International Standard

ISO/IEC/IEEE 8802-1AS

Second edition
2021-11

AMENDMENT 1
2025-04

# Information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks —

## Part 1AS: Timing and synchronization for time-sensitive applications in bridged local area networks

## AMENDMENT 1: Inclusive terminology

*Technologies de l'information — Télécommunications et échange d'information entre systèmes — Réseaux locaux et métropolitains —*

*Partie 1AS: Temporisation et synchronisation pour les applications sensibles au temps des réseaux locaux pontés*

*AMENDEMENT 1: Terminologie inclusive*

Reference number
ISO/IEC/IEEE 8802-1AS:2021/
Amd.1:2025(en)

© IEEE 2024

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

**COPYRIGHT PROTECTED DOCUMENT**

© IEEE 2024

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from IEEE at the address below.

Institute of Electrical and Electronics Engineers, Inc
3 Park Avenue, New York
NY 10016-5997, USA

Email: stds.ipr@ieee.org
Website: www.ieee.org

Published in Switzerland

© IEEE 2024 – All rights reserved

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

ISO/IEC/IEEE 8802-1AS:2021/Amd 1 was prepared by the LAN/MAN of the IEEE Computer Society (as IEEE Std 802.1ASdr-2024) and drafted in accordance with its editorial rules. It was adopted, under the “fast-track procedure” defined in the Partner Standards Development Organization cooperation agreement between ISO and IEEE, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*.

A list of all parts in the ISO/IEC/IEEE 8802 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

© IEEE 2024 – All rights reserved

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.



Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

**IEEE Std 802.1ASdr™-2024**
(Amendment to IEEE Std 802.1AS™-2020
as amended by IEEE Std 802.1AS™-2020/Cor 1-2021)

## IEEE Standard for
## Local and Metropolitan Area Networks—

# Timing and Synchronization for Time-Sensitive Applications

# Amendment 1: Inclusive Terminology

Developed by the

**LAN/MAN Standards Committee**
of the
**IEEE Computer Society**

Approved 15 February 2024

**IEEE SA Standards Board**

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

**Abstract:** This amendment to IEEE Std 802.1AS™-2020 replaces terminology with alternative terminology identified by IEEE Std 1588g™-2022. All changes made in figures are only to replace terminology.

**Keywords:** amendment, best timeTransmitter, frequency offset, Grandmaster Clock, Grandmaster PTP Instance, IEEE 802.1AS™, IEEE 802.1ASdr™, phase offset, PTP End Instance, PTP Relay Instance, synchronization, syntonization, time-aware system

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2024 by The Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 6 March 2024. Printed in the United States of America.

MoCA is a registered trademark of the Multimedia over Coax Alliance.

POSIX is a registered trademark of The Institute of Electrical and Electronics Engineers, Incorporated.

IEEE and IEEE 802 are registered trademarks in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics Engineers, Incorporated.

PDF: ISBN 978-8-8557-0539-3 STD26794
Print: ISBN 978-8-8557-0540-9 STDPD26794

*IEEE prohibits discrimination, harassment and bullying.*

*For more information, visit https://www.ieee.org/about/corporate/governance/p9-26.html.*

*No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.*

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# Important Notices and Disclaimers Concerning IEEE Standards Documents

IEEE Standards documents are made available for use subject to important notices and legal disclaimers. These notices and disclaimers, or a reference to this page (https://standards.ieee.org/ipr/disclaimers.html), appear in all standards and may be found under the heading "Important Notices and Disclaimers Concerning IEEE Standards Documents."

## Notice and Disclaimer of Liability Concerning the Use of IEEE Standards Documents

IEEE Standards documents are developed within IEEE Societies and subcommittees of IEEE Standards Association (IEEE SA) Board of Governors. IEEE develops its standards through an accredited consensus development process, which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE standards are documents developed by volunteers with scientific, academic, and industry-based expertise in technical working groups. Volunteers are not necessarily members of IEEE or IEEE SA and participate without compensation from IEEE. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

IEEE makes no warranties or representations concerning its standards, and expressly disclaims all warranties, express or implied, concerning this standard, including but not limited to the warranties of merchantability, fitness for a particular purpose and non-infringement. IEEE Standards documents do not guarantee safety, security, health, or environmental protection, or guarantee against interference with or from other devices or networks. In addition, IEEE does not warrant or represent that the use of the material contained in its standards is free from patent infringement. IEEE Standards documents are supplied "AS IS" and "WITH ALL FAULTS."

Use of an IEEE standard is wholly voluntary. The existence of an IEEE standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity, nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon their own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

IN NO EVENT SHALL IEEE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: THE NEED TO PROCURE SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE PUBLICATION, USE OF, OR RELIANCE UPON ANY STANDARD, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## Translations

The IEEE consensus balloting process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE is the approved IEEE standard.

## Official statements

A statement, written or oral, that is not processed in accordance with the IEEE SA Standards Board Operations Manual shall not be considered or inferred to be the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that the presenter's views should be considered the personal views of that individual rather than the formal position of IEEE, IEEE SA, the Standards Committee, or the Working Group. Statements made by volunteers may not represent the formal position of their employer(s) or affiliation(s).

## Comments on standards

Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE or IEEE SA. However, **IEEE does not provide interpretations, consulting information, or advice pertaining to IEEE Standards documents**.

Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its Societies and subcommittees of the IEEE SA Board of Governors are not able to provide an instant response to comments, or questions except in those cases where the matter has previously been addressed. For the same reason, IEEE does not respond to interpretation requests. Any person who would like to participate in evaluating comments or in revisions to an IEEE standard is welcome to join the relevant IEEE working group. You can indicate interest in a working group using the Interests tab in the Manage Profile & Interests area of the IEEE SA myProject system.[1] An IEEE Account is needed to access the application.

Comments on standards should be submitted using the Contact Us form.[2]

## Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not constitute compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

## Data privacy

Users of IEEE Standards documents should evaluate the standards for considerations of data privacy and data ownership in the context of assessing and using the standards in compliance with applicable laws and regulations.

[1] Available at: https://development.standards.ieee.org/myproject-web/public/view.html#landing.

[2] Available at: https://standards.ieee.org/content/ieee-standards/en/about/contact/index.html.


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## Copyrights

IEEE draft and approved standards are copyrighted by IEEE under U.S. and international copyright laws. They are made available by IEEE and are adopted for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making these documents available for use and adoption by public authorities and private users, neither IEEE nor its licensors waive any rights in copyright to the documents.

## Photocopies

Subject to payment of the appropriate licensing fees, IEEE will grant users a limited, non-exclusive license to photocopy portions of any individual standard for company or organizational internal use or individual, non-commercial use only. To arrange for payment of licensing fees, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400; https://www.copyright.com/. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

## Updating of IEEE Standards documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect.

Every IEEE standard is subjected to review at least every 10 years. When a document is more than 10 years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit IEEE Xplore or contact IEEE.[3] For more information about the IEEE SA or IEEE's standards development process, visit the IEEE SA Website.

## Errata

Errata, if any, for all IEEE standards can be accessed on the IEEE SA Website.[4] Search for standard number and year of approval to access the web page of the published standard. Errata links are located under the Additional Resources Details section. Errata are also available in IEEE Xplore. Users are encouraged to periodically check for errata.

## Patents

IEEE standards are developed in compliance with the IEEE SA Patent Policy.[5]

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has

[3] Available at: https://ieeexplore.ieee.org/browse/standards/collection/ieee.

[4] Available at: https://standards.ieee.org/standard/index.html.

[5] Available at: https://standards.ieee.org/about/sasb/patcom/materials.html.



Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE SA Website at https://standards.ieee.org/about/sasb/patcom/patents.html. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

## IMPORTANT NOTICE

Technologies, application of technologies, and recommended procedures in various industries evolve over time. The IEEE standards development process allows participants to review developments in industries, technologies, and practices, and to determine what, if any, updates should be made to the IEEE standard. During this evolution, the technologies and recommendations in IEEE standards may be implemented in ways not foreseen during the standard's development. IEEE standards development activities consider research and information presented to the standards development group in developing any safety recommendations. Other information about safety practices, changes in technology or technology implementation, or impact by peripheral systems also may be pertinent to safety considerations during implementation of the standard. Implementers and users of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## Participants

At the time this standard was submitted to the IEEE SA Standards Board for approval, the IEEE 802.1 Working Group had the following membership:

**Glenn Parsons,** *Chair*
**Jessy V. Rouyer,** *Vice Chair*
**János Farkas,** *TSN Task Group Chair*
**Geoffrey Garner,** *Editor IEEE Std 802.1AS*
**Silvana Rodrigues,** *Editor P802.1ASdr*

Katsuyuki Akizuki
Konstantinos Alexandris
Venkat Arunarthi
Ralf Assmann
Huajie Bao
Rudy Belliardi
Jeremias Blendin
Christian Boiger
Paul Bottorff
Radhakrishna Canchi
Feng Chen
Abhijit Choudhury
Paul Congdon
Rodney Cummings
Josef Dorr
Hesham Elbakoury
Anna Engelmann
Thomas Enzinger
Donald Fedyk
Norman Finn
Craig Gunther
Marina Gutierrez
Stephen Haddock
Mark Hantel
Marc Holness
Daniel Hopf
Woojung Huh
Satoko Itaya
Yoshihiro Ito
Michael Karl
Stephan Kehrer
Marcel Kiessling
Gavin Lai
Yizhou Li
Joao Lopes
Lily Lyu
Christophe Mangin
Scott Mansfield
Olaf Mater
David McCall
Larry McMillan
Martin Mittelberger
Hiroki Nakano
Takumi Nomura
Donald R. Pannell
Dieter Proell
Karen Randall
Maximilian Riegel
Atsushi Sato
Frank Schewe
Michael Seaman
Maik Seewald
Ramesh Sivakolundu
Johannes Specht
Nemanja Stamenic
Marius Stanica
Guenter Steindl
Karim Traore
Max Turner
Balazs Varga
Ganesh Venkatesan
Tongtong Wang
Karl Weber
Leon Wessels
Ludwig Winkel
Jordon Woods
Takahiro Yamaura
Nader Zein


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

The following members of the individual balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Thomas Alexander
Boon Chong Ang
Butch Anton
Greg Armstrong
Stefan Aust
Denis Beaudoin
Christian Boiger
Vern Brethour
Ashley Butterworth
Paul Cardinal
Pin Chang
Baw Chng
Paul Congdon
Rodney Cummings
Patrick Diamond
Donald Dunn
Darwin Engwer
Donald Fedyk
Andrew Fieldsend
Geoffrey Garner
Marek Hajduczenia
Marco Hernandez
Werner Hoelzl
C. Huntley
Raj Jain
Pranav Jha
Lokesh Kabra
Piotr Karocki
Stephan Kehrer
Stuart Kerry
Yongbum Kim
Mallory Knodel
Jeff Koftinoff
Gavin Lai
Hyeong Ho Lee
Greg Luri
Jingfei Lv
Christophe Mangin
Scott Mansfield
Sven Meier
David Michel
Ronald Murias
William Rogelio Marchand Nino
Mfanasibili Nkonyane
Satoshi Obara
Bansi Patel
Arumugam Paventhan
Cam Posani
Venkatesha Prasad
Dieter Proell
Alon Regev
Annette Reilly
Denis Reilly
Maximilian Riegel
Markus Rindchen
Silvana Rodrigues
Benjamin Rolfe
Jessy V. Rouyer
Mark Sturza
Caroline Treuthardt
Richard Tse
Max Turner
Dmitri Varsanofiev
Ganesh Venkatesan
John Vergis
James Weaver
Stephen Webb
Scott Willy
Ludwig Winkel
Andreas Wolf
Oren Yuen
Janusz Zalewski

When the IEEE SA Standards Board approved this standard on 15 February 2024, it had the following membership:

**David J. Law,** *Chair*
**Vacant Position,** *Vice Chair*
**Gary Hoffman,** *Past Chair*
**Alpesh Shah**, *Secretary*

Sara R. Biyabani
Ted Burse
Stephen Dukes
Doug Edwards
J. Travis Griffith
Guido R. Hiertz
Ronald W. Hotchkiss
Hao Hu
Yousef Kimiagar
Joseph L. Koepfinger*
Howard Li
Xiaohui Liu
John Haiying Lu
Kevin W. Lu
Hiroshi Mano
Paul Nikolich
Robby Robson
Jon W. Rosdahl
Mark Siira
Lei Wang
F. Keith Waters
Sha Wei
Philip B. Winston
Don Wright

*Member Emeritus


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# Introduction

This introduction is not part of IEEE Std 802.1ASdr-2024, IEEE Standard for Local and Metropolitan Area Networks—Timing and Synchronization for Time-Sensitive Applications—Amendment 1: Inclusive Terminology.

The first edition of IEEE Std 802.1AS was published in 2011. A first corrigendum, IEEE Std 802.1AS-2011/Cor1-2013, provided technical and editorial corrections. A second corrigendum, IEEE Std 802.1AS-2011/Cor2-2015 provided additional technical and editorial corrections.

The second edition, IEEE Std 802.1AS-2020, added support for multiple gPTP domains, Common Mean Link Delay Service, external port configuration, and Fine Timing Measurement for IEEE 802.11 transport. Backward compatibility with IEEE Std 802.1AS-2011 was maintained. A corrigendum, IEEE Std 802.1AS-2020/Cor1-2021, provides technical and editorial corrections.

This amendment to IEEE Std 802.1AS-2020 replaces terminology with alternative terminology identified by IEEE Std 1588g-2022, IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems—Amendment 2: Master-Slave Optional Alternative Terminology. All changes made in figures are only to replace terminology.


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# Contents

4. Acronyms and abbreviations ..... 16

5. Conformance ..... 17

5.4 PTP Instance requirements and options ..... 17
5.6 MAC-specific timing and synchronization methods for IEEE Std 802.11-2016 ..... 18

7. Time-synchronization model for a packet network ..... 19

7.2 Architecture of a time-aware network ..... 19
7.3 Time synchronization ..... 20
7.4 PTP Instance architecture ..... 21
7.5 Differences between gPTP (IEEE Std 802.1AS) and PTP (IEEE Std 1588-2019) ..... 22

8. IEEE 802.1AS concepts and terminology ..... 23

8.4 Messages ..... 23
8.5 Ports ..... 23
8.6 PTP Instance characterization ..... 23

9. Application interfaces ..... 26

9.1 Overview of the interfaces ..... 26
9.2 ClockSourceTime interface ..... 26
9.3 ClockTargetEventCapture interface ..... 26
9.4 ClockTargetTriggerGenerate interface ..... 27
9.5 ClockTargetClockGenerator interface ..... 29
9.6 ClockTargetPhaseDiscontinuity interface ..... 30

10. Media-independent layer specification ..... 31

10.1 Overview ..... 31
10.2 Time-synchronization state machines ..... 33
10.3 Best ~~master~~timeTransmitter clock selection, external port configuration, and announce interval setting state machines ..... 54
10.5 Message attributes ..... 82
10.6 Message formats ..... 82
10.7 Protocol timing characterization ..... 84

11. Media-dependent layer specification for full-duplex point-to-point links ..... 86

11.1 Overview ..... 86
11.2 State machines for MD entity specific to full-duplex point-to-point links ..... 87
11.3 Message attributes ..... 88
11.4 Message formats ..... 88

12. Media-dependent layer specification for IEEE 802.11 links ..... 90

12.1 Overview ..... 90
12.4 Determination of asCapable ..... 95
12.5 State machines ..... 95
12.6 FTM parameters ..... 106
12.7 Format of VendorSpecific information element ..... 107
12.8 Synchronization message interval ..... 108


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

13. Media-dependent layer specification for interface to IEEE 802.3 Ethernet passive optical network link ..... 109

13.1 Overview ..... 109
13.3 Message format ..... 113
13.5 Layering for IEEE 802.3 EPON links ..... 114
13.6 Service interface definitions ..... 114
13.7 MD entity global variables ..... 115
13.8 State machines ..... 115

14. Timing and synchronization management ..... 117

14.1 General ..... 117
14.2 Default Parameter Data Set (defaultDS) ..... 118
14.3 Current Parameter Data Set (currentDS) ..... 119
14.4 Parent Parameter Data Set (parentDS) ..... 121
14.7 Acceptable ~~Master~~TimeTransmitter Table Parameter Data Set (acceptable~~Master~~TimeTransmitterTableDS) ..... 121
14.8 Port Parameter Data Set (portDS) ..... 122
14.10 Port Parameter Statistics Data Set (portStatisticsDS) ..... 123
14.11 Acceptable ~~Master~~TimeTransmitter Port Parameter Data Set (acceptable~~Master~~TimeTransmitterPortDS) ..... 123
14.16 Common Mean Link Delay Service Link Port Parameter Data Set (cmldsLinkPortDS) ..... 124

15. Managed object definitions ..... 125

15.2 Structure of the MIB ..... 125
15.3 Relationship to MIB in IEEE Std 802.1AS-2011 ..... 132
15.4 Security considerations ..... 133
15.5 Textual conventions defined in this MIB ..... 133
15.6 IEEE 802.1AS MIB module, ..... 134

16. Media-dependent layer specification for CSN ..... 205

16.1 Overview ..... 205
16.5 Synchronization messages ..... 205
16.6 Specific CSN requirements ..... 205

Annex A (normative) Protocol Implementation Conformance Statement (PICS) proforma ..... 206

A.5 Major capabilities ..... 206
A.7 Minimal time-aware system ..... 207
A.9 Best ~~master~~timeTransmitter clock ..... 209
A.10 Grandmaster-capable PTP Instance ..... 211
A.11 Media-independent ~~master~~timeTransmitter ..... 212
A.13 Media-dependent, full-duplex point-to-point link ..... 213
A.14 Media-dependent IEEE 802.11 link ..... 216
A.15 Media-dependent IEEE 802.3 EPON link ..... 217
A.16 Media-dependent CSN link ..... 217

Annex F (informative) PTP profile included in this standard ..... 218

F.4 PTP options ..... 218

Annex G (informative) The asymmetry compensation measurement procedure based on line-swapping ..... 219

G.2 Pre-conditions for measurement ..... 219


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

G.3 Measurement procedure ..... 219


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# List of figures

Figure 7-8—PTP Instance model ..... 21
Figure 9-1—Application interfaces ..... 26
Figure 10-1—Model for media-independent layer of PTP Instance ..... 32
Figure 10-2—Time-synchronization state machines—overview and interrelationships..... 34
Figure 10-3—SiteSyncSync state machine..... 44
Figure 10-5—~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine ..... 47
Figure 10-6—~~ClockMasterSyncOffset~~ClockTimeTransmitterSyncOffset state machine..... 49
Figure 10-7—~~ClockMasterSyncReceive~~ClockTimeTransmitterSyncReceive state machine..... 51
Figure 10-8—PortSyncSyncSend state machine ..... 52
Figure 10-9—~~ClockSlave~~ClockTimeReceiverSync state machine ..... 54
Figure 10-10—Example ~~master~~timeTransmitter/~~slave~~timeReceiver hierarchy of PTP Instances ..... 56
Figure 10-11—Best ~~master~~timeTransmitter clock selection state machines—overview and interrelationships..... 63
Figure 10-12—External port configuration state machines—overview and interrelationships ..... 65
Figure 10-14—PortAnnounceInformation state machine..... 75
Figure 10-18—PortAnnounceTransmit state machine ..... 81
Figure 11-2—Transport of time-synchronization information..... 86
Figure 11-3—Model for a PTP Instance of a time-aware system with full-duplex point-to-point links ..... 87
Figure 12-1—Timing measurement procedure for IEEE 802.11 links ..... 91
Figure 12-2—Fine Timing Measurement procedure for IEEE 802.11 links..... 91
Figure 12-4—Media-dependent and lower entities in stations with IEEE 802.11 links ..... 94
Figure 12-5—~~Master~~TimeTransmitter state machine A
(a) For TM, receives information from the PortSync entity and sends to ~~slave~~timeReceiver, and
(b) for FTM, receives and stores information from the PortSync entity ..... 97
Figure 12-6—~~Master~~TimeTransmitter state machine B
(a) For TM, not invoked and
(b) for FTM, receives initial FTM request from ~~slave~~timeReceiver and sends information received from upstream to ~~slave~~timeReceiver in successive FTM frames..... 98
Figure 12-7—~~Slave~~TimeReceiver state machine ..... 103
Figure 13-1—IEEE 802.3 EPON time-synchronization interfaces ..... 112
Figure 13-2—IEEE 802.3 EPON interface model..... 114
Figure G-1—Asymmetry compensation measurement procedure ..... 219


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## List of tables

Table 10-1—Summary of scope of global variables used by time synchronization state machines (see 10.2.4 and 10.2.5) ..... 40
Table 10-2—PTP Port state definitions ..... 55
Table 10-3—Summary of scope of global variables used by best ~~master~~timeTransmitter clock selection, external port configuration, and announce interval setting state machines (see 10.3.9 and 10.3.10) ..... 66
Table 10-9—Values of flag bits ..... 83
Table 12-2—FTM parameters relevant to time-synchronization transport ..... 107
Table 14-2—currentDS table ..... 120
Table 14-6—acceptable~~Master~~TimeTransmitterTableDS table ..... 122
Table 14-7—portState enumeration ..... 122
Table 14-13—acceptable~~Master~~TimeTransmitterPortDS table ..... 124
Table 15-1—IEEE8021-AS-V2V3 MIB structure and object cross reference ..... 126


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# IEEE Standard for Local and Metropolitan Area Networks—

# Timing and Synchronization for Time-Sensitive Applications

# Amendment 1: Inclusive Terminology

(This amendment is based on IEEE Std 802.1AS™-2020 as amended by IEEE Std 802.1AS™-2020/Cor 1-2021.)

NOTE—The editing instructions contained in this amendment define how to merge the material contained therein into the existing base standard and its amendments to form the comprehensive standard.

The editing instructions are shown in ***bold italic***. Four editing instructions are used: change, delete, insert, and replace. ***Change*** is used to make corrections in existing text or tables. The editing instruction specifies the location of the change and describes what is being changed by using ~~strikethrough~~ (to remove old material) and underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material without disturbing the existing material. Deletions and insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. ***Replace*** is used to make changes in figures or equations by removing the existing figure or equation and replacing it with a new one. Editing instructions, change markings, and this NOTE will not be carried over into future editions because the changes will be incorporated into the base standard.[6]

[6] Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard.


Copyright © 2024 IEEE. All rights reserved.
Documents
Document Sharing Hub
Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## 4. Acronyms and abbreviations

***Change the entries for "BMC" and "BMCA" in Clause 4 as follows:***

~~BMC~~BTC best ~~master~~timeTransmitter clock

~~BMCA~~BTCA best ~~master~~timeTransmitter clock algorithm


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# 5. Conformance

## 5.4 PTP Instance requirements and options

### 5.4.1 Summary of requirements

***Change items e) and f) in the lettered list in 5.1.4 (as amended by IEEE Std 802.1AS-2020/Cor 1-2021) as follows:***

e) Implement the Clock~~Slave~~TimeReceiverSync state machine (10.2.13).

f) Support the following best ~~master~~timeTransmitter clock algorithm (~~BMCA~~BTCA) requirements:

1) Implement the ~~BMCA~~BTCA (10.3.1.1, 10.3.1.2, 10.3.2, 10.3.3, 10.3.4, 10.3.5, 10.3.6, 10.3.8, and 10.3.10).
2) For domain 0, implement specifications for externalPortConfigurationEnabled value of FALSE (10.3.1).
3) Implement the PortAnnounceReceive state machine (10.3.11).
4) Implement the PortAnnounceInformation state machine (10.3.12).
5) Implement the PortStateSelection state machine (10.3.13).
6) Have the ~~BMCA~~BTCA as the default mode of operation, with externalPortConfiguration FALSE, on domain 0.
7) Implement at least one of the possibilities for externalPortConfigurationEnabled (i.e., FALSE, meaning the ~~BMCA~~BTCA is used, and TRUE, meaning external port configuration is used) on domains other than domain 0.

### 5.4.2 PTP Instance options

***Change items b), c), and l) in the lettered list in 5.4.2 as follows:***

b) Support the following media-independent ~~master~~timeTransmitter capability on at least one PTP Port:

1) Implement the PortSyncSyncSend state machine (10.2.12).
2) Implement the PortAnnounceTransmit state machine (10.3.16).
3) Implement the AnnounceIntervalSetting state machine (10.3.17).
4) For transmit of the Announce message, support the message requirements as specified in 10.5, 10.6, and 10.7.

c) Support the following for Grandmaster PTP Instance capability:

1) Support the media-independent ~~master~~timeTransmitter capability specified in item b) of 5.4.2.
2) Support the requirements for a grandmaster-capable PTP Instance (10.1.3).
3) Implement the ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine (10.2.9).
4) Implement the ~~ClockMasterSyncOffset~~ClockTimeTransmitterSyncOffset state machine (10.2.10).
5) Implement the ~~ClockMasterReceive~~ClockTimeTransmitterReceive state machine (10.2.11).

l) Implement both ~~BMCA~~BTCA and external port configuration on domains other than domain 0; if both possibilities are implemented on domains other than domain 0, the default value of externalPortConfigurationEnabled shall be FALSE.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

### 5.4.3 PTP Relay Instance requirements

***Change item c) in the lettered list in 5.4.3 as follows:***

c) Support the media-independent ~~master~~timeTransmitter capability specified in item b) of 5.4.2.

### 5.6 MAC-specific timing and synchronization methods for IEEE Std 802.11-2016

***Change item b) in the lettered list in 5.6 as follows:***

b) Support at least one of
   1) the media-dependent ~~master~~timeTransmitter state machines (12.5.1), or
   2) the media-dependent ~~slave~~timeReceiver state machine (12.5.2).


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# 7. Time-synchronization model for a packet network

## 7.2 Architecture of a time-aware network

### 7.2.2 Time-aware network consisting of a single gPTP domain

***Change the second paragraph and related footnote in 7.2.2 as follows:***

Any PTP Instance with clock sourcing capabilities can be a potential Grandmaster PTP Instance, and a selection method (the *best ~~master~~timeTransmitter clock algorithm*, or ~~BMCA~~BTCA) ensures that all of the PTP Instances in a gPTP domain use the same Grandmaster PTP Instance.[7] The ~~BMCA~~BTCA is largely identical to that used in IEEE Std 1588-2019, but somewhat simplified. In Figure 7-1 the ~~BMCA~~BTCA process has resulted in the Grandmaster PTP Instance being on the network backbone. If, however, the access network fails, the systems on a local network automatically switch over to one of the potential Grandmaster PTP Instances on the local network that is as least as "good" as any other. For example, in Figure 7-2, the access network link has failed, and a potential Grandmaster PTP Instance that has a GNSS reference source has become the active Grandmaster PTP Instance. As a result, now two gPTP domains exist where there used to be one. Finally, note that when a time-aware system supports more than one domain, one of the domains supported must be domain 0 for backward compatibility with the 2011 edition of this standard, though domain 0 is not necessarily active in a time-aware system.

### 7.2.3 Time-aware network consisting of multiple gPTP domains

***Change the second paragraph in 7.2.3 as follows:***

As in the single-domain case, any of the network technologies of 7.2.1 can be used. The Grandmaster PTP Instance of each domain is selected by the ~~BMCA~~BTCA; in this case, a separate, independent instance of the ~~BMCA~~BTCA is invoked in each domain.

### 7.2.4 Time-aware networks with redundant Grandmaster PTP Instances and/or redundant paths

***Change the title and text of 7.2.4.2 as follows:***

#### 7.2.4.2 Redundancy specified in this standard (~~BMCA~~BTCA)

This standard provides a basic level of redundancy as follows:

— A detection component that triggers when the current Grandmaster PTP Instance stops working (i.e., loss of Sync messages and Announce messages for a period of time) or if the link to the Grandmaster PTP Instance goes down (i.e., immediate loss of Sync messages and Announce messages).
— A correction component that triggers the ~~Best Master Clock Algorithm~~ best timeTransmitter clock algorithm (~~BMCA~~BTCA) and the sending of Announce messages so that a new Grandmaster PTP Instance can be elected.
— An action component, where the winning Grandmaster PTP Instance starts sending Announce messages and Sync messages and all the PTP Instances listen to this new Grandmaster PTP Instance.

[7] There are, however, short periods during network reconfiguration when more than one Grandmaster PTP Instance might be active while the ~~BMCA~~BTCA process is taking place.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## 7.3 Time synchronization

***Change the title and text of 7.3.4 as follows:***

### 7.3.4 Grandmaster PTP Instance (best ~~master~~timeTransmitter) selection and network establishment

All PTP Instances participate in best ~~master~~timeTransmitter selection so that the IEEE 802.1AS protocol can determine the synchronization spanning tree. This synchronization spanning tree can be different from the forwarding spanning tree determined by IEEE 802.1Q™ Rapid Spanning Tree Protocol (RSTP) since the spanning tree determined by RSTP can be suboptimal or even inadequate for synchronization or can be for a different topology of nodes from the synchronization spanning tree.

gPTP requires that all systems in the gPTP domain be time-aware systems, i.e., the protocol does not transfer timing over systems that are not time-aware (e.g., those that meet the requirements of IEEE Std 802.1Q-2018, but do NOT meet the requirements of the present standard). A time-aware system uses the peer-to-peer delay mechanism on each PTP Port to determine if a non-time-aware system is at the other end of the link or between itself and the Pdelay responder. If, on sending Pdelay_Req,

a) No response is received,
b) Multiple responses are received, or
c) The measured propagation delay exceeds a specified threshold, then

the protocol concludes that a non-time-aware system or end-to-end Transparent Clock (TC) (see IEEE Std 1588-2019) is present. In this case, the link attached to the PTP Port is deemed not capable of running gPTP, and the ~~BMCA~~BTCA ignores it. However, the PTP Port continues to attempt the measurement of propagation delay using the peer-to-peer delay mechanism (for full-duplex IEEE 802.3 links), multipoint control protocol (MPCP) messages (for EPON), or IEEE 802.11 messages (for IEEE 802.11 links), and periodically checks whether the link is or is not capable of running the IEEE 802.1AS protocol.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## 7.4 PTP Instance architecture

***Replace Figure 7-8 with the following figure:***

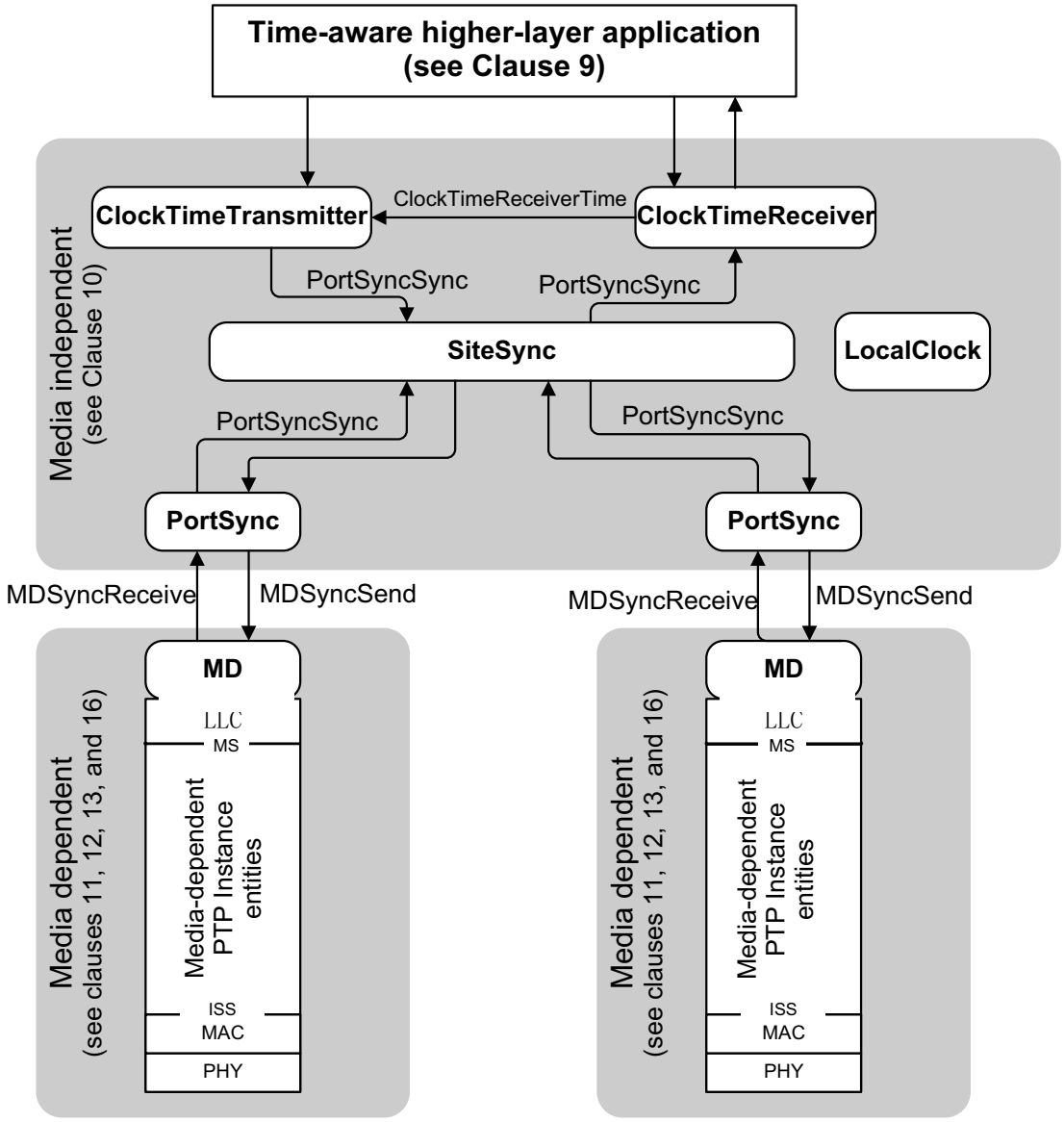



**Figure 7-8—PTP Instance model**

***Change item b) in the lettered list in 7.4 as follows:***

b) A single media-independent part that consists of ~~ClockMaster~~ClockTimeTransmitter, ~~ClockSlave~~ClockTimeReceiver, and SiteSync logical entities, one or more PortSync entities, and a LocalClock entity. The ~~BMCA~~BTCA and forwarding of time information between logical ports and the ~~ClockSlave~~ClockTimeReceiver and ~~ClockMaster~~ClockTimeTransmitter is done by the SiteSync entity, while the computation of PTP Port-specific delays needed for time-synchronization correction is done by the PortSync entities.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

## 7.5 Differences between gPTP (IEEE Std 802.1AS) and PTP (IEEE Std 1588-2019)

***Change item c) in the lettered list in 7.5 as follows:***

c) In gPTP there are only two types of PTP Instances: PTP End Instances and PTP Relay Instances, while IEEE Std 1588-2019 has Ordinary Clocks, Boundary Clocks, end-to-end Transparent Clocks, and P2P Transparent Clocks. A PTP End Instance corresponds to an IEEE 1588 Ordinary Clock, and a PTP Relay Instance is a type of IEEE 1588 Boundary Clock where its operation is very tightly defined, so much so that a PTP Relay Instance with Ethernet ports can be shown to be mathematically equivalent to a P2P Transparent Clock in terms of how synchronization is performed, as shown in 11.1.3. In addition, a PTP Relay Instance can operate in a mode (i.e., the mode where the variable syncLocked is TRUE; see 10.2.5.15) where the PTP Relay Instance is equivalent to a P2P Transparent Clock in terms of when time-synchronization messages are sent. A time-aware system measures link delay and residence time and communicates these in a correction field. In summary, a PTP Relay Instance conforms to the specifications for a Boundary Clock in IEEE Std 1588-2019, but a PTP Relay Instance does not conform to the complete specifications for a P2P Transparent Clock in IEEE Std 1588-2019 because:

   1) When syncLocked is FALSE, the PTP Relay Instance sends Sync according to the specifications for a Boundary Clock, and
   2) The PTP Relay Instance invokes the ~~BMCA~~BTCA and has PTP Port states.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# 8. IEEE 802.1AS concepts and terminology

## 8.4 Messages

### 8.4.3 Generation of event message timestamps

***Change NOTE 2 in 8.4.3 as follows:***

NOTE 2—In general, the timestamps can be generated at a timestamp measurement plane that is removed from the reference plane. Furthermore, the timestamp measurement plane, and therefore the time offset of this plane from the reference plane, is likely to be different for inbound and outbound event messages. To meet the requirement of this clause, the generated timestamps should be corrected for these offsets. Figure 8-2 illustrates these offsets. Based on this model the appropriate corrections are as follows:

egressTimestamp = egressMeasuredTimestamp + egressLatency

ingressTimestamp = ingressMeasuredTimestamp – ingressLatency

where the timestamps relative to the reference plane, egressTimestamp and ingressTimestamp, are computed from the timestamps relative to the timestamp measurement plane, egressMeasuredTimestamp and ingressMeasuredTimestamp, respectively, using their respective latencies, egressLatency and ingressLatency. Failure to make these corrections results in a time offset between the ~~slave~~timeReceiver and ~~master~~timeTransmitter clocks.

## 8.5 Ports

### 8.5.2 Port identity

### 8.5.2.3 Port number

***Change the second paragraph of 8.5.2.3 as follows:***

The portNumber value 0 is assigned to the interface between the ~~ClockMaster~~ClockTimeTransmitter and ClockSource entities (see 10.1 and Figure 10-1). The value 0xFFFF is reserved.

## 8.6 PTP Instance characterization

### 8.6.1 PTP Instance type

***Change the NOTE in 8.6.1 as follows:***

NOTE—Attributes c) through i) can be considered to be associated with the ~~ClockMaster~~ClockTimeTransmitter entity of the PTP Instance.

### 8.6.2 PTP Instance attributes

### 8.6.2.1 priority1

***Change the first paragraph of 8.6.2.1 as follows:***

priority1 is used in the execution of the ~~BMCA~~BTCA (see 10.3). The value of priority1 is an integer selected from the range 0 through 255. The ordering of priority1 in the operation of the ~~BMCA~~BTCA (see 10.3.4 and 10.3.5) is specified as follows. A ~~ClockMaster~~ClockTimeTransmitter A shall be deemed better than a ~~ClockMaster~~ClockTimeTransmitter B if the value of priority1 of A is numerically less than that of B.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

***Change NOTE 2 and NOTE 3 of 8.6.2.1 as follows:***

NOTE 2—The ~~BMCA~~BTCA (see 10.3) considers priority1 before other attributes; the priority1 attribute can therefore be used to force a desired ordering of PTP Instances for best ~~master~~timeTransmitter selection.

NOTE 3—The settings for priority1 in Table 8-1 guarantee that a PTP Instance that is grandmaster-capable is always preferred by the ~~BMCA~~BTCA over a PTP Instance that is not grandmaster-capable.

### 8.6.2.2 clockClass

***Change the first paragraph in 8.6.2.2 as follows:***

The clockClass attribute denotes the traceability of the synchronized time distributed by a ~~ClockMaster~~ClockTimeTransmitter when it is the Grandmaster PTP Instance.

***Change the paragraph after list item b) in 8.6.2.2 as follows:***

The ordering of clockClass in the operation of the best ~~master~~timeTransmitter clock algorithm (see 10.3.4 and 10.3.5) is specified as follows. When comparing clockClass values, PTP Instance A shall be deemed better than PTP Instance B if the value of the clockClass of A is lower than that of B.

### 8.6.2.3 clockAccuracy

***Change the first paragraph in 8.6.2.3 as follows:***

The clockAccuracy attribute indicates the expected time accuracy of a ~~ClockMaster~~ClockTimeTransmitter.

***Change the paragraph after list item b) in 8.6.2.3 as follows:***

The ordering of clockAccuracy in the operation of the best ~~master~~timeTransmitter clock algorithm (see 10.3.4 and 10.3.5) is specified as follows. When comparing clockAccuracy values, PTP Instance A shall be deemed better than PTP Instance B if the value of the clockAccuracy of A is lower than that of B.

### 8.6.2.4 offsetScaledLogVariance

***Change the first paragraph in 8.6.2.4 as follows:***

The offsetScaledLogVariance is a scaled, offset representation of an estimate of the PTP variance. The PTP variance characterizes the precision and frequency stability of the ~~ClockMaster~~ClockTimeTransmitter. The PTP variance is the square of PTP Deviation (PTPDEV) (see B.1.3.2).

***Change the paragraph after list item b) in 8.6.2.4 as follows:***

The ordering of offsetScaledLogVariance in the operation of the best ~~master~~timeTransmitter clock algorithm (see 10.3.4 and 10.3.5) is specified as follows. When comparing offsetScaledLogVariance values, PTP Instance A shall be deemed better than PTP Instance B if the value of the offsetScaledLogVariance of A is lower than that of B.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

### 8.6.2.5 priority2

***Change the first paragraph in 8.6.2.5 as follows:***

priority2 is used in the execution of the ~~BMCA~~BTCA (see 10.3). The value of priority2 shall be an integer selected from the range 0 through 255. The ordering of priority2 in the operation of the ~~BMCA~~BTCA is the same as the ordering of priority1 (see 8.6.2.1).

***Change the NOTE in 8.6.2.5 as follows:***

NOTE—IEEE 802.1AS performance is improved when the number of hops between the Grandmaster PTP Instance and a ~~slave~~timeReceiver PTP End Instance is reduced. When ~~BMCA~~BTCA attributes are equal in a network, the preceding recommendations for priority2 select a PTP Relay Instance in order to reduce the number of hops (rather than use clockIdentity alone).

### 8.6.2.7 timeSource

***Change the first paragraph in 8.6.2.7 as follows:***

The timeSource is an information only attribute indicating the type of source of time used by a ~~ClockMaster~~ClockTimeTransmitter. The value is not used in the selection of the Grandmaster PTP Instance. The data type of timeSource shall be TimeSource, which is an Enumeration8. The values of TimeSource are specified in Table 8-2. These represent categories. For example, the global positioning system (GPS) entry includes not only the GPS system of the U.S. Department of Defense but the European Galileo system and other present and future GNSSs.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# 9. Application interfaces

## 9.1 Overview of the interfaces

***Change the second paragraph in 9.1 as follows:***

The following subclauses define one application interface between the ClockSource entity and ~~ClockMaster~~ClockTimeTransmitter entity (see 10.1.2) and four application interfaces between the ClockTarget entity and ~~ClockSlave~~ClockTimeReceiver entity (see 10.1.2). The ClockSource is an entity that can be used as an external timing source for the gPTP domain. The ClockSource entity either contains or has access to a clock (see 3.3). The ClockTarget entity represents any application that uses information provided by the ~~ClockSlave~~ClockTimeReceiver entity via any of the application interfaces.

***Replace Figure 9-1 with the following figure:***

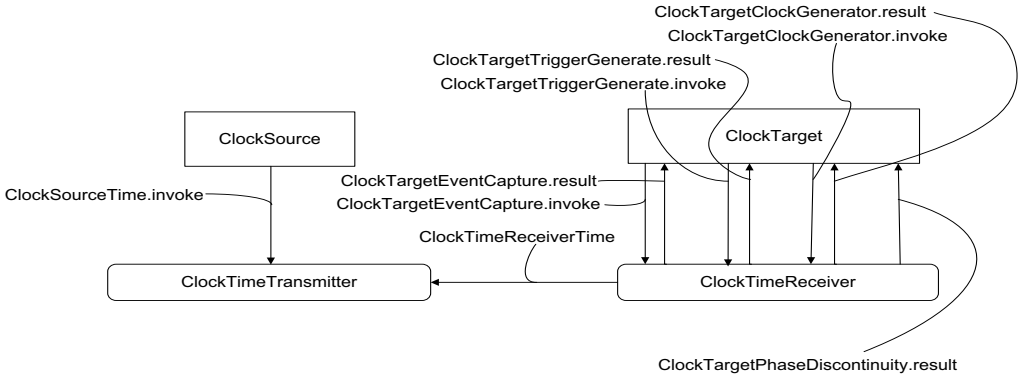



**Figure 9-1—Application interfaces**

## 9.2 ClockSourceTime interface

### 9.2.1 General

***Change 9.2.1 as follows:***

This interface is used by the ClockSource entity to provide time to the ~~ClockMaster~~ClockTimeTransmitter entity of a PTP Instance. The ClockSource entity invokes the ClockSourceTime.invoke function. The function provides the time, relative to the ClockSource, at which the function was invoked.

## 9.3 ClockTargetEventCapture interface

### 9.3.1 General

***Change 9.3.1 as follows:***

This interface is used by the ClockTarget entity to request the synchronized time of an event that it signals to the ~~ClockSlave~~ClockTimeReceiver entity of a PTP Instance. The ClockTarget entity invokes the ClockTargetEventCapture.invoke function to signal an event to the ~~ClockSlave~~ClockTimeReceiver entity. The ~~ClockSlave~~ClockTimeReceiver entity invokes the ClockTargetEventCapture.result function to return


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

the time of the event relative to the current Grandmaster Clock or, if no PTP Instance is grandmaster-capable, the LocalClock. The ClockTargetEventCapture.result function also returns gmPresent, to indicate to the ClockTarget whether a Grandmaster PTP Instance is present.

### 9.3.2 ClockTargetEventCapture.invoke parameters

#### 9.3.2.1 domainNumber (UInteger8)

***Change 9.3.2.1 as follows:***

This parameter is the domain number of the ~~ClockSlave~~ClockTimeReceiver entity that is requested to provide the synchronized time of the signaled event.

### 9.3.3 ClockTargetEventCapture.result parameters

***Change 9.3.3 as follows:***

ClockTargetEventCapture.result {
    domainNumber,
    ~~slave~~timeReceiverTimeCallback,
    gmPresent
}

#### 9.3.3.1 domainNumber (UInteger8)

***Change 9.3.3.1 as follows:***

This parameter is the domain number of the ~~ClockSlave~~ClockTimeReceiver entity that is providing the synchronized time of the signaled event.

***Change the title and text of 9.3.3.2 as follows:***

#### 9.3.3.2 ~~slave~~timeReceiverTimeCallback (ExtendedTimestamp)

The value of ~~slave~~timeReceiverTimeCallback is the time, relative to the Grandmaster Clock, that the corresponding ClockTargetEventCapture.invoke function is invoked.

NOTE—The invocation of the ClockTargetEventCapture.invoke function and the detection of this invocation by the ~~ClockSlave~~ClockTimeReceiver entity are simultaneous in this abstract interface.

## 9.4 ClockTargetTriggerGenerate interface

### 9.4.1 General

***Change 9.4.1 as follows:***

This interface is used by the ClockTarget entity to request that the ~~ClockSlave~~ClockTimeReceiver entity send a result at a specified time relative to the Grandmaster Clock. The ClockTarget entity invokes the ClockTargetTriggerGenerate.invoke function to indicate the synchronized time of the event. The ~~ClockSlave~~ClockTimeReceiver entity invokes the ClockTargetTriggerGenerate.result function to either signal the event at the requested synchronized time or indicate an error condition.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

### 9.4.2 ClockTargetTriggerGenerate.invoke parameters

***Change 9.4.2 as follows:***

ClockTargetTriggerGenerate.invoke {
domainNumber,
~~slave~~timeReceiverTimeCallback
}

#### 9.4.2.1 domainNumber (UInteger8)

***Change 9.4.2.1 as follows:***

This parameter is the domain number of the ~~ClockSlave~~ClockTimeReceiver entity that is requested to signal an event at the specified time.

***Change the title and text of 9.4.2.2 as follows:***

#### 9.4.2.2 ~~slave~~timeReceiverTimeCallback (ExtendedTimestamp)

If ~~slave~~timeReceiverTimeCallback is nonzero, its value is the synchronized time the corresponding ClockTargetTriggerGenerate.result function, i.e., the trigger, is to be invoked. If ~~slave~~timeReceiverTimeCallback is zero, any previous ClockTargetTriggerGenerate.invoke function for which a ClockTargetTriggerGenerate.result function has not yet been issued is canceled.

### 9.4.3 ClockTargetTriggerGenerate.result parameters

#### 9.4.3.1 domainNumber (UInteger8)

***Change 9.4.3.1 as follows:***

This parameter is the domain number of the ~~ClockSlave~~ClockTimeReceiver entity that is triggering an event at the specified time.

#### 9.4.3.2 errorCondition (Boolean)

***Change the NOTE in 9.4.3.2 as follows:***

NOTE—For example, the ClockTargetTriggerGenerate.result function is invoked with errorCondition = TRUE if the requested ~~slave~~timeReceiverTimeCallback is a time prior to the synchronized time when the corresponding ClockTargetTriggerGenerate.invoke function is invoked. As another example, the ClockTargetTriggerGenerate.result function is invoked with errorCondition = TRUE if a discontinuity in the synchronized time causes the requested ~~slave~~timeReceiverTimeCallback to be skipped over.

### 9.4.4 ClockTargetTriggerGenerate interface definition

***Change 9.4.4 as follows:***

The invocation of the ClockTargetTriggerGenerate.invoke function causes the ~~ClockSlave~~ClockTimeReceiver entity to store the value of the ~~slave~~timeReceiverTimeCallback parameter in an internal variable (replacing any previous value of that variable) until the synchronized time, or LocalClock time if gmPresent is FALSE, equals the value of that variable, at which time the ClockTargetTriggerGenerate.result function is invoked with errorCondition = FALSE. If it is not possible to invoke the ClockTargetTriggerGenerate.result function at ~~slave~~timeReceiverTimeCallback, e.g., if ~~slave~~timeReceiverTimeCallback is earlier than the synchronized time (or LocalClock time if gmPresent is


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

FALSE) when the ClockTargetTriggerGenerate.invoke function is invoked, the ClockTargetTriggerGenerate.result function is invoked with errorCondition = TRUE. Invocation of the ClockTargetTriggerGenerate.invoke function with ~~slave~~timeReceiverTimeCallback = 0 (which is earlier than any synchronized time) is used to cancel a pending request.

## 9.5 ClockTargetClockGenerator interface

### 9.5.1 General

***Change 9.5.1 as follows:***

This interface is used by the ClockTarget entity to request that the ~~ClockSlave~~ClockTimeReceiver entity deliver a periodic clock signal of specified period and phase. The ClockTarget entity invokes the ClockTargetClockGenerator.invoke function to request that the ~~ClockSlave~~ClockTimeReceiver entity generate the periodic clock signal. The ~~ClockSlave~~ClockTimeReceiver entity invokes the ClockTargetClockGenerator.result function at significant instants of the desired clock signal.

### 9.5.2 ClockTargetClockGenerator.invoke parameters

***Change 9.5.2 as follows:***

ClockTargetClockGenerator.invoke {
    domainNumber,
    clockPeriod,
    ~~slave~~timeReceiverTimeCallbackPhase
}

#### 9.5.2.1 domainNumber (UInteger8)

***Change 9.5.2.1 as follows:***

This parameter is the domain number of the ~~ClockSlave~~ClockTimeReceiver entity that is requested to deliver a periodic clock signal.

***Change the title and text of 9.5.2.3 as follows:***

#### 9.5.2.3 ~~slave~~timeReceiverTimeCallbackPhase (ExtendedTimestamp)

The value of ~~slave~~timeReceiverTimeCallbackPhase describes phase of the generated clock signal by specifying a point on the timescale in use such that ClockTargetClockGenerator.result invocations occur at synchronized times that differ from ~~slave~~timeReceiverTimeCallbackPhase by $n \times$ clockPeriod, where $n$ is an integer.

NOTE—The value of ~~slave~~timeReceiverTimeCallbackPhase can be earlier or later than the synchronized time the ClockTargetClockGenerator.invoke function is invoked; use of a ~~slave~~timeReceiverTimeCallbackPhase value in the future does not imply that the initiation of the periodic clock signal is suppressed until that synchronized time.

### 9.5.3 ClockTargetClockGenerator.result parameters

***Change 9.5.3 as follows:***

ClockTargetClockGenerator.result {
    domainNumber,
    ~~slave~~timeReceiverTime~~Callback,~~Callback
}


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

#### 9.5.3.1 domainNumber (UInteger8)

***Change 9.5.3.1 as follows:***

This parameter is the domain number of the ~~ClockSlave~~ClockTimeReceiver entity that is delivering a periodic clock signal.

***Change the title and text of 9.5.3.2 as follows:***

#### 9.5.3.2 ~~slave~~timeReceiverTimeCallback (ExtendedTimestamp)

The value of ~~slave~~timeReceiverTimeCallback is the synchronized time of this event.

### 9.6 ClockTargetPhaseDiscontinuity interface

#### 9.6.1 General

***Change 9.6.1 as follows:***

This interface provides discontinuity information, sent from the Grandmaster PTP Instance, to an application within an end station. It is used by the ~~ClockSlave~~ClockTimeReceiver entity to supply sufficient information to the ClockTarget entity to enable the ClockTarget entity to determine whether a phase or frequency discontinuity has occurred. The ~~ClockSlave~~ClockTimeReceiver invokes the ClockTargetPhaseDiscontinuity.result function in the SEND_SYNC_INDICATION block of the ~~ClockSlave~~ClockTimeReceiverSync state machine (see 10.2.13 and Figure 10-9). The invocation occurs when a PortSyncSync structure is received, after the needed information has been computed by the ~~ClockSlave~~ClockTimeReceiverSync state machine.

#### 9.6.2 ClockTargetPhaseDiscontinuity.result parameters

#### 9.6.2.1 domainNumber (UInteger8)

***Change 9.6.2.1 as follows:***

This parameter is the domain number of the ~~ClockSlave~~ClockTimeReceiver entity that is providing discontinuity information.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

# 10. Media-independent layer specification

## 10.1 Overview

### 10.1.2 Model of operation

***Change the first three paragraphs and the NOTE in 10.1.2 as follows:***

A PTP Instance contains a best ~~master~~timeTransmitter selection function and a synchronization function. These functions include PTP Port-specific aspects and aspects associated with the PTP Instance as a whole. The functions are distributed among a number of entities, which together describe the behavior of a compliant implementation. The functions are specified by a number of state machines.

The model for the media-independent layer of a PTP Instance is shown in Figure 10-1. It includes a single SiteSync entity, ~~ClockMaster~~ClockTimeTransmitter entity, and ~~ClockSlave~~ClockTimeReceiver entity for the PTP Instance as a whole, plus one PortSync for each PTP Port. The PTP Instance also includes one MD entity for each PTP Port, which is part of the media-dependent layer. The media-dependent functions performed by the MD entity are described in the clauses for the respective media. In addition to the entities, Figure 10-1 shows the information that flows between the entities via the PortSyncSync, MDSyncSend, and MDSyncReceive structures (see 10.2.2.3, 10.2.2.1, and 10.2.2.2, respectively).

The SiteSync, ~~ClockMaster~~ClockTimeTransmitter, ~~ClockSlave~~ClockTimeReceiver, and PortSync entities each contain a number of cooperating state machines, which are described later in this clause (the MD entity state machines are described in the respective media-dependent clauses). The ~~ClockMaster~~ClockTimeTransmitter entity receives information from an external time source, known as a *ClockSource entity* (see 9.2), via an application interface, and provides the information to the SiteSync entity. The ~~ClockSlave~~ClockTimeReceiver entity receives Grandmaster Clock time-synchronization and current Grandmaster PTP Instance information from the SiteSync entity, and makes the information available to an external application, known as a *clockTarget entity* (see 9.3 through 9.6), via one or more application service interfaces. The SiteSync entity executes the portion of best ~~master~~timeTransmitter clock selection associated with the PTP Instance as a whole, i.e., it uses the best ~~master~~timeTransmitter information received on each PTP Port to determine which PTP Port has received the best information, and updates the states of all the ports (see 10.3.1.1 for a discussion of PTP Port states). It also distributes synchronization information received on the ~~SlavePort~~TimeReceiverPort to all the ports whose state is ~~MasterPort~~TimeTransmitterPort (see 10.3.1.1). The PortSync entity for a ~~SlavePort~~TimeReceiverPort receives best ~~master~~timeTransmitter selection information from the PTP Instance at the other end of the associated link, compares this to the current best ~~master~~timeTransmitter information that it has, and forwards the result of the comparison to the Site Sync entity. The PortSync entity for a ~~SlavePort~~TimeReceiverPort also receives time-synchronization information from the MD entity associated with the PTP Port, and forwards it to the SiteSync entity. The PortSync entity for a ~~MasterPort~~TimeTransmitterPort sends best ~~master~~timeTransmitter selection and time-synchronization information to the MD entity for the PTP Port, which in turn sends the respective messages.

NOTE—This clause does not require a one-to-one correspondence between the PortSync entities of PTP Instances attached to the same gPTP communication path (see 3.11), i.e., more than two PTP Instances can be attached to a gPTP communication path that uses a shared medium and meet the requirements of this clause. However, it is possible for a media-dependent clause to have additional requirements that limit the gPTP communication paths to point-to-point links for that medium; in this case, each link has exactly two PortSync entities, which can be considered to be in one-to-one correspondence. One example of this is the full-duplex point-to-point media-dependent layer specified in ~~11~~Clause 11. In addition, one or more gPTP communication paths can be logically point-to-point but traverse the same shared medium.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

***Replace Figure 10-1 with the following figure:***

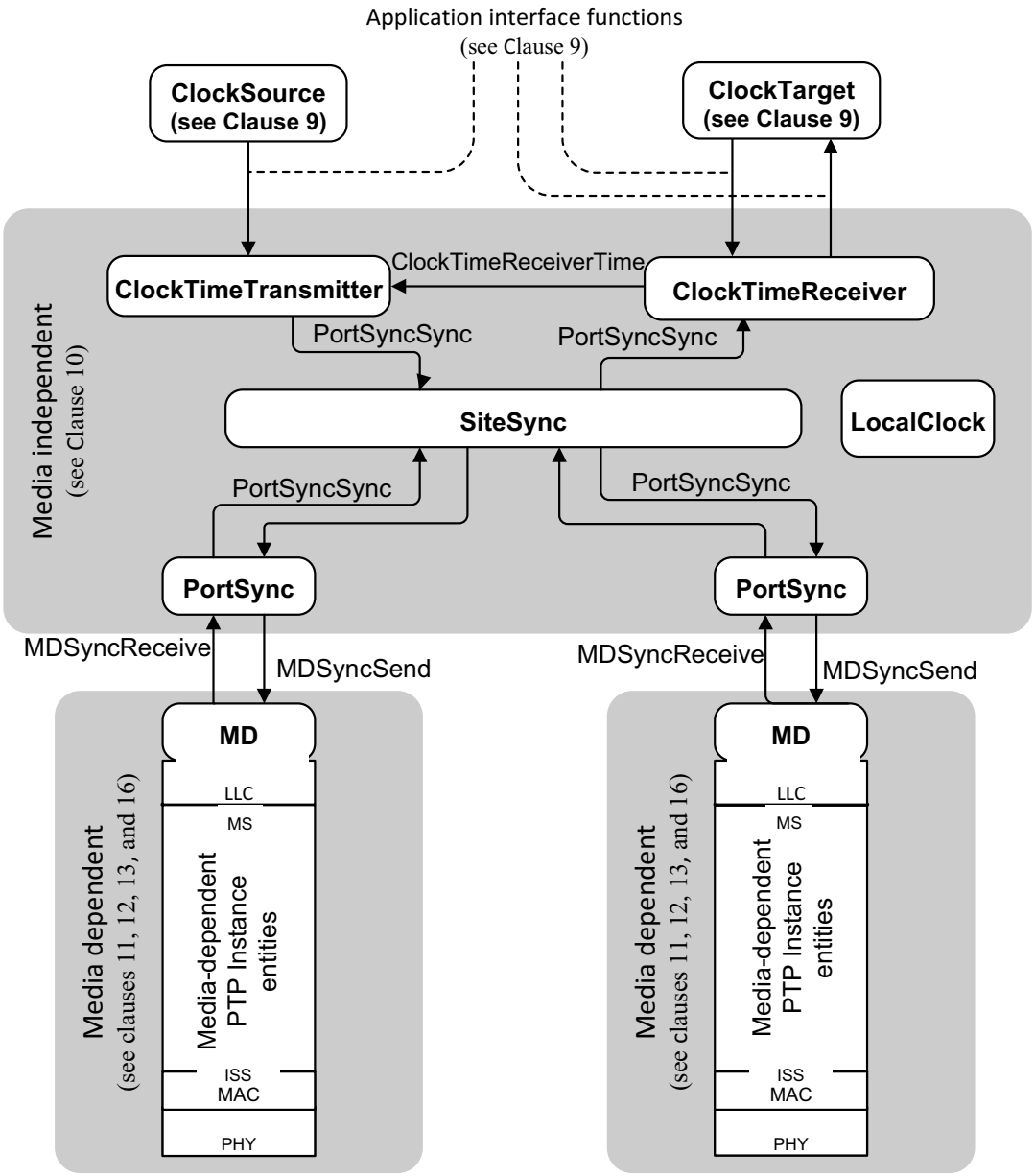



**Figure 10-1—Model for media-independent layer of PTP Instance**

***Change the last paragraph of 10.1.2 as follows:***

The time-synchronization state machines are described in 10.2. The best ~~master~~timeTransmitter clock selection state machines are described in 10.3. The attributes and format of the Announce message are described in 10.5 and 10.6. The timing characterization of the protocol is described in 10.7.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

### 10.1.3 Grandmaster-capable PTP Instance

***Change NOTE 2 in 10.1.3 as follows:***

NOTE 2—While a PTP Instance that is not grandmaster-capable can never be the Grandmaster PTP Instance of the gPTP domain, such a PTP Instance contains a best ~~master~~timeTransmitter selection function, invokes the best ~~master~~timeTransmitter selection algorithm, and conveys synchronization information received from the current Grandmaster PTP Instance.

## 10.2 Time-synchronization state machines

### 10.2.1 Overview

***Change the first two paragraphs of 10.2.1 as follows:***

The time-synchronization function in a PTP Instance is specified by a number of cooperating state machines. Figure 10-2 illustrates these state machines, their local variables, their interrelationships, and the global variables and structures used to communicate between them. The figure indicates the interaction between the state machines and the media-dependent layer and LocalClock entity.

The ~~ClockMasterSyncReceive~~ClockTimeTransmitterSyncReceive, ~~ClockMasterSyncOffset~~ClockTimeTransmitterSyncOffset, and ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machines are optional for PTP Instances that are not grandmaster-capable (see 8.6.2.1 and 10.1.3). These state machines may be present in a PTP Instance that is not grandmaster-capable; however, any information supplied by them, via the ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine, to the SiteSyncSync state machine is not used by the SiteSyncSync state machine if the PTP Instance is not grandmaster-capable.

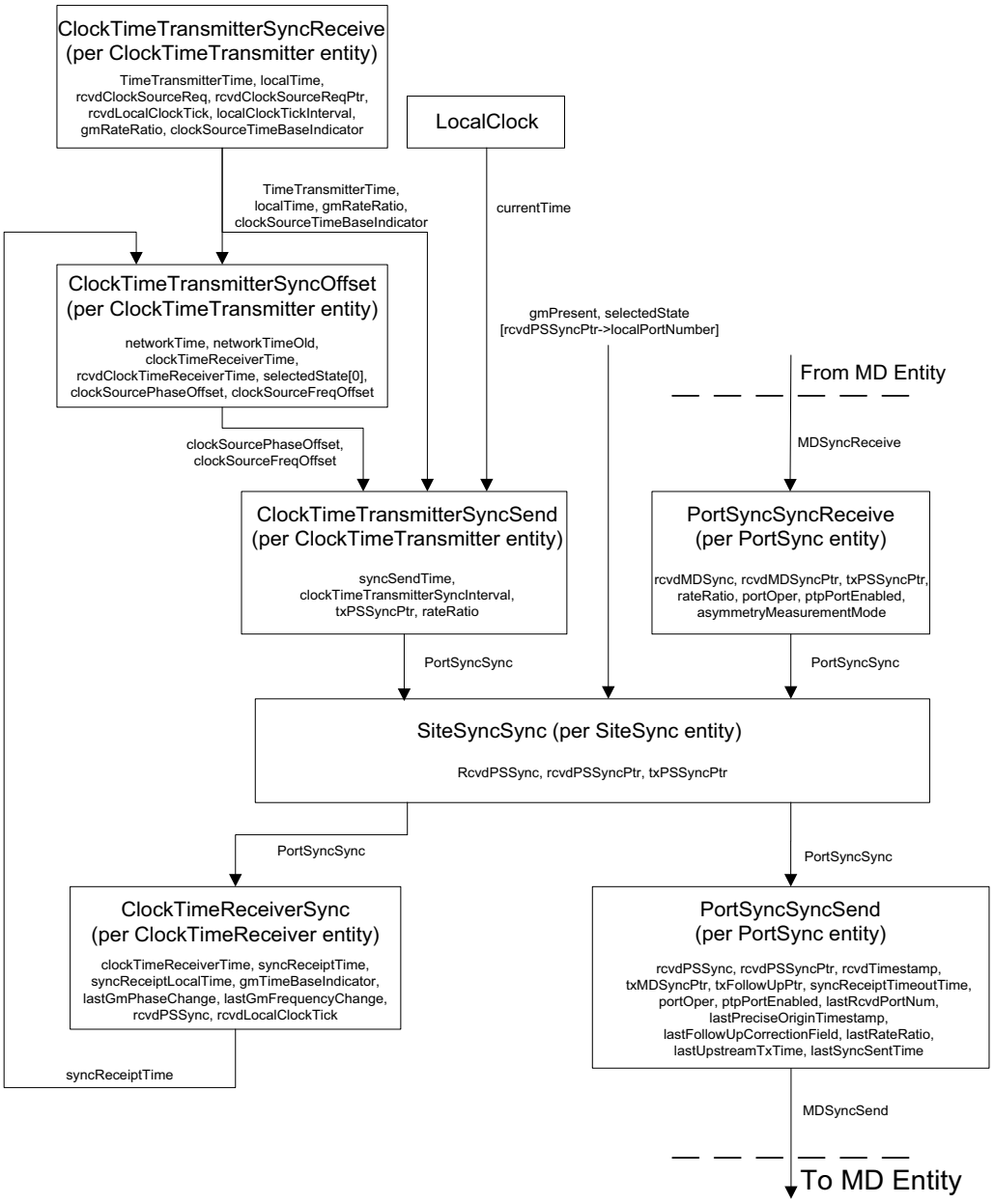

Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

***Replace Figure 10-2 (as amended by IEEE Std 802.1AS-2020/Cor 1-2021) with the following figure:***




Notes:
a) selectedState for each port and gmPresent are set by Port State Selection state machine (see 10.3.12)
b) currentTime is a global variable that is always equal to the current time relative to the local oscillator
c) application interfaces to higher layers are not shown
d) the ClockTimeTransmitterSyncReceive, ClockTimeTransmitterSyncSend, and ClockTimeTransmitterSyncOffset state machines are optional for PTP Instances that are not grandmaster-capable.

**Figure 10-2—Time-synchronization state machines—overview and interrelationships**


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

***Change the NOTE (inserted by IEEE Std 802.1AS-2020/Cor 1-2021), the last paragraph, and the lettered list in 10.2.1 as follows:***

NOTE—Figure 10-2 differs from the 2020 edition of this standard in that the global variable syncSequenceId is removed from the ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend block.

The media-independent layer state machines in Figure 10-2 are as follows:

a) ~~ClockMasterSyncReceive~~ClockTimeTransmitterSyncReceive (one instance per PTP Instance): receives ClockSourceTime.invoke functions from the ClockSource entity and notifications of LocalClock entity ticks (see 10.2.4.18), updates ~~masterTime~~timeTransmitterTime, and provides ~~masterTime~~timeTransmitterTime to ~~ClockMasterSyncOffset~~ClockTimeTransmitterSyncOffset and ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machines.

b) ~~ClockMasterSyncOffset~~ClockTimeTransmitterSyncOffset (one instance per PTP Instance): receives syncReceiptTime from the ~~ClockSlave~~ClockTimeReceiver entity and ~~masterTime~~timeTransmitterTime from the ~~ClockMasterSyncReceive~~ClockTimeTransmitterSyncReceive state machine, computes phase offset and frequency offset between ~~masterTime~~timeTransmitterTime and syncReceiptTime if the PTP Instance is not the Grandmaster PTP Instance, and provides the frequency and phase offsets to the ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine.

c) ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend (one instance per PTP Instance): receives ~~master~~timeTransmitterTime from the ~~ClockMasterSyncReceive~~ClockTimeTransmitterSyncReceive state machine, receives phase and frequency offset between ~~master~~timeTransmitterTime and syncReceiptTime from the ~~ClockMasterSyncOffset~~ClockTimeTransmitterSyncOffset state machine, and provides ~~masterTime~~timeTransmitterTime (i.e., synchronized time) and the phase and frequency offset to the SiteSync entity using a PortSyncSync structure.

d) PortSyncSyncReceive (one instance per PTP Instance, per PTP Port): receives time-synchronization information from the MD entity of the corresponding PTP Port, computes accumulated rateRatio, computes syncReceiptTimeoutTime, and sends the information to the SiteSync entity.

e) SiteSyncSync (one instance per PTP Instance): receives time-synchronization information, accumulated rateRatio, and syncReceiptTimeoutTime from the PortSync entity of the current ~~slave~~timeReceiver port or from the ~~ClockMaster~~ClockTimeTransmitter entity; and sends the information to the PortSync entities of all the ports and to the ~~ClockSlave~~ClockTimeReceiver entity.

f) PortSyncSyncSend (one instance per PTP Instance, per PTP Port): receives time-synchronization information from the SiteSync entity, requests that the MD entity of the corresponding PTP Port send a time-synchronization event message, receives the syncEventEgressTimestamp for this event message from the MD entity, uses the most recent time-synchronization information received from the SiteSync entity and the timestamp to compute time-synchronization information that will be sent by the MD entity in a general message (e.g., for full-duplex IEEE 802.3 media) or a subsequent event message (e.g., for IEEE 802.11 media), and sends this latter information to the MD entity.

g) ~~ClockSlave~~ClockTimeReceiverSync (one instance per PTP Instance): receives time-synchronization information from the SiteSync entity; computes ~~clockSlave~~ClockTimeReceiverTime and syncReceiptTime; sets syncReceiptLocalTime, GmTimeBaseIndicator, lastGmPhaseChange, and lastGmFreqChange; sends ~~clockSlave~~ClockTimeReceiverTime to the ~~ClockMaster~~ClockTimeTransmitter entity; and provides information to the ClockTarget entity (via the ClockTargetPhaseDiscontinuity interface; see 9.6) to enable that entity to determine if a phase or frequency discontinuity has occurred.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

### 10.2.2 Data structures communicated between state machines

#### 10.2.2.1 MDSyncSend

##### 10.2.2.1.6 preciseOriginTimestamp (Timestamp)

***Change 10.2.2.1.6 as follows:***

The preciseOriginTimestamp is the sourceTime of the ~~ClockMaster~~ClockTimeTransmitter entity of the Grandmaster PTP Instance, with any fractional nanoseconds truncated, when the received time-synchronization information was sent by the Grandmaster PTP Instance. The preciseOriginTimestamp is the value of the preciseOriginTimestamp member of the most recently received PortSyncSync structure from the PortSync entity of this PTP Port (see 10.2.2.3.8).

##### 10.2.2.1.7 upstreamTxTime (UScaledNs)

***Change 10.2.2.1.7 as follows:***

The upstreamTxTime is given by the following equation:

$$\text{upstreamTxTime} = \text{syncEventIngressTimestamp} - \frac{\text{meanLinkDelay}}{\text{neighborRateRatio}}$$

where

| | |
|---|---|
| syncEventIngressTimestamp | corresponds to the receipt of the time-synchronization information at the ~~slave~~timeReceiver port of this PTP Instance |
| meanLinkDelay | is defined in 10.2.5.8 |
| neighborRateRatio | is defined in 10.2.5.7 |
| upstreamTxTime | is the value of the upstreamTxTime member of the most recently received PortSyncSync structure from the PortSync entity of this PTP Port (see 10.2.2.3.9) |

#### 10.2.2.2 MDSyncReceive

##### 10.2.2.2.1 General

***Change the first paragraph in 10.2.2.2.1 as follows:***

This structure contains information that is sent by the MD entity of a PTP Port to the PortSync entity of that PTP Port. It provides the PortSync entity with ~~master~~timeTransmitter clock timing information and timestamp of receipt of a time-synchronization event message compensated for propagation time on the upstream link. The information is sent to the PortSync entity upon receipt of time-synchronization information by the MD entity of the PTP Port. The information is in turn provided by the PortSync entity to the SiteSync entity. The information is used by the PortSyncSyncReceive state machine of the PortSync entity to compute the rate ratio of the Grandmaster Clock relative to the local clock and is communicated to the SiteSync entity, and then the SiteSync entity communicates it to the other PortSync entities for use in computing ~~master~~timeTransmitter clock timing information.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

#### 10.2.2.2.4 sourcePortIdentity (PortIdentity)

***Change 10.2.2.2.4 as follows:***

The sourcePortIdentity is the value of the sourcePortIdentity of the time-synchronization event message received by this PTP Port. It is the portIdentity of the upstream ~~MasterPort~~TimeTransmitterPort that sent the event message.

#### 10.2.2.2.5 logMessageInterval (Integer8)

***Change 10.2.2.2.5 as follows:***

The logMessageInterval is the value of the logMessageInterval of the time-synchronization event message received by this PTP Port. It is the currentLogSyncInterval (see 10.7.2.3) of the upstream ~~MasterPort~~TimeTransmitterPort that sent the event message.

#### 10.2.2.2.6 preciseOriginTimestamp (Timestamp)

***Change 10.2.2.2.6 as follows:***

The preciseOriginTimestamp is the sourceTime of the ~~ClockMaster~~ClockTimeTransmitter entity of the Grandmaster PTP Instance, with any fractional nanoseconds truncated, when the time-synchronization event message was sent by the Grandmaster PTP Instance.

#### 10.2.2.2.7 upstreamTxTime (UScaledNs)

***Change 10.2.2.2.7 as follows:***

The upstreamTxTime is given by the following equation:

$$\text{upstreamTxTime} = \text{syncEventIngressTimestamp} - \frac{\text{meanLinkDelay}}{\text{neighborRateRatio}}$$

where

| | |
|---|---|
| syncEventIngressTimestamp | corresponds to the receipt of the time-synchronization information at the ~~slave~~timeReceiver port of this PTP Instance (i.e., at this PTP Port) |
| meanLinkDelay | is defined in 10.2.5.8 |
| neighborRateRatio | is defined in 10.2.5.7 |

### 10.2.2.3 PortSyncSync

#### 10.2.2.3.1 General

***Change 10.2.2.3.1 as follows:***

This structure is sent by the PortSync and ~~ClockMaster~~ClockTimeTransmitter entities to the SiteSync entity and also from the SiteSync entity to the PortSync and ~~ClockSlave~~ClockTimeReceiver entities.

When sent from the PortSync or ~~ClockMaster~~ClockTimeTransmitter entity, it provides the SiteSync entity with ~~master~~timeTransmitter clock timing information, timestamp of receipt of a time-synchronization event message compensated for propagation time on the upstream link, and the time at which sync receipt timeout occurs if a subsequent Sync message is not received by then. The information is used by the SiteSync entity


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

to compute the rate ratio of the Grandmaster Clock relative to the local clock and is communicated to the other PortSync entities for use in computing ~~master~~timeTransmitter clock timing information.

When sent from the SiteSync entity to the PortSync or ~~ClockSlave~~ClockTimeReceiver entity, the structure contains information needed to compute the synchronization information that will be included in respective fields of the time-synchronization event and general messages that will be sent and also to compute the synchronized time that the ~~ClockSlave~~ClockTimeReceiver entity will supply to the ClockTarget entity.

PortSyncSync {
domainNumber,
localPortNumber,
syncReceiptTimeoutTime,
followUpCorrectionField,
sourcePortIdentity,
logMessageInterval,
preciseOriginTimestamp,
upstreamTxTime,
rateRatio,
gmTimeBaseIndicator,
lastGmPhaseChange,
lastGmFreqChange
}

The parameters of the PortSyncSync structure are defined in the following subclauses for when the structure is sent from the PortSync or ~~ClockMaster~~ClockTimeTransmitter entity to the SiteSync entity. If the structure is sent from the SiteSync entity to the PortSync or ~~ClockSlave~~ClockTimeReceiver entity, the member values are copied from the most recently received PortSyncSync structure where the PTP Port that received this structure has PTP Port state of ~~SlavePort~~TimeReceiverPort.

#### 10.2.2.3.3 localPortNumber (UInteger16)

***Change 10.2.2.3.3 as follows:***

If the structure is sent by a PortSync entity, the localPortNumber is the port number of the PTP Port whose PortSync entity sent this structure. If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the localPortNumber is zero.

#### 10.2.2.3.4 syncReceiptTimeoutTime (UScaledNs)

***Change 10.2.2.3.4 as follows:***

If the structure is sent by a PortSync entity, the syncReceiptTimeoutTime is the value of the local time (i.e., the free-running, local clock time) at which sync receipt timeout occurs if a subsequent time-synchronization event message is not received by that time. If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the syncReceiptTimeoutTime is $FFFFFFFFFFFFFFFF_{16}$ [see item h) in 10.2.9.2.1].

#### 10.2.2.3.5 followUpCorrectionField (ScaledNs)

***Change 10.2.2.3.5 as follows:***

If the structure is sent by a PortSync entity, the followUpCorrectionField is the value of the followUpcorrectionField member of the MDSyncReceive structure whose receipt caused the sending of this


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

structure (see 10.2.2.2.2). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the followUpCorrectionField is the sub-nanosecond portion of the ~~ClockMaster~~ClockTimeTransmitter time.

#### 10.2.2.3.6 sourcePortIdentity (PortIdentity)

***Change 10.2.2.3.6 as follows:***

If the structure is sent by a PortSync entity, the sourcePortIdentity is the value of the sourcePortIdentity member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.4). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the clockIdentity member of the sourcePortIdentity is the clockIdentity of this PTP Instance, and the portNumber member of the sourcePortIdentity is 0.

#### 10.2.2.3.7 logMessageInterval (Integer8)

***Change 10.2.2.3.7 as follows:***

If the structure is sent by a PortSync entity, the logMessageInterval is the value of the logMessageInterval member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.5). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the logMessageInterval is the value of ~~clockMasterLogSyncInterval~~clockTimeTransmitterLogSyncInterval (see 10.7.2.4).

#### 10.2.2.3.8 preciseOriginTimestamp (Timestamp)

***Change 10.2.2.3.8 as follows:***

If the structure is sent by a PortSync entity, the preciseOriginTimestamp is the value of the preciseOriginTimestamp member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.6). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the preciseOriginTimestamp is the ~~ClockMaster~~ClockTimeTransmitter time truncated to the next lower nanosecond.

#### 10.2.2.3.9 upstreamTxTime (UScaledNs)

***Change 10.2.2.3.9 as follows:***

If the structure is sent by a PortSync entity, the upstreamTxTime is the value of the upstreamTxTime member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.7). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the upstreamTxTime is the local clock time corresponding to the ~~ClockMaster~~ClockTimeTransmitter time.

#### 10.2.2.3.10 rateRatio (Float64)

***Change 10.2.2.3.10 as follows:***

If the structure is sent by a PortSync entity, the rateRatio is the value of the rateRatio member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.8). It is equal to the ratio of the frequency of the Grandmaster Clock to the frequency of the LocalClock entity of the PTP Instance at the other end of the link attached to this PTP Port, i.e., the PTP Instance that sent the most recently-received time-synchronization event message (see 10.2.8.1.4). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the rateRatio is equal to gmRateRatio (see 10.2.4.14).


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

#### 10.2.2.3.11 gmTimeBaseIndicator (UInteger16)

***Change 10.2.2.3.11 as follows:***

If the structure is sent by a PortSync entity, the gmTimeBaseIndicator is the value of the gmTimeBaseIndicator member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.9). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the gmTimeBaseIndicator is equal to clockSourceTimeBaseIndicator (see 10.2.4.8).

#### 10.2.2.3.12 lastGmPhaseChange (ScaledNs)

***Change 10.2.2.3.12 as follows:***

If the structure is sent by a PortSync entity, the lastGmPhaseChange is the value of the lastGmPhaseChange member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.9). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the lastGmPhaseChange is equal to clockSourcePhaseOffset (see 10.2.4.7).

#### 10.2.2.3.13 lastGmFreqChange (Float64)

***Change 10.2.2.3.13 as follows:***

If the structure is sent by a PortSync entity, the lastGmFreqChange is the value of the lastGmFreqChange member of the MDSyncReceive structure whose receipt caused the sending of this structure (see 10.2.2.2.9). If the structure is sent by a ~~ClockMaster~~ClockTimeTransmitter entity, the lastGmFreqChange is equal to clockSourceFreqOffset (see 10.2.4.6).

### 10.2.3 Overview of global variables used by time synchronization state machines

***Change Table 10-1 as follows:***

**Table 10-1—Summary of scope of global variables used by time synchronization state machines (see 10.2.4 and 10.2.5)**

| Variable name | Subclause of definition | Per PTP Instance (i.e., per domain) | Per PTP Instance, per PTP Port | Instance used by CMLDS (i.e., variable is common across all LinkPorts) | Instance used by CMLDS, per LinkPort |
|---|---|---|---|---|---|
| BEGIN | 10.2.4.1 | Yes | No | Yes | No |
| ~~clockMasterSyncInterval~~clockTimeTransmitterSyncInterval | 10.2.4.2 | Yes | No | No | No |
| ~~clockSlave~~clockTimeReceiverTime | 10.2.4.3 | Yes | No | No | No |
| syncReceiptTime | 10.2.4.4 | Yes | No | No | No |
| syncReceiptLocalTime | 10.2.4.5 | Yes | No | No | No |
| clockSourceFreqOffset | 10.2.4.6 | Yes | No | No | No |
| clockSourcePhaseOffset | 10.2.4.7 | Yes | No | No | No |
| clockSourceTimeBaseIndicator | 10.2.4.8 | Yes | No | No | No |
| clockSourceTimeBaseIndicatorOld | 10.2.4.9 | Yes | No | No | No |


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

**Table 10-1—Summary of scope of global variables used by time synchronization state machines (see 10.2.4 and 10.2.5) *(continued)***

| Variable name | Subclause of definition | Per PTP Instance (i.e., per domain) | Per PTP Instance, per PTP Port | Instance used by CMLDS (i.e., variable is common across all LinkPorts) | Instance used by CMLDS, per LinkPort |
|---|---|---|---|---|---|
| clockSourceLastGmPhaseChange | 10.2.4.10 | Yes | No | No | No |
| clockSourceLastGmFreqChange | 10.2.4.11 | Yes | No | No | No |
| currentTime | 10.2.4.12 | Yes | No | No | No |
| gmPresent | 10.2.4.13 | Yes | No | No | No |
| gmRateRatio | 10.2.4.14 | Yes | No | No | No |
| gmTimeBaseIndicator | 10.2.4.15 | Yes | No | No | No |
| lastGmPhaseChange | 10.2.4.16 | Yes | No | No | No |
| lastGmFreqChange | 10.2.4.17 | Yes | No | No | No |
| localClockTickInterval | 10.2.4.18 | Yes | No | No | No |
| localTime | 10.2.4.19 | Yes | No | No | No |
| selectedState | 10.2.4.20 | Yes | No | No | No |
| ~~masterTime~~timeTransmitterTime | 10.2.4.21 | Yes | No | No | No |
| thisClock | 10.2.4.22 | Yes | No | Yes | No |
| parentLogSyncInterval | 10.2.4.23 | Yes | No | No | No |
| instanceEnable | 10.2.4.24 | Yes | No | No | No |
| syncReceiptTimeoutTime | 10.2.4.25 | Yes | No | No | No |
| asCapable | 10.2.5.1 | No | Yes | No | No |
| asymmetryMeasurementMode | 10.2.5.2 | No | Yes[a] | No | Yes |
| syncReceiptTimeoutTimeInterval | 10.2.5.3 | No | Yes | No | No |
| currentLogSyncInterval | 10.2.5.4 | No | Yes | No | No |
| initialLogSyncInterval | 10.2.5.5 | No | Yes | No | No |
| syncInterval | 10.2.5.6 | No | Yes | No | No |
| neighborRateRatio | 10.2.5.7 | No | Yes[a] | No | Yes |
| meanLinkDelay | 10.2.5.8 | No | Yes[a] | No | Yes |
| delayAsymmetry | 10.2.5.9 | No | Yes[a] | No | Yes |
| computeNeighborRateRatio | 10.2.5.10 | No | Yes[a] | No | Yes |
| computeMeanLinkDelay | 10.2.5.11 | No | Yes[a] | No | Yes |
| portOper[b] | 10.2.5.12 | No | Yes | No | Yes |
| ptpPortEnabled | 10.2.5.13 | No | Yes | No | No |
| thisPort | 10.2.5.14 | No | Yes | No | Yes |
| syncLocked | 10.2.5.15 | No | Yes | No | No |
| neighborGptpCapable | 10.2.5.16 | No | Yes | No | No |
| syncSlowdown | 10.2.5.17 | No | Yes | No | No |


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

**Table 10-1—Summary of scope of global variables used by time synchronization state machines (see 10.2.4 and 10.2.5) *(continued)***

| Variable name | Subclause of definition | Per PTP Instance (i.e., per domain) | Per PTP Instance, per PTP Port | Instance used by CMLDS (i.e., variable is common across all LinkPorts) | Instance used by CMLDS, per LinkPort |
|---|---|---|---|---|---|
| oldSyncInterval | 10.2.5.18 | No | Yes | No | No |
| gPtpCapableMessageSlowdown | 10.2.5.19 | No | Yes | No | No |
| gPtpCapableMessageInterval | 10.2.5.20 | No | Yes | No | No |
| oldGptpCapableMessageInterval | 10.2.5.21 | No | Yes | No | No |
| currentLogGptpCapableMessageInterval | 10.2.5.22 | No | Yes | No | No |
| initialLogGptpCapableMessageInterval | 10.2.5.23 | No | Yes | No | No |

[a] The instance of this variable that is per PTP Instance, per PTP Port exists only for domain 0.
[b] There is one instance of this variable per physical port, which is accessible by all PTP Ports and LinkPorts associated with the physical port.

### 10.2.4 Per PTP Instance global variables

***Change 10.2.4.2, 10.2.4.3, and 10.2.4.4 as follows:***

**10.2.4.2 ~~clockMasterSyncInterval~~clockTimeTransmitterSyncInterval:** A variable containing the mean time interval between successive messages providing time-synchronization information by the ~~ClockMaster~~ClockTimeTransmitter entity to the SiteSync entity. This value is given by $1000000000 \times 2^{\text{~~clockMasterLogSyncInterval~~clockTimeTransmitterLogSyncInterval}}$ ns, where ~~clockMasterLogSyncInterval~~clockTimeTransmitterLogSyncInterval is the logarithm to base 2 of the mean time between the successive providing of time-synchronization information by the ~~ClockMaster~~ClockTimeTransmitter entity (see 10.7.2.4). The data type for ~~clockMasterSyncInterval~~clockTimeTransmitterSyncInterval is UScaledNs.

**10.2.4.3 ~~clockSlave~~ClockTimeReceiverTime:** The synchronized time maintained, at the ~~slave~~timeReceiver, at the granularity of the LocalClock entity [i.e., a new value is computed every localClockTickInterval (see 10.2.4.18) by the ~~ClockSlave~~ClockTimeReceiver entity]. The data type for ~~clockSlave~~ClockTimeReceiverTime is ExtendedTimestamp.

**10.2.4.4 syncReceiptTime:** The synchronized time computed by the ~~ClockSlave~~ClockTimeReceiver entity at the instant time-synchronization information, contained in a PortSyncSync structure, is received. The data type for syncReceiptTime is ExtendedTimestamp.

***Change 10.2.4.8 as follows:***

**10.2.4.8 clockSourceTimeBaseIndicator:** A global variable that is set equal to the timeBaseIndicator parameter of the ClockSourceTime.invoke application interface function (see 9.2.2.3), by the ~~ClockMaster~~ClockTimeTransmitter entity. The parameter timeBaseIndicator of ClockSourceTime.invoke is set by the ClockSource entity and is changed by that entity whenever the time base changes. The data type for clockSourceTimeBaseIndicator is UInteger16.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

***Change 10.2.4.15, 10.2.4.16, and 10.2.4.17 as follows:***

**10.2.4.15 gmTimeBaseIndicator:** The most recent value of gmTimeBaseIndicator provided to the ~~ClockSlave~~ClockTimeReceiverSync state machine via a PortSyncSync structure. The data type for gmTimeBaseIndicator is UInteger16.

**10.2.4.16 lastGmPhaseChange:** The most recent value of lastGmPhaseChange provided to the ~~ClockSlave~~ClockTimeReceiverSync state machine via a PortSyncSync structure. The data type for lastGmPhaseChange is ScaledNs.

**10.2.4.17 lastGmFreqChange:** The most recent value of lastGmFreqChange provided to the ~~ClockSlave~~ClockTimeReceiverSync state machine via a PortSyncSync structure. The data type for lastGmFreqChange is Float64.

***Change 10.2.4.21 as follows:***

**10.2.4.21 ~~masterTime~~timeTransmitterTime:** The time maintained by the ~~ClockMaster~~ClockTimeTransmitter entity, based on information received from the ClockSource and LocalClock entities. The data type for ~~masterTime~~timeTransmitterTime is ExtendedTimestamp.

***Change 10.2.4.23 as follows:***

**10.2.4.23 parentLogSyncInterval:** The most recent logMessageInterval value received on the ~~slave~~timeReceiver port. If this PTP Instance is the Grandmaster PTP Instance, then this is the ~~clockMasterLogSyncInterval~~clockTimeTransmitterLogSyncInterval (see 10.7.2.4). The data type for parentLogSyncInterval is Integer8.

### 10.2.5 Per-port global variables

***Change 10.2.5.15 as follows:***

**10.2.5.15 syncLocked:** A Boolean, set by the PortSyncSyncSend state machine (see 10.2.12.3), that indicates that this PTP Port, when operating as a ~~master~~timeTransmitter port, shall transmit a Sync as soon as possible after the ~~slave~~timeReceiver port received a Sync (ignoring syncInterval). If FALSE, the PTP Port shall use the timing set by syncInterval.

### 10.2.7 SiteSyncSync state machine

### 10.2.7.1 State machine variables

***Change 10.2.7.1.1 as follows:***

**10.2.7.1.1 rcvdPSSyncSSS:** A Boolean variable that notifies the current state machine when a PortSyncSync structure (see 10.2.2.3) is received from the PortSyncSyncReceive state machine of a PortSync entity or from the ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine of the ~~ClockMaster~~ClockTimeTransmitter entity. This variable is reset by this state machine.

### 10.2.7.2 State machine functions

***Change 10.2.7.2.2 as follows:***

**10.2.7.2.2 txPSSync (txPSSyncPtrSSS):** Transmits a copy of the PortSyncSync structure pointed to by txPSSyncPtrSSS to the PortSyncSyncSend state machine of each PortSync entity and the


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

~~ClockSlave~~ClockTimeReceiverSync state machine of the ~~ClockSlave~~ClockTimeReceiver entity of this PTP Instance.

### 10.2.7.3 State diagram

***Change the text of 10.2.7.3 as follows:***

The SiteSyncSync state machine shall implement the function specified by the state diagram in Figure 10-3, the local variables specified in 10.2.7.1, the functions specified in 10.2.7.2, the structure specified in 10.2.2.3, and the relevant global variables and functions specified in 10.2.4 through 10.2.6. The state machine receives time-synchronization information, accumulated rateRatio, and syncReceiptTimeoutTime from the PortSync entity (PortSyncSyncReceive state machine) of the current ~~slave~~timeReceiver port or from the ~~ClockMaster~~ClockTimeTransmitter entity (~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine). If the information was sent by a PortSync entity, the state machine also receives the portIdentity of the PTP Port on the upstream PTP Instance that sent the information to this PTP Instance (if the information was sent by the ~~ClockMaster~~ClockTimeTransmitter entity, the portIdentity is that of the ~~ClockMaster~~ClockTimeTransmitter entity, i.e., it has clockIdentity equal to the clockIdentity of this PTP Instance and portNumber 0). The state machine sends a PortSyncSync structure to the PortSync entities of all the ports and to the ~~ClockSlave~~ClockTimeReceiver entity.

***Replace Figure 10-3 with the following figure:***

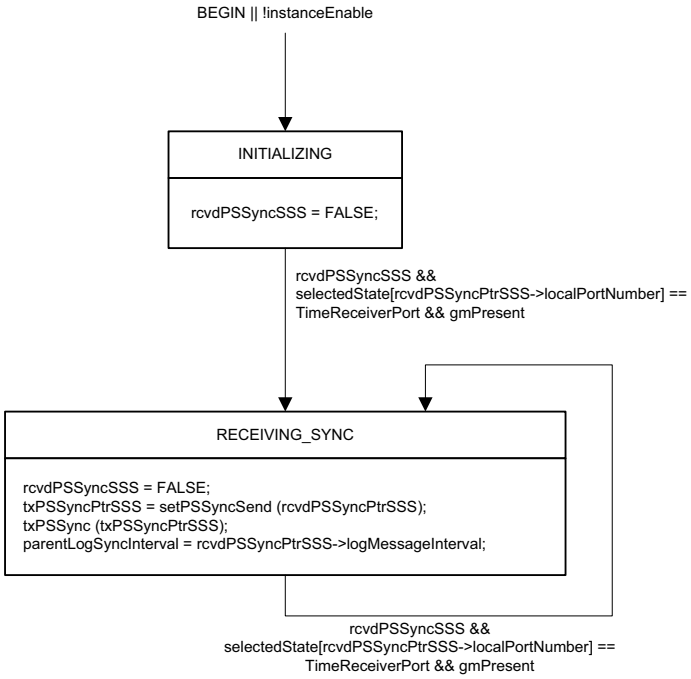



**Figure 10-3—SiteSyncSync state machine**


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

### 10.2.8 PortSyncSyncReceive state machine

#### 10.2.8.1 State machine variables

***Change 10.2.8.1.4 as follows:***

**10.2.8.1.4 rateRatio:** A Float64 variable that holds the ratio of the frequency of the Grandmaster Clock to the frequency of the LocalClock entity. This frequency ratio is computed by:

a) Measuring the ratio of the Grandmaster Clock frequency to the LocalClock frequency at the Grandmaster PTP Instance and initializing rateRatio to this value in the ~~ClockMasterSend~~ClockTimeTransmitterSyncSend state machine of the Grandmaster PTP Instance and

b) Accumulating, in the PortSyncSyncReceive state machine of each PTP Instance, the frequency offset of the LocalClock entity of the PTP Instance at the remote end of the link attached to that PTP Port to the frequency of the LocalClock entity of this PTP Instance.

***Change the title of 10.2.9 as follows:***

### 10.2.9 ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine

#### 10.2.9.2 State machine functions

***Change 10.2.9.2.1 as follows:***

**10.2.9.2.1 setPSSyncCMSS (gmRateRatio):** Creates a PortSyncSync structure to be transmitted, and returns a pointer to this structure. The members are set as follows:

a) localPortNumber is set to 0.

b) preciseOriginTimestamp is set equal to the ~~masterTime~~timeTransmitterTime, with any fractional nanoseconds truncated.

c) followUpCorrectionField is set equal to the sum of

1) The fractional nanoseconds portion of ~~masterTime~~timeTransmitterTime.fractionalNanoseconds and

2) The quantity gmRateRatio × (currentTime – localTime).

d) The clockIdentity member of sourcePortIdentity is set equal to the clockIdentity of this PTP Instance.

e) The portNumber member of the sourcePortIdentity is set to 0.

NOTE 1—This quantity and localPortNumber are redundant; both are retained so that the SiteSync entity can process PortSyncSync structures received from a PortSync entity or the ~~ClockMaster~~ClockTimeTransmitter entity in the same manner.

f) logMessageInterval is set to ~~clockMasterLogSyncInterval~~clockTimeTransmitterLogSyncInterval.

g) upstreamTxTime is set equal to localTime.

h) syncReceiptTimeoutTime is set equal to $FFFFFFFFFFFFFFFF_{16}$, which indicates that there is no sync receipt timeout.

NOTE 2—A ~~ClockMaster~~ClockTimeTransmitter entity does not receive Sync messages, and there is no notion of sync receipt timeout.

i) rateRatio is set equal to gmRateRatio.

j) gmTimeBaseIndicator is set equal to clockSourceTimeBaseIndicator.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025

k) lastGmPhaseChange is set equal to clockSourcePhaseOffset.

l) lastGmFreqChange is set equal to clockSourceFreqOffset.

m) domainNumber is set equal to the domain number of this gPTP domain.

***Change 10.2.9.2.3 as follows:***

**10.2.9.2.3 ~~computeClockMasterSyncInterval~~computeClockTimeTransmitterSyncInterval():** Computes the value of ~~clockMasterSyncInterval~~clockTimeTransmitterSyncInterval (see 10.2.4.2) as $1000000000 \times 2^{\text{clockTimeTransmitterLogSyncInterval}\sout{\text{clockMasterLogSyncInterval}}}$ ns, where ~~clockMasterLogSyncInterval~~clockTimeTransmitterLogSyncInterval is the minimum currentLogSyncInterval value, taken over all the PTP Ports of the PTP Instance (see 10.7.2.4).

### 10.2.9.3 State diagram

***Change the text of 10.2.9.3 as follows:***

The ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine shall implement the function specified by the state diagram in Figure 10-5, the local variables specified in 10.2.9.1, the functions specified in 10.2.9.2, the structure specified in 10.2.2.3, and the relevant global variables and functions specified in 10.2.4 through 10.2.6. The state machine receives ~~masterTime~~timeTransmitterTime and clockSourceTimeBaseIndicator from the ~~ClockMasterSyncReceive~~ClockTimeTransmitterSyncReceive state machine, and phase and frequency offset between ~~masterTime~~timeTransmitterTime and syncReceiptTime from the ~~ClockMasterSyncOffset~~ClockTimeTransmitterSyncOffset state machine. It provides ~~masterTime~~timeTransmitterTime (i.e., synchronized time) and the phase and frequency offset to the SiteSync entity via a PortSyncSync structure.

The ~~ClockMasterSyncSend~~ClockTimeTransmitterSyncSend state machine is optional for PTP Instances that are not grandmaster-capable (see 8.6.2.1, 10.1.3, and 10.2.1). This state machine may be present in a PTP Instance that is not grandmaster-capable; however, any information supplied by it to the SiteSyncSync state machine is not used by the SiteSyncSync state machine if the PTP Instance is not grandmaster-capable.


Copyright © 2024 IEEE. All rights reserved.

Documents
Document Sharing Hub

Authorized licensed use limited to: Tsinghua University. Downloaded on February 26,2026 at 08:01:10 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 8802-1AS-2021_Amd1:2025-2025