any Pdelay_Req messages, the destination field **is not** 01-80-C2-00-00-0E.
A:9	FAIL	In any Pdelay_Resp messages, the destination field **is not** 01-80-C2-00-00-0E.
A:9	FAIL	In any Pdelay_Resp_Follow_Up messages, the destination field **is not** 01-80-C2-00-00-0E.
A:10	FAIL	In any Pdelay_Req messages, the PCP field **is not** 4.
A:10	FAIL	In any Pdelay_Resp messages, the PCP field **is not** 4.
A:10	FAIL	In any Pdelay_Resp_Follow_Up messages, the PCP field **is not** 4.
A:11	FAIL	In any Pdelay_Req messages, the VID field **is not** 0.
A:11	FAIL	In any Pdelay_Resp messages, the VID field **is not** 0.
A:11	FAIL	In any Pdelay_Resp_Follow_Up messages, the VID field **is not** 0.
A:11	PASS	In all messages, the unicastFlag field is FALSE, the transportSpecific field is '0' at offset zero, the EtherType field is 0x88F7, the destination field is 01-80-C2-00-00-0E, the PCP field is 4 and the VID field is 0.

Possible Problems: None

Test PWR.c.5.4 – Multiple Priorities

Purpose: To verify the highest priority is used for event messages when the transport mechanism allows for multiple priorities.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC, Multiple Priorities Allowed
B	All	Multiple Priorities Allowed

References: [1] IEEE Std 1588-2008: Annex F.4
[2] IEEE Std 802.1Q-2011

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2012-01-17	Preview release

Discussion: This test will validate the highest priority is used for event messages when the transport mechanism allows for multiple priorities by observing the value of the priority field in multiple messages emitted from the DUT [1]. The 3 bit Priority Code Point (PCP) field is at offset 16 in the 802.1Q header [2]. Of the messages observed, the event messages, Sync, Delay_Req, Pdelay_Req and Pdelay_Resp, should have the highest priority.

Refer to Appendix C Table 6: IEEE C37.238 Message Fields
and Table 9: IEEE 802.1 Q Header Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
*Part A: Announce, **Sync**, Follow_Up & **Pdelay_Req** Messages*
A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s for at least one message of each type to be received from the DUT.
A:3. Observe the PCP field in all messages.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No messages received from the DUT.
A:3	FAIL	The priority of any Sync and Pdelay_Req messages received is less than that of any Announce, and Follow_Up messages received.
A:3	PASS	The priority of all event messages is highest.

*Part B: **Pdelay_Req, Delay_Req, Pdelay_Resp** & Pdelay_Resp_Follow_Up Messages*

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Wait up to 10 s for at least one Pdelay_Req message to be received from the DUT.
B:3. Send Announce, Sync and Pdelay_Req messages from TS1.
B:4. Wait up to 10 s for Delay_Req, Pdelay_Resp and Pdelay_Resp_Follow_Up messages to be received from the DUT.
B:5. Observe the PCP field in all messages.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req messages received from the DUT.
B:4	FAIL	No messages received from the DUT.
B:4	FAIL	The priorities of any Pdelay_Req, Delay_Req and Pdelay_Resp messages received are less than that of any Pdelay_Resp_Follow_Up messages received.
B:5	PASS	The priority of all event messages is highest.

Possible Problems: None

Test PWR.c.5.5 – IEEE Std 802.1Q Tags

Purpose: To verify only messages with IEEE Std 802.1Q tags are accepted.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	None

References: [1] IEEE Std C37.238-2011: sub-clause 5.6
[2] IEEE Std 1588-2008: Annex F.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2012-11-20	Preview release

Discussion: This test will validate that the DUT only accepts IEEE C37.238 messages with IEEE 802.1Q tags by observing whether the DUT responds to Pdelay_Req messages [1]. IEEE C37.238 messages with IEEE 802.1 Q tags will have a 32-bit field between MAC addresses and the EtherType field. The first 16 bits are the Tag Protocol Identifier (TPID) set to a value of 0x8100 [2].

The August 11, 2013 revision of the IEEE Std. C37.238 removes the requirement for all devices to accept IEEE C37.238 messages that have had their IEEE 802.1Q tags removed. The concern is that the requirement currently implies that untagged C37.238 messages may circulate and that it is unclear if the device has to forward these messages with tags or without.

Refer to Appendix C Table 6: IEEE C37.238 Message Fields
and Table 9: IEEE 802.1 Q Header Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: With IEEE 802.1Q tags
A:1. Capture traffic received and sent by TS1 for the duration of this test.
A:2. Have TS1 send Pdelay_Req messages with the *IEEE 802.1Q tag*. The tag should have values listed in the table above (8100 4 0 0).
A:3. Wait up to 10 s or for 3 Pdelay_Resp Messages to be received from the DUT.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	Pdelay_Resp messages are not received from the DUT.
A:3	PASS	Pdelay_Resp messages are received from the DUT.

Part B: Without IEEE 802.1Q tags

B:1. Capture traffic received and sent by TS1 for the duration of this test.

B:2. Have TS1 send Pdelay_Req messages without the *IEEE 802.1Q tag*.

B:3. Wait up to 10 s or for 3 Pdelay_Resp Messages to be received from the DUT.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	Pdelay_Resp messages are received from the DUT.
B:3	PASS	Pdelay_Resp messages are not received from the DUT.

Possible Problems: None

Test PWR.c.5.6 – TransportSpecific field checking upon receipt

Purpose: To verify only messages with recognized transportSpecific subtypes are processed.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	IEEE 802.1 AS (gPTP) support must be reported if present.
B	GMC	IEEE 802.1 AS (gPTP) support must be reported if present.

References: [1] IEEE Std C37.238-2011: sub-clause 5.6
[2] IEEE Std 1588-2008: Annex F.4

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-11-12 Preview release

Discussion: This test will validate that the DUT only accepts PTP frames with a recognized transportSpecific subtype. By Default, including for Power Profile devices, the transportSpecific bit should be zero (0). For generalized PTP (IEEE 802.1AS) devices, a value of 1 should be used. If a DUT does not recognize a subtype (per [2]): "then the message is treated as any other message with an unrecognized EtherType", thus the PTP frame should not be acted upon (just as though the EtherType was not 88-F7).

Refer to Appendix C Table 6: IEEE C37.238 Message Fields
and Table 9: IEEE 802.1 Q Header Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: Pdelay_Req messages with transportSpecific of one
A:1. If the DUT supports 802.1AS then this test should stop here.
A:2. Have TS1 send Pdelay_Req messages with the *transportSpecific* field set to 1.
A:3. Wait 10 s observing all Pdelay messages from the DUT.

Observable Results:

Part:Step	Status	Description
A:1	NA	The DUT supports 802.1AS
A:3	FAIL	The DUT responds to received Pdelay_Req messages with transportSpecific set to 1.
A:3	PASS	The DUT ignores Pdelay_Req messages with transportSpecific set to 1.

Part B: Announce messages with transportSpecific of one

B:1. If the DUT supports 802.1AS then this test should stop here.
B:2. Wait for TS1 to receive Announce messages. Note the *priority1* value in the received messages.
B:3. From TS1, send Announce messages every second with a *prioirty1* value less (better) than that transmitted by the DUT. The *transportSpecific* field in the transmitted Announce should be one (1).
B:4. Wait 10 s observing all Announce messages received at TS1 from the DUT.

Observable Results:

Part:Step	Status	Description
B:1	NA	The DUT supports 802.1AS
B:2	FAIL	TS1 does not receive an Announce message.
B:4	FAIL	TS1 does not receive an Announce message within 1.5 s of the last received announce message.
B:4	PASS	The DUT ignores better Announce messages with transportSpecific set to 1.

Possible Problems: None

GROUP 6: Timescale

Overview:

This group covers requirements defined in the IEEE Std C37.238-2011 sub-clause 5.9, identifying the PTP timescale to be used. The tests in this group validate the value of the currentUtcOffset, clockClass, clockAccuracy, grandmasterIdentity fields by observing the corresponding fields in Announce messages. Tests in this group will also validate the re-synchronization behavior of devices by observing the specified offset threshold, slew rate and offset from grandmaster over time.

Notes:

Test PWR.c.6.1 – PTP Timescale

Purpose: To verify that devices in a substation use the PTP timescale.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.9.5
[2] IEEE Std 1588-2008: sub-clause 8.2.4.8
[3] IEEE Std 1588-2008: sub-clause 7.6.2.6

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-14	Preview release

Discussion: This test will validate that ordinary and boundary clocks in a substation system support the PTP timescale in the event they become grandmaster by observing the value of the timeSource field of Announce messages emitted from the DUT and by request [1]. For grandmaster clocks, the value of the timePropertiesDS.ptpTimescale should be TRUE [2]. In particular, per [3], the timeSource value must be 0x40, deduced from:

Refer to Appendix C Table 10: timeSource

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Timescale

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Make the DUT grandmaster by sending 3 Announce messages from TS1 with higher priority1 values.
A:3. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
A:4. Observe the originTimestamp and timeSource fields.
A:5. Observe the DUT's timeSource,
  a. by requesting ieeeC37238timePropDS.TimeSource, if SNMP is supported, or
  b. by means provided, if observable.
A:6. Observe the DUT's timescale,
  a. by requesting ieeeC37238timePropDS.PTPTimescale, if SNMP is supported, or
  b. by means provided, if observable.
A:7. Observe the DUT's FrqTraceable,
  a. by requesting ieeeC37238timePropDS.FrqTraceable, if SNMP is supported, or
  b. by means provided, if observable.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	No announce message is received.
A:4	FAIL	The originTimestamp observed is in the PTP timescale.
A:5	FAIL	The value of the timeSource observed and requested is not 0x40.
A:6	FAIL	The timescale is not TRUE or PTP.
A:7	FAIL	The FrqTraceable is FALSE indicating the frequency is not traceable to a primary reference.
A:7	PASS	The device supports the PTP timescale.

Possible Problems: None

Test PWR.c.6.2 – Current Utc Offset

Purpose: To verify the value of timePropertiesDS.currentUtcOffset is TAI – UTC.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	SNMP or means of observing timePropertiesDS.currentUtcOffset

References: [1] IEEE Std C37.238-2011: sub-clause 5.9
[2] IEEE Std 1588-2008: sub-clause 7.2.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-16 Preview release

Discussion: This test will validate the value of timePropertiesDS.currentUtcOffset is TAI-UTC by request. The value should be TAI – UTC (35 s as of 2014).

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: Timescale
A:1. Observe the DUT's timePropertiesDS.currentUtcOffset,
  a. by requesting ieeeC37238timPropDS.CurUTCOfst, if SNMP is supported, or
  b. by means provided, if observable.

Observable Results:

Part:Step	Sta-tus	Description
A:1	FAIL	The currentUtcOffset is not 35+n, where n is the number of leap seconds after July 1, 2012.
A:1	PASS	The currentUtcOffset is TAI – UTC.

Possible Problems: None

Test PWR.c.6.3 – Grandmaster Clock Class

Purpose: To verify the clockClass of a device.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGM

References: [1] IEEE Std 1588-2008: sub-clause 5.3.7
[2] IEEE Std C37.238-2011: sub-clause 5.9.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-16	Preview release

Discussion: This test will validate the DUT's clockClass value by observing the first octet of the clock quality field in Announce messages emitted from the DUT [1]. The clock quality is made up of three fields; clockClass (UInteger8), clockAccuracy (Enumeration8) and offsetScaledLogVariance (UInteger16). Any preferred grandmaster clock shall operate as a clockClass 6 clock while synchronized to a primary reference time source with a PTP timescale distribution [2].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: ClockClass 6

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
A:2. Observe the clockClass field.
A:3. Observe the DUT's clockClass,
  a. by requesting ieeeC37238timePropDS. clockClass, if SNMP is supported, or
  b. by means provided, if observable.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	No Announce messages are received.
A:3	PASS	The clockClass observed and requested is 6.

Possible Problems: None

Test PWR.c.6.4 – Grandmaster Degradation of Clock Class

Purpose: To verify the device properly degrades the clockClass.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGM

References:
[1] IEEE Std 1588-2008: sub-clause 5.3.7
[2] IEEE Std C37.238-2011: sub-clause 5.9.2
[3] IEEE Std C37.238-2011: sub-clause 5.9.3
[4] IEEE Std 1588-2008: sub-clause 7.6.2.4

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-16	Preview release

Discussion: This test will validate the DUT's clockClass value degrades properly by observing the first octet of the clock quality field in Announce messages emitted from the DUT [1]. Any preferred grandmaster clock shall operate as a clockClass 6 clock while synchronized to a primary reference time source with a PTP timescale distribution [2]. When the clock loses the ability to synchronize to a primary reference time source and is in holdover mode, it should degrade to clockClass 7 [3]. A degradation alternative for clockClass 7 is clockClass 187 for clocks that are not within the holdover specification [4]. A clock with clockClass 187 may be a slave to another clock in the domain.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: ClockClass 6 to 7 to 187

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
A:3. Observe the clockClass field in each Announce message.
A:4. Observe the DUT's clockClass,
  a. by requesting ieeeC37238timePropDS.clockClass, if SNMP is supported, or
  b. by means provided, if observable.
A:5. Disable any of the DUT's primary reference time sources.
A:6. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
A:7. Observe the clockClass field in each Announce message.
A:8. Observe the DUT's clockClass,
  a. by requesting ieeeC37238timePropDS.clockClass, if SNMP is supported, or
  b. by means provided, if observable.
A:9. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
A:10. Ensure that the DUT is not within holdover specification by observing the clockAccuracy field of Announce messages or requesting the ieeeC37238defaultDS.ClkAccuracy until the value exceeds 0x24.
A:11. Observe the clockClass field of Announce messages.
A:12. Observe the DUT's clockClass,
  a. by requesting ieeeC37238timePropDS.clockClass, if SNMP is supported, or
  b. by means provided, if observable.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Announce messages are received.
A:3	FAIL	The clockClass is not 6.
A:4	FAIL	The clockClass is not 6.
A:5	FAIL	The DUT has the ability to synchronize to a primary reference time source.
A:6	FAIL	No Announce messages are received.
A:7	FAIL	The clockClass does not degrade to 7.
A:8	FAIL	The clockClass does not degrade to 7.
A:9	FAIL	No Announce messages are received.
A:10	FAIL	The DUT has the ability to synchronize to a primary reference time source.
A:10	FAIL	The DUT is within the holdover specification (i.e. clockAccuracy is less than 0x24).
A:11	FAIL	The clockClass does not degrade to 187.
A:12	FAIL	The clockClass does not degrade to 187.
A:12	PASS	The clockClass degrades properly.

Possible Problems: If the clockAccuracy value is unknown, 0xFE, a different technique of verifying holdover mode will need to be used.

Test PWR.c.6.5 – Slave-Only Clock Class

Purpose: To verify the clockClass of a device.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	Slave-Only, SNMP or means of observing the DUT's clockClass

References: [1] IEEE Std C37.238-2011: sub-clause 5.4.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-16	Preview release

Discussion: This test will validate that slave-only clocks have a clockClass of 255 by requesting the defaultDS.clockQuality.clockClass [1].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: ClockClass 255

A:1. Observe the DUT's clockClass,
   a. by requesting ieeeC37238defaultDS.ClkClass, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The clockClass is not 255.
A:1	PASS	The clockClass is 255.

Possible Problems: None

Test PWR.c.6.6 – Clock Accuracy

Purpose: To verify the clock accuracy of a device.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGM, Capable of disconnecting primary reference

References: [1] IEEE Std 1588-2008: sub-clause 5.3.7
[2] IEEE Std C37.238-2011: sub-clause 5.9.4
[3] IEEE Std 1588-2008: sub-clause 7.6.2.5

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames. *Precision Pulse Moniter*: A two channel oscilloscope or equivalent system that can observe the PTP time-synced output (1PPS, etc) of the DUT's output versus that of the DUT's primary reference with known accuracy.

Modification History:	2013-01-08	Preview release

Discussion: This test will validate the DUT’s clockAccuracy by observing the second octet of the clock quality field in Announce messages emitted from the DUT [1]. Any preferred grandmaster clock shall be designed to degrade its clockAccuracy when appropriate [2]. First, when the DUT is synchronized to a primary reference its clockAccuracy should be in the range of 0x20 – 0x23 (25 ns – 1 µs). Second, when the device loses the ability to synchronize to a primary reference but is still in holdover mode its clockAccuracy should be in the range of 0x20 – 0x24 (25 ns – 2.5 µs). Grandmaster-capable devices shall be in holdover mode within 2 µs for up to 5 s at a constant temperature. After it is not in holdover mode the DUT’s clockAccuracy should be in the range of 0x24 – 0xFE (2.5 µs to Unknown). Refer to Appendix C Table 11: clockAccuracy enumeration

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP. For this test connect the *Precision Pulse Monitor* to the time reference (1PPS) of the DUT and that of it's primary reference.

Test Procedure:

Part A: ClockAccuracy

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Measure the instantaneous absolute value of the delta between the DUT's 1PPS signal vs. that of its primary reference for the duration of this test.
A:3. Wait up to 5 s or for 3 Announce messages to be received from the DUT.
A:4. Observe the clockAccuracy field.
A:5. Observe the DUT's clockAccuracy,
  a. by requesting ieeeC37238timePropDS.clockAccuracy, if SNMP is supported, or
  b. by means provided, if observable.
A:6. Disable any of the DUT's primary reference time sources.
A:7. Wait up to 5 s or for 3 Announce messages to be received from the DUT.
A:8. Observe the clockAccuracy field.
A:9. Observe the DUT's clockAccuracy,
  a. by requesting ieeeC37238timePropDS.clockAccuracy, if SNMP is supported, or
  b. by means provided, if observable.
A:10. Wait until the DUT's clockAccuracy exceeds 2us to ensure that the DUT is not within holdover specification.
A:11. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
A:12. Observe the clockAccuracy field.
A:13. Observe the DUT's clockAccuracy,
  a. by requesting ieeeC37238timePropDS.clockAccuracy, if SNMP is supported, or
  b. by means provided, if observable.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	Any measured delta value exceeds 1 µs plus the maximum possible difference between the reference time and true time.
A:3	FAIL	No Announce messages are received.
A:4	FAIL	The clockAccuracy is not in the range of 0x20–0x23.
A:4	FAIL	The clockAccuracy reported is not accurate according to the measured delta value.
A:5	FAIL	The clockAccuracy is not in the range of 0x20–0x23.
A:5	FAIL	The clockAccuracy reported is not accurate according to the measured delta value.
A:6	FAIL	The DUT has the ability to synchronize to a primary reference time source.
A:7	FAIL	No Announce messages are received.
A:8	FAIL	The clockAccuracy is not in the range of 0x20–0x24.
A:8	FAIL	The clockAccuracy reported is not accurate according to the measured delta value.
A:9	FAIL	The clockAccuracy is not in the range of 0x20–0x24.
A:9	FAIL	The clockAccuracy reported is not accurate according to the measured delta value.
A:10	FAIL	The DUT is within the holdover specification.
A:11	FAIL	No Announce messages are received.
A:12	FAIL	The clockAccuracy is not in the range of 0x24–0x31 or 0xFE.
A:12	FAIL	The clockAccuracy reported is not accurate according to the measured delta value.
A:13	FAIL	The clockAccuracy is not in the range of 0x24–0x31 or 0xFE.
A:13	FAIL	The clockAccuracy reported is not accurate according to the measured delta value.
A:13	FAIL	Any clockAccuracy reported by the DUT is within the range of 0x80–0xFD indicating use by alternate PTP profiles.
A:13	WARN	Any clockAccuracy reported by the DUT is within the range of 0x00–0x1F, 0x32–0x7F or 0xFF indicating it is reserved.
A:13	PASS	The clockAccuracy degrades properly.

This publication is available free of charge from http://dx.doi.org/10.6028/NIST.IR.8002

Possible Problems: The clockAccuracy will be 0x00-0x1F, 0x32-0x7F or 0xFF if reserved. The clockAccuracy will be 0x80-0xFD for use by alternate PTP profiles. The clockAccuracy will also be 0xFE if unknown.

Test PWR.c.6.7 – Holdover Drift for Grandmasters

Purpose: To verify the holdover drift of grandmaster capable devices.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

References: [1] IEEE Std C37.238-2011: Annex B

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames. *Precision Pulse Monitor*: A two channel oscilloscope or equivalent system that can observe the PTP time-synced output (1PPS, etc) of the grandmaster Test Station's output (1PPS or equivalent) vs that of the Slave DUT's with 40 ns or better accuracy.

Modification History: 2013-05-20 Preview release

Discussion: When a clock is no longer synchronized to another clock, it is free running on its own internal oscillator. As long as the free running clock stays within its accuracy requirements, it is considered in holdover mode. This test will validate that grandmaster capable devices do not drift beyond 2 µs for up to 5 s at a constant temperature [1].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP. Connect the *Precision Pulse Monitor* to the time reference (1PPS or 1PulsePer125µs) reference of the DUT and that of the TS1. The DUT's only time reference is TS1; therefore TS1 should be a very stable reference.

Test Procedure:

Part A: Holdover Drift

A:1. Set the grandmaster priority1 value of TS1 to be less than that of the DUT.
A:2. Measure the instantaneous absolute value of the delta between TS1's 1PPS signal vs. that of the DUT.
A:3. Repeat step A:2 until they are within 1µs of each other.
A:4. Disconnect the DUT from TS1.
A:5. Measure the instantaneous absolute value of each of the deltas between TS1's 1PPS signal vs. that of the DUT.
A:6. Repeat step A:5 for 5 s.
A:7. Report temperature and environment of test location.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	The final measured delta value exceeds 1 µs.
A:6	FAIL	Before the DUT has been free running for 5 s, any measured delta value exceeds 2 µs plus the maximum possible difference between the reference time and true time.
A:6	PASS	The DUT remains in holdover mode for at least 5 s.
A:7	INFO	The temperature and environment of the test location are:

Possible Problems: None

Test PWR.c.6.8 – GrandmasterID

Purpose: To verify that grandmaster clocks have configurable grandmaster IDs.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.9.6
[2] IEEE Std C37.238-2011: sub-clause 5.12.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-08	Preview release

Discussion: This test will validate that the grandmasterID is configurable by varying the value of the grandmasterID field and observing it in Announce messages emitted from the DUT. Grandmaster clocks shall have a configurable 1-byte ID communicated through the IEEE_C37_238 TLV that appends Announce messages [1]. Therefore grandmaster clocks should allow us to write the defaultDS.GMIdentity to any value in the range of 0x0003-0x00FE [2].

This test will also validate that Announce messages are not sent before a unique grandmasterID has been assigned by observing the value of the grandmasterID field in the first 5 Announce messages emitted from the DUT and comparing it with others. This will require the knowledge of the grandmasterIDs assigned to the other devices on the system.

Refer to Appendix C Table 12: TLV Organization Extension Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Unique Grandmaster Identity

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s or for 5 Announce messages to be received from the DUT.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Announce messages are received.
A:2	FAIL	The grandmaster identity in each message is not in the range of 0x0003-0x00FE.
A:2	PASS	The grandmaster identity in all five Announce messages is unique to that of other device's grandmaster identity.

Part B: Configurable Grandmaster Identity

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Make the DUT's grandmaster identity 0x0003 by writing ieeeC37238defaultDS.GMIdentity.
B:3. Wait up to 10 s or for 3 Announce messages to be received from the DUT.
B:4. Observe the grandmasterIdentity field.
B:5. Observe the DUT's grandmaster identity by requesting ieeeC37238defaultDS.GMIdentity.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	No Announce messages are received.
B:4	FAIL	The grandmaster identity is not 0x0003.
B:5	PASS	The grandmaster identity is 0x0003.

Possible Problems: None

Test PWR.c.6.9 – Re-synchronization Behavior

Purpose: To verify that the re-synchronization behavior is specified and performed correct.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	OC, BC	GMC
C, D	OC, BC	GMC, SNMP or means of observing the DUT's offset from grandmaster

References: [1] IEEE Std C37.238-2011: sub-clause 5.9

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-16	Preview release

Discussion: This test will first validate that the threshold of the device is documented by requesting the value. The threshold is offset from primary reference where the clock will step its time [1]. Second, this test will validate that the slew rate is specified by requesting the slew rate. The slew rate is the rate at which the clock constantly slews after passing the threshold [1].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Offset Threshold

A:1. Observe the DUT's threshold,
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The threshold cannot be observed.
A:1	PASS	The threshold is documented.

Part B: Slew Rate

B:1. Observe the DUT's slew rate
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.

Observable Results:

Part:Step	Status	Description
B:1	FAIL	The slew rate cannot be observed.
B:1	PASS	The slew rate is specified.

Part C: Step, offset > threshold

C:1. Capture traffic received by TS1 for the duration of this test.
C:2. Generate and send Announce, Sync and Follow_Up messages once per second from TS1 for the duration of this test. Ensure the priority1 value of the Announce message is better (less) than that of the DUTs.
C:3. After each message is sent, observe the DUT's offset from master by requesting ieeeC37238currentDS.OfstFrMaster.
C:4. Ensure the DUT has reached steady state where the offset from master is within the reported threshold.
C:5. Cease all communication with the DUT for 1 minute.
C:6. Resume sending Announce, Sync and Follow_Up again. This time, add two times the value of the reported threshold to the originTimestamp in Sync messages and to the preciseOriginTimestamp in Follow_Up messages. This will ensure the DUT's offset from the TS1 is greater than the reported threshold where it should step its time.
C:7. After each message is sent, observe the DUT's offset from master by requesting ieeeC37238currentDS.OfstFrMaster. Observe whether the DUT steps or slews its time.
C:8. Continue steps C:6 and C:7 until the offset from grandmaster is less than the reported threshold.

Observable Results:

Part:Step	Status	Description
C:3	FAIL	The offset from master is not observed.
C:4	FAIL	The offset from master is within the reported threshold.
C:7	FAIL	The offset from master is not observed.
C:7	FAIL	The DUT slews its time.
C:8	FAIL	No MasterStepChange notification is received to indicate that a step change occurred in the current grandmaster time.
C:8	FAIL	No OfstExceedsLimit notification is received to indicate the offset from master exceeds the configurable limit.
C:8	FAIL	The offset from grandmaster does not decrease to inside the threshold in a step.
C:8	PASS	The DUT steps its time properly when outside the reported threshold.

Part D: Slew, offset < threshold

D:1. Capture traffic received by TS1 for the duration of this test.
D:2. Generate and send Announce, Sync and Follow_Up messages once per second from TS1 for the duration of this test. Ensure the priority1 value of the Announce message is better (less) than that of the DUTs.
D:3. After each message is sent, observe the DUT's offset from master by requesting ieeeC37238currentDS.OfstFrMaster.
D:4. Ensure the DUT has reached steady state where the offset from master is within the reported threshold.
D:5. Cease all communication with the DUT for 1 minute.
D:6. Resume sending Announce, Sync and Follow_Up again. This time, add half the value of the reported threshold to the originTimestamp in Sync messages and to the preciseOriginTimestamp in Follow_Up messages. This will ensure the DUT's offset from the TS1 is less than the reported threshold where it should slew its time with a constant slew rate that it reported.
D:7. After each message is sent, observe the DUT's offset from master by requesting ieeeC37238currentDS.OfstFrMaster. Observe whether the DUT steps or slews its time and the rate at which it slews its time.
D:8. Continue steps D:6 and D:7 until the offset from grandmaster is less than a quarter of the reported threshold.

Observable Results:

Part:Step	Status	Description
D:3	FAIL	The offset from master is not observed.
D:4	FAIL	The offset from master is within the reported threshold.
D:7	FAIL	The offset from master is not observed.
D:7	FAIL	The DUT steps its time.
D:8	FAIL	A MasterStepChange notification is received to indicate that a step change occurred in the current grandmaster time.
D:8	FAIL	An OfstExceedsLimit notification is received to indicate the offset from master exceeds the configurable limit.
D:8	FAIL	The offset from grandmaster does not decrease at the specified rate.
D:8	PASS	The offset from grandmaster decreases at the specified slew rate when inside the reported threshold.

Possible Problems: Parts C and D work for the case that the primary reference can be a PTP device and not just a GPS type of reference.

GROUP 7: TLVs

Overview:

This group covers requirements defined in the IEEE Std C37.238-2011 sub-clause 5.10, identifying the two TLVs appended to Announce messages. The first TLV is the Organization Extension defined in the IEEE 1588-2008 sub-clause 5.3.8. The second is a profile specific TLV called the Alternate Time Offset Indicator.

The tests in this group validate the order of the two TLVs and the field values of both TLVs by observing the fields in Announce messages emitted from the DUT. The tests in this group validate that the Organization TLV is disregarded when the organizationId or the organizationSubType is not recognized by the DUT. The tests in this group validate that Announce messages without the two mandatory TLVs are discarded. The tests in this group also validate that the correct behavior is followed by the DUT when jumps in the Alternate Time Offset Indicator happen before or after the receipt of the Announce message. Finally, the tests in this group validate that boundary clocks forward information contained in all Alternate Time Offset Indicator TLV entities in the most recent Announce messages received from their master.

Notes:

Test PWR.c.7.1 – Order of TLVs

Purpose: To verify the order of the mandatory TLVs.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC
B	TC	None

References: [1] IEEE Std C37.238-2011: sub-clause 5.12.1 [3] Ibid.: sub-clause 16.3
[2] IEEE Std 1588-2008: sub-clause 14.3

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-09-20 Preview release

Discussion: This test will validate the order of the two mandatory TLVs by observing them in Announce messages emitted from the DUT. The TLVs that append IEEE C37.238 Announce messages should start with ORGANIZATION_ EXTENSION then ALTERNATE_TIME_OFFSET_INDICATOR. These two mandatory TLVs will be followed by any additional TLVs [1].

The message header and the Announce message fields are 64 octets long. The first TLV, ORGANIZATION_EXTENSION, is 22 octets long starting with a tlvType of value 0x0003, at offset 64. Therefore the second TLV, ALTERNATE_TIME_OFFSET_INDICATOR, should have a tlvType of value 0x0009 at offset 86.

The Working Group H24/SubC7 is currently reviewing the need to require specific locations and orders of the TLVs. Pending further notice from the working group, this test may need to be revised or removed.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: TLV Order

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s or for 3 Announce Messages to be received from the DUT.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Announce message is received.
A:2	FAIL	The Announce message is appended by one TLV or none.
A:2	PASS	The first TLV after the Announce Message has a tlvType of 0x003, and the second TLV has a tlvType of 0x0009.

Part B: TLV Order in Forwarded Announce Message

B:1. Capture traffic received by TS2 for the duration of this test.
B:2. Send Announce messages from TS1 to the DUT.
B:3. Wait up to 10 s or for 3 Announce Messages to be received by TS2 from the DUT.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	No Announce message is received.
B:3	FAIL	The Announce message is appended by one TLV or none.
B:3	PASS	The first TLV after the Announce Message has a tlvType of 0x003, and the second TLV has a tlvType of 0x0009.

Possible Problems: None

Test PWR.c.7.2 – Profile-Specific TLV Default Field Values

Purpose: To validate the IEEE_C37.238 TLV default field values.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	Not Slave-Only

References: [1] IEEE Std C37.238-2011: sub-clause 5.12.2 [3] IEEE Std 1588-2008: sub-clause 16.3
[2] IEEE Std 1588-2008: sub-clause 5.3.8

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-15 Preview release

Discussion: This test will validate the IEEE_C37.238 field values by observing the TLV fields in Announce messages emitted from the DUT. When an announce message is received or sent it comprises at least 64 octets before any TLVs.
The ORGANIZATION_EXTENSION TLV is 22 octets long. There are five fields in this TLV extension; tlvType, lengthField, organizationId, organizationSubType and the dataField. The details and values for each field are located in Appendix C Table 12: TLV Organization Extension Fields. The dataField is used to communicate the grandmaster ID, grandmasterTimeInaccuracy and the networkTimeInaccuracy. The grandmasterTimeInaccuracy and networkTimeInaccuracy values are 0X00000000-FFFFFFFF, in nanoseconds, where FFFFFFFF indicates that the maximum value of approximately 4.29 s has been exceeded.
The details and values for each field of the ALTERNATE_TIME_OFFSET_INDICATOR TLV are located in Appendix C Table 13: TLV Alternate Time Offset Indicator Fields. There are eight fields in this TLV extension; tlvType, lengthField, keyField, currentOffset, jumpSeconds, timeOfNextJump, displayName and the pad [3]. The lengthField value should be two plus the number of octets in the dataField which is even. The displayName field should not be bigger than ten bytes. The pad field is either 1 or 0 byte in order to make the TLV an even length of octets [2].

The Working Group H24/SubC7 is reviewing the lengthField value. For this test we will assume the lengthField value of 0x000C to is incorrect and will instead use the lengthField value of 0x0012.

This test will also validate that Announce messages are not sent until the ALTERNATE_TIME_OFFSET_INDICATOR TLV has been configured by observing the values of the corresponding field in the first 5 Announce messages emitted from the TLV.

Refer to Appendix C Table 12: TLV Organization Extension Fields
and Table 13: TLV Alternate Time Offset Indicator Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Encoding of ORGANIZATION_EXTENSION TLV

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.
A:2. If the DUT is a transparent clock, send Announce messages from TS1 to the DUT.
A:3. Wait up to 10 s or for 5 Announce messages to be received from the DUT.
A:4. Find a TLV that starts with the tlvType value (2 octets) that is 0x0003.
A:5. Repeat steps A:3-4 for the first 5 Announce messages emitted from the DUT.

Observable Results:

Part:Step	Status	Description
A:4	FAIL	No TLV is found with the tlvType value (2 octets at tlv-specific offset 0) 0x0003.
A:4	FAIL	The lengthField value (2 octets at tlv-specific offset 2) is anything other than 0x0012.
A:4	FAIL	The organizationId value (3 octets at tlv-specific offset 4) is anything other than 0x1C129D.
A:4	FAIL	The organizationSubType value (4 octets at tlv-specific offset 7) is anything other than 0x000001.
A:4	FAIL	The grandmasterID value (2 octets at tlv-specific offset 10) is anything other than a value in the range 0x0003-00FE.
A:4	FAIL	The reserved field value (2 octets at tlv-specific offset 20) is anything other than 0x00.
A:4	FAIL	The default values of the ORGANIZATION_EXTENSION TLV for the following fields are **not** valid: tlvType, lengthField, organizationId, organizationSubType, grandmasterId, reserved.
A:5	PASS	The default values of the ORGANIZATION_EXTENSION TLV are correct in all 5 Announce messages.

Part B: Encoding of ALTERNATE_TIME_OFFSET_INDICATOR TLV

B:1. Capture traffic received by TS1 and TS2 for the duration of this test.
B:2. If the DUT is a transparent clock, send Announce messages from TS1 to the DUT.
B:3. Wait up to 10 s or for 5 Announce Messages to be received from the DUT.
B:4. Find a TLV that starts with the tlvType value (2 octets) that is 0x0009.
B:5. Repeat steps B:3-4 for the first 5 Announce messages emitted from the DUT.

Observable Results:

Part:Step	Status	Description
B:4	FAIL	Notlv is found with the tlvType value (2 octets at tlv-specific offset 0) 0x0009.
B:4	FAIL	The lengthField value (2 octets at tlv-specific offset 2) is anything other than 15 + display-Name.PTPText.lengthField + pad (0 or 1 to ensure lengthField is an even value).
B:4	FAIL	The keyField value (1 octet at tlv-specific offset 4) is anything other than the alternate time.
B:4	FAIL	The currentOffset value (4 octets at tlv-specific offset 5) is anything other than the difference, in seconds, between the node's time and the alternate time.
B:4	FAIL	The jumpSeconds value (4 octets at tlv-specific offset 9) is anything other than the size, in seconds, of the next discontinuity.
B:4	FAIL	The timeOfNextJump value (6 octets at tlv-specific offset 13) is anything other than the transmitting node's time at the time the next discontinuity will occur.
B:4	FAIL	The displayName.PTPText.lengthField (1 octet at tlv-specific offset 19) is anything other than a value in the range 0x01 to 0x0a.
B:4	FAIL	The displayName.PTPText.textField (at tlv-specific offset 20) is anything other than the text name of the alternate timescale.
B:4	FAIL	The pad field value (1 octet at tlv-specific offset 20+L) is present and is anything other than 0x00.
B:4	FAIL	The pad field value (1 octet at tlv-specific offset 20+L) is **not** set to make the ALTERNATE_TIME_OFFSET_INDICATOR_TLV lengthField value even.
B:4	FAIL	The default values of the ALTERNATE_TIME_OFFSET_INDICATOR TLV for the following fields are **not** valid: tlvType, lengthField, displayName.PTPText.lengthField, and pad.
B:5	PASS	The default values of the ALTERNATE_TIME_OFFSET_INDICATOR TLV are correct in all 5 Announce messages.

Possible Problems: None

Test PWR.c.7.3 – OrganizationId and OrganizationSubType Recognition

Purpose: To verify the contents of the Organization TLV are disregarded if the PTP nodes do not recognize the organizationId or the organizationSubType.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	SNMP or means of observing the DUT's grandmasterTimeInaccuracy

References: [1] IEEE Std 1588-2008: sub-clause 14.3.2.3
[2] IEEE Std C37.238-2011: sub-clause 5.12.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-16	Preview release

Discussion: This test will validate that the contents of TLVs, except for the lengthField field, are disregarded when the PTP node does not recognize a particular organizationId or organizationSubType [1]. This will be validated by sending incorrect values in the organizationId and organizationSubType fields through Announce messages from TS1. Normally the value of the organizationId field should be 0x1C129D and the value of the organizationSubType should be 0x000001 [2]. Also the values of the grandmasterTimeInaccuracy field in the Announce messages will vary. To ensure the contents of the TLV are disregarded, the test will observe the grandmasterTimeInaccuracy. If the lengthField of the Organization Extension TLV is not disregarded the Alternate Time Offset Indicator TLV will start at offset 86 of the forwarded Announce message.

Refer to Appendix C Table 12: TLV Organization Extension Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Encoding of ORGANIZATION_EXTENSION TLV

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Observe the DUT's grandmasterTimeInaccuracy,
  a. by requesting ieeeC37238parentsDS.GMTimeInacc, if SNMP is supported, or
  b. by means provided, if observable.

A:3. Generate and send Announce messages from TS1, with the following values:
  a. organizationId = 0x1C129**D**
  b. organizationSubType = 0x00000**1**
  c. grandmasterTimeInaccuracy = 0x00000**1**

A:4. Observe the DUT's grandmasterTimeInaccuracy,
  a. by requesting ieeeC37238parentsDS.GMTimeInacc, if SNMP is supported, or
  b. by means provided, if observable.

A:5. Generate and send Announce messages from TS1, with the following values:
  a. organizationId = 0x1C129**E**
  b. organizationSubType = 0x00000**3**
  c. grandmasterTimeInaccuracy = 0x00000**6**

A:6. Observe the DUT's grandmasterTimeInaccuracy,
  a. by requesting ieeeC37238parentsDS.GMTimeInacc, if SNMP is supported, or
  b. by means provided, if observable.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	The grandmasterTimeInaccuracy is not observable.
A:4	FAIL	The grandmasterTimeInaccuracy is not 0x000001.
A:4	FAIL	The contents of the TLV, except the lengthField, are disregarded.
A:6	FAIL	The grandmasterTimeInaccuracy is 0x000006.
A:6	PASS	The contents of the TLV, except the lengthField, are disregarded.

Possible Problems: None

Test PWR.c.7.4 – Announce Messages without TLVs

Purpose: To verify that Announce messages without the two mandatory TLVs attached are ignored by the BMCA.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-D	OC, BC	GMC
E-H	OC, BC	SNMP or means of observing the DUT's grandmaster

References: [1] IEEE Std C37.238-2011: sub-clause 5.12.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-09-20 Preview release

Discussion: This test will validate Announce messages without two mandatory TLVs attached are ignored by the BMCA by sending Announce messages with varied TLVs and priority fields then observing the DUTs behavior. The BMCA makes the device with the lowest grandmasterPriority1 value grandmaster. If either or both of the mandatory TLVs are not appended to an Announce message, the message should be ignored by the BMCA. Therefore the grandmaster will remain the same, regardless of the grandmasterPriority1 value sent in the Announce message. According to reference [1], the two mandatory TLVs are ORGANIZATION_ EXTENSION and ALTERNATE_TIME_OFFSET_INDICATOR.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Both TLVs Present, TS1 < DUT

A:1. Capture traffic received by TS1 for the duration of this test part.
A:2. Wait for an Announce Messages to be received from the DUT.
A:3. Other than the grandmasterPriority1 field, all fields that influence the BMCA are identical in TS1 and DUT messages.
A:4. TS1 sends Announce messages with both mandatory TLVs attached and with the grandmasterPriority1 field set to a value less than the DUT's grandmasterPriority1 value.
A:5. Observe which device becomes grandmaster.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Announce message received
A:5	PASS	DUT does **not** continue to transmit Announce messages.

Part B: Only ORGANIZATION_EXTENSION TLV present, TS1 < DUT

B:1. Capture traffic received by TS1 for the duration of this test part.
B:2. Wait for an Announce Messages to be received from the DUT.
B:3. Other than the grandmasterPriority1 field, all fields that influence the BMCA are identical in TS1 and DUT messages.
B:4. TS1 sends Announce messages with only the ORGANIZATION_EXTENSION TLV attached and with the grandmasterPriority1 field set to a value less than the DUT's grandmasterPriority1 value.
B:5. Observe which device becomes grandmaster.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Announce message received
B:5	PASS	DUT continues to transmit Announce messages.

Part C: Only ALTERNATE_TIME_OFFSET_INDICATOR TLV present, TS1 < DUT

C:1. Capture traffic received by TS1 for the duration of this test part.
C:2. Wait for an Announce Messages to be received from the DUT.
C:3. Other than the grandmasterPriority1 field, all fields that influence the BMCA are identical in TS1 and DUT messages.
C:4. TS1 sends Announce messages with only the ALTERNATE_TIME_OFFSET_INDICATOR TLV attached and with the grandmasterPriority1 field set to a value less than the DUT's grandmasterPriority1 value.
C:5. Observe which device becomes grandmaster.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	No Announce message received
C:5	PASS	DUT continues to transmit Announce messages.

Part D: Both TLVs absent, TS1 < DUT

D:1. Capture traffic received by TS1 for the duration of this test part.
D:2. Wait for an Announce Messages to be received from the DUT.
D:3. Other than the grandmasterPriority1 field, all fields that influence the BMCA are identical in TS1 and DUT messages.
D:4. TS1 sends Announce messages without either of the mandatory TLVs attached and with the grandmaster-Priority1 field set to a value less than the DUT's grandmasterPriority1 value.
D:5. Observe which device becomes grandmaster.

Observable Results:

Part:Step	Status	Description
D:2	FAIL	No Announce message received
D:5	PASS	DUT continues to transmit Announce messages.

Part E: Both TLVs present, TS1 < TS2 < DUT

E:1. Have TS1 and TS2 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS2 to a value less than the DUT's grandmasterPriority1 value. Set the grandmasterPriority1 field of TS1 to a value less than the TS2's grandmasterPriority1 value.

E:2. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
E:2	FAIL	The DUT's grandmaster is not TS1.
E:2	PASS	The DUT's grandmaster is TS1.

Part F: Only ORGANIZATION_EXTENSION TLV present, TS1 < TS2 < DUT

F:1. Have TS2 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS2 to a value less than the DUT's grandmasterPriority1 value.

F:2. Have TS1 send Announce messages with only the ORGANIZATION_EXTENSION TLV attached. Set the grandmasterPriority1 field of TS1 to a value less than the TS2's grandmasterPriority1 value.

F:3. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
F:3	FAIL	The DUT's grandmaster is not TS2.
F:3	PASS	The DUT's grandmaster is TS2.

Part G: Only ALTERNATE_TIME_OFFSET_INDICATOR TLV present, TS1 < TS2 < DUT

G:1. Have TS2 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS2 to a value less than the DUT's grandmasterPriority1 value.

G:2. Have TS1 send Announce messages with only the ALTERNATE_TIME_OFFSET_INDICATOR TLV attached. Set the grandmasterPriority1 field of TS1 to a value less than the TS2's grandmasterPriority1 value.

G:3. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
G:3	FAIL	The DUT's grandmaster is not TS2.
G:3	PASS	The DUT's grandmaster is TS2.

Part H: Both TLVs absent, TS1 < TS2 < DUT

H:1. Have TS2 send Announce messages with both mandatory TLVs attached. Set the grandmasterPriority1 field of TS2 to a value less than the DUT's grandmasterPriority1 value.

H:2. Have TS1 send Announce messages with both mandatory TLVs absent. Set the grandmasterPriority1 field of TS1 to a value less than the TS2's grandmasterPriority1 value.

H:3. Observe the DUT's grandmaster,
   a. by requesting ieeeC37238defaultDS.GMIdentity, if SNMP is supported, or
   b. by means provided, if observable

Observable Results:

Part:Step	Status	Description
H:3	FAIL	The DUT's grandmaster is not TS2.
H:3	PASS	The DUT's grandmaster is TS2.

Possible Problems: None

Test PWR.c.7.5 – ALTERNATE_TIME_OFFSET_INDICATOR TLV with Discontinuity

Purpose: To verify that ALTERNATE_TIME_OFFSET_INDICATOR TLV prevents errors due to discontinuity.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-B	OC, BC	SNMP or means of observing the DUT's alternate time

References: [1] IEEE Std C37.238-2011: sub-clause 5.12.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2012-11-27	Preview release

Discussion: This test will validate that nodes ignore certain TLVs by sending Announce messages with varying values of the jumpSeconds and timeOfNextJump fields, then observing the alternate time of the DUT. The time and magnitude of a discontinuity are indicated using the jumpSeconds and timeOfNextJump fields of received ALTERNATE_TIME_OFFSET_INDICATOR TLVs that append Announce messages. A positive value in the jumpSeconds field indicates a discontinuity will cause the currentOffset of the alternate time to increase. Nodes shall ignore TLVs if the value of the jumpSeconds field of a received ALTERNATE_TIME_OFFSET_INDICATOR TLV entity is non-zero and the time of the receiving node is greater than the value of timeOfNextJump field of the received TLV [1]. This means the supposed forthcoming discontinuity has already passed.

In aim to ensure that the discontinuity does not impair the DUT, we observe pdelay response. However this may not be a complete validation as the impairment may be elsewhere.

Refer to Appendix C Table 13: TLV Alternate Time Offset Indicator Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Jump In Future

A:1. Capture traffic sent and received by TS1 for the duration of this test part.
A:2. Observe the DUT's alternate time,
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.
A:3. Observe that the DUT is responding to Pdelay.
A:4. Send an Announce message from TS1 with,
  a. jumpSeconds set to a value of 60 s and
  b. timeOfNextJump set to a value 180 s **greater** than the receiving node time.
A:5. Before timeOfNextJump, observe the DUT's alternate time,
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.
A:6. Before timeOfNextJump, confirm that the DUT is responding to Pdelay.
A:7. After timeOfNextJump, observe the DUT's alternate time,
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.
A:8. After timeOfNextJump, observe the DUT is responding to Pdelay.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	The DUT is not responding to Pdelay.
A:5	FAIL	If able to view, the alternate time found in step A:5 is not the same as in step A2.
A:6	FAIL	The DUT is not responding to Pdelay.
A:7	FAIL	If able to view, the alternate time found in step A:7 is not 60 less than in step A:2.
A:8	PASS	The DUT is responding to Pdelay.

Part B: Jump In Past

B:1. Capture traffic sent and received by TS1 for the duration of this test part.
B:2. Observe the DUT's alternate time,
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.
B:3. Observe that the DUT is responding to Pdelay.
B:4. Send an Announce message from TS1 with,
  a. jumpSeconds set to a value of 60 s and
  b. timeOfNextJump set to a value 180 s **less** than the receiving node time
B:5. Observe the DUT's alternate time,
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.
B:6. Observe the DUT is responding to Pdelay.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	The DUT is not responding to Pdelay.
B:5	FAIL	If able to view, the alternate time found in step B:5 is not the same as in step B2.
B:6	PASS	The DUT is responding to Pdelay.

Possible Problems:

Test PWR.c.7.6 – Sequence of Announce Messages before Discontinuity

Purpose: To verify that a contiguous sequence of Announce messages are transmitted before a discontinuity.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	PrefGMC
B	OC, BC	GMC

References: [1] IEEE Std 1588-2008: sub-clause 8.2.5.4.2
[2] IEEE Std C37.238-2011: sub-clause 5.2
[3] IEEE Std 1588-2008: sub-clause 16.3.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-11-28 Preview release

Discussion: This test will validate that a contiguous sequence of Announce messages are transmitted before a discontinuity by setting values for the jumpSeconds and timeOfNextJump fields and observing how many Announce messages are emitted from the DUT before the timeOfNextJump. The announceReceiptTimeout specifies the number of announceInterval that have to pass without receipt of an Announce message before the occurrence of the event ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES [1]. The range of this value is 2 to 255. The value of portDS.announceReceiptTimeout shall be 2 for all preferred grandmaster clocks, 3 for all other grandmaster-capable devices [2]. If a discontinuity (jump) is about to occur, the node shall indicate this in a contiguous sequence of at least portDS.announceReceiptTimeout+1 Announce messages transmitted immediately before the discontinuity [3]. Therefore immediately before the discontinuity, all preferred grandmaster clocks should send at least 3 Announce messages and all other grandmaster-capable devices should send at least 4 Announce messages indicating the discontinuity.

Refer to Appendix C Table 13: TLV Alternate Time Offset Indicator Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Preferred Grandmaster Clocks

A:1. Capture traffic received by TS1 for the duration of this test part.

A:2. Observe the currentOffset in Announce messages sent from the DUT and by requesting ieeeC37238timePropDS.LocalTCurOfs.

A:3. Change jumpSeconds by writing the object ieeeC37238timePropDS.LocalTJumpS to be 60 s. Changes the timeOfNextJump by writing the object ieeeC37238timePropDS.LocalTNtJump to be 180 s greater than the receiving node time.

A:4. Before timeOfNextJump, observe the Announce messages sent from the DUT.

A:5. After timeOfNextJump, observe the currentOffset in Announce messages sent from the DUT and by requesting ieeeC37238timePropDS.LocalTCurOfs.

Observable Results:

Part:Step	Sta-tus	Description
A:4	FAIL	In at least 3 messages, the jumpSeconds value is not 60 s.
A:4	FAIL	In at least 3 messages, the timeOfNextJump is not greater than the receiving node time by 180 s.
A:4	FAIL	At least 3 messages were not received indicating the discontinuity.
A:5	PASS	The currentOffset in the Announce message is 60 s greater than in step A:2.

Part B: Other Grandmaster-Capable Devices

B:1. Capture traffic received by TS1 for the duration of this test part.

B:2. Observe the currentOffset in Announce messages sent from the DUT and by requesting ieeeC37238timePropDS.LocalTCurOfs.

B:3. Change jumpSeconds by writing the object ieeeC37238timePropDS.LocalTJumpS to be 60 s. Changes the timeOfNextJump by writing the object ieeeC37238timePropDS.LocalTNtJump to be 180 s greater than the receiving node time.

B:4. Before timeOfNextJump, observe the Announce messages sent from the DUT.

B:5. After timeOfNextJump, observe the currentOffset in Announce messages sent from the DUT and by requesting ieeeC37238timePropDS.LocalTCurOfs.

Observable Results:

Part:Step	Status	Description
B:4	FAIL	In at least 4 messages, the jumpSeconds value is not 60 s.
B:4	FAIL	In at least 4 messages, the timeOfNextJump is not greater than the receiving node time by 180 s.
B:4	FAIL	At least 4 messages were not received indicating the discontinuity.
B:5	PASS	The currentOffset in the Announce message is 60 s greater than in step B:2.

Possible Problems: None

Test PWR.c.7.7 – ALTERNATE_TIME_OFFSET_INDICATOR TLV is not UTC

Purpose: To verify that alternate time offset indicator does not indicate the offset or pending changes in offset of UTC from the PTP timescale.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A-B	OC, BC	GMC

References: [1] IEEE Std 1588-2008: sub-clause 16.3.1
[2] IEEE Std C37.238-2011: sub-clause 5.5.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-11-27 Preview release

Discussion: This test will validate that the alternate time offset indicated is not being used to indicate the offset or pending changes in the offset of UTC from the PTP timescale by comparing the value of the currentUtcOffset field in Announce messages and the value of the currentOffset field in the Alternate Time Offset Indicator TLV emitting from the DUT [1]. The UTC timescale is the standard for clocks around the world and includes leap seconds. The PTP timescale, also known as the TAI timescale is an average of time reported from atomic clocks around the world and does not adjust for leap seconds. Therefore UTC timescale will be behind the TAI timescale (35 s as of 2014).

The offset between TAI and UTC, in seconds, is the currentUtcOffset, at offset 44 of the announce message. The offset between the node's time and the alternate time, in seconds, is the currentOffset, at offset 5 of the ALTERNATE_TIME_OFFSET_INDICATOR TLV [2]. If the node time is UTC or TAI the currentOffset TLV field will be a multiple of 1800 s.

The Working Group H24/SubC7 is currently reviewing the definition of currentOffset with regards to its value being the offset of the alternate time, in seconds, from the node's time rather than from the UTC time. Once the relevant requirements in [1] and the IEEE Std 1588-2008 sub-clause 16.3.3.3.4 are clarified this test may need revision.

Refer to Appendix C Table 5: Announce Message Fields
and Table 13: TLV Alternate Time Offset Indicator Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: Comparing CurUTCOfst and currentOffset
A:1. Capture traffic received by TS1 for the duration of this test part.
A:2. Wait for an Announce Messages to be received from the DUT.
A:3. Observe these fields of the Announce messages:
   a. CurUTCOfst at offset 44 of Announce message
   b. currentOffset at offset 5 of the Alternate Time Offset Indicator TLV

Observable Results:

Part:Step	Sta-tus	Description
A:2	FAIL	No Announce message received.
A:3	PASS	The CurUTCOfst is not the same as currentOffset.

Part B: Multiple of 1800

B:1. Capture traffic received by TS1 for the duration of this test part.
B:2. Wait for an Announce Messages to be received from the DUT.
B:3. Observe the currentOffset, at offset 5 of the Alternate Time Offset Indicator TLV attached to the Announce messages.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Announce message received.
B:3	PASS	The currentOffset is a multiple of 1800.

Possible Problems: None

Test PWR.c.7.8 – Boundary Clocks Forwarding ALTERNATE_TIME_OFFSET_INDICATOR

Purpose: To verify boundary clocks forward the information contained in the ALTERNATE_TIME_OFFSET_INDICATOR TLV in transmitted Announce messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC	None

References: [1] IEEE Std 1588-2008: sub-clause 16.3.1

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-12-04 Preview release

Discussion: This test validates that boundary clocks, that are not the grandmaster and that implement the alternate timescale option, forward the information contained in all ALTERNATE_TIME_OFFSET_INDICATOR TLV entities contained in the most recent Announce message received from its master in any Announce message that it transmits [1]. This is validated by emitting Announce messages from TS1 and observing whether the DUT forwards the same entities to TS2.

Refer to Appendix C Table 13: TLV Alternate Time Offset Indicator Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A:

A:1. Capture traffic sent by TS1 and received by TS2 for the duration of this test part.

A:2. TS1 sends Announce messages, with both mandatory TLVs attached and with the grandmasterPriority1 field set to a value less than the DUT's grandmasterPriority1 value, to the DUT.

A:3. Observe the Announce messages that are received by TS2.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No announce messages sent from TS1.
A:3	FAIL	No announce messages received by TS2.
A:3	FAIL	The ALTERNATE_TIME_OFFSET_INDICATOR TLV is not attached to each message.
A:3	PASS	Each ALTERNATE_TIME_OFFSET_INDICATOR TLV entity is identical to entity when sent from TS1.

Possible Problems: Test may not validate the Boundary clocks are forwarding the most recent data received.

GROUP 8: Time Inaccuracy

Overview:

This group covers requirements defined in sub-clause 5.13 of the IEEE Std C37.238-2011, identifying how the time inaccuracy operates by using the profile specific IEEE_C37_238 TLV. The time inaccuracy comprises three parts: grandmaster, network and local.

Notes:

Test PWR.c.8.1 – LocalTimeInaccuracy for Grandmaster Clocks

Purpose: To validate the LocalTimeInaccuracy for a grandmaster clock does not exceed the specified maximum value and takes the correct action in the case the clock's offset exceeds the threshold.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.13
[2] IEEE Std C37.238-2011: Annex B

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-05-29 Preview release

Discussion: This test will validate the LocalTimeInaccuracy does not exceed the specified maximum value by requesting the currentDS.LocTimeInacc value or observing as specified [1].
The TimeInaccuracy should be less than 200 ns for grandmaster clocks [2]. The only exception for that requirement is the case that it is being used for shorter PTP communication paths (less than 16 network hops) with an accumulated TimeInaccuracy better than 1us.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: Reported Inaccuracy
A:1. Observe the DUT's LocalTimeInaccuracy by requesting ieeeC37238defaultDS.LocTimeInacc.
A:2. Observe the DUT's current LocalTimeInaccuracy by requesting ieeeC37238currentDS.LocTimeInacc.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The LocalTimeInaccuracy is not observed.
A:1	NOTE	The LocalTimeInaccuracy is not less than 200 ns.
A:2	PASS	The DUT's current LocalTimeInaccuracy is less than or equal to the value found in step A:1.

Possible Problems: *"Grandmaster clocks with TimeInaccuracy that exceeds 0.2 µs may be used for shorter PTP communication paths (less than 16 network hops), if the TimeInaccuracy accumulated in the communication path is better than 1 µs."

Test PWR.c.8.2 – TimeInaccuracy for Grandmaster Clocks

Purpose: To validate the TimeInaccuracy of a grandmaster clock does not exceed the specified maximum value.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, OC	1PPS input, GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.13
[2] IEEE Std C37.238-2011: Annex B

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames.
Traffic Generator capable of arbitrary frame generation with hardware timestamping of transmitted frames.
Traffic Monitor capable of capturing traffic and hardware timestamping of received frames.
Error of time source must be known. One cable with known delay.

Modification History:	2013-08-28	Preview release

Discussion: This test will validate the TimeInaccuracy of the grandmaster capable device does not exceed the maximum value by comparing the values in messages emitted from the DUT with calculations of those values. The TimeInaccuracy should be less than 0.2 µs for grandmaster clocks [2]. The only exception for that requirement is the case that it is being used for shorter PTP communication paths (less than 16 network hops) with an accumulated TimeInaccuracy better than 1µs.

The TimeInaccuracy is calculated as follows:

$$TimeInaccuracy = trx_{sync} - POT_{sync}$$

Where trx_{sync} is the time the sync message was transmitted and POT_{sync} is the Precise Origin Timestamp in the Sync message.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP. In addition, synchronize the DUT to the test station using the DUT's 1PPS input. Connect the DUT to TS1 with a cable of known delay. The speed of light in Ethernet cables is approximately 2/3 *c*, or 5.0 ns/m, so every 10 m of cable length difference contributes a difference in meanPathDelay of approximately 50 ns.

Test Procedure:

Part A: Observed TimeInaccuracy through Sync Precise Origin Timestamp

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Capture all Sync messages and if two-step, Follow_Up messages from DUT and observe the Precise Origin Timestamp (POT_{sync}) in each.
A:3. Acquire the actual time (trx_{sync}) the Sync message was transmitted by subtracting the link delay from TS1's hardware timestamp.
A:4. Calculate the TimeInaccuracy from at least 10 Sync messages.
A:5. Calculate the mean of the TimeInaccuracies.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Sync messages were observed.
A:3	FAIL	Less than 10 trx_{sync}values were acquired.
A:5	FAIL	The mean of all TimeInaccuracy calculations is greater than or equal to 200 ns.
A:5	PASS	The mean of all TimeInaccuracy calculations is less than 200 ns.

Possible Problems: None.

Test PWR.c.8.3 – LocalTimeInaccuracy for Transparent Clocks

Purpose: To validate the TimeInaccuracy of a transparent clock does not exceed the specified maximum value and takes the correct action when in the case that it does.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	TC	No SNMP, Means of observing the DUT's LocalTimeInaccuracy
B	TC	SNMP
C	TC	Means of setting the DUT's LocalTimeInaccuracy

References: [1] IEEE Std C37.238-2011: sub-clause 5.13
[2] IEEE Std C37.238-2011: Annex B

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-05-29	Preview release

Discussion: This test will first validate the LocalTimeInaccuracy is specified by the device manufacturer. Second, this test will validate the LocalTimeInaccuracy that is specified does not exceed 50 ns.

This test also validates that transparent clocks do not forward messages when their LocalTimeInaccuracy exceeds the specified maximum LocalTimeInaccuracy. This is tested by observing the DUT forward messages, then setting the LocalTimeInaccuracy greater than the maximum and then observing whether it continues to forward messages.

The requirement that TCs shall not forward any IEEE C37.238 messages, while their LocalTimeInaccuracy exceeds its specified maximum LocalTimeInaccuracy is currently under review by the Working Group H24/SubC7. Upon final decision part C of this test may need removal.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Reported LocalTimeInaccuracy less than 50 ns

A:1. Observe the DUT's LocalTimeInaccuracy by means provided, if observable.

Observable Results:

Part:Step	Status	Description
A:1	FAIL	The LocalTimeInaccuracy is not reported.
A:1	PASS	The LocalTimeInaccuracy is less than 50 ns.

Part B: Reported LocalTimeInaccuracy less than 50 ns, SNMP

B:1. Observe the DUT's LocalTimeInaccuracy by requesting ieeeC37238TCdefaultDS.LocTimeInacc.

Observable Results:

Part:Step	Status	Description
B:1	FAIL	The LocalTimeInaccuracy is not reported.
B:1	PASS	The LocalTimeInaccuracy is less than 50 ns.

Part C: Forwarding Messages

C:1. Capture traffic received by TS1 and TS2 for the duration of this test.

C:2. Send Announce messages once a second from TS1 to the DUT for the duration of this test.

C:3. Observe the Announce messages emitted from the DUT to TS2.

C:4. Set the DUT's LocalTimeInaccuracy greater than 50 ns,

   a. by means provided, if observable, or
   b. by set, if SNMP is supported.

C:5. Observe the Announce messages emitted from the DUT to TS2.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	Announce messages are not forwarded from the DUT.
C:5	FAIL	Announce messages continue to be forwarded from the DUT.
C:5	PASS	The DUT stops forwarding Announce messages when LocalTimeInaccuracy exceeds 50 ns.

Possible Problems: The LocalTimeInaccuracy may not be observable for transparent clocks.

Test PWR.c.8.4 – TimeInaccuracy for Transparent Clocks

Purpose: To validate the TimeInaccuracy of a transparent clock does not exceed the specified maximum value.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	TC	None

References: [1] IEEE Std C37.238-2011: sub-clause 5.13
[2] IEEE Std C37.238-2011: Annex B

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames.
Traffic Generator capable of arbitrary frame generation with hardware timestamping of transmitted frames.
Traffic Monitor capable of capturing traffic and hardware timestamping of received frames.
Two-link Precision Full-Duplex In-line Traffic Monitor capable of time-stamping received frames on two different ports with the same time-base, with sub-40 ns granularity (functionality may be equivalently achieved if both test stations share a common time-base.

Modification History:	2013-05-29	Preview release

Discussion: This test will verify that TimeInaccuracy does not exceed the specified maximum value, LocalTimeInaccuracy, by comparing values observed in messages emitted from the DUT with the calculations of those values [1]. The specified LocalTimeInaccuracy should be less than 50 ns for transparent clocks [2].

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Observed TimeInaccuracy through Follow_Up correctionField value

A:1. Connect TS1 to the DUT using a Short-Cable with precisely known delay. Two cables (designated ShortCable and Long-Cable will be used, approximately 300 ns or more difference in latency.

A:2. Capture traffic received by TS1 and TS2 for the duration of this test.

A:3. Wait at most 12 s for TS1 to receive an Announce message, and observe its priority1 value.

A:4. From TS1, send Announce messages once a second with a lower (better) priority1 value so that the DUT becomes a slave to TS1.

A:5. From TS2, send Announce messages once a second with a higher (worse) priority1 value so that the DUT becomes a master to TS2.

A:6. From TS1, send valid Sync and Follow_Up messages every 125 ms. Follow_Up message's correctionField value is initially cF=3000 (0xBB8).

A:7. Wait 1 second.

A:8. Capture all Sync and Follow_Up messages from TS1 and sent by DUT.TS2 on a common time base. This can be done using the Two-link Precision Full-Duplex In-line Traffic Monitor or just TS1 and TS2 if they share a common time-base.

A:9. From the captures made in step A:8:

a. Calculate $t_{residence}$ as the residence time that a Sync message takes to traverse the bridge. That is the time from the first Sync message sent from TS1 to the first Sync message sent by DUT.TS2. Continue in this manner for each subsequent set of Sync messages.

b. Add 80 ns possible traffic monitor timestamp error to $t_{residence}$.

c. Add to $t_{residence}$ the known cable delay of the cable between the DUT and TS1 and the DUT and TS2.

d. Add 100 ns to $t_{residence}$ due to allowable error in DUT's Pdelay measurement.

e. Convert to ScaledNS (multiply $t_{residence}$ by 2^{16}).

f. Add the value cF in the Follow_Up message sent by TS1 (initially 0xBB8).

g. Compare this value ($t_{residence}$) with the correctionField value sent in the Follow_Up message from port DUT.TS2.

h. Do this calculation for at least 3 sets of Sync and Follow_Up messages.

A:10. Repeat steps A:6 through A:9, where cF is now 30000 (0x7530).

A:11. Repeat steps A:6 through A:9, with a different cable of known length (Long-cable), where cF is now 3000 (0xBB8).

Observable Results:

Part:Step	Status	Description
A:9	FAIL	Any observed Follow_Up message's correctionField value must never exceed the calculated value from step A:9.
A:9	FAIL	Any observed Follow_Up message's correctionField value must never exceed 50 ns.
A:9	FAIL	When the Long-Cable is used, the observed Follow_Up message's mean correctionField value is not observed to increase by at least 100 ns from the mean value when the Short-Cable is used.
A:9	PASS	The average of observed Follow_Up message's correctionField values does not exceed ±50 % of 50 ns.

Possible Problems: None

Test PWR.c.8.5 – GrandmasterTimeInaccuracy and NetworkTimeInaccuracy for Grandmaster Clocks

Purpose: To validate the IEEE_C37.238 TLV default field values when devices are in grandmaster state.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC

References: [1] IEEE Std C37.238-2011: sub-clause 5.13
[2] IEEE Std C37.238-2011: sub-clause 5.12.2

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-09-20 Preview release

Discussion: This test will validate the grandmasterTimeInaccuracy and the networkTimeInaccuracy by observing the values of the corresponding fields in Announce messages emitted from the grandmaster capable DUT's. The grandmasterTimeInaccuracy field is at offset 12 of the profile specific IEEE_C37_238 TLV that appends Announce messages. The value should be a more precise value of the clockAccuracy of the clockQuality field at offset 48 of Announce messages [1]. The networkTimeInaccuracy field is at offset 16 of the profile specific IEEE_C37_238 TLV that appends Announce messages. The value should be set to zero by default.

Refer to Appendix C Table 12: TLV Organization Extension Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Proper GrandmasterTimeInaccuracy and NetworkTimeInaccuracy Values

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s or for 3 Announce Messages to be received from the DUT.
A:3. Observe the ClockQuality, grandmasterTimeInaccuracy and networkTimeInaccuracy fields.
A:4. Observe the DUT's grandmasterTimeInaccuracy by requesting ieeeC37238parentDS.GMTimeInacc.
A:5. Observe the DUT's networkTimeInaccuracy by requesting ieeeC37238parentDS.NetTimeInacc and ieeeC37238defaultDS.NetTimeInacc.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Announce messages are received.
A:4	FAIL	The grandmasterTimeInaccuracy value observed (4 octets at tlv-specific offset 12) and requested is anything other than a more precise value of the clockQuality field between 0x00000000 and 0xFFFFFFFF.
A:4	FAIL	The value of the grandmasterTimeInaccuracy field observed and requested is 0xFFFFFFFF and the maximum value has not been exceeded.
A:5	FAIL	The networkTimeInaccuracy value observed (4 octets at tlv-specific offset 16) and requested is anything other than TimeInaccuracy accumulated in the worst network path between 0x00000000 and 0xFFFFFFFF.
A:5	FAIL	The value of the networkTimeInaccuracy field observed and requested is 0xFFFFFFFF and the maximum value has not been exceeded.
A:5	PASS	The grandmasterTimeInaccuracy and networkTimeInaccuracy fields are correct.

Possible Problems: None

This publication is available free of charge from http://dx.doi.org/10.6028/NIST.IR.8002

Test PWR.c.8.6 – GrandmasterTimeInaccuracy and NetworkTimeInaccuracy

Purpose: To validate the IEEE_C37.238 TLV default field values when devices are not in the grandmaster state.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	BC, TC	Means of observing the DUT's LocalTimeInaccuracy
B	BC, TC	SNMP
C	BC, TC	SNMP or means of observing the DUT's LocalTimeInaccuracy

References: [1] IEEE Std C37.238-2011: sub-clause 5.13
[2] IEEE Std C37.238-2011: sub-clause 5.12.2

Resource Requirements: Two test stations capable of transmitting and receiving arbitrary MAC frames

Modification History: 2012-09-20 Preview release

Discussion: This test will validate the grandmasterTimeInaccuracy and the networkTimeInaccuracy by observing the values of the corresponding fields in Announce messages emitted from the grandmaster capable DUT's. The grandmasterTimeInaccuracy field is at offset 12 of the profile specific IEEE_C37_238 TLV that appends Announce messages. The value should be a more precise value of the clockAccuracy of the clockQuality field at offset 48 of Announce messages [1]. The networkTimeInaccuracy field is at offset 16 of the profile specific IEEE_C37_238 TLV that appends Announce messages. The value should be set to zero by default when emitted from the TS. The DUT may increment the value with its own LocalTimeInaccuracy. When either of these fields has a value of 0xFFFFFFFF, in nanoseconds, the maximum value has been exceeded [2].

Refer to Appendix C Table 12: TLV Organization Extension Fields

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Proper GrandmasterTimeInaccuracy and NetworkTimeInaccuracy Values

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.
A:2. Send Announce Messages every second from TS1 to the DUT for the duration of this test.
A:3. Wait up to 10 s or for 3 Announce Messages to be received by TS2 from the DUT.
A:4. Observe the ClockQuality, grandmasterTimeInaccuracy and networkTimeInaccuracy fields.
A:5. Observe the DUT's LocalTimeInaccuracy by means provided, if observable.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	No Announce messages are received.
A:4	FAIL	The grandmasterTimeInaccuracy value (4 octets at tlv-specific offset 12) is anything other than that of the Announce Message emitted from TS1.
A:5	WARN	The networkTimeInaccuracy value (4 octets at tlv-specific offset 16) is not zero or the value of the LocalTimeInaccuracy of the DUT.
A:5	PASS	The grandmasterTimeInaccuracy and networkTimeInaccuracy fields are correct.

Part B: Proper GrandmasterTimeInaccuracy and NetworkTimeInaccuracy Values, SNMP

B:1. Capture traffic received by TS1 and TS2 for the duration of this test.
B:2. Send Announce Messages every second from TS1 to the DUT for the duration of this test.
B:3. Wait up to 10 s or for 3 Announce Messages to be received by TS2 from the DUT.
B:4. Observe the ClockQuality, grandmasterTimeInaccuracy and networkTimeInaccuracy fields.
B:5. Observe the DUT's LocalTimeInaccuracy by requesting ieeeC37238currentDS.LocTimeInacc.
B:6. Observe the DUT's grandmasterTimeInaccuracy by requesting ieeeC37238parentDS.GMTimeInacc.
B:7. Observe the DUT's networkTimeInaccuracy by requesting ieeeC37238parentDS.NetTimeInacc.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	No Announce messages are received.
B:5	FAIL	The LocalTimeInaccuracy is anything other than the TimeInaccuracy contribution of the device in nanoseconds.
B:6	FAIL	The grandmasterTimeInaccuracy value observed (4 octets at tlv-specific offset 12) and requested is anything other than that of the Announce Message emitted from TS1.
B:7	WARN	The networkTimeInaccuracy value observed (4 octets at tlv-specific offset 16) and requested is not zero or the value of the LocalTimeInaccuracy of the DUT.
B:7	PASS	The grandmasterTimeInaccuracy and networkTimeInaccuracy fields are correct.

Part C: NetworkTimeInaccuracy Exceeds Limit

C:1. Capture traffic received by TS1 and TS2 for the duration of this test.
C:2. Observe the DUT's LocalTimeInaccuracy,
  a. by means provided, if observable, or
  b. by request, if SNMP is supported.
C:3. Send Announce Messages every second from TS1 to the DUT for the duration of this test with the networkTimeInaccuracy field **greater** than 4.29 s subtracted by the device's LocalTimeInaccuracy.
C:4. Wait up to 10 s or for 3 Announce Messages to be received by TS2 from the DUT.
C:5. Observe the networkTimeInaccuracy fields.

Observable Results:

Part:Step	Status	Description
C:4	FAIL	No Announce messages are received.
C:5	WARN	The value of the networkTimeInaccuracy field is not 0xFFFFFFFF indicating the maximum value has not been exceeded.
C:5	PASS	The value of the networkTimeInaccuracy field is 0xFFFFFFFF indicating the maximum value has been exceeded.

Possible Problems: None

GROUP 9: Miscellaneous

Overview:

This group covers requirements defined in sub-clauses 5.7, 5.8 and 5.11 of IEEE Std C37.238-2011, identifying clock types, communication model and the clock identity. The tests in this group validate the operation of one-step and two-step modes from sub-clause 5.7, the multicast communication model from sub-clause 5.8 and how the clock identity is constructed from sub-clause 5.11.

Notes:

Test PWR.c.9.1 – Clock Identity

Purpose: To verify the clockIdentity values are EUI-64 constructed based on EUI-48.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	OC, BC	GMC
B	All	None

References: [1] IEEE Std C37.238-2011: sub-clause 5.11
[2] IEEE Std 1588-2008: sub-clause 7.5.2.2
[3] IEEE Std 1588-2008: sub-clause F.3

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-02-11	Preview release

Discussion: This test will validate the clockIdentity is EUI-64 constructed based on EUI-48 by observing the sourcePortIdentity field in messages emitted from the DUT [1]. The clockIdentity is the first eight octets of the ten octet long sourcePortIdentity field [2]. It is valid if the first three octets are the MAC address of the DUT, the following two octets are 0xFFFE, then the last three are the rest of the MAC address. In all peer delay messages the source MAC Address shall be the egress port's MAC Address [3]. The format of the sourcePortIdentity is shown in Table 14: sourcePortIdentity

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:
Part A: Announce, Sync & Follow_Up Messages
A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Wait up to 10 s for at least one message of each type to be received from the DUT.
A:3. Observe the first eight octets of the sourcePortIdentity field.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	In any messages, the first three octets are not the first half of the DUT's MAC address.
A:3	FAIL	In any messages, the middle two octets are not 0xFFFE.
A:3	FAIL	In any messages, the last three octets are not the second half of the DUT's MAC address.
A:3	FAIL	In any peer delay messages, the MAC address in the clock identity is not the egress port's MAC address.
A:3	PASS	In all messages, the clockIdentity is constructed correctly.

Part B: Pdelay_Req, Pdelay_Resp & Pdelay_Resp_Follow_Up Messages

B:1. Capture traffic received by TS1 for the duration of this test.
B:2. Wait up to 10 s for Pdelay_Req messages to be received from the DUT.
B:3. Send Pdelay_Req messages from TS1.
B:4. Wait up to 10 s for Pdelay_Resp and Pdelay_Resp_Follow_Up messages to be received from the DUT.
B:5. Observe the first eight octets of the sourcePortIdentity field.

Observable Results:

Part:Step	Status	Description
B:5	FAIL	In any messages, the first three octets are not the first half of the DUT's MAC address.
B:5	FAIL	In any messages, the middle two octets are not 0xFFFE.
B:5	FAIL	In any messages, the last three octets are not the second half of the DUT's MAC address.
B:5	FAIL	In any peer delay messages, the MAC address in the clock identity is not the egress port's MAC address.
B:5	PASS	In all messages, the clockIdentity is constructed correct.

Possible Problems: None

Test PWR.c.9.2 – Peer Delay One-Step and Two-Step Ingress Ports

Purpose: To verify all devices' ingress ports support both one-step and two-step modes of peer delay response.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	SNMP or means of observing the DUT's offset from master

References: [1] IEEE Std C37.238-2011: sub-clause 5.7.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-01-14	Preview release

Discussion: This test will validate that ingress ports support both one-step and two-step modes by simulating a jump in the delay through Pdelay_Resp and Pdelay_Resp_Follow_Up messages and observing whether the offset from grandmaster is updated [1]. First, the test station will respond to Pdelay_Req messages emitted from the DUT, with Pdelay_Resp messages, while incrementing the value of the correctionField by 1 ns each message. Second, the test station will respond to Pdelay_Req messages emitted from the DUT, with Pdelay_Resp messages, while incrementing the value of the correctionField by 5 ns each message. If the device supports one-step mode it will accept Pdelay_Resp messages and update its offsetFromMaster with this 4 nanosecond jump.

A similar process will be followed to validate the device supports two-step mode. First, the test station will respond to Pdelay_Req messages emitted from the DUT, with Pdelay_Resp and Pdelay_Resp_Follow_Up messages, while incrementing the values of the requestReceiptTimestamp and responseOriginTimestamp by 1 ns each message. Second, the test station will respond to Pdelay_Req messages emitted from the DUT, with Pdelay_Resp and Pdelay_Resp_Follow_Up messages, while incrementing the values of the requestReceiptTimestamp and responseOriginTimestamp by 5 ns each message. If the device supports two-step mode it will accept Pdelay_Resp and Pdelay_Resp_Follow_Up messages and update its offsetFromMaster with this 4 nanosecond jump.

For one-step clocks the calculations for the meanPathDelay and offsetFromMaster are shown below.

$$mPD = \frac{(t_4 - t_1) - correctionField\ of\ Pdelay\_Resp}{2}$$

$$offsetFromMaster = syncEventIngressTimestamp - originTimestamp - mPD - correctionField\ of\ sync\ message$$

For two-step clocks the calculations for the meanPathDelay and offsetFromMaster are shown below.

$$mPD = \frac{(t_4 - t_1) - (responseOriginTimestamp - requestReceiptTimestamp) - correctionFields}{2}$$

$$offsetFromMaster = syncEventIngressTimestamp - preciseOriginTimestamp - mPD - correctionField\ of\ sync\ message - correctionField\ of\ Follow\_Up\ message$$

Where, *mPD* is the value of the meanPathDelay measured and computed for each instance of a peer delay request-response measurement.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Pdelay_Resp, One-step mode

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Wait up to 10 s for at least one Pdelay_Req message to be received from the DUT.

A:3. In response, generate and send 30 Pdelay_Resp messages from TS1, while incrementing the correctionField by 1 ns in each message.

A:4. After each message is sent, observe the DUT's offset from master,

a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
b. by means provided, as specified.

A:5. Generate and send 30 Pdelay_Resp messages from TS1, while incrementing the correctionField by 5 ns.

A:6. After each message is sent, observe the DUT's offset from master,

a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
b. by means provided, as specified.

A:7. Repeat steps A:1-5 for each ingress port on the DUT.

A:8. Observe the DUT's logAnnounceInterval by requesting ieeeC37238portDS.LogAnnounceInt.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Pdelay_Req messages are received.
A:3	FAIL	The correctionField in each Pdelay_Resp message sent is 1 ns greater than the last.
A:4	FAIL	The offset from the master is not observed.
A:5	FAIL	The correctionField in each Pdelay_Resp message sent is 5 ns greater than the last.
A:6	FAIL	A jump in the offset from the master has occurred.
A:7	PASS	All ingress ports on the device support one-step mode.

Part B: Pdelay_Resp and Pdelay_Resp_Follow_Up, Two-step mode

B:1. Capture traffic received by TS1 for the duration of this test.

B:2. Wait up to 10 s for at least one Pdelay_Req message to be received from the DUT.

B:3. In response, generate and send 30 Pdelay_Resp messages from TS1, while incrementing the requestReceiptTimestamp by 1 ns in each message. Also, generate and send 30 Pdelay_Resp_Follow_Up messages from TS1, while incrementing the responseOriginTimestamp by 1 ns in each message

B:4. After each message is sent, observe the DUT's offset from master,
   a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
   b. by means provided, as specified.

B:5. Generate and send 30 Pdelay_Resp messages from TS1, while incrementing the requestReceiptTimestamp by 5 ns in each message. Also, generate and send 30 Pdelay_Resp_Follow_Up messages from TS1, while incrementing the responseOriginTimestamp by 5 ns in each message.

B:6. After each message is sent, observe the DUT's offset from master,
   a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
   b. by means provided, as specified.

B:7. Repeat steps B:1-6 for each ingress port on the DUT.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req messages are received.
B:3	FAIL	The requestReceiptTimestamp in each Pdelay_Resp message sent is 1 ns greater than the last.
B:3	FAIL	The responseOriginTimestamp in each Pdelay_Resp_Follow_Up message sent is 1 ns greater than the last.
B:4	FAIL	The offset from the master is not observed.
B:5	FAIL	The requestReceiptTimestamp in each Pdelay_Resp message sent is 5 ns greater than the last.
B:5	FAIL	The responseOriginTimestamp in each Pdelay_Resp_Follow_Up message sent is 5 ns greater than the last.
B:6	FAIL	A jump in the offset from the master has occurred.
B:7	PASS	All ingress ports on the device support two-step mode.

Possible Problems: None

Test PWR.c.9.3 – Sync One-Step and Two-Step Ingress Ports

Purpose: To verify all devices' ingress ports support both one-step and two-step modes of sync messages.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A, B	All	SNMP or means of observing the DUT's offset from master

References: [1] IEEE Std C37.238-2011: sub-clause 5.7.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-01-14 Preview release

Discussion: This test will validate that ingress ports support both one-step and two-step modes by simulating a jump in the delay through Sync and Follow_Up messages and observing whether the offset from grandmaster is updated [1]. First, the test station will generate and send Sync messages while incrementing the value of the originTimestamp by 1 ns each message. Second, the test station will generate and send Sync messages while incrementing the value of the originTimestamp by 5 ns each message. If the device supports one-step mode it will accept Sync messages and update its offsetFromMaster with this 4 nanosecond jump.

A similar process will be followed to validate the device supports two-step mode. First, the test station will generate and send Sync and Follow_Up messages, while incrementing the values of the originTimestamp and preciseOriginTimestamp by 1 ns each message. Second, the test station will generate and send Sync and Follow_Up messages, while incrementing the values of the originTimestamp by 1 ns and the preciseOriginTimestamp by 5 ns each message. If the device supports two-step mode it will accept Sync and Follow_Up messages and update its offsetFromMaster with this 4 ns jump.

For one-step clocks the calculations for the meanPathDelay and offsetFromMaster are shown below.

$$offsetFromMaster = syncEventIngressTimestamp - originTimestamp - mPD - correctionField\ of\ sync\ message$$

For two-step clocks the calculations for the meanPathDelay and offsetFromMaster are shown below.

$$offsetFromMaster = syncEventIngressTimestamp - preciseOriginTimestamp - mPD - correctionField\ of\ sync\ message - correctionField\ of\ Follow\_Up\ message$$

Where, *mPD* is the value of the meanPathDelay measured and computed for each instance of a peer delay request-response measurement.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: Sync, One-step mode

A:1. Capture traffic received by TS1 for the duration of this test.

A:2. Generate and send 30 Sync messages from TS1, while incrementing the originTimestamp by 1 ns in each message.

A:3. After each message is sent, observe the DUT's offset from master,
   a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
   b. by means provided, as specified.

A:4. Generate and send 30 Sync messages from TS1, while incrementing the originTimestamp by 5 ns.

A:5. After each message is sent, observe the DUT's offset from master,
   a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
   b. by means provided, as specified.

A:6. Repeat steps A:1-5 for each ingress port on the DUT.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	The offset from the master is not observed.
A:5	FAIL	A jump in the offset from the master has not occurred.
A:6	PASS	All ingress ports on the device support one-step mode.

Part B: Sync and Follow_Up, Two-step mode

B:1. Capture traffic received by TS1 for the duration of this test.

B:2. Generate and send 30 Sync messages from TS1, while incrementing the originTimestamp by 1 ns in each message. Also, generate and send 30 Follow_Up messages from TS1, while incrementing the precise-OriginTimestamp by 1 ns in each message

B:3. After each message is sent, observe the DUT's offset from master,
   a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
   b. by means provided, as specified.

B:4. Generate and send 30 Sync messages from TS1, while incrementing the originTimestamp by 1 ns in each message. Also, generate and send 30 Follow_Up messages from TS1, while incrementing the precise-OriginTimestamp by 5 ns in each message.

B:5. After each message is sent, observe the DUT's offset from master,
   a. by requesting ieeeC37238currentDS.OfstFrMaster or, if SNMP is supported, or
   b. by means provided, as specified.

B:6. Repeat steps B:1-6 for each ingress port on the DUT.

Observable Results:

Part:Step	Status	Description
B:3	FAIL	The offset from the master is not observed.
B:5	FAIL	A jump in the offset from the master has not occurred.
B:6	PASS	All ingress ports on the device support two-step mode.

Possible Problems: None

Test PWR.c.9.4 – One-Step or Two-Step Mode Egress Ports

Purpose: To verify all devices' egress ports support either one-step or two-step mode.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	All	None
B	TC	One-step Clock
C	TC	Two-step Clock

References: [1] IEEE Std C37.238-2011: sub-clause 5.7.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History:	2013-05-30	Preview release

Discussion: This test will validate that egress ports support either one-step or two-step modes by sending Pdelay_Req messages from the test station and observing the response messages emitted from the DUT [1].

This test will also validate that one-step transparent clocks modify the correctionField of outgoing Sync messages while two-step transparent clocks modify the correctionField of outgoing Follow_Up messages.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: One-step or Two-step mode

A:1. Capture traffic received by TS1 for the duration of this test.
A:2. Send Pdelay_Req messages from TS1.
A:3. Wait up to 10 s for messages to be received from the DUT.

Observable Results:

Part:Step	Status	Description
A:3	FAIL	No Pdelay_Resp messages are received.
A:3	PASS	Either just Pdelay_Resp messages or both Pdelay_Resp and Pdelay_Resp_Follow_Up messages are received.

Part B: One-Step Transparent Clocks

B:1. Capture traffic received by TS1 and TS2 for the duration of this test.
B:2. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.
B:3. Respond to the Pdelay_Req message as a one step clock with a Pdelay_Resp message.
B:4. Generate and send 10, one-step, Sync messages from TS1, with the correctionField set to '0'.
B:5. After each message is sent, observe all messages forwarded from the DUT to TS2.
B:6. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.
B:7. Respond to the Pdelay_Req message as a two step clock with a Pdelay_Resp and a Pdelay_Resp_Follow_Up message.
B:8. Generate and send 10, two-step, Sync and 10 Follow_Up messages from TS1, with the correctionField set to '0'.
B:9. After each message is sent, observe all messages forwarded from the DUT to TS2.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req messages are received by the TS1.
B:5	FAIL	No Sync messages are received by the TS2.
B:5	FAIL	The correctionField in each Sync message sent by the DUT is '0'.
B:5	FAIL	The correctionField in any Follow_Up message sent by the DUT is changed.
B:9	FAIL	No Sync messages are received by the TS2.
B:9	FAIL	The correction field in each Sync message sent by the DUT is '0'.
B:9	FAIL	The correctionField in any Follow_Up message sent by the DUT is not '0'.
B:9	PASS	The DUT modifies the correctionField of all Sync messages received and does not modify the correctionField of any Follow_Up messages.

Part C: Two-Step Transparent Clocks

C:1. Capture traffic received by TS1 and TS2 for the duration of this test.
C:2. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.
C:3. Respond to the Pdelay_Req message as a one step clock with a Pdelay_Resp message.
C:4. Generate and send 10 Sync messages from TS1, with the correctionField set to '0'.
C:5. After each message is sent, observe all messages forwarded from the DUT to TS2.
C:6. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.
C:7. Respond to the Pdelay_Req message as a two step clock with a Pdelay_Resp and a Pdelay_Resp_Follow_Up message.
C:8. Generate and send 10 Sync and 10 Follow_Up messages from TS1, with the correctionFields set to '0'.
C:9. After each message is sent, observe all messages forwarded from the DUT to TS2.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	No Pdelay_Req messages are received by the TS1.
C:5	FAIL	No Sync messages are received by the TS2.
C:5	FAIL	The correctionField in each Sync message sent by the DUT is not '0'.
C:5	FAIL	No Follow_Up messages are received by the TS2.
C:5	FAIL	The correctionField in any Follow_Up message received is '0'.
C:9	FAIL	No Sync messages are received by the TS2.
C:9	FAIL	The correctionField in each Sync message sent by the DUT is not '0'.
C:9	FAIL	No Follow_Up messages are received by the TS2.
C:9	FAIL	The correctionField in any Follow_Up message received is '0'.
C:9	PASS	The DUT does not modify the correctionField of Sync messages received and generates and sends Follow_Up messages with the correctionField set to the meanPathDelay and any residence time corrections.

Possible Problems: None

Test PWR.c.9.5 – One-Step or Two-Step Flags

Purpose: To verify all devices' egress ports support either one-step or two-step mode with varying flag fields.

Device Type Prerequisites and Certification Classifier:

Part	Applies To Device Type	Prerequisite Conditions
A	TC	One-step Clock
B	TC	One-step Clock
C	TC	Two-step Clock
D	TC	Two-step Clock

References: [1] IEEE Std C37.238-2011: sub-clause 5.7.1

Resource Requirements: One test station capable of transmitting and receiving arbitrary MAC frames

Modification History: 2013-05-30 Preview release

Discussion: This test will validate the behavior of a device when it receives a Sync and Follow_Up message even with the two-step flag FALSE. When a transparent clock receives a Sync message with the two-step flag FALSE it should utilizze the time provided in the Sync message and ignore the Follow_Up message.
Similarly, this test will validate the behavior of a device when it receives a Sync and Follow_Up message with the two-step flag TRUE. When a transparent clock receives a Sync message with the two-step flag TRUE it should utilize the time provided in the Follow_Up message. The time provided in the Sync and Follow_Up messages sent by the TS will have a large difference, so the observed message forwarded by the transparent clock should be obvious.

Test Setup: Refer to Appendix A: DEFAULT TEST SETUP.

Test Procedure:

Part A: One-Step Transparent Clocks receiving Syncs with Two-Step Flag FALSE

A:1. Capture traffic received by TS1 and TS2 for the duration of this test.

A:2. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.

A:3. Respond to the Pdelay_Req message as a one step clock with a Pdelay_Resp message.

A:4. Generate and send messages from TS1 as a one step clock, however send both Sync and Follow_Up messages. Send ten of each message type. In the Sync message set the two-step flag to FALSE and the originTimestamp to 0x000000000001(seconds portion) 0x00000001 (nanoseconds portion). In the Follow_Up messages set the preciseOriginTimestamp to 0x999999999999(seconds portion) 0x99999999 (nanoseconds portion).

A:5. After each message is sent, observe all messages forwarded from the DUT to TS2. The DUT is a One-Step clock and should therefore only send Sync messages, utilizing the time provided in the Sync messages received because they had the two-step flag FALSE.

Observable Results:

Part:Step	Status	Description
A:2	FAIL	No Pdelay_Req messages are received by the TS1.
A:5	FAIL	No Sync messages are received by the TS2 from the DUT.
A:5	FAIL	The two-step flag field in each Sync message sent by the DUT is "TRUE".
A:5	FAIL	The correctionField in each Sync message sent by the DUT is '0'.
A:5	FAIL	The correctionField in any Follow_Up message forwarded by the DUT is changed.
A:5	FAIL	The originTimestamp in Sync messages forwarded by the DUT is not close to 0x000000000001(seconds portion) 0x00000001 (nanoseconds portion).
A:5	PASS	The DUT takes notice of the two-step flag value (FALSE) and forwards Sync messages with values from the Sync messages received and not from the Follow_Up messages.

Part B: One-Step Transparent Clocks receiving Syncs with Two-Step Flag TRUE

B:1. Capture traffic received by TS1 and TS2 for the duration of this test.

B:2. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.

B:3. Respond to the Pdelay_Req message as a two step clock with a Pdelay_Resp and Pdelay_Resp_Follow_Up message.

B:4. Generate and send messages from TS1 as a two step clock, sending both Sync and Follow_Up messages. Send ten of each message type. In the Sync message set the two-step flag to TRUE and the originTimestamp to 0x000000000001(seconds portion) 0x00000001 (nanoseconds portion). In the Follow_Up messages set the preciseOriginTimestamp to 0x999999999999(seconds portion) 0x99999999 (nanoseconds portion).

B:5. After each message is sent, observe all messages forwarded from the DUT to TS2. The DUT is a One-Step clock and should therefore only send Sync messages, utilizing the time provided in the Follow_Up messages received because the Sync messages received had the two-step flag TRUE.

Observable Results:

Part:Step	Status	Description
B:2	FAIL	No Pdelay_Req messages are received by the TS1.
B:5	FAIL	No Sync messages are received by the TS2 from the DUT.
B:5	FAIL	The two-step flag field in each Sync message sent by the DUT is "TRUE".
B:5	FAIL	The correctionField in each Sync message sent by the DUT is '0'.
B:5	FAIL	The correctionField in any Follow_Up message forwarded by the DUT is changed.
B:5	FAIL	The originTimestamp in Sync messages forwarded by the DUT is not close to 0x999999999999(seconds portion) 0x99999999 (nanoseconds portion).
B:5	PASS	The DUT takes notice of the two-step flag value (TRUE) and forwards Sync messages with values from the Follow_Up messages received and not from the Sync messages.

Part C: Two-Step Transparent Clocks receiving Syncs with Two-Step Flag FALSE

C:1. Capture traffic received by TS1 and TS2 for the duration of this test.

C:2. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.

C:3. Respond to the Pdelay_Req message as a one step clock with a Pdelay_Resp message.

C:4. Generate and send messages from TS1 as a one step clock, however send both Sync and Follow_Up messages. Send ten of each message type. In the Sync message set the two-step flag to FALSE and the originTimestamp to 0x000000000001(seconds portion) 0x00000001 (nanoseconds portion). In the Follow_Up messages set the preciseOriginTimestamp to 0x999999999999(seconds portion) 0x99999999 (nanoseconds portion).

C:5. After each message is sent, observe all messages forwarded from the DUT to TS2. The DUT is a Two-Step clock and should therefore send both Sync and Follow_Up messages, utilizing the time provided in the Sync messages received because they had the two-step flag FALSE.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	No Pdelay_Req messages are received by the TS1.
C:5	FAIL	No Sync messages are received by the TS2 from the DUT.
C:5	FAIL	The two-step flag field in each Sync message sent by the DUT is "FALSE".
C:5	FAIL	The correctionField in each Sync message forwarded by the DUT is changed.
C:5	FAIL	No Follow_Up messages are received by the TS2 from the DUT.
C:5	FAIL	The correctionField in any Follow_Up message sent by the DUT is '0'.
C:5	FAIL	The preciseOriginTimestamp in Follow_Up messages forwarded by the DUT is not close to 0x000000000001(seconds portion) 0x00000001 (nanoseconds portion).
C:5	PASS	The DUT takes notice of the two-step flag value (FALSE) and forwards Sync and Follow_Up messages with values from the Sync messages received and not from the Follow_Up messages.

Part D: Two-Step Transparent Clocks receiving Syncs with Two-Step Flag TRUE

D:1. Capture traffic received by TS1 and TS2 for the duration of this test.

D:2. Wait 10 s for Pdelay_Req message to be received by TS1 from the DUT.

D:3. Respond to the Pdelay_Req message as a one step clock with a Pdelay_Resp message.

D:4. Generate and send messages from TS1 as a two step clock, sending both Sync and Follow_Up messages. Send ten of each message type. In the Sync message set the two-step flag to TRUE and the originTimestamp to 0x000000000001(seconds portion) 0x00000001 (nanoseconds portion). In the Follow_Up messages set the preciseOriginTimestamp to 0x999999999999(seconds portion) 0x99999999 (nanoseconds portion).

D:5. After each message is sent, observe all messages forwarded from the DUT to TS2. The DUT is a Two-Step clock and should therefore send both Sync and Follow_Up messages, utilizing the time provided in the Follow_Up messages received because they had the two-step flag FALSE.

Observable Results:

Part:Step	Status	Description
C:2	FAIL	No Pdelay_Req messages are received by the TS1.
C:5	FAIL	No Sync messages are received by the TS2 from the DUT.
C:5	FAIL	The two-step flag field in each Sync message sent by the DUT is "FALSE".
C:5	FAIL	The correctionField in each Sync message forwarded by the DUT is changed.
C:5	FAIL	No Follow_Up messages are received by the TS2 from the DUT.
C:5	FAIL	The correctionField in any Follow_Up message sent by the DUT is '0'.
C:5	FAIL	The preciseOriginTimestamp in Follow_Up messages forwarded by the DUT is not close to 0x999999999999(seconds portion) 0x99999999 (nanoseconds portion).
C:5	PASS	The DUT takes notice of the two-step flag value (TRUE) and forwards Sync and Follow_Up messages with values from the Follow_Up messages received and not from the Sync messages.

Possible Problems: None

Appendix A: DEFAULT TEST SETUP

Except where otherwise specified, all tests will require the DUT to have the following default configuration at the beginning of each test case:

PTP Settings:
- The Power Profile is enabled by default on the ports under test
- The default values for the Power Profile are in place including (but not limited to) Priority1 and Priority2 set to 128 for GMC or 255 for slave-only; logAnnounceInterval set to 0; logSyncInterval set to 0; slave-Only set to FALSE for GMC or TRUE for slave-only; domainNumber set to 0.
- The DUT is connected to a primary reference and has reached steady state.

VLAN Settings:
- The ports connected to the test stations should not be members of any VLAN
  - If this isn't possible, then only a member of VLAN 1 (and other VID labels cannot have a VID of 1)
- The ports connected to the test stations should not be members of the untagged set for any VLAN
- Port Settings
  - Enable Ingress Filtering – Disabled or Enabled
  - Acceptable Frame Types – Accept All Frames

Spanning Tree Settings (If DUT is an IEEE 802.1 Bridge):
- RSTP or MSTP is enabled
- DUT priority should be default (0x8000)
- Hello Time must be 2 s (if configurable)
- Port Settings
  - AdminEdge – False
  - AutoEdge – False

FDB Settings (If DUT is an IEEE 802.1 Bridge):
- Ageing time – 300 s (default)

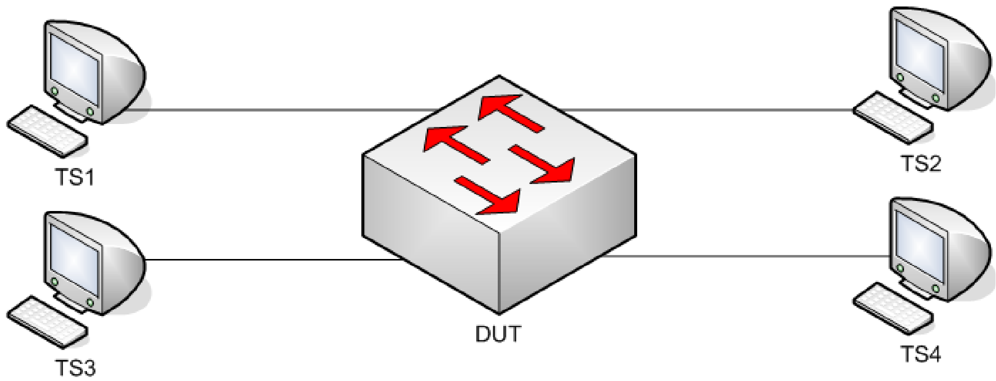
Choice of VLAN component type:
Each test case presumes that C-VLAN components are in use, not S-VLAN components. The Destination MAC address in MVRPDUs transmitted to the DUT must be selected accordingly.

VID Translation Tables:
Disabled (i.e. all VLANs map to themselves)

Dynamic Port Parameters (If DUT is an IEEE 802.1 Bridge):
- DUT is root of the spanning tree
- Port Conditions
  - Port Role = Designated
  - Port State = Forwarding
  - No Dynamic VLAN Registration Entries exist
  - No Dynamic MAC Address Registration Entries exist
    - At a minimum, no Dynamic MAC Address Registration Entries may exist for any of the MAC addresses used in test traffic

Default Test Topology

Unless otherwise stated, all tests will use the following topology. Many tests require less than 4 Test Stations. In those cases, the additional test stations may or may not be actually included in the physical topology, at the discretion of the test technician.



Test Traffic:

Some tests specify the transmission of test traffic from a test station. Unless otherwise specified, this test traffic shall be well-formed 64-byte VLAN-tagged broadcast traffic sent at a rate of 10 frames per second. For each test, each test station will be configured for a unique source address amongst the set of test stations.

Appendix B: NOTES ON TEST PROCEDURES

There are scenarios where in test procedures it is desirable to leave certain aspects of the testing procedure as general as possible. In these cases, the steps in the described test procedure may use placeholder values, or may intentionally use non-specific terminology, and the final determination of interpretation or choice of values is left to the discretion of the test technician. The following is an attempt to capture and describe all such instances used throughout the procedures.

Ports on DUT

In general, any Bridge Port on the DUT may be used as an interface with a test station. There is assumed to be no difference in behavior, with respect to the protocols involved in this test suite, between any two Bridge Ports on the DUT. Hence, actual ports used may be chosen for convenience, on a basis such as media type or link speed. Specific Bridge Ports are designated by the Test Station (TS) connected to them via the notation DUT.TS (e.g. Test Station 1 (TS1) is connected to Bridge Port DUT.TS1)

VLAN IDs

Many of the procedures call for registration of certain VIDs, or for traffic to be transmitted tagged with a certain VID. Including a specific VID in the procedures of these cases limits the generality of the test case, and may preclude test execution on certain devices which treat the VID with a special significance. Given this, placeholders are used for VIDs when appropriate.

In general, any VID may be used in these cases, with the following stipulations: Each placeholder must use a different VID (e.g. VLAN "A" and VLAN "B" may not both be VID 204). A given placeholder must remain consistent within a given test part (e.g. VLAN "A" must refer to the same VID throughout a test part).

Where there exists a constraint on the VID chosen for a given placeholder, the test case will specify the constraint.

Use of "various"

To maintain generality, some steps will specify that "various other values" (or the like) should be used in place of a given parameter. Ideally, all possible values would be tested in this case. However, limits on available time may constrain the ability of the test technician to attempt this. Given this, a subset of the set of applicable values must generally be used.

When deciding how many values should be used, it should be noted that the more values that are tested, the greater the confidence of the results obtained (although there is a diminishing return on this).

When deciding which specific values to use, it is generally recommended to choose them at pseudo-randomly yet deterministically. However, if there exist subsets of the applicable values with special significance, values from each subset should be attempted.

Inter-test health check

Where possible and time permitting, the health and operation of the DUT can be evaluated between each test, and potentially even between each test part. The purpose of such 'health checking' is simply to establish the device as operational. This may take the form of monitoring for periodic transmissions (Pdelay_Req messages, BPDU Messages, etc.) or actively probing the device depending on its capabilities (e.g.: ICMP Echo Requests). These optional inter-test health checking aids in detecting 'silent failures' caused by previous tests that may appear to pass but otherwise result in following test cases to fail.

Appendix C: Discussion Tables

Table 1: Pdelay_Req Message Fields

Fields		Octets	Offset	Values
transportSpecific	messageType	1	0	
reserved	versionPTP	1	1	
messageLength		2	2	
domainNumber		1	4	varies
Reserved		1	5	
flagField		2	6	
correctionField		8	8	0
Reserved		4	16	
sourcePortIdentity		10	20	
sequenceId		2	30	
controlField		1	32	
logMessageInterval		1	33	
originTimestamp		10	34	0
reserved		10	44	

Table 2: Pdelay_Resp Message Fields

Fields		Octets	Offset	One-Step Clock Values	Two-Step Clock Values
transportSpecific	messageType	1	0		
reserved	versionPTP	1	1		
messageLength		2	2		
domainNumber		1	4	domainNumber from Pdelay_Req	domainNumber from Pdelay_Req
Reserved		1	5		
flagField		2	6		
correctionField		8	8	correctionField from Pdelay_Req + (t_3-t_2)	0 - any fractional nanosecond portion of t_2
Reserved		4	16		
sourcePortIdentity		10	20		
sequenceId		2	30	sequenceId from Pdelay_Req	sequenceId from Pdelay_Req
controlField		1	32		
logMessageInterval		1	33		
requestReceiptTimestamp		10	34	0	Seconds and nanoseconds portion of t_2
requestingPortIdentity		10	44	sourcePortIdentity from Pdelay_Req	sourcePortIdentity from Pdelay_Req

Table 3: Pdelay_Resp_Follow_Up Message Fields

Fields		Octets	Offset	Values
transportSpecific	messageType	1	0	
reserved	versionPTP	1	1	
messageLength		2	2	
domainNumber		1	4	domainNumber from Pdelay_Req
Reserved		1	5	
flagField		2	6	
correctionField		8	8	correctionField from Pdelay_Req + any fractional nanosecond portion of t_3
Reserved		4	16	
sourcePortIdentity		10	20	
sequenceId		2	30	sequenceId from Pdelay_Req
controlField		1	32	
logMessageInterval		1	33	
responseOriginTimestamp		10	34	Seconds and nanoseconds portion of t_3
requestingPortIdentity		10	44	sourcePortIdentity from Pdelay_Req

Table 4: Action after Receipt of Pdelay_Resp Message

Number of Received Pdelay_Resp	Action
0	Retransmit a Pdelay_Req message
1	Protocol of 11.4 should be executed as specified.
Multiple	Enter FAULTY state or fault condition depending on clock type. Discard Sync and Follow_Up messages.

Table 5: Announce Message Fields

Field	Octets	Offset	Influences the BMCA	Coverage
header	34	0	–	
originTimestamp	10	34	–	
currentUtcOffset	2	44	–	Test PWR.c.7.7
reserved	1	46	–	
grandmasterPriority1	1	47	yes	
grandmasterClockQuality	4	48	yes	
grandmasterPriority2	1	52	yes	
grandmasterIdentity	8	53	yes	
stepsRemoved	2	61	yes	
timeSource	1	63	–	

This publication is available free of charge from http://dx.doi.org/10.6028/NIST.IR.8002

Table 6: IEEE C37.238 Message Fields

	Field		Octets	Offset	Coverage
Ethernet Specific	DA		6	-18	
	SA		6	-12	
	802.1Q Header		4	-6	Test PWR.c.5.6, Test PWR.c.5.7, Test PWR.c.5.8
	Ethertype		2	-2	Test PWR.c.5.3
Message Header	transportSpecific	messageType	1	0	Test PWR.c.5.2
	reserved	versionPTP	1	1	
	messageLength		2	2	
	domainNumber		1	4	
	reserved		1	5	
	flagField		2	6	Test PWR.c.5.1
	correctionField		8	8	
	reserved		4	16	
	sourcePortIdentity		10	20	Test PWR.c.3.10
	sequenceId		2	30	
	controlField		1	32	
	logMessageInterval		1	33	

Table 7: Multicast MAC Addresses

Message types	Address (hex)
All except peer delay mechanism messages	01-1B-19-00-00-00
Peer delay mechanism messages	01-80-C2-00-00-0E

Table 8: Ethernet Transport Specific Field

Enumeration	Value (hex)	Specification
DEFAULT	0	All PTP layer 2 Ethernet transmissions not covered by another enumeration value.
ETHERNET_AVB	1	This value is reserved for use in connection with the standard being developed by the IEEE 802.1 AVB Task Group as P802.1AS.
Reserved	2-F	Reserved for assignment in future versions of this standard.

Table 9: IEEE 802.1 Q Header Fields

Field	Bits	Value	Coverage
TPID	16	8100	
PCP	3	4	Test PWR.c.5.6, Test PWR.c.5.7
DEI	1	0	
VID	12	0	Test PWR.c.5.7

Table 10: timeSource

timeSource	Specification	timeSource	Specification
0x10	ATOMIC CLOCK	0x50	NTP
0x20	GPS	0x60	HAND SET
0x30	TERRESTRIAL RADIO	0x90	OTHER
0x40	PTP	0xA0	INTERNAL OSCILLATOR

Table 11: clockAccuracy enumeration

Value	Specification	Value	Specification
0x20	The time is accurate to within 25 ns	0x29	The time is accurate to within 1 ms
0x21	The time is accurate to within 100 ns	0x2A	The time is accurate to within 2.5 ms
0x22	The time is accurate to within 250 ns	0x2B	The time is accurate to within 10 ms
0x23	The time is accurate to within 1 µs	0x2C	The time is accurate to within 25 ms
0x24	The time is accurate to within 2.5 µs	0x2D	The time is accurate to within 100 ms
0x25	The time is accurate to within 10 µs	0x2E	The time is accurate to within 250 ms
0x26	The time is accurate to within 25 µs	0x2F	The time is accurate to within 1 s
0x27	The time is accurate to within 100 µs	0x30	The time is accurate to within 10 s
0x28	The time is accurate to within 250 µs	0x31	The time is accurate to >10 s
		0xFE	Unknown

Any other value for clockAccuracy is either reserved, or set by IEEE 1588 PTP profiles to not used by PTP.

Table 12: TLV Organization Extension Fields

Field	Octets	Offset	Value	Coverage
tlvType	2	0	0x0003	Test PWR.c.7.2
lengthField	2	2	0x0012	Test PWR.c.7.2
organizationId	3	4	0x1C129D	Test PWR.c.7.2, Test PWR.c.7.3
organizationSubType	3	7	0x000001	Test PWR.c.7.2, Test PWR.c.7.3
dataField -grandmasterId	2	10	0x0003-00FE	Test PWR.c.6.7, Test PWR.c.7.2
dataField -grandmasterTimeInaccuracy	4	12		Test PWR.c.7.3, Test PWR.c.8.3, Test PWR.c.8.4
dataField -networkTimeInaccuracy	4	16		Test PWR.c.8.3, Test PWR.c.8.4
dataField -reserved	2	20	0x00	Test PWR.c.7.2

Table 13: TLV Alternate Time Offset Indicator Fields

Field	Octets	Offset	Default Field Values	Coverage
tlvType	2	0	0x0009	
lengthField	2	2	15 + displayName.PTPText.lengthField + pad	
dataField - keyField	1	4	Alternate timescale	
dataField - currentOffset	4	5	Node's time – alternate time	Test PWR.c.7.7
dataField - jumpSeconds	4	9	Size of next discontinuity	60 s for Test PWR.c.7.5 and Test PWR.c.7.6
dataField - timeOfNextJump	6	13	Time next discontinuity will occur	+/- 180 s for Test PWR.c.7.5. +180 s for Test PWR.c.7.6
dataField – displayName – PTPText.lengthfield	1	19	1 + L	Test PWR.c.7.6
dataField – displayName – PTPText.textfield	L	20	Text name of Alternate timescale	
dataField - pad	0-1	20+L	1 or 0	

Table 14: sourcePortIdentity

Fields	Contents	Octets	Offset
clockIdentity	First half of MAC address	3	0
	0xFFFE	2	3
	Second half of MAC address	3	5
portNumber		2	8

Table 15: SNMP MIB Test Coverage

No.	Name	IEEE Std 1588-2008 subclause	R/W	Test Coverage
Boundary and Ordinary clock objects				
1	**ieeeC37238defaultDS**	8.2.1		
2	TwoStepFlag	8.2.1.2.1	R/W	Test PWR.c.4.2
3	ClkIdentity	8.2.1.2.2	R	Test PWR.c.4.2
4	NumberPorts	8.2.1.2.3	R	Test PWR.c.4.2
5	ClkClass	8.2.1.3.1.1	R	Test PWR.c.6.3, Test PWR.c.6.4, Test PWR.c.6.5
6	ClkAccuracy	8.2.1.3.1.2	R	Test PWR.c.6.6
7	OfsScdLogVar	8.2.1.3.1.3	R	Test PWR.c.4.2
8	Priority1	8.2.1.4.1	R/W	Test PWR.c.4.2
9	Priority2	8.2.1.4.2	R/W	Test PWR.c.4.2
10	DomainNumber	8.2.1.4.3	R/W	Test PWR.c.4.2
11	SlaveOnly	8.2.1.4.4	R/W	Test PWR.c.4.2
12	GMIdentity	GM identity to be transmitted in IEEE_C37_238 TLV.	R/W	Test PWR.c.6.7
13	NetTimeInacc	networkTimeInaccuracy to be transmitted in IEEE_C37_238 TLV. This value is set at grandmaster capable clocks.	R/W	Test PWR.c.8.3
14	EngTimeInacc	Engineered networkTimeInaccuracy in nanoseconds. This value is set at the end device to represent the worst networkTimeInaccuracy from this device to all preferred grandmasters.	R/W	Test PWR.c.4.2
15	LocTimeInacc	Maximum TimeInaccuracy that the device contributes to the networkTimeInaccuracy.	R	Test PWR.c.8.1
16	OfstFrMLimit	Offset from Master Limit to generate OfstExceedsLimit event.	R/W	Test PWR.c.4.2
17	**ieeeC37238currentDS**	8.2.2		
18	StepsRemoved	8.2.2.2	R	Test PWR.c.3.4
19	OfstFrMaster	8.2.2.3	R	Test PWR.c.9.2, Test PWR.c.9.3
20	LocTimeInacc	TimeInaccuracy contribution of the device in nanoseconds.	R	Test PWR.c.8.1
21	**ieeeC37238parentDS**	8.2.3		
22	ClkIdentity	8.2.3.2	R	Test PWR.c.3.1
23	PortNumber	8.2.3.2	R	Test PWR.c.4.3
24	Stats	8.2.3.3	R	Test PWR.c.4.3
25	ObsOfstScdLVar	8.2.3.4	R	Test PWR.c.4.3
26	ObsPhChgRate	8.2.3.5	R	Test PWR.c.4.3
27	GMClkIdentity	8.2.3.6	R	Test PWR.c.3.6
28	GMClkClass	8.2.3.7	R	Test PWR.c.3.6
29	GMClkAccuracy	8.2.3.7	R	Test PWR.c.3.6
30	GMOfstScdLVar	8.2.3.7	R	Test PWR.c.3.6

No.	Name	IEEE Std 1588-2008 subclause	R/W	Test Coverage
31	GMPriority1	8.2.3.8	R	Test PWR.c.3.6
32	GMPriority2	8.2.3.9	R	Test PWR.c.3.6
33	GMIdentity	GM identity received in IEEE_C37_238 TLV	R	Test PWR.c.3.8
34	GMTimeInacc	grandmasterTimeInaccuracy received in IEEE_C37_238 TLV, nanoseconds	R	Test PWR.c.8.3, Test PWR.c.4.1
35	NetTimeInacc	networkTimeInaccuracy received in IEEE_C37_238 TLV, nanoseconds	R	Test PWR.c.8.3
36	**ieeeC37238timePropDS**	8.2.4		
37	CurUTCOfst	8.2.4.2	R	Test PWR.c.6.2
38	CurUTCOfstVd	8.2.4.3	R	Test PWR.c.4.4
39	Leap59	8.2.4.4	R	Test PWR.c.4.4
40	Leap61	8.2.4.5	R	Test PWR.c.4.4
41	TmeTraceable	8.2.4.6	R	Test PWR.c.4.1
42	FrqTraceable	8.2.4.7	R	Test PWR.c.6.1
43	PTPTimescale	8.2.4.8	R	Test PWR.c.6.1
44	TimeSource	8.2.4.9	R	Test PWR.c.6.1
45	LocalTCurOfs	16.3.3.4	R/W	Test PWR.c.7.6
46	LocalTJumpS	16.3.3.5	R/W	Test PWR.c.7.6
47	LocalTNtJump	16.3.3.6	R/W	Test PWR.c.7.6
48	LocalTName	16.3.3.7	R/W	Test PWR.c.4.4
49	LeapEvLatest	The seconds portion of PTP time for the second prior to the latest IERS-announced leap-second event (may be past or future).	R/W	
50	UTCOfstNext	Seconds offset between TAI and UTC time-scales after LeapEvLatest (same as CurUTCOfst after LeapEvLatest time).	R/W	
51	LeapEvExpiry	The seconds portion of PTP time for the expiry of the latest IERS-announced leap-second event. If PTP time > LeapEvExpiry, devices shall set CurUTCOfstVd to False.	R/W	
52	**ieeeC37238portDS**	8.2.5		
53	ClkIdentity	8.2.5.2.1	R	Test PWR.c.4.5
54	PortNumber	8.2.5.2.1	R	Test PWR.c.4.5
55	PortState	8.2.5.3.1	R	Test PWR.c.4.5
56	MPathDly	8.2.5.3.3	R	Test PWR.c.2.8, Test PWR.c.2.9
57	LogAnnounceInt	8.2.5.4.1	R/W	Test PWR.c.4.5
58	AnnounceRctTout	8.2.5.4.2	R/W	Test PWR.c.4.5
59	LogSyncInt	8.2.5.4.3	R/W	Test PWR.c.4.5
60	DelayMech	8.2.5.4.4	R/W	Test PWR.c.4.5
61	LogMinPdlyRInt	8.2.5.4.5	R/W	Test PWR.c.4.5
62	VersionNumber	8.2.5.4.6	R	Test PWR.c.4.5
63	PortEnabled	True if port is enabled.	R/W	Test PWR.c.4.5
64	DlyAsymmetry	Path delay asymmetry.	R/W	Test PWR.c.4.5
65	ProfileId	Indicates the PTP Profile in use.	R/W	Test PWR.c.4.5
66	NetProtocol	Indicates Network Protocol in use.	R/W	Test PWR.c.4.5
67	VlanId	Port VLAN ID	R/W	Test PWR.c.4.5
68	Priority	Port Priority	R/W	Test PWR.c.4.5
Transparent clock objects				
69	**ieeeC37238TCDefaultDS**	8.3.2		

No.	Name	IEEE Std 1588-2008 subclause	R/W	Test Coverage
70	ClkIdentity	8.3.2.2.1	R	Test PWR.c.4.6
71	NumberPorts	8.3.2.2.2	R	Test PWR.c.4.6
72	DelayMech	8.3.2.3.1	R/W	Test PWR.c.4.6
73	PriDomain	8.3.2.3.2	R/W	Test PWR.c.4.6
74	Syntonize	True if syntonization is enabled.	R/W	Test PWR.c.4.6
75	CurGMaster	Comprises current grandmaster identity.	R	Test PWR.c.4.6
76	TwoStepFlag	8.2.1.2.1	R/W	Test PWR.c.4.6
77	GMIdentity	GM identity received in IEEE_C37_238 TLV.	R	Test PWR.c.4.6
78	NetProtocol	Indicates Network Protocol in use.	R/W	Test PWR.c.4.6
79	VlanId	Port VLAN ID	R/W	Test PWR.c.4.6
80	Priority	Port Priority	R/W	Test PWR.c.4.6
81	GMTimeInacc	grandmasterTimeInaccuracy received in IEEE_C37_238 TLV.	R	Test PWR.c.8.4
82	NetTimeInacc	networkTimeInaccuracy received in IEEE_C37_238 TLV.	R	Test PWR.c.8.4
83	LocTimeInacc	TimeInaccuracy contribution of the local clock in nanoseconds.	R	Test PWR.c.8.2
84	**ieeeC37238TCPortDS**	8.3.3		
85	PortNumber	8.3.3.2.1	R	Test PWR.c.4.7
86	LMinPdlyRInt	8.3.3.3.1	R/W	Test PWR.c.1.4
87	Faulty	8.3.3.3.2	R	Test PWR.c.4.7
88	MeanPDly	8.3.3.3.3	R	Test PWR.c.2.8
89	DlyAsymm	Path delay asymmetry.	R/W	Test PWR.c.4.7
Boundary and Ordinary clock objects (optional for transparent clocks)				
90	**ieeeC37238Events**			
91	ChangeOfMaster	Indicates that new grandmaster has been selected.		Test PWR.c.4.8
92	MasterStepChange	Indicates that a step change occurred in current grandmaster time.		Test PWR.c.6.8
93	FaultyState	Indicates that a clock has entered faulty state.		Test PWR.c.2.7
94	PortStateChange	Indicates that port state has changed.		Test PWR.c.4.8
95	OfstExceedsLimit	Indicates that, for a clock in a slave state, Offset from Master exceeds configurable limit.		Test PWR.c.6.8
96	OtherProfileDetect	Indicates that another PTP profile has been detected.		Test PWR.c.4.8
97	LeapSecAnnounced	Indicates that a leap second has been announced.		Test PWR.c.4.4
98	PTPServiceStarted	Indicates that PTP service has started.		Test PWR.c.4.8
99	PTPServiceStopped	Indicates that PTP service has stopped.		Test PWR.c.4.8

Appendix D: Calculations

Test PWR.c.1.1 – logAnnounceInterval, Test PWR.c.1.2 – logSyncInterval, Test PWR.c.1.4 – logMinPdelayReqInterval

In a sequence of messages from the DUT, the mean, variance and standard deviation of the intervals, I, are calculated as follows.

$$mean = \frac{1}{n}(I_1 + I_2 + I_3 + \cdots + I_n)$$

$$variance = \frac{1}{n}((I_1 - mean)^2 + (I_2 - mean)^2 + (I_3 - mean)^2 + \cdots + (I_n - mean)^2)$$

$$standard\ deviation = s = \sqrt{variance}$$

With the sample mean ($mean$) and the sample std.dev (s) computed, the next step is to compute, with 90 % confidence, the range the true mean is within as follows.

$$mean - 1.645\left(\frac{s}{\sqrt{n}}\right) < \mu < mean + 1.645\left(\frac{s}{\sqrt{n}}\right)$$

Test PWR.c.2.4 – Peer Delay Turnaround Timestamps, One-Step Clock

In a sequence of Pdelay_Resp messages from the DUT the mean and variance of the correctionField *cF* are calculated as follows.

$$mean = \frac{1}{n}(cF_1 + cF_2 + cF_3 + \cdots + cF_n)$$

$$variance = \frac{1}{n}((cF_1 - mean)^2 + (cF_2 - mean)^2 + (cF_3 - mean)^2 + \cdots + (cF_n - mean)^2)$$

Test PWR.c.2.5 – Peer Delay Message Field Values, Two-Step Clock

In a sequence of Pdelay_Resp and Pdelay_Resp_Follow_Up messages from the DUT, the mean and variance of the turnaround time *tT* are calculated as follows.

$$mean = \frac{1}{n}(tT_1 + tT_2 + tT_3 + \cdots + tT_n)$$

$$variance = \frac{1}{n}((tT_1 - mean)^2 + (tT_2 - mean)^2 + (tT_3 - mean)^2 + \cdots + (tT_n - mean)^2)$$

Test PWR.c.2.8 – Mean Path Delay

The mean and variance are calculated as follows.

$$mean = \frac{1}{n}(mPD_1 + mPD_2 + mPD_3 + \cdots + mPD_n)$$

$$variance = \frac{1}{n}((mPD_1 - mean)^2 + (mPD_2 - mean)^2 + (mPD_3 - mean)^2 + \cdots + (mPD_n - mean)^2)$$

Where *mPD* is the meanPathDelay computed for successive peer delay measurements.

Appendix E: Acronyms and Abbreviations

BC	Boundary Clock
BMCA	Best Master Clock Algorithm
BPDU	Bridge Protocol Data Unit
DA	Destination Address
DEI	Drop Eligible Indicator
DUT	Device Under Test
EUI (EUI-64, EUI-48)	Extended Unique Identifier
GMC	Grandmaster Capable
GPS	Global Positioning System
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IOL	InterOperability Lab
MIB	Management Information Base
MSTP	Multiple Spanning Tree Protocol
MVRP	Multiple VLAN Registration Protocol
MVRPDU	Multiple VLAN Registration Protocol Data Unit
N/A	Not Applicable
NIST	Nation Institute of Standards and Technology
OC	Ordinary Clock
PCP	Priority Code Point
PPS	Pulse Per Second
PrefGM	Preferred Grandmaster
PSRC	Power System Relaying Committee
PTP	Precision Time Protocol
R/W	Read/Write
RSTP	Rapid Spanning Tree Protocol
SA	Source Address
SNMP	Simple Network Management Protocol
SO	Slave Only
TAI	International Atomic Time
TBD	To Be Determined
TC	Transparent Clock
TCI	Tag Control Information
TLV	Type-Length-Value
TPID	Tag Protocol Identifier
TS (TS1)	Test Station
UNH	University of New Hampshire
UTC	Coordinated Universal Time
VID	VLAN Identifier
VLAN	Virtual Local Area Network