IEC 61588

Edition 2.0 2009-02

INTERNATIONAL STANDARD

IEEE 1588™

Precision clock synchronization protocol for networked measurement and control systems

IEC 61588:2009(E) IEEE Std. 1588(E):2008

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2008 IEEE

All rights reserved. IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by the Institute of Electrical and Electronics Engineers, Inc.

Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the IEC Central Office.

Any questions about IEEE copyright should be addressed to the IEEE. Enquiries about obtaining additional rights to this publication and other information requests should be addressed to the IEC or your local IEC member National Committee.

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland
Email: inmail@iec.ch
Web: www.iec.ch

The Institute of Electrical and Electronics Engineers, Inc
3 Park Avenue
US-New York, NY10016-5997
USA
Email: stds-info@ieee.org
Web: www.ieee.org

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

- Catalogue of IEC publications: www.iec.ch/searchpub
The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub
Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.

- Electropedia: www.electropedia.org
The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv
If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch
Tel.: +41 22 919 02 11

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

IEC 61588

Edition 2.0 2009-02

IEEE 1588™

INTERNATIONAL STANDARD

Precision clock synchronization protocol for networked measurement and control systems

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

PRICE CODE XH

ICS 25.040.40; 35.110; 35.240.50

ISBN 978-2-88910-546-5

Warning! Make sure that you obtained this publication from an authorized distributor.

AVA Documents
Document Sharing Hub

Full document details: IEC 61588 2009

CONTENTS

Foreword ... xi
IEEE introduction ... xiv

1. Overview ... 1
   1.1 Scope ... 1
   1.2 Purpose ... 2
   1.3 Layout of the document ... 2

2. Normative references ... 3

3. Definitions, acronyms, and abbreviations ... 4
   3.1 Definitions ... 4
   3.2 Acronyms and abbreviations ... 7

4. Conventions ... 8
   4.1 Descriptive lexical form syntax ... 8
   4.2 Word usage ... 9
   4.3 Behavioral specification notation ... 10

5. Data types and on-the-wire formats in a PTP system ... 11
   5.1 General ... 11
   5.2 Primitive data type specifications ... 11
   5.3 Derived data type specifications ... 12
   5.4 On-the-wire formats ... 15

6. Clock synchronization model ... 16
   6.1 General ... 16
   6.2 Principle assumptions about the network and implementation recommendations ... 16
   6.3 PTP systems ... 17
   6.4 PTP message classes ... 17
   6.5 PTP device types ... 18
   6.6 Synchronization overview ... 29
   6.7 PTP communications overview ... 37

7. Characterization of PTP entities ... 41
   7.1 Domains ... 41
   7.2 PTP timescale ... 41
   7.3 PTP communications ... 42
   7.4 PTP communication media ... 46
   7.5 PTP ports ... 47
   7.6 PTP device characterization ... 53
   7.7 PTP timing characterization ... 61

8. PTP data sets ... 63
   8.1 General specifications for data set members ... 63
   8.2 Data sets for ordinary and boundary clocks ... 65
   8.3 Data sets for transparent clocks ... 74

9. PTP for ordinary and boundary clocks ... 76
   9.1 General protocol requirements for PTP ordinary and boundary clocks ... 76
   9.2 State protocol ... 76
   9.3 Best master clock algorithm ... 83
   9.4 Grandmaster clocks ... 92
   9.5 Message processing semantics ... 93
   9.6 Changes in the local clock ... 107

10. PTP for transparent clocks ... 107
   10.1 General requirements for both end-to-end and peer-to-peer transparent clocks ... 107
   10.2 End-to-end transparent clock requirements ... 108

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

10.3 Peer-to-peer transparent clock requirements ... 108

11. Clock offset, path delay, residence time, and asymmetry corrections ... 108
11.1 General specifications ... 108
11.2 Computation of clock offset in ordinary and boundary clocks ... 109
11.3 Delay request-response mechanism ... 110
11.4 Peer delay mechanism ... 112
11.5 Transparent clock residence time correction for PTP version 2 events ... 117
11.6 Asymmetry correction for PTP version 2 event messages ... 120

12. Synchronization and syntonization of clocks ... 122
12.1 Syntonization ... 122
12.2 Synchronization ... 123

13. PTP message formats ... 124
13.1 General ... 124
13.2 General message format requirements ... 124
13.3 Header ... 124
13.4 Suffix ... 128
13.5 Announce message ... 128
13.6 Sync and Delay_Req messages ... 130
13.7 Follow_Up message ... 130
13.8 Delay_Resp message ... 130
13.9 Pdelay_Req message ... 131
13.10 Pdelay_Resp message ... 131
13.11 Pdelay_Resp_Follow_Up message ... 132
13.12 Signaling message ... 132
13.13 Management message ... 133

14. TLV entity specifications ... 133
14.1 General requirements ... 133
14.2 Experimental TLVs ... 134
14.3 Vendor and standard organization extension TLVs ... 135

15. Management ... 135
15.1 General ... 135
15.2 PTP management mechanism ... 136
15.3 Processing of management messages ... 136
15.4 Management message format ... 137
15.5 Management TLVs ... 138

16. General optional features ... 158
16.1 Unicast message negotiation (optional) ... 158
16.2 Path trace (optional) ... 163
16.3 Alternate timescales (optional) ... 165

17. State configuration options ... 169
17.1 General ... 169
17.2 Data types for options ... 169
17.3 Grandmaster clusters (optional) ... 170
17.4 Alternate master (optional) ... 172
17.5 Unicast discovery (optional) ... 173
17.6 Acceptable master table (optional) ... 175

18. Compatibility requirements ... 177
18.1 Compatibility between version 2 and future versions ... 177
18.2 Compatibility between version 1 and version 2 ... 177
18.3 Message formats and data types ... 178
18.4 Naming changes ... 183
18.5 Restrictions on mixed version 1 and version 2 systems ... 183

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

19. Conformance 184
19.1 Conformance objective 184
19.2 PTP conformance requirements 184
19.3 PTP profiles 185
Annex A (informative) Using PTP 187
Annex B (informative) Timescales and epochs in PTP 197
Annex C (informative) Examples of residence and asymmetry corrections 200
Annex D (normative) Transport of PTP over User Datagram Protocol over Internet Protocol Version 4 . 219
Annex E (normative) Transport of PTP over User Datagram Protocol over Internet Protocol Version 6.. 221
Annex F (normative) Transport of PTP over IEEE 802.3 /Ethernet 223
Annex G (normative) Transport of PTP over DeviceNET 225
Annex H (normative) Transport of PTP over ControlNET 228
Annex I (normative) Transport of PTP over IEC 61158 Type 10 230
Annex J (normative) Default PTP profiles 237
Annex K (informative) Security protocol (experimental) 241
Annex L (informative) Transport of cumulative frequency scale factor offset (experimental) 264
Annex M (informative) Bibliography 268
Annex N (informative) List of partcipants 270

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

TABLES

Table 1 —Primitive PTP data types ... 12
Table 2 —domainNumber ... 41
Table 3 —networkProtocol enumeration ... 46
Table 4 —Non-EUI-64 addressTechnology enumeration ... 51
Table 5 —clockClass specifications ... 55
Table 6 —clockAccuracy enumeration ... 56
Table 7 —timeSource enumeration ... 57
Table 8 —PTP state enumeration ... 73
Table 9 —Delay mechanism enumeration ... 74
Table 10 —PTP portState definition ... 77
Table 11 —Event applicability in boundary clocks ... 83
Table 12 —Information sources for data set comparison algorithm ... 88
Table 13 —Updates for state decision code M1 and M2 ... 91
Table 14 —Updates for state decision code M3 ... 91
Table 15 —Updates for state decision code P1, and P2 ... 91
Table 16 —Updates for state decision code S1 ... 92
Table 17 —Source identity comparisons ... 95
Table 18 —Common message header ... 124
Table 19 —Values of messageType field ... 125
Table 20 —Values of flagField ... 126
Table 21 —correctionField semantics ... 127
Table 22 —References for sequenceId value exceptions ... 127
Table 23 —controlField enumeration ... 128
Table 24 —Values of logMessageInterval field ... 128
Table 25 —Announce message fields ... 129
Table 26 —Sync and Delay_Req message fields ... 130
Table 27 —Follow_Up message fields ... 130
Table 28 —Delay_Resp message fields ... 130
Table 29 —Pdelay_Req message fields ... 131
Table 30 —Pdelay_Resp message fields ... 131
Table 31 —Pdelay_Resp_Follow_Up message fields ... 132
Table 32 —Acceptance of signaling messages ... 132
Table 33 —Signaling message fields ... 133

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Table 34 —tlvType values ... 134
Table 35 —Organization specific TLV fields ... 135
Table 36 —Acceptance of management messages ... 136
Table 37 —Management message fields ... 137
Table 38 —Values of the actionField ... 138
Table 39 —Management TLV fields ... 139
Table 40 —managementId values ... 140
Table 41 —CLOCK_DESCRIPTION management TLV data field ... 141
Table 42 —clockType specification ... 142
Table 43 —USER_DESCRIPTION management TLV data field ... 144
Table 44 —INITIALIZE management TLV data field ... 145
Table 45 —INITIALIZATION_KEY enumeration ... 145
Table 46 —Fault log severityCode enumeration ... 145
Table 47 —FAULT_LOG management TLV data field ... 146
Table 48 —TIME management TLV data field ... 147
Table 49 —CLOCK_ACCURACY management TLV data field ... 147
Table 50 —DEFAULT_DATA_SET management TLV data field ... 148
Table 51 —PRIORITY1 management TLV data field ... 148
Table 52 —PRIORITY2 management TLV data field ... 149
Table 53 —DOMAIN management TLV data field ... 149
Table 54 —SLAVE_ONLY management TLV data field ... 149
Table 55 —CURRENT_DATA_SET management TLV data field ... 149
Table 56 —PARENT_DATA_SET management TLV data field ... 150
Table 57 —TIME_PROPERTIES_DATA_SET management TLV data field ... 151
Table 58 —UTC_PROPERTIES management TLV data field ... 152
Table 59 —TRACEABILITY_PROPERTIES management TLV data field ... 152
Table 60 —TIMESCALE_PROPERTIES management TLV data field ... 152
Table 61 —PORT_DATA_SET management TLV data field ... 153
Table 62 —LOG_ANNOUNCE_INTERVAL management TLV data field ... 154
Table 63 —ANNOUNCE_RECEIPT_TIMEOUT management TLV data field ... 154
Table 64 —LOG_SYNC_INTERVAL management TLV data field ... 154
Table 65 —DELAY_MECHANISM management TLV data field ... 155
Table 66 —LOG_MIN_PDELAY_REQ_INTERVAL management TLV data field ... 155
Table 67 —VERSION_NUMBER management TLV data field ... 155
Table 68 —TRANSPARENT_CLOCK_DEFAULT_DATA_SET management TLV data field ... 156

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Table 69 —PRIMARY_DOMAIN management TLV data field ... 156
Table 70 —TRANSPARENT_CLOCK_PORT_DATA_SET management TLV data field ... 157
Table 71 —MANAGEMENT_ERROR_STATUS TLV format ... 157
Table 72 —managementErrorId enumeration ... 158
Table 73 —REQUEST_UNICAST_TRANSMISSION TLV format ... 160
Table 74 —GRANT_UNICAST_TRANSMISSION TLV format ... 161
Table 75 —CANCEL_UNICAST_TRANSMISSION TLV format ... 161
Table 76 —ACKNOWLEDGE_CANCEL_UNICAST_TRANSMISSION TLV format ... 162
Table 77 —UNICAST_NEGOTIATION_ENABLE management TLV data field ... 162
Table 78 —PATH_TRACE TLV format ... 164
Table 79 —PATH_TRACE_LIST management TLV data field ... 164
Table 80 —PATH_TRACE_ENABLE management TLV data field ... 164
Table 81 —ALTERNATE_TIME_OFFSET_INDICATOR TLV format ... 166
Table 82 —ALTERNATE_TIME_OFFSET_ENABLE management TLV data field ... 167
Table 83 —ALTERNATE_TIME_OFFSET_NAME management TLV data field ... 167
Table 84 —ALTERNATE_TIME_OFFSET_MAX_KEY management TLV data field ... 168
Table 85 —ALTERNATE_TIME_OFFSET_PROPERTIES management TLV data field ... 168
Table 86 —GRANDMASTER_CLUSTER_TABLE management TLV data field ... 171
Table 87 —Alternate master attributes ... 173
Table 88 —ALTERNATE_MASTER management TLV data field ... 173
Table 89 —UNICAST_MASTER_TABLE management TLV data field ... 174
Table 90 —UNICAST_MASTER_MAX_TABLE_SIZE management TLV data field ... 175
Table 91 —Operation of acceptable master table option ... 176
Table 92 —ACCEPTABLE_MASTER_TABLE management TLV data field ... 176
Table 93 —ACCEPTABLE_MASTER_MAX_TABLE_SIZE management TLV data field ... 177
Table 94 —ACCEPTABLE_MASTER_TABLE_ENABLED management TLV data field ... 177
Table 95 —Version 1 stratum to version 2 class ... 178
Table 96 —Version 2 clockClass to version 1 stratum ... 178
Table 97 —Version 1 to version 2 translation of grandmasterIsPreferred field ... 179
Table 98 —Version 2 to version 1 translation of the priority1 field ... 179
Table 99 —Version 1 clock identifier to version 2 clockAccuracy ... 179
Table 100 —Version 2 clockAccuracy to version 1 clock identifier ... 179
Table 101 —Version 1 to version 2 translation of grandmasterIsBoundaryClock field ... 180
Table 102 —Version 2 to version 1 translation of the priority2 field ... 180
Table 103 —Version 1 control field and version 2 messageType field mappings ... 180
Table 104 —Translation of flagField from version 1 to version 2 ... 181

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Table 105 —Translation of flagField from version 2 to version 1 ... 181
Table 106 —Version 2 fields with no version 1 counterpart ... 182
Table 107 —Version 1 fields with no version 2 counterpart ... 183
Table 108 —Name correspondence ... 183
Table 109 —Mixed system restrictions ... 184
Table B.1—Relationships between timescales ... 199
Table C.1—Interpretation of Figure C.1 key values ... 202
Table C.2—Interpretation of Figure C.2 key values ... 203
Table C.3—Interpretation of Figure C.3 key values ... 205
Table C.4—Interpretation of Figure C.4 key values ... 207
Table C.5—Interpretation of Figure C.5 key values ... 209
Table C.6—Interpretation of Figure C.6 key values ... 210
Table C.7—Interpretation of Figure C.7 key values ... 211
Table C.8—Interpretation of Figure C.8 key values ... 213
Table C.9—Interpretation of Figure C.9 key values ... 215
Table C.10—Interpretation of Figure C.10 key values ... 217
Table C.11—Interpretation of Figure C.11 key values ... 218
Table D.1—IPv4 multicast addresses ... 219
Table D.2—transportSpecific field values ... 220
Table E.1—IPv6 multicast addresses ... 222
Table F.1—Multicast MAC addresses ... 223
Table F.2—Ethernet transport specific field ... 224
Table G.1—DeviceNet clockIdentity octets 0 through 7 ... 226
Table G.2—DeviceNet headers for all PTP message packets ... 226
Table H.1—ControlNet clockIdentity octets 2 through 7 ... 228
Table I.1—Mapping of messages ... 231
Table I.2—IEEE 802.3 DLPDU syntax ... 232
Table I.3—Multicast MAC address ... 233
Table I.4—LT (Length/Type) ... 234
Table I.6—Mapping of the parameter and attribute names ... 235
Table I.7—Translation of flagField from PTP version 2 to PROFINET ... 236
Table K.1—flagField.SECURE flag ... 242
Table K.2—AUTHENTICATION TLV ... 260
Table K.3—algorithmId values ... 261
Table K.4—ICV and pad length ... 261
Table K.5—AUTHENTICATION_CHALLENGE TLV ... 262

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Table K.6—challengeType values 262
Table K.7—SECURITY_ASSOCIATION_UPDATE TLV 263
Table K.8—addressType values 263
Table L.1—CUM_FREQ_SCALE_FACTOR_OFFSET TLV format 266

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

FIGURES

Figure 1 —Mealy state transition diagram ... 10
Figure 2 —Model of an ordinary clock ... 19
Figure 3 —Model of a boundary clock ... 21
Figure 4 —Model of an end-to-end transparent clock ... 22
Figure 5 —End-to-end residence time correction model ... 23
Figure 6 —Combined ordinary and end-to-end transparent clock ... 25
Figure 7 —Model of a peer-to-peer transparent clock ... 27
Figure 8 —Peer-to-peer residence time and link delay correction model ... 28
Figure 9 —Combined ordinary and peer-to-peer transparent clock. ... 30
Figure 10 —Simple master–slave clock hierarchy ... 32
Figure 11 —Pruned mesh topology ... 33
Figure 12 —Basic synchronization message exchange ... 34
Figure 13 —Link delay measurement ... 35
Figure 14 —Timestamp generation model ... 36
Figure 15 —Hierarchical topology ... 37
Figure 16 —Linear topology ... 38
Figure 17 —Multiply connected topology ... 39
Figure 18 —Bridging disparate technologies ... 40
Figure 19 —Definition of latency constants ... 45
Figure 20 —Propagation asymmetry ... 47
Figure 21 —Port model ... 48
Figure 22 —Scaled log variance hysteresis ... 60
Figure 23 —State machine for a full implementation ... 78
Figure 24 —State machine for a slave-only implementation ... 79
Figure 25 —STATE_DECISION_EVENT logic ... 81
Figure 26 —State decision algorithm ... 87
Figure 27 —Data set comparison algorithm, part 1 ... 89
Figure 28 —Data set comparison algorithm, part 2 ... 90
Figure 29 —Receipt of Announce message logic ... 96
Figure 30 —Receipt of Sync message logic ... 98
Figure 31 —Receipt of Follow_Up message logic ... 100
Figure 32 —Receipt of Delay_Req message logic ... 101
Figure 33 —Receipt of Delay_Resp message logic ... 103
Figure 34 —Delay request-response path length measurement ... 110

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Figure 35 —Peer delay link measurement ... 113

Figure 36 —Permitted mixed system configuration ... 183

Figure 37 —Profile print form ... 186

Figure C.1—Master, end-to-end, and slave one-step clocks; no asymmetry correction ... 201

Figure C.2—Master, end-to-end, and slave one-step clocks; with asymmetry correction ... 202

Figure C.3—Master two-step and end-to-end transparent and slave one-step clocks; with asymmetry correction ... 204

Figure C.4—Master and end-to-end transparent, two-step, and one-step, slave clocks; with asymmetry correction ... 206

Figure C.5—One-step master, two-step end-to-end transparent, and one-step slave clocks; with asymmetry correction ... 208

Figure C.6—One-step peer responder, end-to-end transparent, and peer requestor clocks; with asymmetry correction ... 210

Figure C.7—One-step peer responder, two-step end-to-end transparent, and one-step peer requestor clocks; with asymmetry correction ... 211

Figure C.8—Two-step peer responder, two-step end-to-end transparent, and one-step peer requestor clocks: option 1; with asymmetry correction ... 212

Figure C.9—Two-step peer responder, two-step end-to-end transparent, and one-step peer requestor clocks: option 2; with asymmetry correction ... 214

Figure C.10—Two-step peer responder, one-step end-to-end transparent, and one-step peer requestor clocks: option 2; with asymmetry correction ... 216

Figure C.11—One-step peer master, two-step peer-to-peer transparent, and one-step peer slave clocks; time computation ... 217

Figure G.1— Event message timestamp point ... 225

Figure I.1—PROFINET region combined with domains ... 231

Figure K.1—PTP secure message processing ... 252

Figure K.2—Challenge processing ... 254

Figure K.3—Secure transmit processing ... 256

Figure K.4—Secure event processing ... 258

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

INTERNATIONAL ELECTROTECHNICAL COMMISSION

PRECISION CLOCK SYNCHRONIZATION PROTOCOL FOR NETWORKED MEASUREMENT AND CONTROL SYSTEMS

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
6) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC/IEEE 61588 has been processed through subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation.

This second edition cancels and replaces the first edition published in 2004. It constitutes a technical revision.

The text of this standard is based on the following documents:

IEEE Std	FDIS	Report on voting
1588 (2008)	65C/510/FDIS	65C/517/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

The committee has decided that the contents of this publication will remain unchanged until 2013.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

IEC/IEEE Dual Logo International Standards

This Dual Logo International Standard is the result of an agreement between the IEC and the Institute of Electrical and Electronics Engineers, Inc. (IEEE). The original IEEE Standard was submitted to the IEC for consideration under the agreement, and the resulting IEC/IEEE Dual Logo International Standard has been published in accordance with the ISO/IEC Directives.

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.

Use of an IEC/IEEE Dual Logo International Standard is wholly voluntary. The IEC and IEEE disclaim liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEC or IEEE Standard document.

The IEC and IEEE do not warrant or represent the accuracy or content of the material contained herein, and expressly disclaim any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEC/IEEE Dual Logo International Standards documents are supplied “AS IS”.

The existence of an IEC/IEEE Dual Logo International 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 IEC/IEEE Dual Logo International Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard.

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

In publishing and making this document available, the IEC and IEEE are not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Neither the IEC nor IEEE is undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEC/IEEE Dual Logo International Standards or IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations – Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments for revision of IEC/IEEE Dual Logo International Standards are welcome from any interested party, regardless of membership affiliation with the IEC or IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA and/or General Secretary, IEC, 3, rue de Varembé, PO Box 131, 1211 Geneva 20, Switzerland.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

NOTE – 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 with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems

Sponsor

Technical Committee on Sensor Technology (TC-9)

of the

IEEE Instrumentation and Measurement Society

Approved 27 March 2008

IEEE-SA Standards Board

Abstract: A protocol is provided in this standard that enables precise synchronization of clocks in measurement and control systems implemented with technologies such as network communication, local computing, and distributed objects. The protocol is applicable to systems communicating via packet networks. Heterogeneous systems are enabled that include clocks of various inherent precision, resolution, and stability to synchronize. System-wide synchronization accuracy and precision in the sub-microsecond range are supported with minimal network and local clock computing resources. Simple systems are installed and operated without requiring the management attention of users because the default behavior of the protocol allows for it.

Keywords: boundary clock, clock, distributed system, master clock, measurement and control system, real-time clock, synchronized clock, transparent clock

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

IEEE Introduction

This standard defines a protocol enabling precise synchronization of clocks in measurement and control systems implemented with technologies such as network communication, local computing, and distributed objects. The clocks communicate with each other over a communication network. The protocol generates a master–slave relationship among the clocks in the system. All clocks ultimately derive their time from a clock known as the grandmaster clock. In its basic form, this protocol is intended to be administration free.

History

Measurement and control applications are increasingly using distributed system technologies such as network communication, local computing, and distributed objects. Without a standardized protocol for synchronizing the clocks in these devices, it is unlikely that the benefits will be realized in the multivendor system component market. Existing protocols for clock synchronization are not optimum for these applications. For example, Network Time Protocol (NTP) targets large distributed computing systems with millisecond synchronization requirements. The protocol proposed in this standard specifically addresses the following needs of measurement and control systems:

— Spatially localized

— Microsecond to sub-microsecond accuracy and precision

— Administration free

— Accessible for both high-end devices and low-cost, low-end devices

Notice to users

Laws and regulations

Users of these documents should consult all applicable laws and regulations. Compliance with the provisions of this standard does not imply 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.

Copyrights

This document is copyrighted by the IEEE. It is made available 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 this document available for use and adoption by public authorities and private users, the IEEE does not waive any rights in copyright to this document.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Updating of IEEE documents

Users of IEEE standards 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. 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 the IEEE Standards Association Web site at http://ieeexplore.ieee.org/xpl/standards.jsp, or contact the IEEE at the address listed previously.

For more information about the IEEE Standards Association or the IEEE standards development process, visit the IEEE-SA Web site at http://standards.ieee.org.

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: http://standards.ieee.org/reading/ieee/updates/errata/. Users are encouraged to check this URL for errata periodically.

Interpretations

Current interpretations can be accessed at the following URL: http://standards.ieee.org/reading/ieee/interp/.

Patents

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 with respect to the existence or validity of any patent rights in connection therewith. A patent holder or patent applicant has filed a statement of assurance that it will grant licenses under these 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. Other Essential Patent Claims may exist for which a statement 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 are reasonable or non-discriminatory. Further information may be obtained from the IEEE Standards Association.

IMPORTANT NOTICE: This standard is not intended to assure safety, security, health, or environmental protection in all circumstances. Implementers of the standard are responsible for determining appropriate safety, security, environmental, and health practices or regulatory requirements.

This IEEE document is made available for use subject to important notices and legal disclaimers. These notices and disclaimers appear in all publications containing this document and may be found under the heading "Important Notice" or "Important Notices and Disclaimers Concerning IEEE Documents." They can also be obtained on request from IEEE or viewed at http://standards.ieee.org/IPR/disclaimers.html.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

PRECISION CLOCK SYNCHRONIZATION PROTOCOL FOR NETWORKED MEASUREMENT AND CONTROL SYSTEMS

1. Overview

1.1 Scope

This standard defines a protocol enabling precise synchronization of clocks in measurement and control systems implemented with technologies such as network communication, local computing, and distributed objects. The protocol is applicable to systems communicating by local area networks supporting multicast messaging including, but not limited to, Ethernet. The protocol enables heterogeneous systems that include clocks of various inherent precision, resolution, and stability to synchronize to a grandmaster clock. The protocol supports system-wide synchronization accuracy in the sub-microsecond range with minimal network and local clock computing resources. The default behavior of the protocol allows simple systems to be installed and operated without requiring the administrative attention of users. The standard includes mappings to User Datagram Protocol (UDP)/Internet Protocol (IP), DeviceNet, and a layer-2 Ethernet implementation. It includes formal mechanisms for message extensions, higher sampling rates, correction for asymmetry, a clock type to reduce error accumulation in large topologies, and specifications on how to incorporate the resulting additional data into the synchronization protocol. The standard permits synchronization accuracies better than 1 ns. The protocol has features to address applications where redundancy and security are a requirement. The standard defines conformance and management capability. There is provision to support unicast as well as multicast messaging. The standard includes an annex on recommended practices. Annexes defining communication-medium-specific implementation details for additional network implementations are expected to be provided in future versions of this standard.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

1.2 Purpose

Measurement and control applications are increasingly employing distributed system technologies such as network communication, local computing, and distributed objects. Many of these applications will be enhanced by having an accurate system-wide sense of time achieved by having local clocks in each sensor, actuator, or other system device. Without a standardized protocol for synchronizing these clocks, it is unlikely that the benefits will be realized in the multivendor system component market. Existing protocols for clock synchronization are not optimum for these applications. For example, the Network Time Protocol (NTP) targets large distributed computing systems with millisecond synchronization requirements. The protocol in this standard specifically addresses the needs of measurement and control and operational systems in the fields of test and measurement, industrial automation, military systems, manufacturing systems, power utility systems, and certain telecommunications applications. These applications need:

— Spatially localized systems with options for larger systems

— Microsecond to sub-microsecond accuracy

— Administration-free operation

— Applicability for both high-end devices and low-cost, low-end devices

— Provisions for the management of redundant and fault-tolerant systems

Several different application areas such as industrial automation, telecommunication, semiconductor manufacturing, military systems, and utility power generation have emerged that require the standard to be revised.

1.3 Layout of the document

This standard, which defines the Precision Time Protocol (PTP), is divided into 19 clauses:

Clause	Purpose
1	Provides the scope and benefits of this standard
2	Lists references to other standards
3	Provides definitions that are either not found in other standards or have been modified for use with this standard
4	Provides conventions for the notation used in this standard
5	Defines the data types used in this standard
6	Provides an overview of PTP
7	Defines characteristics of PTP entities
8	Defines PTP data sets
9	Defines PTP for ordinary and boundary clocks
10	Defines PTP for transparent clocks
11	Specifies PTP time computations and corrections
12	Specifies how to syntonize and synchronize clocks
13	Defines the format of messages passed between participating clocks
14	Specifies type, length, value (TLV) formats
15	Specifies management TLVs
16	Defines general optional features of this standard
17	Defines state configuration options of this standard
18	Defines forward and backward compatibility between versions
19	Defines requirements for conformance

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Annexes are provided as follows:

Annex	Purpose
Annex A	Using PTP
Annex B	Defines timescales and epochs in PTP
Annex C	Examples of timing computations and message fields
Annex D	Defines mappings of PTP to User Datagram Protocol (UDP) over Internet Protocol version 4 (IPv4)
Annex E	Defines mappings of PTP to UDP over Internet Protocol version 6 (IPv6)
Annex F	Defines mappings of PTP over IEEE 802.3
Annex G	Defines mappings of PTP to DeviceNet™[1]
Annex H	Defines mappings of PTP to ControlNet™[2]
Annex I	Defines mappings of PTP to PROFINET™[3]
Annex J	Default PTP Profile
Annex K	Defines an experimental security option
Annex L	Defines an experimental cumulative frequency TLV
Annex M	Bibliography

2. Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

IEC 61158-3-2:2007, Industrial communication networks—Fieldbus specifications—Part 3-2: Data-link layer service definition—Type 2 elements.[4]

IEC 61158-4-2:2007, Industrial communication networks—Fieldbus specifications—Part 4-2: Data-link layer protocol specification—Type 2 elements.

IEC 61158-5-2:2007, Industrial communication networks—Fieldbus specifications—Part 5-2: Application layer service definition—Type 2 elements.

IEC 61158-5-10:2007, Industrial communication networks—Fieldbus specifications—Part 5-10: Application layer service definition—Type 10 elements.

IEC 61158-6-2:2007, Industrial communication networks—Fieldbus specifications—Part 6-2: Application layer protocol specification—Type 2 elements.

IEC 61158-6-10:2007, Industrial communication networks—Fieldbus specifications—Part 6-10: Application layer protocol specification—Type 10 elements.

---

[1]DeviceNet™ is a trade name of Open DeviceNet Vendor Association, Inc. This information is given for the convenience of users of this standard and does not constitute an endorsement by the IEEE or IEC of these products. Equivalent products may be used if they can be shown to lead to the same results.

[2]ControlNet™ is a trade name of ControlNet International, Ltd. This information is given for the convenience of users of this standard and does not constitute an endorsement by the IEEE or IEC of these products. Equivalent products may be used if they can be shown to lead to the same results.

[3]PROFINET™ is the trade name of the non-profit organization PROFIBUS Nutzerorganisation e.V. (PNO). This information is given for the convenience of users of this standard and does not constitute an endorsement by the IEEE or IEC of these products. Equivalent products may be used if they can be shown to lead to the same results.

[4]IEC publications are available from the Sales Department of the International Electrotechnical Commission, Case Postale 131, 3, rue de Varembé, CH-1211, Genève 20, Switzerland/Suisse (http://www.iec.ch/). IEC publications are also available in the United States from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (http://www.ansi.org/).

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

IEC 61784-1:2007, Industrial communication networks—Profiles—Part 1: Fieldbus profiles.

IEC 61784-2:2007, Industrial communications networks—Profiles—Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3.

IEC 62026-3:2008, Low-voltage switchgear and controlgear—Controller-device interfaces (CDIs)—Part 3: DeviceNet.

IEEE Std 802®, IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture.[5, 6]

IEEE Std 802.1AB™, IEEE Standard for Local and Metropolitan Area Networks—Part 1AB: Station and Media Access Control Connectivity Discovery.

IEEE Std 802.1Q™-2005, IEEE Standard for Local and Metropolitan Area Networks—Part 1Q: Virtual bridged local area networks.

IEEE Std 802.3™-2005, IEEE Standard for Information Technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and Physical Layer specifications.

ISO/IEC 10646:2003, Information technology—Universal Multiple-Octet Coded Character Set (UCS)[7].

3. Definitions, acronyms, and abbreviations

3.1 Definitions

For the purposes of this document, the following terms and definitions apply.

3.1.1 accuracy: The mean of the time or frequency error between the clock under test and a perfect reference clock, over an ensemble of measurements. Stability is a measure of how the mean varies with respect to variables such as time, temperature, and so on. The precision is a measure of the deviation of the error from the mean.

3.1.2 atomic process: A process is atomic if the values of all inputs to the process are not permitted to change until all of the results of the process are instantiated, and the outputs of the process are not visible to other processes until the processing of each output is complete.

3.1.3 boundary clock: A clock that has multiple Precision Time Protocol (PTP) ports in a domain and maintains the timescale used in the domain. It may serve as the source of time, i.e., be a master clock, and may synchronize to another clock, i.e., be a slave clock.

3.1.4 clock: A node participating in the Precision Time Protocol (PTP) that is capable of providing a measurement of the passage of time since a defined epoch.

[5]IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards/ieee.org/).
[6] The IEEE standards or products referred to in this clause are trademarks of the Institute of Electrical and Electronics Engineers, Inc.
[7]ISO/IEC publications are available from the ISO Central Secretariat, 1 chemin de la Voie-Creuse, CP 56, CH-1211, Genève 20, Switzerland/Suisse (http://www.iso.ch/). ISO/IEC publications are also available in the United States from Global Engineering Documents, 15 Inverness Way East, Englewood, Colorado 80112, USA (http://global.ihs.com/). Electronic copies are available in the United States from the American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (http://www.ansi.org/).

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

NOTE—In the case of PTP ordinary and boundary clocks that are properly synchronized, the epoch is the epoch of the timescale in use. In the case of PTP transparent clocks, the epoch is locally defined and not necessarily aligned with the timescale.[8]

3.1.5 default: When applied to attribute values and options means the configuration of a Precision Time Protocol (PTP) device as it is delivered from the manufacturer.

3.1.6 direct communication: A communication of Precision Time Protocol (PTP) information between two ordinary or boundary clocks with no intervening boundary clock.

3.1.7 domain: A logical grouping of clocks that synchronize to each other using the protocol, but that are not necessarily synchronized to clocks in another domain.

3.1.8 end-to-end transparent clock: A transparent clock that supports the use of the end-to-end delay measurement mechanism between slave clocks and the master clock.

3.1.9 epoch: The origin of a timescale.

3.1.10 event: An abstraction of the mechanism by which signals or conditions are generated and represented.

3.1.11 foreign master: An ordinary or boundary clock sending Announce messages to another clock that is not the current master recognized by the other clock.

3.1.12 fractional frequency offset: The fractional frequency offset (FFO) between a measured frequency and a reference frequency is defined as follows:

$$\mathrm{FFO} = \frac{(\mathrm{FO} - \mathrm{FR})}{\mathrm{FR}}$$

where FO is the measured frequency and FR is the reference frequency.

3.1.13 grandmaster clock: Within a domain, a clock that is the ultimate source of time for clock synchronization using the protocol.

3.1.14 holdover: A clock previously synchronized/syntonized to another clock (normally a primary reference or a master clock) but now free-running based on its own internal oscillator, whose frequency is being adjusted using data acquired while it had been synchronized/syntonized to the other clock. It is said to be in holdover or in the holdover mode, as long as it is within its accuracy requirements.

3.1.15 link: A network segment between two Precision Time Protocol (PTP) ports supporting the peer delay mechanism of this standard. The peer delay mechanism is designed to measure the propagation time over such a link.

3.1.16 management node: A device that configures and monitors clocks.

3.1.17 master clock: In the context of a single Precision Time Protocol (PTP) communication path, a clock that is the source of time to which all other clocks on that path synchronize.

3.1.18 message timestamp point: A point within a Precision Time Protocol (PTP) event message serving as a reference point in the message. A timestamp is defined by the instant a message timestamp point passes the reference plane of a clock.

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

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

3.1.19 multicast communication: A communication model in which each Precision Time Protocol (PTP) message sent from any PTP port is capable of being received and processed by all PTP ports on the same PTP communication path.

3.1.20 node: A device that can issue or receive Precision Time Protocol (PTP) communications on a network.

3.1.21 one-step clock: A clock that provides time information using a single event message.

3.1.22 ordinary clock: A clock that has a single Precision Time Protocol (PTP) port in a domain and maintains the timescale used in the domain. It may serve as a source of time, i.e., be a master clock, or may synchronize to another clock, i.e., be a slave clock.

3.1.23 parent clock: The master clock to which a clock is synchronized.

3.1.24 peer-to-peer transparent clock: A transparent clock that, in addition to providing Precision Time Protocol (PTP) event transit time information, also provides corrections for the propagation delay of the link connected to the port receiving the PTP event message. In the presence of peer-to-peer transparent clocks, delay measurements between slave clocks and the master clock are performed using the peer-to-peer delay measurement mechanism.

3.1.25 phase change rate: The observed rate of change in the measured time with respect to the reference time. The phase change rate is equal to the fractional frequency offset between the measured frequency and the reference frequency. *See also:* **fractional frequency offset.**

3.1.26 portNumber: An index identifying a specific Precision Time Protocol (PTP) port on a PTP node.

3.1.27 precision: *See:* **accuracy**.

3.1.28 Precision Time Protocol (PTP): The protocol defined by IEEE Std 1588-2008/IEC 61588:2009. As an adjective, it indicates that the modified noun is specified in or interpreted in the context of IEEE Std 1588-2008/IEC 61588:2009.

3.1.29 primary reference: A source of time and or frequency that is traceable to international standards. *See also:* **traceability.**

3.1.30 profile: The set of allowed Precision Time Protocol (PTP) features applicable to a device.

3.1.31 Precision Time Protocol (PTP) communication: Information used in the operation of the protocol, transmitted in a PTP message over a PTP communication path.

3.1.32 Precision Time Protocol (PTP) communication path: The signaling path portion of a particular network enabling direct communication among ordinary and boundary clocks.

3.1.33 Precision Time Protocol (PTP) message: One of the message types defined in IEEE Std 1588-2008/ IEC 61588:2009.

3.1.34 Precision Time Protocol (PTP) node: A PTP ordinary, boundary, or transparent clock or a device that generates or parses PTP messages.

3.1.35 Precision Time Protocol (PTP) port: A logical access point of a clock for PTP communications to the communications network.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

3.1.36 recognized standard time source: A recognized standard time source is a source external to Precision Time Protocol (PTP) that provides time and/or frequency as appropriate that is traceable to the international standards laboratories maintaining clocks that form the basis for the International Atomic Time (TAI) and Universal Coordinated Time (UTC) timescales. Examples of these are Global Positioning System (GPS), NTP, and National Institute of Standards and Technology (NIST**)** timeservers.

3.1.37 requestor: The port implementing the peer-to-peer delay mechanism that initiates the mechanism by sending a Pdelay_Req message.

3.1.38 responder: The port responding to the receipt of a Pdelay_Req message as part of the operation of the peer-to-peer delay mechanism.

3.1.39 stability: *See:* **accuracy.**

3.1.40 synchronized clocks: Two clocks are synchronized to a specified uncertainty if they have the same epoch and their measurements of the time of a single event at an arbitrary time differ by no more than that uncertainty.

3.1.41 syntonized clocks: Two clocks are syntonized if the duration of the second is the same on both, which means the time as measured by each advances at the same rate. They may or may not share the same epoch.

3.1.42 timeout: A mechanism for terminating requested activity that, at least from the requester's perspective, does not complete within the specified time.

3.1.43 timescale: A linear measure of time from an epoch.

3.1.44 traceability: A property of the result of a measurement or the value of a standard whereby it can be related to stated references, usually national or international standards, through an unbroken chain of comparisons all having stated uncertainties. (adapted from the ISO/IEC Guide 99:2007. [B15][9])

3.1.45 translation device: A boundary clock or, in some cases, a transparent clock that translates the protocol messages between regions implementing different transport and messaging protocols, between different versions of IEEE Std 1588-2008/IEC 61588:2009, or different Precision Time Protocol (PTP) profiles.

3.1.46 transparent clock: A device that measures the time taken for a Precision Time Protocol (PTP) event message to transit the device and provides this information to clocks receiving this PTP event message. *See also:* **end-to-end transparent clock**; **peer-to-peer transparent clock**.

3.1.47 two-step clock: A clock that provides time information using the combination of an event message and a subsequent general message. *See also:* **one-step clock**.

3.2 Acronyms and abbreviations

ARB	arbitrary
BMC	best master clock
CAN	Controller Area Network
CP	Communication Profile [according to IEC 61784-1:2007[10]]
CPF	Communication Profile Family [according to IEC 61784-1:2007]
DS	differentiated service
E2E	end-to-end

[9]The numbers in square brackets correspond to those of the bibliography in Annex M.
[10]Information on references can be found in Clause 2.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

GPS	Global Positioning System
IANA	Internet Assigned Numbers Authority
ICV	integrity check value
ID	identification
IPv4, IPv6	Internet Protocol version 4 / 6
JD	Julian Date
JDN	Julian Day Number
MAC	media access control [according to IEEE Std 802.3-2005]
MJD	Modified Julian Day
NIST	National Institute of Standards and Technology (see www.nist.gov)
NTP	Network Time Protocol (see IETF RFC 1305 [B7])
OUI	organizational unique identifier (allocated by the IEEE)
	NOTE—The OUI is typically used in specifications or the implementation of devices for the purpose of identification. It identifies the organization that owns the OUI-dependent subidentifier and may not necessarily be the organization that defines the specification or provides the hardware. The IEEE OUI listing can be obtained at http://standards.ieee.org/regauth/oui/index.shtml.
P2P	peer-to-peer
PHY	physical layer [according to IEEE Std 802.3-2005]
POSIX	Portable Operating System Interface (see ISO/IEC 9945:2003 [B16])
PPS	pulse per second
PTP	Precision Time Protocol
SA	security associations
SNTP	Simple Network Time Protocol
SOF	Start of Frame
TAI	International Atomic Time
TC	traffic class
TLV	type, length, value [according to IEEE Std 802.1AB]
ToS	type of service
UCMM	UnConnect Message Manager
UDP/IP	User Datagram Protocol (see IETF RFC 768 [B5]) / Internet Protocol (see IETF RFC 791 [B6])
UTC	Coordinated Universal Time

4. Conventions

4.1 Descriptive lexical form syntax

4.1.1 Lexical form syntax

A lexical form refers to:

a) A name

b) A data type

The conventions illustrated in the following list regarding lexical forms are used in this standard:

— Type names: e.g., ClockQuality (no word separation, initial letter of each word capitalized)

— Enumeration members and global constants: e.g., ATOMIC_CLOCK (underscore word separation, all letters capitalized)

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

— Fields within messages, instances of structures, and variables: e.g., secondsField, clockQuality, clockIdentity (two-word field names at a minimum, no word separation, initial word not capitalized, initial letter capitalization on subsequent words)

— Members of a structure: e.g., clockQuality.clockClass (structure name followed by a period followed by the member name)

— Data set name: e.g., defaultDS, parentDS, portDS, currentDS, timePropertiesDS (no word separation, initial letter of each word not capitalized followed by the letters DS)

— Data set members: e.g., defaultDS.clockQuality.clockClass (Data set name followed by a period followed a type name followed by a period followed by the variable name)

— Message names: e.g., Sync, Delay_Req (underscore word separation, initial letter of each word capitalized)

— <localNameForSomething>: text enclosed in angle brackets < > is used where the standard needs to refer to something whose syntax or lexical form is dependent on the local implementation and language

When a lexical form appears in text, as opposed to in a type, or a format definition, the form is to be interpreted as singular, plural, or possessive as appropriate to the context of the text.

4.2 Word usage

4.2.1 Shall

The word “shall,” which is equivalent to “is required to,” is used to indicate mandatory requirements, strictly to be followed in order to conform to the standard and from which no deviation is permitted.

4.2.2 Recommended

The word “recommended” is used to indicate flexibility of choice with a strong preference alternative.

4.2.3 Must

The word “must” indicates an unavoidable situation.

4.2.4 Should

The word “should,” which is equivalent to “is recommended that,” is used to indicate:

— Among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others

— That a certain course of action is preferred but not required

— That (in the negative form) a certain course of action is deprecated but not prohibited

4.2.5 May

The word “may,” which is equivalent to “is permitted,” is used to indicate a course of action permissible within the limits of the standard.

4.2.6 Can

The word “can,” which is equivalent to “is able to,” is used to indicate possibility and capability, whether material or physical.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

4.2.7 Optional

Clauses and text marked optional are not required to be implemented. If the option is implemented, then all specifications of the clause or text, respectively, shall be implemented according to this standard.

NOTE— This definition is recursive, which means that options within options obey these same rules.

4.2.8 Reserved

The word "reserved" indicates:

— If used in an assignment of values to an enumeration or an attribute that the values indicated are reserved for use in future editions of this standard and shall not be used for any other purpose.

— If used in a field of a message that the field is reserved for use in future editions of this standard. The field shall be present in the message with the size specified. No interpretation of reserved fields is to be made for this edition of this standard, and the fields shall not be used for any other purpose.

4.3 Behavioral specification notation

State transition diagrams are used to specify behavioral characteristics as illustrated in Figure 1. Each state transition diagram is composed of the following components:

— Named boxes, representing states

— Directed arrows, indicating transitions from one state to the next.

Each transition is labeled with:

a) The enabling event or predicate label for a transition

b) The transition action label for a transition

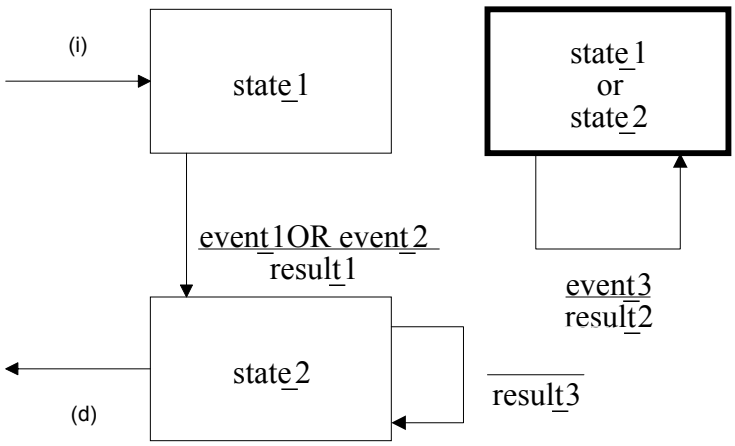

Figure 1—Mealy state transition diagram

The notation used describes state transition diagrams using the Mealy style, where actions are associated with the transition from one state to another.

Events, for example "event_1," "event_2," and "event_3," identify the inputs to the state machine. They can be operation requests and responses, or internal occurrences such as timer expirations.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Predicates, for example "event_1 OR event_2," identify enabling conditions for transitions. The first predicate encountered, evaluated from left to right, which is TRUE, selects the transition to execute and therefore the next state.

Transition actions, for example "result_1," are the actions that are executed before transitioning to the next state.

The next state identifies the state for the state machine after the selected transition action completes. The value of the current state changes as the transition to the next state occurs.

A bold line for a state box indicates that the box represents multiple states. Any transition shown that begins and terminates in such a state box indicates that there has been no change in state.

Transitions, for example the transition resulting in result_3, which have no indicated enabling conditions, occur via unspecified mechanisms. Unless otherwise stated, in PTP, the events giving rise to these mechanisms are implementation-specific and outside the scope of the standard.

A transition into a state machine, for example "(i)," is indicated by a transition arrow that has no source state. A transition out of a state machine, for example "(d)," is indicated by a transition arrow with no destination state.

NOTE—For example, As a result of either event_1 or event_2 becoming TRUE, state_1 is replaced with the value of the next state. In this example, the next state is state_2, which is specified as the name of the state box that is the target of the transition arrow. Before the transition, result_1 occurs. "event_3" can occur in either state_1 or state_2. The state is unchanged, but an action, result_2, occurs as the result of event_3.

5. Data types and on-the-wire formats in a PTP system

5.1 General

The data types specified for the various PTP variables and message fields define logical properties that are necessary for correct operation of the protocol or interpretation of PTP message content.

Implementations are free to use any internal representation of PTP data types; however, the internal representation shall not change the semantics of any quantity visible via PTP communications or the semantics of any specified operation of the protocol.

5.2 Primitive data type specifications

All nonprimitive PTP data types are derived from the primitive types listed in Table 1. These types are not tied to any specific programming language. The essential properties of each type shall be as follows:

— Integer: All integer types are of finite length as indicated by the number associated with each, e.g., UInteger48, and as signed or unsigned as indicated by the absence or presence of a leading U. Numbers with these data types obey the laws of arithmetic within the range represented by the length. Arithmetic operations are treated as modulo the capacity of the data type; e.g., the sum of two UInteger48 values is computed modulo 2^{+48}. Signed integers are represented in two's complement form.

— Enumeration: All enumerations are of finite field length as indicated by the number associated with each, e.g., Enumeration4. Unless otherwise stated in this standard, the only interpretations of the bit pattern in the enumeration field are the associations between the bit patterns and the assigned meanings of the enumeration.

— Boolean: The only interpretation is as logical values within the context of Boolean algebra.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

— Nibble and Octet: These are 4 and 8 bit fields, respectively. The only interpretations are those explicitly defined within this standard.

Table 1—Primitive PTP data types

Data type	Definition
Boolean	TRUE or FALSE.
Enumeration4	4-bit enumerated value
Enumeration8	8-bit enumerated value
Enumeration16	16-bit enumerated value
UInteger4	4-bit unsigned integer
Integer8	8-bit signed integer
UInteger8	8-bit unsigned integer
Integer16	16-bit signed integer
UInteger16	16-bit unsigned integer
Integer32	32-bit signed integer
UInteger32	32-bit unsigned integer
UInteger48	48-bit unsigned integer
Integer64	64-bit signed integer
Nibble	4-bit field not interpreted as a number
Octet	8-bit field not interpreted as a number

5.3 Derived data type specifications

5.3.1 General

Arrays of any of the primitive data types are represented in the format <data type>[lengthField] <label>, where <lengthField> indicates the number of instances of the data type in the array and <label> is the lexical name for the array data type so defined.

Structures consisting of an ordered list of members are indicated with the following syntax:

```
struct <StructureName>
{
    <DataType1> <memberName1>;

    <DataType2> <memberName2>;

    …
};
```

where <StructureName> is the lexical name for the data type so defined, <DataType1> is the data type of the first member, <memberName1> is the lexical name of the first member, and so forth.

The syntax typedef <DataType> <TypeName> is interpreted as defining a derived data type with the same properties as the data type defined by <DataType> but with a new name given by <TypeName>.

The syntax typedef <DataType> [lengthField] <TypeName> is interpreted as defining a derived data type consisting of an array of elements of type <DataType>, but with a new name given by <TypeName>.

5.3.2 TimeInterval

The TimeInterval type represents time intervals.

```
struct TimeInterval
{
                    Integer64 scaledNanoseconds;
};
```

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

The scaledNanoseconds member is the time interval expressed in units of nanoseconds and multiplied by 2^{+16}.

Positive or negative time intervals outside the maximum range of this data type shall be encoded as the largest positive and negative values of the data type, respectively.

For example, 2.5 ns is expressed as $0000\ 0000\ 0002\ 8000_{16}$.

5.3.3 Timestamp

The Timestamp type represents a positive time with respect to the epoch.

```
struct Timestamp
{
        UInteger48 secondsField;
        UInteger32 nanosecondsField;
};
```

The secondsField member is the integer portion of the timestamp in units of seconds.

The nanosecondsField member is the fractional portion of the timestamp in units of nanoseconds.

The nanosecondsField member is always less than 10^9.

For example:

+2.000000001 seconds is represented by secondsField = $0000\ 0000\ 0002_{16}$ and nanosecondsField= $0000\ 0001_{16}$.

5.3.4 ClockIdentity

The ClockIdentity type identifies a clock.

```
typedef Octet[8] ClockIdentity;
```

5.3.5 PortIdentity

The PortIdentity type identifies a PTP port.

```
struct PortIdentity
{
        ClockIdentity clockIdentity;
        UInteger16 portNumber;
};
```

5.3.6 PortAddress

The PortAddress type represents the protocol address of a PTP port.

```
struct PortAddress
{
        Enumeration16 networkProtocol;
        UInteger16 addressLength;
        Octet[addressLength] addressField;
};
```

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

The value of the networkProtocol member shall be taken from the networkProtocol enumeration; see 7.4.1. The addressLength is the length in octets of the address. The range shall be 1 to 16 octets.

The addressField member holds the protocol address of a port in the format defined by the mapping annex of the protocol as identified by the networkProtocol member. The most significant octet of the addressField is mapped into the octet of the addressField member with index 0.

5.3.7 ClockQuality

The ClockQuality represents the quality of a clock.

```
struct ClockQuality
{
        UInteger8 clockClass;
        Enumeration8 clockAccuracy;
        UInteger16 offsetScaledLogVariance;
};
```

5.3.8 TLV

The TLV type represents TLV extension fields.

```
struct TLV
{
        Enumeration16 tlvType;
        UInteger16 lengthField;
        Octet[lengthField ] valueField;
};
```

The length of all TLVs shall be an even number of octets.

5.3.9 PTPText

The PTPText data type is used to represent textual material in PTP messages.

```
struct PTPText
{
        UInteger8 lengthField;
        Octet[lengthField] textField;
};
```

The textField member shall be encoded as UTF-8 symbols as specified by ISO/IEC 10646:2003. The most significant octet of the leading text symbol shall be the element of the array with index 0.

NOTE—A single UTF-8 symbol can be 1–4 octets long. Therefore, the lengthField value can be larger than the number of symbols.

5.3.10 FaultRecord

The FaultRecord type is used to construct fault logs.

```
struct FaultRecord
{
    UInteger16 faultRecordLength;
    Timestamp faultTime;
    Enumeration8 severityCode;
    PTPText faultName;
    PTPText faultValue;
```

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

```
        PTPText faultDescription;
};
```

The faultRecordLength member shall indicate the number of octets in the FaultRecord not including the 2 octets of the faultRecordLength member.

5.4 On-the-wire formats

5.4.1 General

PTP protocol data units consist of the PTP messages defined in Clause 13 through Clause 17 based on the data types defined in 5.2, 5.3, and 17.2. The internal ordering of the fields of the PTP protocol data units is specified in 5.4.2 to 5.4.4.

5.4.2 Primitive data types

Numeric primitive data types defined in 5.2 shall be formatted with the most significant octet nearest the beginning of the protocol data unit followed in order by octets of decreasing significance.

The Boolean data type TRUE shall be formatted as a single bit equal to 1 and FALSE as a single bit equal to 0.

Enumerations of whatever length shall be formatted as though the assigned values are unsigned integers of the same length; e.g., Enumeration16 shall be formatted as though the value had type UInteger16.

5.4.3 Arrays of primitive types

All arrays shall be formatted with the member having the lowest numerical index closest to the start of the protocol data unit followed by successively higher numbered members. In octet arrays, the octet with the lowest numerical index is termed the most significant octet.

When a field containing more than one octet is used to represent a numeric value, the most significant octet shall be nearest the start of the protocol data unit, followed by successively less significant octets.

When a single octet contains multiple fields of primitive data types, the bit positions within the octet of each of the primitive types as defined in the message field specification shall be preserved. For example, the first field of the header of the PTP messages is a single octet composed of two fields, one of type Nibble bits 4-7, and one of type Enumeration4 bits 0-3 (see 13.3.1).

5.4.4 Derived data types

Derived data types defined as structs shall be formatted with the first member of the struct closest to the beginning of the protocol data unit followed by each succeeding member. Each member shall be formatted according to its data type.

Derived data types defined as typedefs shall be formatted according to its referenced data type.

5.4.5 Mapping of PTP protocol data units into their on-the-wire formats

Unless otherwise specified, PTP protocol data units shall be mapped to and from their on-the-wire format based on the rules of the underlying physical layer transport. Any exceptions are specified in one of the transport-specific annexes of this standard or in an applicable PTP profile.

NOTE—PTP protocol mechanisms operate in the upper layers of the protocol stack (i.e., PTP is an “application” that uses the services of the network or link layer). The physical layer transport dictates the on-the-wire format.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

6. Clock synchronization model

6.1 General

This clause provides a model for understanding the operation of the Precision Time Protocol. The exact specifications of these interactions are found in subsequent clauses.

The PTP standard specifies a clock synchronization protocol. This protocol is applicable to distributed systems consisting of one or more nodes, communicating over a network. Nodes are modeled as containing a real-time clock that may be used by applications within the node for various purposes such as generating timestamps for data or ordering events managed by the node. The protocol provides a mechanism for synchronizing the clocks of participating nodes to a high degree of accuracy and precision. This standard specifies:

a) The Precision Time Protocol

b) The node, system, and communication properties necessary to support PTP.

6.2 Principle assumptions about the network and implementation recommendations

The following principle assumptions and recommendations should be followed in order to ensure correct operation of the protocol. These are discussed in greater detail in later clauses:

a) PTP assumes that the network eliminates cyclic forwarding of PTP messages within each communication path (e.g., by using a spanning tree protocol). PTP eliminates cyclic forwarding of PTP messages between communication paths.

b) PTP is tolerant of an occasional missed message, duplicated message, or message that arrived out of order. However, PTP assumes that such impairments are relatively rare.

c) PTP was designed assuming a multicast communication model. PTP also supports a unicast communication model as long as the behavior of the protocol is preserved. PTP assumes that Announce messages are periodically sent by one port and delivered to all other ports of ordinary or boundary clocks within a communication path. If the communication path consists of more than two ports, the assumption is that Announce messages are either sent in multicast or the Announce information is replicated to all ports in the communication path using unicast messages. PTP ports discover other ports within a communication path through the receipt of multicast Announce messages. When multicast communication is not available, another form of discovery (e.g., by configuration) is required; see for example 17.5. PTP management messages sent to all ports also require either multicast messaging or replication of the management message to all ports within the communication path.

d) Like all message-based time transfer protocols, PTP time accuracy is degraded by asymmetry in the paths taken by event messages; see 7.4.2. Specifically the time offset error is 1/2 of the asymmetry. Asymmetry is not detectable by PTP; however, if known, PTP corrects for asymmetry. Asymmetry can be introduced in the physical layer, e.g., via transmission media asymmetry, by bridges and routers, and in large systems by the forward and reverse paths traversed by event messages taking different routes through the network. Systems should be configured and components selected to minimize these effects guided by the required timing accuracy. In single subnet systems with distances of a few meters, asymmetry is not usually a concern for time accuracies above a few 10s of ns.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

e) If two-step clocks are used, then the network has to be designed such that the general message takes the same path as the event message through a transparent clock. Failure to do this will result in a condition where the transparent clock does not calculate path delay properly. This condition is undetectable and may introduce additional jitter and wander, but it will not break the protocol.

f) PTP assumes that the number of boundary clocks forming the master–slave synchronization hierarchy from the grandmaster clock to any slave clock is less than 255; see 9.3.2.5.

g) Network components, for example bridges, introduce timing jitter and wander that if uncorrected can degrade time transfer accuracy. Since the jitter and wander are often traffic dependent, the network traffic patterns should be designed to minimize the traffic and to minimize the variation in the traffic load. It is also recommended that PTP event messages be sent in high priority compared with other data; see A.5.3.3. Whenever possible such devices should be replaced by PTP boundary or transparent clocks.

h) The network's protocol is structured such that a message timestamp point can be defined.

6.3 PTP systems

A PTP system is a distributed, networked system consisting of a combination of PTP and non-PTP devices. PTP devices include ordinary clocks, boundary clocks, end-to-end transparent clocks, peer-to-peer transparent clocks, and management nodes. Non-PTP devices include bridges, routers, and other infrastructure devices, and possibly devices such as computers, printers, and other application devices.

The protocol is a distributed protocol that specifies how the real-time clocks in the system synchronize with each other. These clocks are organized into a master–slave synchronization hierarchy with the clock at the top of the hierarchy—the grandmaster clock—determining the reference time for the entire system. The synchronization is achieved by exchanging PTP timing messages, with the slaves using the timing information to adjust their clocks to the time of their master in the hierarchy.

Devices in a PTP system communicate with each other via a communication network. The network may include translation devices between segments implementing different network communication protocols.

The protocol executes within a logical scope called a domain. Unless otherwise specified, all PTP messages, data sets, state machines, and all other PTP entities are always associated with a particular domain. A given physical network and individual devices connected to the network can be associated with multiple domains. Within this standard, the time established within one domain by the protocol is independent of the time in other domains.

6.4 PTP message classes

The protocol defines event and general PTP messages. Event messages are timed messages in that an accurate timestamp is generated at both transmission and receipt as specified in 6.6.5. General messages do not require accurate timestamps.

The set of event messages consists of:

a) Sync (see 13.6)

b) Delay_Req (see 13.6)

c) Pdelay_Req (see 13.9)

d) Pdelay_Resp (see 13.10)

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

The set of general messages consists of:

— Announce (see 13.5)

— Follow_Up (see 13.7)

— Delay_Resp (see 13.8)

— Pdelay_Resp_Follow_Up (see 13.11)

— Management (see Clause 15)

— Signaling (see 13.12)

The Sync, Delay_Req, Follow_Up, and Delay_Resp messages are used to generate and communicate the timing information needed to synchronize ordinary and boundary clocks using the delay request-response mechanism.

The Pdelay_Req, Pdelay_Resp, and Pdelay_Resp_Follow_Up messages are used to measure the link delay between two clock ports implementing the peer delay mechanism. The link delay is used to correct timing information in Sync and Follow_Up messages in systems composed of peer-to-peer transparent clocks. Ordinary and boundary clocks that implement the peer delay mechanism can synchronize using the measured link delays and the information in the Sync and Follow_Up messages.

The Announce message is used to establish the synchronization hierarchy.

The management messages are used to query and update the PTP data sets maintained by clocks. These messages are also used to customize a PTP system and for initialization and fault management. Management messages are used between management nodes and clocks.

The signaling messages are used for communication between clocks for all other purposes. For example, signaling messages can be used for negotiation of the rate of unicast messages between a master and its slaves.

All messages can be extended by means of a standard type, length, value (TLV) extension mechanism. For example, the PATH_TRACE message extensions can be used to detect rogue frames; see 16.2.1 for more detail on rogue frames.

6.5 PTP device types

6.5.1 General

There are five basic types of PTP devices, as follows:

a) Ordinary clock

b) Boundary clock

c) End-to-end transparent clock

d) Peer-to-peer transparent clock

e) Management node

All five types implement one or more aspects of the protocol.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

Two mechanisms are used in PTP to measure the propagation delay between PTP ports. The first one, the delay request-response mechanism, uses the messages Sync, Delay_Req, Delay_Resp, and if required, Follow_Up; see 11.3. The second one, the peer delay mechanism, uses the messages Pdelay_Req, Pdelay_Resp, and if required, Pdelay_Resp_Follow_Up; see 11.4. Ports on ordinary and boundary clocks can be implemented using either mechanism. Ports on end-to-end transparent clocks are independent of these mechanisms. Ports on peer-to-peer transparent clocks use the peer delay mechanism. These two mechanisms do not interwork on the same communication path. In addition, the peer delay mechanism is restricted to topologies where each peer-to-peer port communicates PTP messages with at most one other such port; see 11.4.4.

The use of the various clock types is limited as follows:

— Ordinary and boundary clocks with ports implementing the peer delay mechanism, and peer-to-peer transparent clocks, can only be connected in topologies where ports implementing the peer delay mechanism communicate PTP messages to and from a single port also implementing the peer delay mechanism; see 11.4.4. Except in carefully designed networks, this will preclude the use of end-to-end transparent clocks and bridges that do not support PTP using the peer delay mechanism, e.g., a conventional bridge.

— Ordinary and boundary clock ports implementing the delay request-response mechanism, and end-to-end transparent clocks, can be connected in any topology that excludes ports using the peer delay mechanism. This precludes the use of peer-to-peer transparent clocks in such a system.

— A boundary clock with ports supporting each of the two mechanisms may be used to bridge between regions supporting the different mechanisms.

6.5.2 Ordinary clocks

The model of an ordinary clock is illustrated in Figure 2.

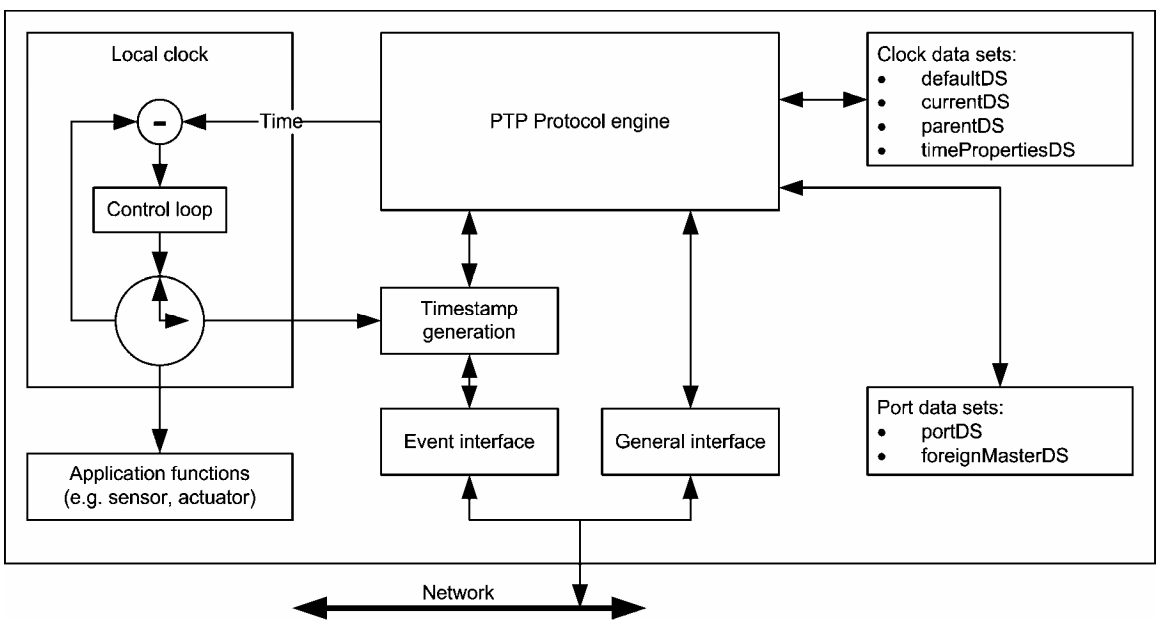

Figure 2—Model of an ordinary clock

An ordinary clock communicates with the network via two logical interfaces based on a single physical port. The event interface is used to send and receive event messages, which are timestamped by the timestamp generation block based on the value of the local clock. The general interface is used to send and

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub
Full document details: IEC 61588 2009

receive general messages. An ordinary clock in a domain supports a single copy of the protocol and has a single PTP state. The ordinary clock can be the grandmaster clock in a system, or it can be a slave clock in the master–slave hierarchy.

An ordinary clock maintains two types of data sets, referred to as clock data sets and port data sets, respectively.

The clock data sets are as follows:

a) defaultDS: Attributes describing the ordinary clock.

b) currentDS: Attributes related to synchronization.

c) parentDS: Attributes describing the parent (the clock to which the ordinary clock synchronizes) and the grandmaster (the clock at the root of the master–slave hierarchy).

d) timePropertiesDS: Attributes of the timescale.

The port data sets contain attributes of the port, including the PTP state.

The protocol engine:

— Sends and receives PTP messages.

— Maintains the data sets.

— Executes the state machine associated with the port.

— If the port is in the slave state (synchronizes to a master), it computes the master's time based on the received PTP timing messages and timestamps that were generated.

The control loop in the local clock adjusts the clock to agree with the time of its master if the ordinary clock port is in the slave state. If the port is in the master state, the local clock is free running or possibly synchronized to an external source of time such as the Global Positioning System (GPS). If the port is in the master state and the ordinary clock is the grandmaster clock of the domain, then the local clock is typically synchronized to an external source of time traceable to TAI and UTC such as the GPS system.

In some applications such as industrial automation, an ordinary clock may also be associated with an application device such as a sensor or an actuator. In a telecommunications application, an ordinary clock may be associated with a timing demarcation device. The local clock provides time support to such a device as illustrated in Figure 2.

6.5.3 Boundary clocks

The model of a boundary clock is illustrated in Figure 3.

The boundary clock typically has several physical ports with each physical port communicating with the network via two logical interfaces: event and general. Each port of a boundary clock is like the port of an ordinary clock with the following exceptions:

a) The clock data sets are common to all ports of the boundary clock.

b) The local clock is common to all ports of the boundary clock.

c) Each protocol engine has the additional function of resolving the states of all ports to determine which port provides the time signal used to synchronize the local clock.

The messages related to synchronization, establishing the master–slave hierarchy, and signaling (see 6.4), terminate in the protocol engine of a boundary clock and are not forwarded. Management messages are

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

forwarded by other ports on the boundary clock subject to restrictions to limit the propagation of these messages within the system.

The boundary clock model of Figure 3 is applicable only to PTP messages. For all non-PTP messages, the boundary clock behaves as a normal network component, e.g., bridge, repeater, or router.

A boundary clock is typically used only as a network element and is not normally associated with application devices such as sensors or actuators.

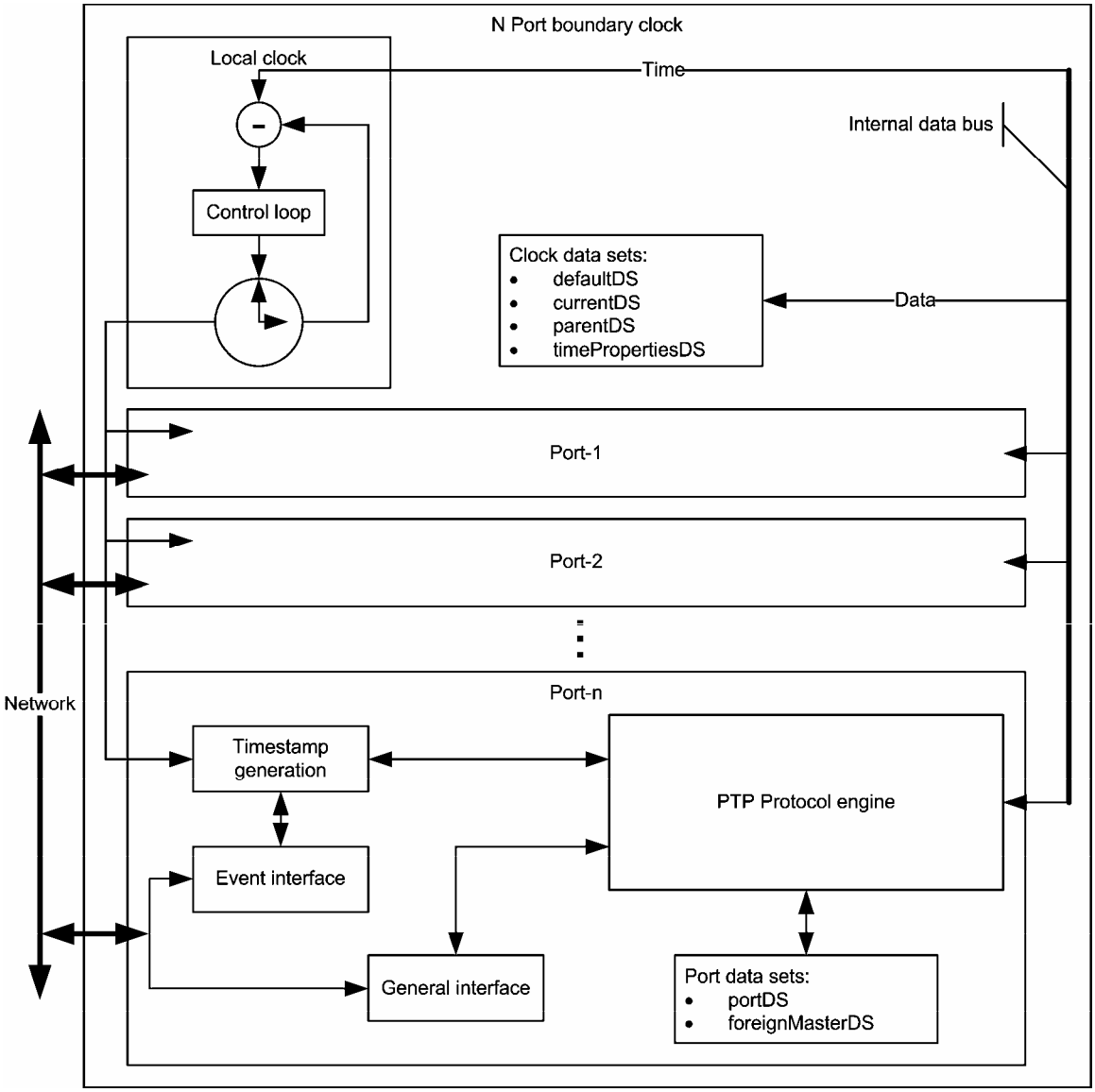

Figure 3—Model of a boundary clock

6.5.4 End-to-end transparent clocks

The model of an end-to-end transparent clock is illustrated in Figure 4.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub
Full document details: IEC 61588 2009

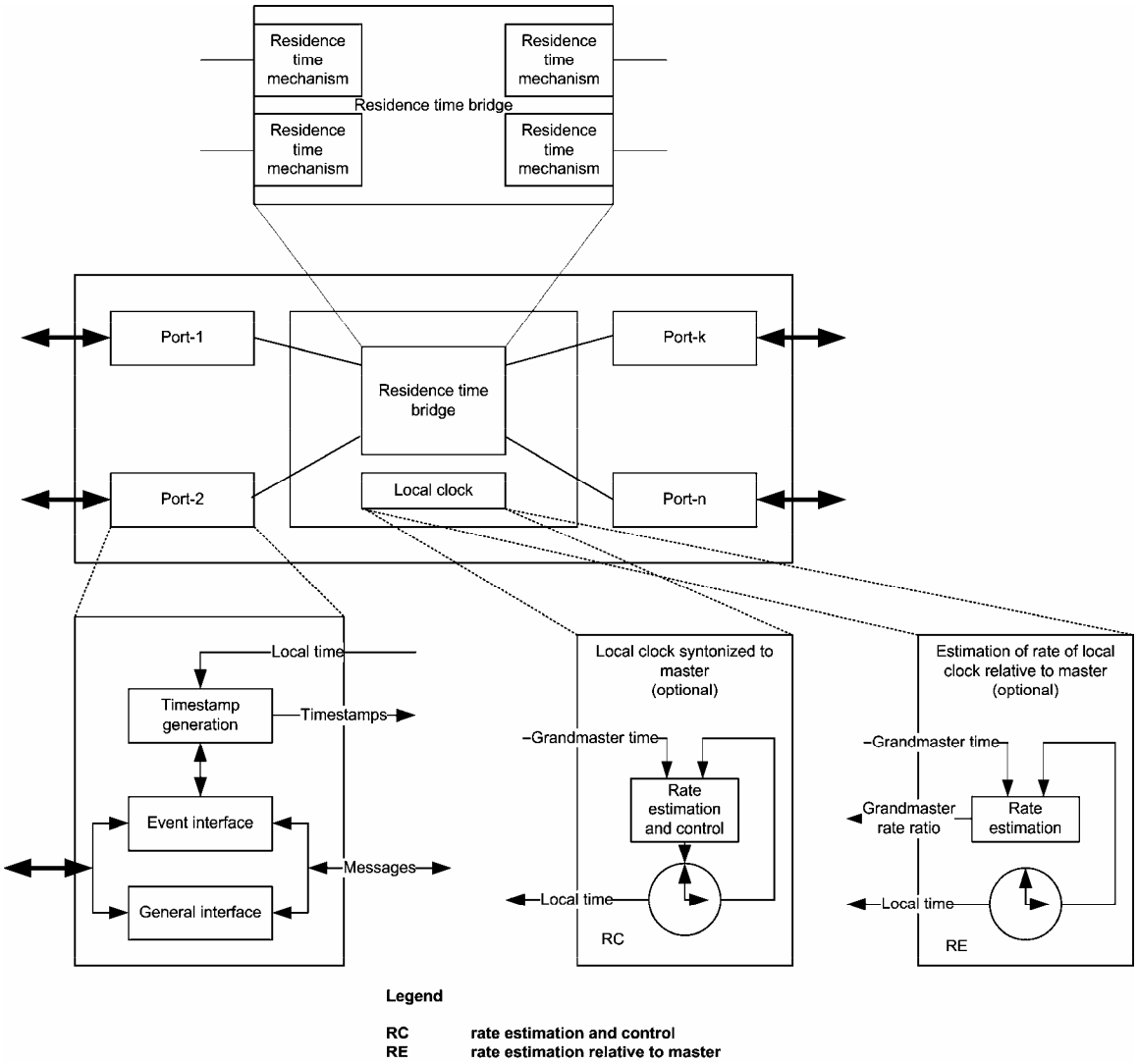

Figure 4—Model of an end-to-end transparent clock

The end-to-end transparent clock forwards all messages just as a normal bridge, router, or repeater. However, for PTP event messages, the residence time bridge, shown in Figure 4, measures the residence time of PTP event messages (the time the message takes to traverse the transparent clock). These residence times are accumulated in a special field, the correctionField, of the PTP event message or the associated follow-up message (Follow_Up or Pdelay_Resp_Follow_Up). This correction is based on the difference in the timestamp generated when the event message enters and leaves the transparent clock. Any updates to checksums required by the network protocol are made. Note that the value of the correction update and checksums are specific to each output port and message since the residence times are not necessarily the same for all paths through the transparent clock or for successive messages on the same path. The correction process is illustrated for an arbitrary pair of ports in Figure 5.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

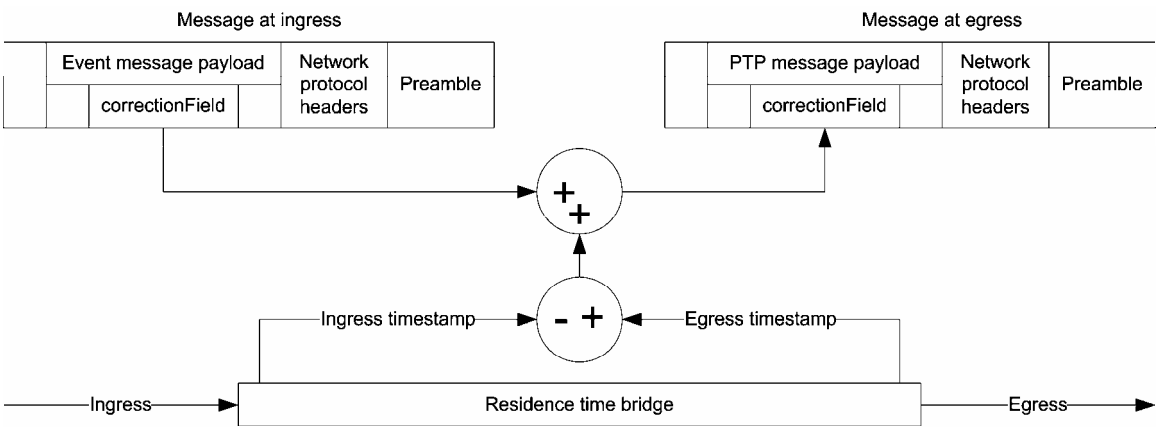

Figure 5—End-to-end residence time correction model

The timestamps used in computing the residence time are based on timestamps generated from the local clock. Since these accumulated residence times are used by a slave to adjust the time provided by a master, it is important that any errors resulting from differences in the rates of the master and the transparent clocks be negligible for the accuracy required by the application. Since it is possible that the rates of the master and local clocks (essentially the definition of the second on the two clocks) can differ by 0.02%, the error introduced can be 0.02% of the measured residence time. Thus, for a residence time of 1 ms, the maximum error is 200 ns. This error may be unacceptably large, and there are several possible ways of reducing it.

One way, the method discussed extensively in 11.3, is to make the rate of the local clock equal to that of the master, i.e., syntonize the local clock to the master. This is done by observing the timing information, corrected for any upstream residence times, in received Sync and, if present, Follow_Up messages. This is illustrated in Figure 4 in block RC (rate control). The grandmaster time, corrected for any upstream residence times, is input to the Rate estimation and control block in RC (shown by the arrow labeled "Grandmaster time"). The corresponding local time is also input to this block. The Rate estimation and control block uses the sequences of master and corresponding local times to estimate the ratio of the master and local clock rates. The Rate estimation and control block then uses this estimated rate ratio to adjust (i.e., control) the local clock rate. Note that the adjustment of the local clock rate need not be a physical changing of the oscillator frequency (i.e., an analog implementation); it is equally acceptable to use a fixed frequency local oscillator, compute the ratio of the master and local rates, and multiply the timestamps taken with the local clock by this ratio (i.e., a digital implementation). The key aspect of the scheme illustrated by RC is that it operates closed loop; i.e., the local times used in the rate estimation and residence time measurement are relative to the rate-adjusted local oscillator. This means that rate adjustments to an oscillator at one node influence the adjustments at downstream nodes, because the residence time is used to adjust the master times at downstream nodes.

A second way to reduce the effect of residence time error that may be useful in some applications is to use free-running local clocks. This is illustrated in the block RE (rate estimation) in Figure 4. In this case, the frequency of the local oscillator is not adjusted, but it is allowed to free-run. Each clock measures the local residence time of the Sync message based on the free-running clock. This measured residence time is added to the correctionField of the Sync or Follow_Up message as discussed in 11.3. Each local clock also computes the ratio of the rate of the local free-running clock to the rate of the grandmaster. This ratio is computed based on the timestamps in received Sync and Follow_Up messages corrected by adding the correctionField. Even though the correctionField does not reflect the exact residence time, the resulting ratio will still have greatly reduced jitter compared with simply using the raw timestamps, and it will still have the correct mean value. In addition, if the residence time is sufficiently large relative to the rate ratio measurement interval, the worst-case phase error accumulation over a succession of transparent clocks will be less than that using the first method (above), but at the expense of greater phase accumulation for many non-worst-case scenarios. The local value of the rate ratio must be maintained in each clock for use in correcting the residence times for Delay_Req messages, as discussed below. In addition, each local clock

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub
Full document details: IEC 61588 2009

computes the difference between the residence time corrected by this ratio and the uncorrected residence times. These differences must also be accumulated and passed to the slave clock. If this method is to be used, it should be specified in a profile (see 19.3) along with a profile-specific TLV used for the accumulation. With this method, the slave must add these accumulated differences to the correctionField of received Sync or Follow_Up messages before computing the offset of the slave clock from the grandmaster. End-to-end transparent clocks also correct for the residence time of Delay_Req messages. However, since with this method the oscillators are free-running, the measured value of the residence time of the Delay_Req message must be corrected using the ratio of the local clock rate to that of the grandmaster that was computed and saved as mentioned earlier. This is the only change in the operation of the clock for the Delay_Req messages. With this technique, Pdelay_Resp messages are treated exactly like Sync messages and must have a TLV specified for this purpose. The Pdelay_Req messages are handled exactly like the Delay_Req messages.

NOTE 1—Before deciding which method to use, it is important to evaluate the likely errors compared with the application error budget. For example, if the total residence time over all clocks between a master and slave is 10 ms, then the maximum error in not correcting for the local clock rate at each intermediate clock is no worse than 200 ppm of this value, or 2 µs. For many designs, the total residence time will be considerably less, and either mechanism is likely to produce satisfactory results.

NOTE 2—Implementers of PTP nodes need to consider the resources required to support multiple domains. The inability to process the protocol in a timely fashion due to resource limitations may lead to deterioration in the synchronization performance, thrashing, or failure of the protocol. Users need to be aware of this limitation when selecting nodes and designing their systems.

An end-to-end transparent clock may be used as a network element, or it may be associated with application devices such as sensors or actuators. In the latter case, an ordinary clock is combined with the transparent clock to provide real-time support for the application device. A model of such a combined device is illustrated in Figure 6.

In Figure 6, when the ordinary clock is a slave, the timing information includes the ingress timestamp. When it is a master, it includes the egress timestamp. The residence time bridge delivers the incoming PTP timing messages, the Announce messages, the ingress timestamp generated by the incoming Sync message, and any internal timing corrections to the protocol engine of the ordinary clock. The protocol engine computes the correct time based on this information and sends it as input to the local clock. If the ordinary clock were a master, it would originate Sync and Follow_Up messages with the message timestamps referenced to the local clock of the ordinary clock, and based on internal timing corrections and the egress timestamp. In practice the end-to-end transparent and ordinary clock functions would share a common physical clock.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

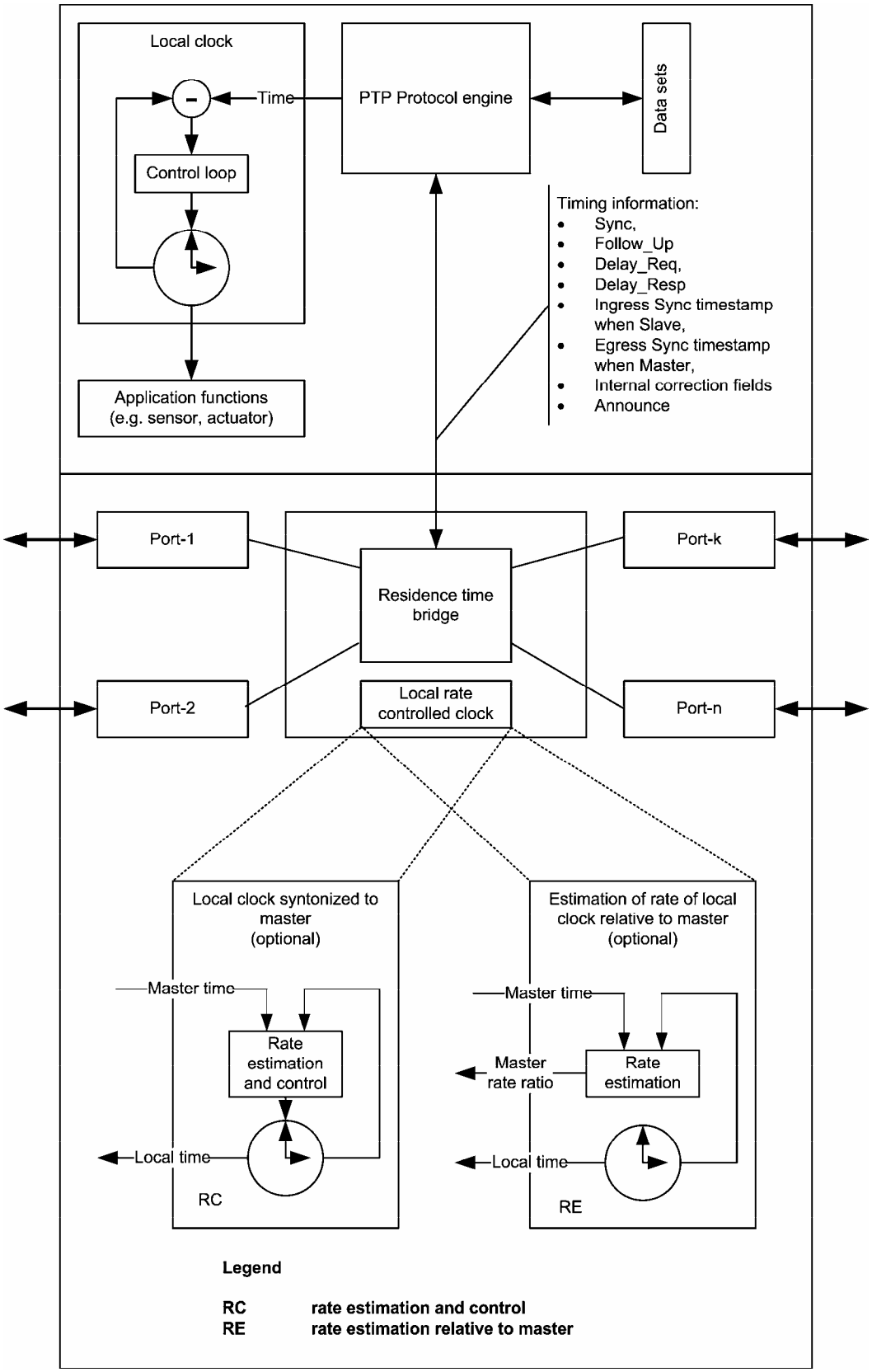

Figure 6—Combined ordinary and end-to-end transparent clock

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub
Full document details: IEC 61588 2009

6.5.5 Peer-to-peer transparent clocks

The model of a peer-to-peer transparent clock is illustrated in Figure 7.

The peer-to-peer transparent clock differs from the end-to-end transparent clock in the way it corrects and handles the PTP timing messages. In all other respects, it is identical with the end-to-end transparent clock.

The peer-to-peer transparent clock has an additional per port block as indicated in Figure 7. This block is used to compute the link delay between each port and a similarly equipped port on another node sharing the link, i.e., the link peer. The link peer exists in another clock supporting the peer delay mechanism since non-peer-to-peer devices are not expected between peer-to-peer transparent clocks. The computation of link delay is based on an exchange of Pdelay_Req, Pdelay_Resp, and possibly Pdelay_Resp_Follow_Up messages with the link peer. As a result of these exchanges, the link delay is known for each port of the peer-to-peer transparent clock. Unlike the end-to-end transparent clock, which corrects and forwards all PTP timing messages, the peer-to-peer transparent clock only corrects and forwards Sync and Follow_Up messages. The appropriate correctionField in these messages is updated for both the residence time of the Sync message within the peer-to-peer transparent clock and the link delay on the port receiving the Sync message.

This correction process is illustrated in Figure 8. The egress PTP timing message contains the residence time and link delay corrections for the actual ingress link traversed by a Sync packet. A change in Sync packet routing through a system of peer-to-peer clocks may result in the Sync packet entering the peer-to-peer transparent clock from a different link. Since the peer-to-peer transparent clock corrects the residence and path times based on the actual link and internal path taken by each Sync packet, the egress timing information is always correct to within the accuracy of the measurements. Therefore, the timing information provided to the slave always reflects the actual path through the network of peer-to-peer transparent clocks. By contrast, in the case of the end-to-end corrections, the slave waits for a new path delay value based on the combined Sync and Delay_Resp messages, which takes more time since the messages need to traverse the entire new path before correct values are present at the slave.

The measurement of path delay using the peer delay mechanism does not interwork with path delay measurements based on the delay request-response mechanism. As a result, a peer-to-peer transparent clock can only work with clock ports supporting the peer delay mechanism. With peer-to-peer transparent clocks, the master only needs to issue Sync and Follow_Up messages and respond to Pdelay_Req messages. It does not receive Delay_Req messages, and therefore, the processing workload is independent of the number of peer-to-peer clocks served by a given port. Likewise, end-to-end transparent clocks support only the delay request-response mechanism and not the peer delay mechanism. If a network contains peer-to-peer transparent clocks in one region and end-to-end transparent clocks in another region, the regions can be connected only through a boundary clock.

The timestamps used in computing the residence and link delay times are based on timestamps generated from the local clocks. Since these accumulated residence and link delay times are used by a slave to adjust the time provided by a master, it is important that any errors resulting from differences in the rates of the master and the transparent clocks be negligible for the accuracy required by the application. Since it is possible that the rates of the master and local clocks (essentially the definition of the second on the two clocks) can differ by 0.02%, the error introduced can be 0.02% of the measured residence time. Thus, for a residence time of 1 ms, the maximum error is 200 ns. This error may be unacceptably large, and there are several possible ways of reducing it.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

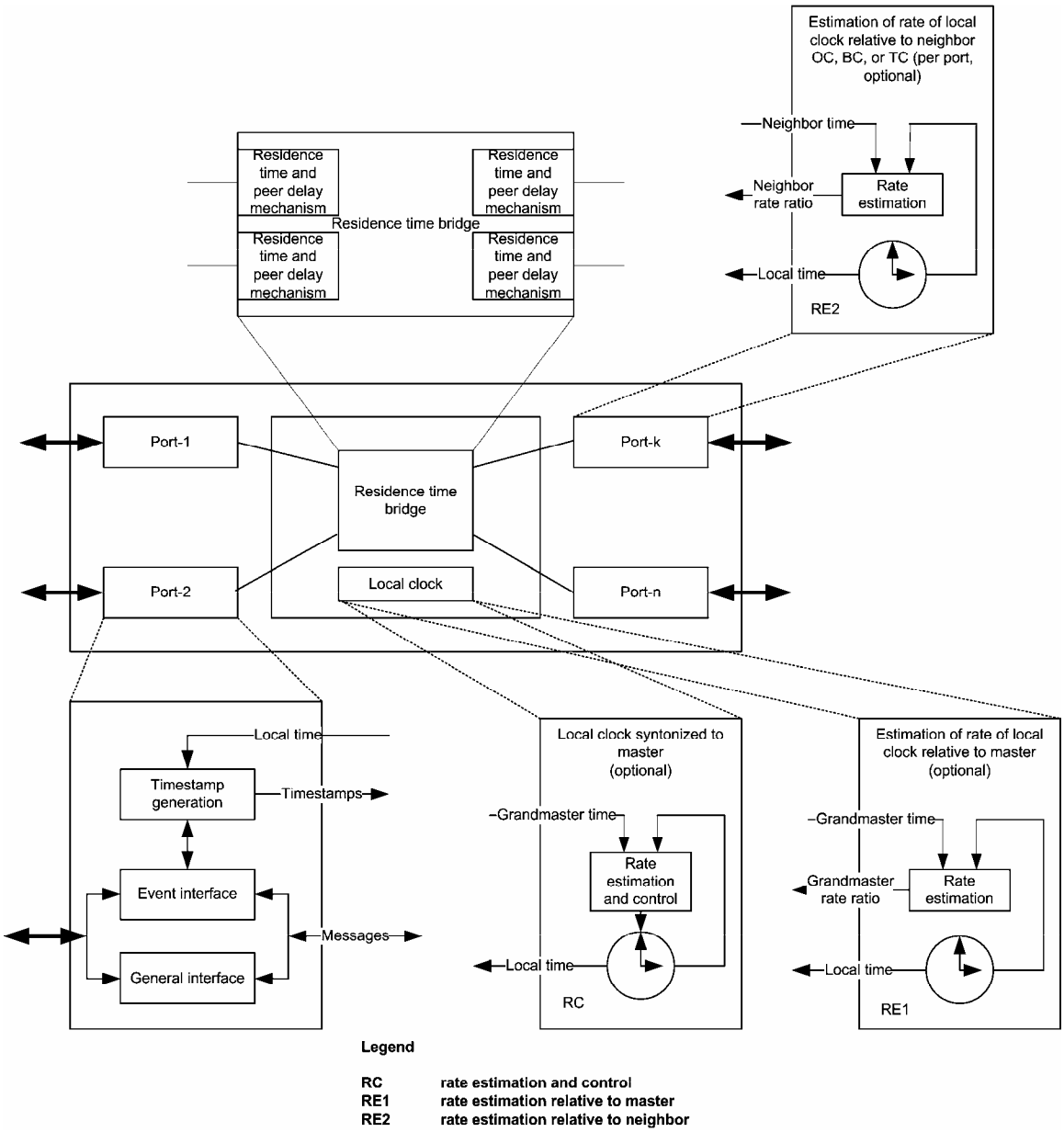

Figure 7—Model of a peer-to-peer transparent clock

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub
Full document details: IEC 61588 2009

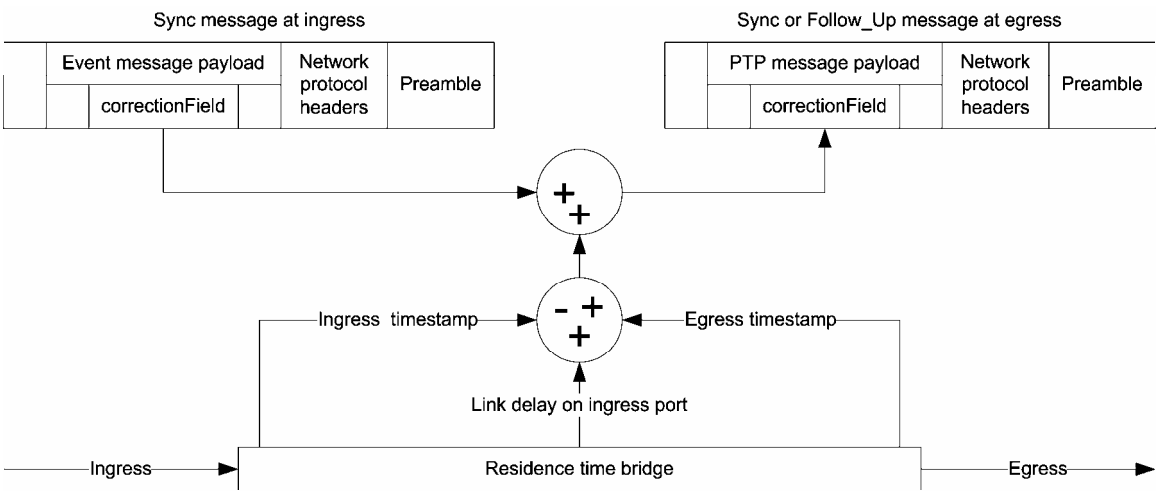

Figure 8—Peer-to-peer residence time and link delay correction model

One way of reducing the maximum error, the method discussed extensively in 11.4, is to make the rate of the local clock equal to that of the master, i.e., syntonize the local clock to the master, by observing the timing information in received Sync and, if present, Follow_Up messages, corrected for any upstream residence and link delay times. This is illustrated in Figure 7 in block RC (rate control). The grandmaster time, corrected for any upstream residence and link delay times, is input to the Rate estimation and control block in RC (shown by the arrow labeled "grandmaster time"). The corresponding local time is also input to this block. The rate estimation and control block uses the sequences of the grandmaster and corresponding local times to estimate the ratio of the grandmaster and local clock rates. The rate estimation and control block then uses this estimated rate ratio to adjust (i.e., control) the local clock rate. Note that the adjustment of the local clock rate need not be a physical changing of the oscillator frequency (i.e., an analog implementation); it is equally acceptable to use a fixed frequency local oscillator, compute the ratio of the master and local rates, and multiply the timestamps taken with the local clock by this ratio (i.e., a digital implementation). The key aspect of the scheme illustrated by RC is that it operates closed loop; i.e., the local times used in the rate estimation and residence time measurement are relative to the rate-adjusted local oscillator. This means that rate adjustments to an oscillator at one node influence the adjustments at downstream nodes, because the residence time is used to adjust the master times at downstream nodes.

A second way to reduce the effect of residence time error that may be useful in some applications is to use free-running local clocks. This is illustrated in the block RE1 (master rate ratio estimation) in Figure 7. In this case, the frequency of the local oscillator is not adjusted but is allowed to free-run. Each clock measures the local residence time of the Sync message based on the free-running clock. This measured residence time is added to the correctionField of the Sync or Follow_Up message as discussed in 11.4. Each local clock also computes the ratio of the rate of the local free-running clock to the rate of the grandmaster. This ratio is computed based on the timestamps in received Sync and Follow_Up messages corrected by adding the correctionField. Even though the correctionField does not reflect the exact residence time, the resulting ratio will still have greatly reduced jitter compared with simply using the raw timestamps and will still have the correct mean value. In addition, if the residence time is sufficiently large relative to the rate ratio measurement interval, the worst-case phase error accumulation over a succession of transparent clocks will be less than that using the first method (above), but at the expense of greater phase accumulation for many non-worst-case scenarios. In addition, each local clock computes the difference between the residence time corrected by this ratio and the uncorrected residence times. These differences must also be accumulated and passed to the slave clock. If this method is to be used, it should be specified in a profile; see 19.3, along with a profile-specific TLV used for the accumulation. With this method, the slave must add these accumulated differences to the correctionField of received Sync or Follow_Up messages before computing the offset of the slave clock from the grandmaster.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

With both of these methods, the correctionField of either the Sync or Follow_Up message must also be corrected for the link delay as discussed in 11.4. The link delays are measured using the peer delay mechanism (see 6.6.4). If the peer-to-peer transparent clocks are syntonized to the master as shown in block RC, the peer delay measurement is also done using syntonized clocks. However, if the rate of the transparent clock relative to the grandmaster is measured and used as in block RE1, the peer delay measurement is done using the free-running local clocks. In this case, there is an error in the link delay equal to the elapsed time between the receipt of the Pdelay_Req message and the sending of the Pdelay_Resp message, t_3–t_2 (see 6.6.4 and Figure 13) multiplied by the fractional frequency offset between the transparent clock and its neighbor. To eliminate this component of error in link delay, the frequency offset between the transparent clock and its neighbor can be measured using the timestamp information contained in successive Pdelay_Resp, and possibly Pdelay_Resp_Follow_Up messages. This is illustrated in the block RE2 (neighbor rate ratio estimation) in Figure 7. Note that to make this measurement, it is necessary to transmit the Pdelay_Req receipt time (t_2) and Pdelay_Resp transmit time (t_3) separately (see 11.4.3).

A peer-to-peer transparent clock may be used as a network element, or it may be associated with application devices such as sensors or actuators. In the latter case, an ordinary clock is combined with the transparent clock to provide real-time support for the application device. A model of such a combined device is illustrated in Figure 9.

In Figure 9, when the ordinary clock is a slave, the timing information includes the ingress timestamp. When it is a master, it includes the egress timestamp. When the clock is a slave, the residence time bridge delivers the incoming PTP timing messages, the Announce messages, and the ingress timestamp generated by the incoming Sync message, and any internal timing corrections to the protocol engine of the ordinary clock. The protocol engine computes the correct time based on these messages and sends it as input to the local clock. If the ordinary clock were a master, then it would originate Sync and Follow_Up messages with the sending timestamps referenced to the local clock of the ordinary clock based on internal timing corrections and the egress timestamp. In practice the peer-to-peer transparent and ordinary clock functions would share a common physical clock.

6.5.6 Management nodes

A management node is a PTP device that:

a) Has one or more physical connections to the network

b) Serves as an human or programmatic interface to PTP management messages

c) May be combined with any of the clock types

6.6 Synchronization overview

6.6.1 General

There are two phases in the normal execution of the protocol:

— Establishing the master-slave hierarchy

— Synchronizing the clocks

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

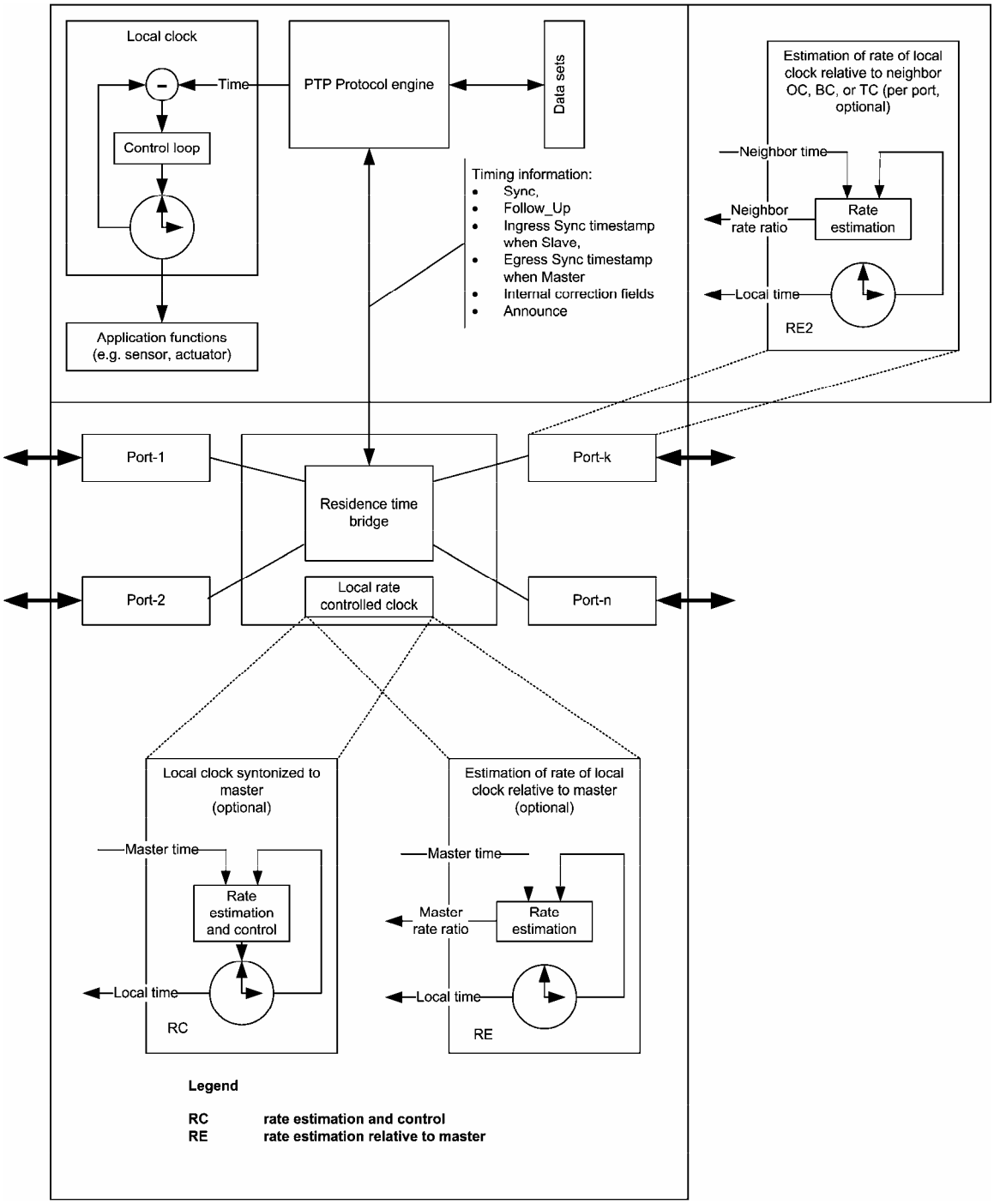

Figure 9—Combined ordinary and peer-to-peer transparent clock.

6.6.2 Establishing the master–slave hierarchy

6.6.2.1 General

Within a domain, each port of an ordinary and boundary clock executes an independent copy of the protocol state machine. For "state decision events," each port examines the contents of all Announce messages received on the port. Using the best master clock algorithm, the Announce message contents and

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub
Full document details: IEC 61588 2009

the contents of the data sets associated with the ordinary or boundary clock are analyzed to determine the state of each port of the clock.

6.6.2.2 PTP state machine

Each port of an ordinary and boundary clock maintains a separate copy of the PTP state machine. This state machine defines the allowed states of the port and the transition rules between states. The principal "state decision events" determining the master–slave hierarchy are the receipt of an Announce message and the end of an announceInterval (the interval between Announce messages). The port states determining the master–slave hierarchy are as follows:

a) MASTER: The port is the source of time on the path served by the port.

b) SLAVE: The port synchronizes to the device on the path with the port that is in the MASTER state.

c) PASSIVE: The port is not the master on the path nor does it synchronize to a master.

6.6.2.3 Best master clock algorithm

The best master clock algorithm compares data describing two clocks to determine which data describes the better clock. This algorithm is used to determine which of the clocks described in several Announce messages received by a local clock port is the best clock. It is also used to determine whether a newly discovered clock—a foreign master—is better than the local clock itself. The data describing a foreign master is contained in the grandmaster fields of an Announce message. The data describing the local clock is contained in the defaultDS data set of the clock.

The best master clock algorithm is composed of two separate algorithms:

— Data set comparison algorithm

— State decision algorithm

The data set comparison algorithm is based on pair-wise comparisons of attributes with the following precedence:

a) priority1: A user configurable designation that a clock belongs to an ordered set of clocks from which a master is selected

b) clockClass: An attribute defining a clock's TAI traceability

c) clockAccuracy: An attribute defining the accuracy of a clock

d) offsetScaledLogVariance: An attribute defining the stability of a clock

e) priority2: A user configurable designation that provides finer grained ordering among otherwise equivalent clocks

f) clockIdentity: A tie-breaker based on unique identifiers

In addition to this precedence order, the "distance" measured by the number of boundary clocks between the local clock and the foreign master is used when two Announce messages reflect the same foreign master. The distance is indicated in the stepsRemoved field of Announce messages. This condition can occur in PTP systems with cyclic paths not removed by a protocol outside of PTP. The data set comparison algorithm unambiguously selects one of the two clocks as "better" or as "topologically better. "

The state decision algorithm determines whether the next state of the port—the recommended state—will be MASTER, SLAVE, or PASSIVE based on the results of the data set comparison algorithm and whether the local clock's class is less than 128; see 7.6.2.4. This recommended state is then evaluated by the port

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009

protocol engine based on the current state of the protocol state machine to determine the actual next port state.

6.6.2.4 Simple master–slave hierarchy

The process establishing a master–slave hierarchy among the ordinary and boundary clocks in a domain is illustrated in Figure 10.

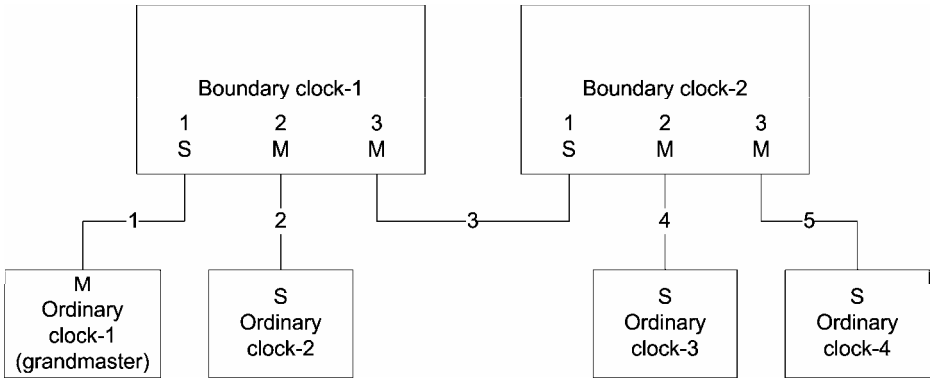

Figure 10—Simple master–slave clock hierarchy

In this example, ordinary clock-1 is at the root of the hierarchy and is called the grandmaster clock. Port-1 of boundary clock-1 is a slave (as indicated by the S) to the grandmaster clock. All other ports on boundary clock-1 are masters to the clocks connected to them. Thus, port-1 of boundary clock-2 is a slave to boundary clock-1 and so forth. Only ordinary and boundary clocks maintain this form of state, and only boundary clocks establish the branch points in the master–slave hierarchy (i.e., paths 1, 2, 3, 4, and 5 may contain transparent clocks, but these clocks do not participate in the master–slave hierarchy and do not maintain this form of state).

6.6.2.5 Pruning mesh topologies

Figure 11 illustrates the case where a mesh network is reduced to a tree-structured master–slave hierarchy by the protocol. This occurs when the underlying bridging or routing protocols do not eliminate cyclic paths. In Figure 11, ordinary clock-1 is assumed to have been selected as the grandmaster clock by the best master clock algorithm. In the boundary clocks, the port states have been selected by the best master clock algorithm as shown to construct the tree. The pruned paths are shown in dashed lines. For each boundary clock, one port is selected by the best master clock algorithm as the slave port. The other ports are set to either the master or the passive state. The best master clock algorithm ensures that a single master port is selected on each segment.

6.6.3 Synchronizing ordinary and boundary clocks

In a PTP system, ordinary or boundary clocks synchronize by exchanging PTP timing messages on the communication path linking the two clocks. For example, in Figure 10, boundary clock-1 synchronizes to ordinary clock-1 by exchanging messages on communication path-1.

The basic pattern of synchronization message exchange is illustrated in Figure 12.

Published by IEC under licence from IEEE. © 2008 IEEE. All rights reserved.

Documents
Document Sharing Hub

Full document details: IEC 61588 2009