INTERNATIONAL STANDARD **ISO/IEEE 11073-10408**

Second edition
2022-12

Health informatics — Device interoperability —

Part 10408: Personal health device communication — Device specialization — Thermometer

Informatique de santé — Interopérabilité des dispositifs —

Partie 10408: Communication entre dispositifs de santé personnels — Spécialisation des dispositifs — Thermomètre

Reference number
ISO/IEEE 11073-10408:2022(E)

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

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

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

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

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

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

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

ISO/IEEE 11073-10408 was prepared by the *IEEE 11073 Standards Committee of the IEEE Engineering in Medicine and Biology Society* (as IEEE Std 11073-10408-2019) and drafted in accordance with its editorial rules. It was adopted, under the "fast-track procedure" defined in the Partner Standards Development Organization cooperation agreement between ISO and IEEE, by Technical Committee ISO/TC 215, *Health informatics*.

This second edition cancels and replaces the first edition (ISO/IEEE 11073-10408:2010), which has been technically revised.

A list of all parts in the ISO/IEEE 11073 series can be found on the ISO website.

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

IEEE Std 11073-10408™-2019
(Revision of IEEE Std 11073-10408-2008)

Health informatics—Personal health device communication

Part 10408: Device specialization—Thermometer

Developed by the

IEEE 11073™ Standards Committee
of the
IEEE Engineering in Medicine and Biology Society

Approved 7 November 2019

IEEE SA Standards Board

Abstract: Within the context of the ISO/IEEE 11073 family of standards for device communication, this standard establishes a normative definition of communication between personal telehealth thermometer devices and compute engines (e.g., cell phones, personal computers, personal health appliances, and set top boxes) in a manner that enables plug-and-play interoperability. It leverages appropriate portions of existing standards including ISO/IEEE 11073 terminology, information models, application profile standards, and transport standards. It specifies the use of specific term codes, formats, and behaviors in telehealth environments restricting optionality in base frameworks in favor of interoperability. This standard defines a common core of communication functionality for personal telehealth thermometer devices.

Keywords: IEEE 11073-10408™, medical device communication, personal health devices, thermometer

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

PDF: ISBN **978-1-5044-6361-4** **STD24006**
Print: ISBN **978-1-5044-6362-1** **STDPD24006**

Important Notices and Disclaimers Concerning IEEE Standards Documents

IEEE documents are made available for use subject to important notices and legal disclaimers. These notices and disclaimers, or a reference to this page, appear in all standards and may be found under the heading "Important Notices and Disclaimers Concerning IEEE Standards Documents." They can also be obtained on request from IEEE or viewed at http://standards.ieee.org/ipr/disclaimers.html.

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

IEEE Standards documents (standards, recommended practices, and guides), both full-use and trial-use, are developed within IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association ("IEEE SA") Standards Board. IEEE ("the Institute") develops its standards through a consensus development process, approved by the American National Standards Institute ("ANSI"), which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE Standards are documents developed through scientific, academic, and industry-based technical working groups. Volunteers in IEEE working groups are not necessarily members of the Institute and participate without compensation from IEEE. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

IEEE Standards do not guarantee or ensure safety, security, health, or environmental protection, or ensure against interference with or from other devices or networks. Implementers and users of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

IEEE does not warrant or represent the accuracy or content of the material contained in its standards, and expressly disclaims all warranties (express, implied and statutory) not included in this or any other document relating to the standard, including, but not limited to, the warranties of: merchantability; fitness for a particular purpose; non-infringement; and quality, accuracy, effectiveness, currency, or completeness of material. In addition, IEEE disclaims any and all conditions relating to: results; and workmanlike effort. IEEE standards documents are supplied "AS IS" and "WITH ALL FAULTS."

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

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

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

Translations

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

Official statements

A statement, written or oral, that is not processed in accordance with the IEEE SA Standards Board Operations Manual shall not be considered or inferred to be the official position of IEEE or any of its committees and shall not be considered to be, or be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position of IEEE.

Comments on standards

Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE. However, IEEE does not provide consulting information or advice pertaining to IEEE Standards documents. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to comments or questions except in those cases where the matter has previously been addressed. For the same reason, IEEE does not respond to interpretation requests. Any person who would like to participate in revisions to an IEEE standard is welcome to join the relevant IEEE working group.

Comments on standards should be submitted to the following address:

Secretary, IEEE SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854 USA

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not 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

IEEE draft and approved standards are copyrighted by IEEE under US and international copyright laws. They are made available by IEEE and are adopted for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making these documents available for use and adoption by public authorities and private users, IEEE does not waive any rights in copyright to the documents.

Photocopies

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

Updating of IEEE Standards documents

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

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

In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit IEEE Xplore at https://ieeexplore.ieee.org or contact IEEE at the address listed previously. For more information about the IEEE SA or IEEE's standards development process, visit the IEEE SA Website at http://standards.ieee.org.

Errata

Errata, if any, for IEEE standards can be accessed via https://standards.ieee.org/standard/index.html. Search for standard number and year of approval to access the web page of the published standard. Errata links are located under the Additional Resources Details section. Errata are also available in IEEE Xplore: https://ieeexplore.ieee.org/browse/standards/collection/ieee/. Users are encouraged to periodically check for errata.

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

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

Participants

At the time this standard was completed, the Personal Health Devices Working Group had the following membership:

Daidi Zhong, *Co-chair*
Michael J. Kirwan, *Co-chair*

Karsten Aalders
Charles R. Abbruscato
Nabil Abujbara
Maher Abuzaid
James Agnew
Haidar Ahmad
Manfred Aigner
Jorge Alberola
Murtaza Ali
Rolf Ambuehl
David Aparisi
Paolo Ariano
Lawrence Arne
Diego B. Arquillo
Serafin Arroyo
Muhammad Asim
Merat Bagha
Doug Baird
David Baker
Anindya Bakshi
Ananth Balasubramanian
Sunlee Bang
M. Jonathan Barkley
Gilberto Barrón
David Bean
John Bell
Rudy Belliardi
Kathryn M. Bennett
Daniel Bernstein
George A. Bertos
Chris Biernacki
Ola Björsne
Thomas Blackadar
Marc Blanchet
Thomas Bluethner
Douglas P. Bogia
Xavier Boniface
Shannon Boucousis
Julius Broma
Lyle G. Bullock, Jr.
Bernard Burg
Chris Burns
Anthony Butt
Jeremy Byford-Rew
Satya Calloji
Xiaoying Cao
Carole C. Carey
Craig Carlson
Santiago Carot-Nemesio
Randy W. Carroll
Simon Carter
Seungchul Chae
Rahul Chauhan
Wenjuan Chen
James Cheng
Peggy Chien
David Chiu
Jinyong Choi
Chia-Chin Chong
Saeed A. Choudhary
Jinhan Chung
Malcolm Clarke
John A. Cogan
John T. Collins
Cory Condek
Todd H. Cooper
David Cornejo
Douglas Coup
Nigel Cox
Hans Crommenacker
Tomio Crosley
Allen Curtis
Ndifor Cyril Fru
Jesús Daniel Trigo
Eyal Dassau
David Davenport
Russell Davis
Sushil K. Deka
Ciro de la Vega
Pedro de-las-Heras-Quiros
Jim Dello Stritto
Matthew d'Entremont
Kent Dicks
Hyoungho Do
Alistair Donaldson
Xiaolian Duan
Brian Dubreuil
Sourav Dutta
Jakob Ehrensvard
Fredrik Einberg
Michihiro Enokida
Javier Escayola Calvo
Mark Estes
Leonardo Estevez
Roger Feeley
Hailing Feng
Bosco T. Fernandes
Christoph Fischer
Morten Flintrup
Joseph W. Forler
Russell Foster
Eric Freudenthal
Matthias Frohner
Ken Fuchs
Jing Gao
Qi Gao
Marcus Garbe
John Garguilo
Rick Geimer
Igor Gejdos
Ferenc Gerbovics
Alan Godfrey
Nicolae Goga
Julian Goldman
Raul Gonzalez Gomez
Chris Gough
Channa Gowda
Charles M. Gropper
Amit Gupta
Jeff Guttmacher
Rasmus Haahr
Christian Habermann
Michael Hagerty
Jerry Hahn
Robert Hall
Shu Han
Nathaniel Hamming
Rickey L. Hampton
Sten Hanke
Aki Harma
Jordan Hartmann
Kai Hassing
Wolfgang Heck
Nathaniel Heintzman
Charles Henderson
Jun-Ho Her
Helen B. Hernandez
Takashi Hibino
Timothy L. Hirou
Allen Hobbs
Alex Holland
Arto Holopainen
Kris Holtzclaw
Xinyi Hong
Robert Hoy
Frank Hsu
Anne Huang
Sen-Der Huang
Ron Huby
David Hughes
Robert D. Hughes
Jiyoung Huh
Hugh Hunter
Hitoshi Ikeda
Yutaka Ikeda
Philip O. Isaacson
Atsushi Ito
Michael Jaffe
Praduman Jain
Danny Jochelson
Phaneeth Junga
Akiyoshi Kabe

Steve Kahle
Tomio Kamioka
James J. Kang
Kei Kariya
Andy Kaschl
Junzo Kashihara
Kohichi Kashiwagi
Ralph Kent
Laurie M. Kermes
Ikuo Keshi
Ahmad Kheirandish
Junhyung Kim
Minho Kim
Min-Joon Kim
Taekon Kim
Tetsuya Kimura
Alfred Kloos
Jeongmee Koh
Jean-Marc Koller
Fei Kong
John Koon
Patty Krantz
Raymond Krasinski
Alexander Kraus
Ramesh Krishna
Geoffrey Kruse
Falko Kuester
Rafael Lajara
Shing C. B. Lam
Pierre Landau
Jaechul Lee
JongMuk Lee
Kyong Ho Lee
Rami Lee
Sungkee Lee
Woojae Lee
Yonghee Lee
Joe Lenart
Kathryn A. Lesh
Catherine Li
Jingli Li
Qiong Li
Xiangchen Li
Zhuofang Li
Patrick Lichter
Jisoon Lim
Joon-Ho Lim
John Lin
Hongliang Niu
Xiaoming Liu
Wei-Jung Lo
Charles Lowe
Don Ludolph
Christian Luszick
Songdong Lv
Bob MacWilliams
Srikkanth Madhurbootheswaran
Miriam L. Makhlouf
Romain Marmot
Sandra Martinez
Miguel Martínez de EsproncedaCámara
Peter Mayhew
Jim McCain
László Meleg
Alexander Mense
Behnaz Minaei
Jinsei Miyazaki
Erik Moll
Darr Moore
Carsten Mueglitz
Piotr Murawski
Soundharya Nagasubramanian
Jae-Wook Nah
Alex Neefus
Trong-Nghia Nguyen-Dobinsky
Michael E. Nidd
Tetsu Nishimura
Jim Niswander
Hongliang Niu
Hiroaki Niwamoto
Thomas Norgall
Anand Noubade
Yoshiteru Nozoe
Abraham Ofek
Brett Olive
BegonyaOtal
Marco Paleari
Charles Palmer
Bud Panjwani
Carl Pantiskas
Harry P. Pappas
Hanna Park
Jong-Tae Park
Myungeun Park
Soojun Park
Phillip E. Pash
TongBi Pei
Lucian Pestritu
Soren Petersen
James Petisce
Peter Piction
Michael Pliskin
Varshney Prabodh
Jeff Price
Harald Prinzhorn
Harry Qiu
Arif Rahman
Tanzilur Rahman
Steve Ray
Phillip Raymond
Terrie Reed
Tim Reilly
Barry Reinhold
Brian Reinhold
Melvin I. Reynolds
John G. Rhoads
Jeffrey S. Robbins
Chris Roberts
Moskowitz Robert
Scott M. Robertson
Timothy Robertson
David Rosales
Gary Sagiv
Fatemeh Saki
Bill Saltzstein
Benedikt Salzbrunn
Giovanna Sannino
Jose A. Santos-Cadenas
Stefan Sauermann
John Sawyer
Guillaume Schatz
Alois Schloegl
Paul S. Schluter
Lars Schmitt
Mark G. Schnell
Richard A. Schrenker
Antonio Scorpiniti
KwangSeok Seo
Riccardo Serafin
Sid Shaw
Frank Shen
Min Shih
Mazen Shihabi
Redmond Shouldice
Sterrly K. Simon
Marjorie Skubic
Robert Smith
Ivan Soh
Motoki Sone
Emily Sopensky
Rajagopalan Srinivasan
Andreas Staubert
Nicholas Steblay
Lars Steubesand
John (Ivo) Stivoric
Raymond A. Strickland
Chandrasekaran Subramaniam
Hermanni Suominen
Lee Surprenant
Ravi Swami
Ray Sweidan
Jin Tan
Yi Tang
Haruyuyki Tatsumi
Isabel Tejero
John W. Thomas
Jonas Tirén
Alexandra Todiruta
Janet Traub
Gary Tschautscher
Masato Tsuchid
Ken Tubman
Yoshihiro Uchida
Akib Uddin
Sunil Unadkat
Fabio Urbani
Philipp Urbauer
Laura Vanzago
Alpo Värri
Andrei Vasilateanu
Dalimar Velez
Martha Velezis
Rudi Voon
Barry Vornbrock
Isobel Walker

David Wang
Jerry P. Wang
Yao Wang
Yi Wang
Steve Warren
Fujio Watanabe
Toru Watsuji
Mike Weng
Yuefeng Weng
Kathleen Wible
Paul Williamson
Jan Wittenber
Jia-Rong Wu
Will Wykeham
Ariton Xhafa
Qifeng Yan
Ricky Yang
Shaoqin Ye
Melanie S. Yeung
Qiang Yin
Done-Sik Yoo
Jianchao Zeng
Jason Zhang
Zhiqiang Zhang
Thomas Zhao
Jia Zheng
Miha Zoubek
Szymon Zyskoter

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

Bjoern Andersen
Lyle G. Bullock, Jr.
Keith Chow
Malcolm Clarke
David Fuschi
Randall Groves
Werner Hoelzl
Noriyuki Ikeuchi
Atsushi Ito
Raj Jain
Piotr Karocki
Raymond Krasinski
H. Moll
Beth Pumo
Stefan Schlichting
Janek Schumann
Walter Struppler
Oren Yuen
Janusz Zalewski
Daidi Zhong

When the IEEE SA Standards Board approved this standard on 7 November 2019, it had the following membership:

Gary Hoffman, *Chair*
Ted Burse, *Vice Chair*
Jean-Philippe Faure, *Past Chair*
Konstantinos Karachalios, *Secretary*

Masayuki Ariyoshi
Stephen D. Dukes
J. Travis Griffith
Guido Hiertz
Christel Hunter
Joseph L. Koepfinger*
Thomas Koshy
John D. Kulick
David J. Law
Joseph Levy
Howard Li
Xiaohui Liu
Kevin Lu
Daleep Mohla
Andrew Myles
Annette Reilly
Dorothy Stanley
Sha Wei
Phil Wennblom
Philip Winston
Howard Wolfman
Feng Wu
Jingyi Zhou

*Member Emeritus

Introduction

This introduction is not part of IEEE Std 11073-10408-2019, Health informatics—Personal health device communication—Part 10408: Device specialization—Thermometer.

ISO/IEEE 11073 standards enable communication between medical devices and external computer systems. This document uses the optimized framework created in IEEE Std 11073-20601[a] and describes a specific, interoperable communication approach for thermometers. These standards align with and draw on the existing clinically focused standards to provide support for communication of data from clinical or personal health devices.

[a] Information on references can be found in Clause 2.

Contents

Health informatics—Personal health device communication

Part 10408: Device specialization—Thermometer

1. Overview

1.1 Scope

Within the context of the ISO/IEEE 11073 family of standards for device communication, this standard establishes a normative definition of communication between personal telehealth thermometer devices and compute engines (e.g., cell phones, personal computers, personal health appliances, and set top boxes) in a manner that enables plug-and-play interoperability. It leverages appropriate portions of existing standards, including ISO/IEEE 11073 terminology, information models, application profile standards, and transport standards. It specifies the use of specific term codes, formats, and behaviors in telehealth environments restricting optionality in base frameworks in favor of interoperability. This standard defines a common core of communication functionality for personal telehealth thermometers.

1.2 Purpose

This standard addresses a need for an openly defined, independent standard for controlling information exchange to and from personal health devices and compute engines (e.g., cell phones, personal computers, personal health appliances, and set top boxes). Interoperability is the key to growing the potential market for these devices and to enabling people to be better informed participants in the management of their health.

1.3 Context

See IEEE Std 11073-20601™ for an overview of the environment within which this standard is written.[1]

This document, IEEE Std 11073-10408, defines the device specialization for the thermometer, being a specific agent type, and it provides a description of the device concepts, its capabilities, and its implementation according to this standard.

This standard is based on IEEE Std 11073-20601, which in turn draws information from both ISO/IEEE 11073-10201:2004 [B6] and ISO/IEEE 11073-20101:2004 [B7].[2] The medical device encoding rules (MDERs) used within this standard are fully described in IEEE Std 11073-20601.

[1] Information on normative references can be found in Clause 2.
[2] The numbers in brackets correspond to the numbers of the bibliography in Annex A.

This standard defines specialized nomenclature codes that will be collected in future versions of IEEE Std 11073-10101. Between this standard, IEEE Std 11073-10101, IEEE Std 11073-20601, and IEEE Std 11073-104zz, all required nomenclature codes for implementation are documented. New codes may be defined in newer versions/revisions of each of these documents. In the case of a conflict, where one term code has been assigned to two separate semantic concepts with different RefIDs, in general the oldest definition that is in actual use should take precedence. The same policy applies when one RefID has two different code values assigned in different specifications. The resolution of such conflicts will be determined through joint action by the responsible working groups and other stakeholders, and any corrective action published as corrigenda.

NOTE—In this standard, the term *IEEE Std 11073-104zz* is used to refer to the collection of device specialization standards that utilize IEEE Std 11073-20601, where zz can be any number from 01 to 99, inclusive.[3]

1.4 Word usage

The word *shall* indicates mandatory requirements strictly to be followed in order to conform to the standard and from which no deviation is permitted (shall equals is required to).[4,5]

The word *should* indicates that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required (should equals is recommended that).

The word *may* is used to indicate a course of action permissible within the limits of the standard (may equals is permitted to).

The word *can* is used for statements of possibility and capability, whether material, physical, or causal (can equals is able to).

2. Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so that 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.

IEEE Std 11073-20601™, Health informatics—Personal health device communication—Part 20601: Application profile—Optimized Exchange Protocol.[6,7]

IEEE Std 11073-10101™, Health informatics—Point-of-care medical device communication—Part 10101: Nomenclature.

[3] Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard.
[4] The use of the word *must* is deprecated and cannot be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.
[5] The use of *will* is deprecated and cannot be used when stating mandatory requirements; *will* is used only in statements of fact.
[6] The IEEE standards or products referred to in this clause are trademarks of The Institute of Electrical and Electronics Engineers, Inc.
[7] IEEE publications are available from The Institute of Electrical and Electronics Engineers (http://standards.ieee.org/).

3. Definitions, acronyms, and abbreviations

3.1 Definitions

For the purposes of this standard, the following terms and definitions apply. The *IEEE Standards Dictionary Online* should be referenced for terms not defined in this clause.[8]

agent: A node that collects and transmits personal health data to an associated manager.

body temperature: The measurement of the core body temperature of the person.

class: In object-oriented modeling, it describes the attributes, methods, and events that objects instantiated from the class utilize.

compute engine: *See:* **manager.**

device: A physical device implementing either an agent or a manager role.

extremity body temperature: The measurement of the temperature at the extremities of the body of the person, such as finger or toe.

handle: An unsigned 16-bit number that is locally unique and identifies one of the object instances within an agent.

manager: A node receiving data from one or more agent systems. Examples of managers include a cellular phone, health appliance, set top box, or a computer system.

obj-handle: *See:* **handle.**

object: A unit that represents some functionality or item in a device whose properties are described by attributes.

personal health device: A device used in personal health applications.

personal telehealth device: *See:* **personal health device.**

3.2 Acronyms and abbreviations

APDU	application protocol data unit
ASN.1	Abstract Syntax Notation One
DIM	domain information model
EUI-64	extended unique identifier (64 bits)
ICS	implementation conformance statement
MDC	medical device communication
MDER	medical device encoding rules
MDS	medical device system
MOC	managed object class
PDU	protocol data unit
PHD	personal health device
RT-SA	real-time sample array
VMO	virtual medical object
VMS	virtual medical system

[8] *IEEE Standards Dictionary Online* is available at http://dictionary.ieee.org.

4. Introduction to ISO/IEEE 11073 personal health devices

4.1 General

This standard and the remainder of the series of ISO/IEEE 11073 personal health device (PHD) standards fit in the larger context of the ISO/IEEE 11073 series of standards. The full suite of standards enables agents to interconnect and interoperate with managers and with computerized health-care information systems. See IEEE Std 11073-20601 for a description of the guiding principles for this series of ISO/IEEE 11073 personal health device standards.

IEEE Std 11073-20601 supports the modeling and implementation of an extensive set of personal health devices. This standard defines aspects of the thermometer device. It describes all aspects necessary to implement the application layer services and data exchange protocol between an ISO/IEEE 11073 PHD thermometer agent and a manager. This standard utilizes a subset of the classes and functionality defined in IEEE Std 11073-20601, defines the objects needed to model a thermometer, and adds new modeling definitions where appropriate. All new definitions are given in Annex B in Abstract Syntax Notation One (ASN.1) [B8].

All nomenclature codes referenced in this standard are collected in Annex C. Annex C may contain definitions of codes that are used by this standard and that are not yet present in the referenced versions of IEEE Std 11073-10101 and IEEE Std 11073-20601.

4.2 Introduction to IEEE 11073-20601 modeling constructs

4.2.1 General

The ISO/IEEE 11073 series of standards, and in particular IEEE Std 11073-20601, is based on an object-oriented systems management paradigm. The overall system model is divided into three principal components: the domain information model (DIM), the service model, and the communication model. See IEEE Std 11073-20601 for a detailed description of the modeling constructs.

4.2.2 Domain information model

The DIM is a hierarchical model that describes an agent as a set of objects. These objects and their attributes represent the elements that control behavior and report on the status of the agent and data that an agent can communicate to a manager. Communication between the agent and the manager is defined by the application protocol in IEEE Std 11073-20601.

4.2.3 Service model

The service model defines the conceptual mechanisms for the data exchange services. Such services are mapped to messages that are exchanged between the agent and the manager. Protocol messages within the ISO/IEEE 11073 series of standards are defined in ASN.1. The messages defined in IEEE Std 11073-20601 can coexist with messages defined in other standard application profiles defined in the ISO/IEEE 11073 series of standards.

4.2.4 Communication model

In general, the communication model supports the topology of one or more agents communicating over logical point-to-point connections to a single manager. For each logical point-to-point connection, the dynamic system behavior is defined by a connection state machine as specified in IEEE Std 11073-20601.

4.2.5 Implementing the models

An agent implementing this standard shall implement all mandatory elements of the information, service, and communication models as well as all conditional elements where the condition is met. The agent should implement the recommended elements, and it may implement any combination of the optional elements. A manager implementing this standard shall utilize at least one of the mandatory, conditional, recommended, or optional elements. In this context, "utilize" means to use the element as part of the primary function of the manager device. For example, a manager whose primary function is to display data would need to display a piece of data in the element in order to utilize it.

4.3 Compliance with other standards

Devices that comply with this standard may also be required to comply with other domain- and device-specific standards that supersede the requirements of this standard with respect to issues including safety, reliability, and risk management. A user of this standard is expected to be familiar with all other such standards that apply and to comply with any higher specifications thus imposed.

Typically, medical devices should comply with the IEC 60601-1 [B1] base standards with respect to electrical and mechanical safety and any device-specific standard as might be defined in the IEC 60601-2 [B2] series of standards. Software aspects may apply through standards such as IEC 62304 [B3]. Devices that comply with this standard implement higher layers of network software and utilize lower layers as appropriate to the application. The requirements on performance of such applications and conformance are defined elsewhere and are outside the scope of this standard. Moreover, the use of any medical equipment is subject to risk assessment and risk management appropriate to the application. Some relevant examples are ISO 14971 [B5] and IEC 80001-1 [B4]. The requirements of such risk assessment, risk management, and conformance are outside the scope of this standard.

5. Thermometer device concepts and modalities

5.1 General

This clause presents the general concepts of thermometer devices. In the context of personal health devices in this family of standards, a thermometer is a device that measures the temperature at some point on the body of a person. In general, the thermometer will be taking a measurement representative of the core temperature of the body, and traditionally oral or rectal measurements are used for spot checks or attached under the armpit (axillary) for extended monitoring. Band thermometers are placed on exposed areas of skin, often the forehead, but these are not considered as accurate. Typically, the thermometer is placed at the measurement site for sufficient time for the measuring probe to reach the same temperature as the body site, and when stable, a direct digital reading of the probe temperature is taken.

Methods to determine the temperature of the probe vary, but common methods include change in the properties of materials with heat such as resistance or semiconductor bandgap voltage. Tympanic thermometers measure the temperature of the tympanum by infrared measurement, which is noncontact and immediate. Unless relevant, the method used to determine temperature is ignored in this standard.

Thermometers may be designed for specialist monitoring purposes. For example, thermometers embedded in capsules may be swallowed and their data transmitted for monitoring during periods of high physical activity for signs of overheating.

Measurements may be taken at the extremities of the body (fingers, toes) and would typically be used to monitor for signs of problems due to circulation or hypothermia.

5.2 Body temperature

This standard assumes that a temperature measurement is normally taken as representative of the core body temperature, and therefore, the actual site of its measurement is not relevant. For this reason, the type attribute for the temperature object in the standard configuration is set to generic body temperature.

When the site or the method is significant, this will be indicated by use of a separate type for the temperature in an extended configuration.

The main units used in medicine are the Celsius and the Fahrenheit scales.

6. Thermometer domain information model

6.1 Overview

This clause describes the domain information model of the thermometer.

6.2 Class extensions

In this standard, no class extensions are defined with respect to IEEE Std 11073-20601.

6.3 Object instance diagram

The object instance diagram of the thermometer domain information model, defined for the purposes of this standard, is shown in Figure 1.

The objects of the DIM, as shown in Figure 1, are described in 6.4 to 6.12. This includes the medical device system (MDS) object (see 6.5) and the numeric objects (see 6.6). There are no real-time sample array (RT-SA) objects (see 6.7), enumeration objects (see 6.8), PM-store objects (see 6.9), or scanner objects (see 6.10) in the thermometer. See 6.11 for rules for extending the thermometer information model beyond elements as described in this standard. Each clause that describes an object of the thermometer contains the following information:

- The nomenclature code used to identify the class of the object. One example of where this code is used is the configuration event, where the object class is reported for each object. This allows the manager to determine whether the class of the object being specified is a numeric, real-time sample array, enumeration, scanner, or PM-store class.
- The attributes of the object. Each object has attributes that represent and convey information on the physical device and its data sources. Each object has a Handle attribute that identifies the object instance within an agent. Attribute values are accessed and modified using methods such as GET and SET. Attributes types are defined using ASN.1. The ASN.1 definitions for new attribute types specific to this standard are in Annex B, and the ASN.1 definitions for existing attribute types referenced in this standard are in IEEE Std 11073-20601.
- The methods available on the object.
- The potential events generated by the object. Data are sent to the manager using events.
- The available services such as getting or setting attributes.

The attributes for each class are defined in tables that specify the name of the attribute, its value, and its qualifier. The qualifiers mean M — Attribute is Mandatory, C — Attribute is Conditional and depends on the condition stated in the Remark or Value column (if IEEE Std 11073-20601 is referenced, then it

contains the conditions), R — Attribute is Recommended, NR — Attribute is Not Recommended, NA — Attribute is Not Allowed, and O — Attribute is Optional. Mandatory attributes shall be implemented by the agent. Conditional attributes shall be implemented if the condition applies and may be implemented otherwise. Recommended attributes should be implemented by the agent. Not recommended attributes should not be implemented by the agent. Optional attributes may be implemented by the agent. If any attribute (from the DIM of IEEE Std 11073-20601) is not included in the definition of that object in this standard, it shall not be included in that object by an implementation, unless it is a vendor-specific attribute extended according to 6.12.

An attribute is further qualified as static, dynamic, or observational.

— Static attributes shall not change value during the life of an association.

— Dynamic attributes have a value that may change during the life of an association. The dynamic attribute value should be sent at configuration time and shall be sent at or before the time when the value would be needed for interpreting a reported observation.

— Observational attributes have a value that may change during the life of an association. When a set of observational attribute values are received, these values are combined with the available context information (i.e., all related dynamic and static attribute values) to represent the observation at the observation time.

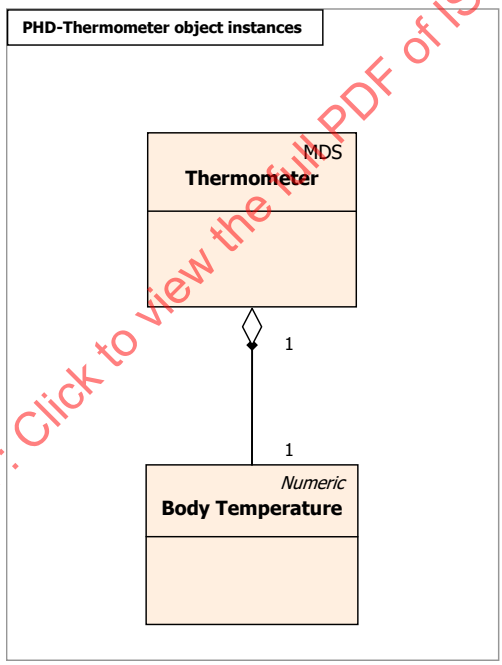

Figure 1—Thermometer—domain information model

6.4 Types of configuration

6.4.1 General

As specified in IEEE Std 11073-20601, there are two styles of configuration available. Subclauses 6.4.2 and 6.4.3 briefly introduce standard and extended configurations.

6.4.2 Standard configuration

Standard configurations are defined in the IEEE 11073-104zz specializations (such as this standard) and are assigned a well-known identifier (Dev-Configuration-Id). The usage of a standard configuration is negotiated at association time between the agent and the manager. If the manager recognizes and selects to operate using the configuration, then the agent can send measurements immediately. If the manager does not recognize the configuration, the agent provides the configuration prior to transmitting measurement information.

6.4.3 Extended configuration

In extended configurations, the configuration of the agent is not predefined in a standard. The agent determines the objects, attributes, and values that will be used in a configuration and assigns a configuration identifier. When the agent associates with a manager, an acceptable configuration is negotiated. Typically, the manager does not recognize the configuration of the agent on the first connection, so the manager responds that the agent must send its configuration information as a configuration event report. If, however, the manager recognizes the configuration, either because it was preloaded in some way or the agent had previously associated with the manager, then the manager responds that the configuration is known and no further configuration information needs to be sent.

6.5 Medical device system object

6.5.1 MDS object attributes

Table 1 summarizes the attributes of the thermometer MDS object. The nomenclature code to identify the MDS class is MDC_MOC_VMS_MDS_SIMP.

Table 1—MDS object attributes

Attribute name	Value	Qualifier		
		Extended configuration	Standard configuration (Dev-Configuration-Id = 0x0320)	Standard configuration (Dev-Configuration-Id = 0x0321)
Handle	0	M	M	M
System-Type	Attribute not present. See IEEE Std 11073-20601.	C	C	C
System-Model	{“Manufacturer”,”Model”}.	M	M	M
System-Id	Extended unique identifier (64 bits) (EUI-64).	M	M	M
Dev-Configuration-Id	Standard config: 0x0320 (800) Standard config: 0x0321 (801) Extended configs: 0x4000-0x7FFF.	M	M	M
Attribute-Value-Map	See IEEE Std 11073-20601.	C	C	C
Production-Specification	See IEEE Std 11073-20601.	O	O	O
Mds-Time-Info	See IEEE Std 11073-20601.	C	C	M
Date-and-Time	See IEEE Std 11073-20601.	C	M	NA
Base-Offset-Time	See IEEE Std 11073-20601.	R	NA	M
Relative-Time	See IEEE Std 11073-20601.	C	NR	NR
HiRes-Relative-Time	See IEEE Std 11073-20601.	C	NR	NR
Date-and-Time-Adjustment	See IEEE Std 11073-20601.	C	C	C
Power-Status	onBattery or onMains.	R	R	R
Battery-Level	See IEEE Std 11073-20601.	R	R	R

Table 1—MDS object attributes *(continued)*

Attribute name	Value	Qualifier		
		Extended configuration	Standard configuration (Dev-Configuration-Id = 0x0320)	Standard configuration (Dev-Configuration-Id = 0x0321)
Remaining-Battery-Time	See IEEE Std 11073-20601.	R	R	R
Reg-Cert-Data-List	See IEEE Std 11073-20601.	O	O	O
System-Type-Spec-List	{MDC_DEV_SPEC_PROFILE_TEMP, 2}.	M	M	M
Confirm-Timeout	See IEEE Std 11073-20601.	O	O	O

NOTE—See IEEE Std 11073-20601 for information on whether an attribute is static or dynamic.

If a standard configuration is chosen, the timestamp type of MDS shall be consistent with the timestamp type of the metric objects in the chosen standard configuration.

In the response to a Get MDS object command, only implemented attributes and their corresponding values are returned.

See IEEE Std 11073-20601 for descriptive explanations of the individual attributes as well as for information on attribute ID and attribute type.

The Dev-Configuration-Id attribute holds a locally unique 16-bit identifier that identifies the device configuration instance. For a thermometer agent with extended configuration, this identifier is chosen in the range of extended-config-start to extended-config-end (see IEEE Std 11073-20601) as shown in Table 1.

The agent sends the Dev-Configuration-Id during the Associating state (see 8.3) to identify its configuration for the duration of the association. If the manager already holds the configuration information relating to the Dev-Configuration-Id, it recognizes the Dev-Configuration-Id and the Configuring state (see 8.4) is skipped; the agent and manager then enter the Operating state. If the manager does not recognize the Dev-Configuration-Id, the agent and manager enter the Configuring state.

If an agent implements multiple IEEE 11073-104zz specializations, System-Type-Spec-List is a list of type/version pairs, each referencing the respective device specialization and version of that specialization.

6.5.2 MDS object methods

Table 2 defines the methods (actions) of the MDS object. These methods are invoked using the Action service. In Table 2, the Subservice type name column defines the name of the method; the Mode column defines whether the method is invoked as an unconfirmed action (i.e., roiv-cmip-action from IEEE Std 11073-20601) or a confirmed action (i.e., roiv-cmip-confirmed-action); the Subservice type (action-type) column defines the nomenclature code to use in the action-type field of an action request and response (see IEEE Std 11073-20601); the Parameters (action-info-args) column defines the associated ASN.1 data structure (see IEEE Std 11073-20601 for ASN.1 definitions) to use in the action message for the action-info-args field of the request; and the Results (action-info-args) column defines the structure to use in the action-info-args of the response.

Table 2—MDS object methods

Service	Subservice type name	Mode	Subservice type (action-type)	Parameters (action-info-args)	Results (action-info-args)
ACTION	Set-Time	Confirmed	MDC_ACT_SET_TIME	SetTimeInvoke	—
	Set-Base-Offset-Time	Confirmed	MDC_ACT_SET_BO_TIME	SetBOTimeInvoke	—

— **Set-Time:**
This method allows the manager to set a real-time clock in the agent with the absolute time. The agent indicates whether the Set-Time command is valid using the mds-time-capab-set-clock bit in the Mds-Time-Info attribute (see IEEE Std 11073-20601). Agents with an internal real-time clock (RTC) shall indicate this capability by also setting the mds-time-capab-real-time-clock bit in the Mds-Time-Info attribute.

The Set-Time method can be supported only if the Absolute-Time-Stamp attribute is supported.

— **Set-Base-Offset-Time:**
This method allows the manager to set a real-time clock in the agent with the base time and offset. The agent indicates whether the Set-Base-Offset-Time command is valid using the mds-time-capab-set-clock bit in the Mds-Time-Info attribute (see IEEE Std 11073-20601).

The Set-Base-Offset-Time method can be supported only if the Base-Offset-Time-Stamp attribute is supported.

6.5.3 MDS object events

Agents following only this device specialization and no others shall send event reports using agent-initiated measurement data transmission. During the association procedure (see 8.3), DataReqModeCapab shall be set to the appropriate value for the event report style (set to data-req-supp-init-agent). The manager shall assume the thermometer agent does not support any of the MDS-Data-Request features (see IEEE Std 11073-20601 for additional information). The data-req-init-manager-count shall be set to zero, and the data-req-init-agent-count shall be set to 1.

Agents following this device specialization as well as others shall send event reports in the appropriate fashion. During the association procedure (see 8.3), DataReqModeCapab shall be set to the appropriate value for the event report style.

Table 3 defines the events that can be sent by the thermometer MDS object.

Table 3—Thermometer MDS object events

Service	Subservice type name	Mode	Subservice type (event-type)	Parameters (event-info)	Results (event-reply-info)
EVENT REPORT	MDS-Configuration-Event	Confirmed	MDC_NOTI_CONFIG	ConfigReport	ConfigReportRsp
	MDS-Dynamic-Data-Update-Var	Confirmed	MDC_NOTI_SCAN_REPORT_VAR	ScanReportInfoVar	—
	MDS-Dynamic-Data-Update-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_FIXED	ScanReportInfoFixed	—
	MDS-Dynamic-Data-Update-MP-Var	Confirmed	MDC_NOTI_SCAN_REPORT_MP_VAR	ScanReportInfoMPVar	—
	MDS-Dynamic-Data-Update-MP-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_MP_FIXED	ScanReportInfoMPFixed	—

— **MDS-Configuration-Event:**
This event is sent by the thermometer agent during the configuring procedure if the manager does not already know the configuration of the thermometer agent from past associations or because the manager has not been implemented to recognize the configuration according to the thermometer

device specialization. The event provides static information about the supported measurement capabilities of the thermometer agent.

— **MDS-Dynamic-Data-Update-Var:**
This event provides dynamic measurement data from the thermometer agent for the body temperature numeric object. These data are reported using a generic attribute list variable format. The event is sent as an unsolicited message by the agent (i.e., an agent-initiated measurement data transmission). See 8.5.3 for more information on unsolicited event reporting. MDS events for temperature readings shall be sent no faster than once per second.

— **MDS-Dynamic-Data-Update-Fixed:**
This event provides dynamic measurement data from the thermometer agent for the body temperature numeric object. These data are reported in the fixed format defined by the Attribute-Value-Map attribute of the object. The event is sent as an unsolicited message by the agent (i.e., an agent-initiated measurement data transmission). See 8.5.3 for more information on unsolicited event reporting.

— **MDS-Dynamic-Data-Update-MP-Var:**
This is the same as MDS-Dynamic-Data-Update-Var but allows inclusion of data from multiple people.

— **MDS-Dynamic-Data-Update-MP-Fixed:**
This is the same as MDS-Dynamic-Data-Update-Fixed but allows inclusion of data from multiple people.

NOTE—IEEE Std 11073-20601 requires that managers support all of the MDS object events listed above.

6.5.4 Other MDS services

6.5.4.1 GET service

A thermometer agent shall support the GET service, which is provided by the MDS object to retrieve the values of all implemented MDS object attributes. The GET service can be invoked only after the manager has confirmed selection of the configuration of the agent.

The manager may request the MDS object attributes of the thermometer agent; in which case, the manager shall send the “Remote Operation Invoke | Get” message (see roiv-cmip-get in IEEE Std 11073-20601) with the reserved MDS handle value of 0. The thermometer agent shall report its MDS object attributes to the manager using the “Remote Operation Response | Get” message (see rors-cmip-get in IEEE Std 11073-20601). See Table 4 for a summary of the GET service including some message fields.

Table 4—Thermometer MDS object GET service

Service	Subservice type name	Mode	Subservice type	Parameters	Results
GET	<na>	<implied confirmed>	<na>	GetArgumentSimple = (obj-handle = 0), attribute-id-list <*optional*>	GetResultSimple = (obj-handle = 0), attribute-list

See 8.5.2 for details on the procedure for getting the MDS object attributes.

6.5.4.2 SET service

The thermometer specialization does not require an implementation to support the MDS object SET service.

6.6 Numeric objects

6.6.1 General

The thermometer DIM (see Figure 1) contains one required object for body temperature, which is described in 6.6.2.

Sometimes, the interpretation of one attribute value in an object depends on other attribute values in the same object. For example, Unit-Code and Unit-LabelString provide context for the observed values. Whenever a contextual attribute changes, the agent shall report these changes to the manager using an MDS object event (see 6.5.3) prior to reporting any of the dependent values.

6.6.2 Temperature

Table 5 summarizes the attributes of the body temperature numeric object. The nomenclature code to identify the numeric class is MDC_MOC_VMO_METRIC_NU. The body temperature numeric object shall be supported by a thermometer agent.

For a thermometer agent with standard configuration, the AttrValMap structure (see IEEE Std 11073-20601) of the Attribute-Value-Map attribute shall contain the attribute ID and attribute length information of the Basic-Nu-Observed-Value and Absolute-Time-Stamp (or Base-Offset-Time-Stamp) attribute in the same order as indicated in Table 1.

The body temperature numeric object does not support any methods, events, or other services.

See ISO/IEEE 11073-20601 for descriptive explanations on the individual attributes as well as for information on attribute ID and attribute type.

{MDC_PART_SCADA, MDC_TEMP_BODY} is the standard configuration and denotes a general body temperature measurement.

{MDC_PART_SCADA, MDC_TEMP_zzz} is used in extended configurations and might report a specific type of measurement, site, or method-specific temperature measurement types as defined in Table 6.

Table 5—Body temperature numeric object attributes

Attribute name	Extended configuration		Standard configuration (Dev-Configuration-Id = 0x0320)		Standard configuration (Dev-Configuration-Id = 0x0321)	
	Value	Qual.	Value	Qual.	Value	Qual.
Handle	See IEEE Std 11073-20601.	M	1	M	1	M
Type	{MDC_PART_SCADA, MDC_TEMP_zzz}.	M	{MDC_PART_SCADA, MDC_TEMP_BODY}.	M	{MDC_PART_SCADA, MDC_TEMP_BODY}.	M
Supplemental-Types	See IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Metric-Spec-Small	mss-avail-intermittent, mss-avail-stored-data, mss-msmt-aperiodic, mss-acc-agent-initiated.	M	mss-avail-intermittent, mss-avail-stored-data, mss-upd-aperiodic, mss-msmt-aperiodic, mss-acc-agent-initiated.	M	mss-avail-intermittent, mss-avail-stored-data, mss-upd-aperiodic, mss-msmt-aperiodic, mss-acc-agent-initiated.	M
Metric-Structure-Small	See IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Measurement-Status	See IEEE Std 11073-20601.	R	Attribute not initially present. If present, follow IEEE Std 11073-20601.	O	Attribute not initially present. If present, follow IEEE Std 11073-20601.	O
Metric-Id	{MDC_TEMP_zzz} See IEEE Std 11073-20601.	O	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Metric-Id-List	See IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Metric-Id-Partition	See IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Unit-Code	MDC_DIM_DEGC or MDC_DIM_FAHR.	M	MDC_DIM_DEGC.	M	MDC_DIM_DEGC.	M
Attribute-Value-Map	See IEEE Std 11073-20601.	C	MDC_ATTR_NU_VAL_OBS_BASIC, then MDC_ATTR_TIME_STAMP_ABS.	M	MDC_ATTR_NU_VAL_OBS_BASIC, then MDC_ATTR_TIME_STAMP_BO.	M
Source-Handle-Reference	See IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Label-String	See IEEE Std 11073-20601.	O	Attribute not initially present. If present, follow IEEE Std 11073-20601.	O	Attribute not initially present. If present, follow IEEE Std 11073-20601.	O
Unit-LabelString	See IEEE Std 11073-20601.	O	Attribute not initially present. If present, follow IEEE Std 11073-20601.	O	Attribute not initially present. If present, follow IEEE Std 11073-20601.	O
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C	See IEEE Std 11073-20601.	M	Attribute not allowed.	NA
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	R	Attribute not allowed.	NA	See IEEE Std 11073-20601.	M

Table 5—Body temperature numeric object attributes *(continued)*

Attribute name	Extended configuration		Standard configuration (Dev-Configuration-Id = 0x0320)		Standard configuration (Dev-Configuration-Id = 0x0321)	
	Value	Qual.	Value		Value	Qual.
Relative-Time-Stamp	See IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Measure-Active-Period	See IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601. If fixed format is used and the standard configuration is unchanged, this attribute is mandatory; otherwise, the conditions from IEEE Std 11073-20601 apply.	R	Attribute not initially present. If present, follow IEEE Std 11073-20601. If fixed format is used and the standard configuration is unchanged, this attribute is mandatory; otherwise, the conditions from IEEE Std 11073-20601 apply.	R
Compound-Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	C
Compound-Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Compound-Nu-Observed-Value	See IEEE Std 11073-20601.	C	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR	Attribute not initially present. If present, follow IEEE Std 11073-20601.	NR
Accuracy	See IEEE Std 11073-20601.	R	Attribute not initially present. If present, follow IEEE Std 11073-20601.	R	Attribute not initially present. If present, follow IEEE Std 11073-20601.	R

NOTE—See IEEE Std 11073-20601 for information on whether an attribute is static or dynamic.

Table 6—Extended configuration: Temperature type

Type	Value	Measurement site
MDC_TEMP_AXILLA	57380	Axillary (armpit)
MDC_TEMP_BODY	19292	General body temperature measurement
MDC_TEMP_EAR	57356	Ear (usually earlobe)
MDC_TEMP_FINGER	57360	Finger
MDC_TEMP_GIT	57384	Gastro-intestinal tract
MDC_TEMP_ORAL	57352	Mouth
MDC_TEMP_RECT	57348	Rectum
MDC_TEMP_TOE	57376	Toe
MDC_TEMP_TYMP	19320	Tympanum (ear drum)

6.7 Real-time sample array objects

Real-time sample array objects are not required by this standard.

6.8 Enumeration objects

Enumeration objects are not required by this standard.

6.9 PM store objects

PM-store objects are not required by this standard.

6.10 Scanner objects

Scanner objects are not required by this standard.

6.11 Class extension objects

In this standard, no class extension objects are defined with respect to IEEE Std 11073-20601.

6.12 Thermometer information model extensibility rules

The thermometer domain information model of this standard may be extended by including vendor-specific metrics and attributes as required. Any object or attribute extensions implemented should follow the guidelines of this standard as closely as possible. Such vendor-specific attributes shall be identified by assigning nomenclature codes from the private numbering space (0xF000–0xFFFF) within the corresponding partition as defined in IEEE Std 11073-20601.

A thermometer agent having a configuration with extensions beyond the standard configuration, as specified in this standard, shall use a configuration ID in the range of IDs reserved for extended configurations (see IEEE Std 11073-20601).

7. Thermometer service model

7.1 General

The service model defines the conceptual mechanisms for data exchange services. These services are mapped to messages that are exchanged between the agent and the manager. Protocol messages within the ISO/IEEE 11073 series of standards are defined in ASN.1. See IEEE Std 11073-20601 for a detailed description of the personal health device service model. Subclause 7.2 through 7.3 define the specifics of object access and event reporting services for a thermometer agent according to this standard.

7.2 Object access services

The object access services of IEEE Std 11073-20601 are used to access the objects defined in the domain information model of the thermometer.

The following generic object access services are supported by a thermometer agent according to this standard:

— GET service: used by the manager to retrieve the values of the agent MDS object attributes. The list of thermometer MDS object attributes is given in 6.5.4.1.

— SET service: used by the manager to set the values of the agent object attributes. There are no settable attributes defined for a thermometer agent according to this standard.

— Event report service: used by the agent to send configuration reports and measurement data to the manager. The list of event reports for the thermometer device specialization is given in 6.5.3.

— Action service: used by the manager to invoke actions (or methods) supported by the agent. An example is Set-Time action, which is used to set a real-time clock with the absolute time at the agent.

Table 7 summarizes the object access services described in this standard.

Table 7—Thermometer object access services

Service	Subservice type name	Mode	Subservice type	Parameters	Result	Remarks
GET	<na>	<implied Confirmed>	<na>	GetArgumentSimple = (obj-handle = 0), attribute-id-list <*optional*>	GetResultSimple = (obj-handle = 0), attribute-list	Allows the manager to retrieve the attribute values of the agent's MDS object.
EVENT REPORT	MDS-Configuration-Event	Confirmed	MDC_NOTI_CONFIG	ConfigReport	ConfigReportRsp	Configuration Report to inform manager of the configuration of the agent.
	MDS-Dynamic-Data-Update-Var	Confirmed	MDC_NOTI_SCAN_REPORT_VAR	ScanReportInfoVar	—	Data Report to provide dynamic data to manager for some or all of the agent's objects in variable format.
	MDS-Dynamic-Data-Update-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_FIXED	ScanReportInfoFixed	—	Data Report to provide dynamic data to manager for some or all of the agent's objects in fixed format.
	MDS-Dynamic-Data-Update-MP-Var	Confirmed	MDC_NOTI_SCAN_REPORT_MP_VAR	ScanReportInfoMPVar	—	This is the same as MDS-Dynamic-Data-Update-Var but allows inclusion of data from multiple people.
	MDS-Dynamic-Data-Update-MP-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_MP_FIXED	ScanReportInfoMPFixed	—	This is the same as MDS-Dynamic-Data-Update-Fixed but allows inclusion of data from multiple people.
ACTION	Set-Time	Confirmed	MDC_ACT_SET_TIME	SetTimeInvoke	—	Manager method to invoke the agent to set time to requested value.
	Set-Base-Offset-Time	Confirmed	MDC_ACT_SET_BO_TIME	SetBOTimeInvoke	—	Manager method to invoke the agent to set time to requested value.

7.3 Object access event report services

The event report service (see Table 7) is used by the agent to report its information (e.g., measurements). Event reports in this standard are a property of the MDS object only. The event reports used in this standard are defined in IEEE Std 11073-20601.

The following conditions apply for a thermometer agent according to this standard:

— Event reports shall be used in confirmed mode.

— Agent-initiated mode shall be supported for measurement data transmission.

A thermometer agent, which is designed to operate in an environment where data may be collected from multiple people, may use one of the multiple-person event report styles to transmit all the data from each person in a single event report. If this functionality is not required, the agent shall use the single-person event report styles, which have reduced overhead.

A manager shall support both single-person and multiple-person event reports. A thermometer agent may support single-person and/or multiple-person event reports. The formats for single- and multiple-person reports are described in IEEE Std 11073-20601.

8. Thermometer communication model

8.1 Overview

This clause describes the general communication model and procedures of the thermometer agent as defined in IEEE Std 11073-20601. Therefore, the respective parts of IEEE Std 11073-20601 are not reproduced; rather the specific choices and restrictions with respect to optional elements (e.g., objects, attributes, and actions) and specific extensions (e.g., nomenclature terms) are specified.

For an illustrative overview of the various message transactions during a typical measurement session, see the sequence diagram for the example use case in Annex D and the corresponding protocol data unit (PDU) examples in Annex E.

8.2 Communications characteristics

In this subclause, limits on the size of an application protocol data unit (APDU) transmitted or to be received by a thermometer agent are defined. Small limits allow for simple implementations in terms of low cost and complexity.

A thermometer agent implementing only this device specialization shall not transmit any APDU larger than N_{tx} and shall be capable of receiving any APDU up to a size of N_{rx}. For this standard, N_{tx} shall be 896 octets and N_{rx} shall be 224 octets.

For a thermometer agent implementing functions from other device specializations, an upper bound estimation of the APDU sizes brings the following: An agent shall not transmit any APDU larger than the sum of N_{tx} of all the device specializations implemented and shall be capable of receiving any APDU up to the sum of N_{rx} of all the device specializations implemented. If these numbers are higher than the maximum size determined in IEEE Std 11073-20601, the latter shall be applied.

In case the APDU size limit does not allow for the inclusion of a certain amount of multiple pending measurements at the agent, they shall be sent using multiple event reports. See 8.5.3 for the maximum number of measurements allowed for inclusion in a single event report.

8.3 Association procedure

8.3.1 General

The association procedure for a thermometer agent and manager according to this standard shall be pursued as specified in IEEE Std 11073-20601.

Table 8 lists the valid combinations of protocol version and nomenclature version. In the association procedure, an agent indicating support to a specific protocol version shall indicate support to the corresponding nomenclature version as well. In the association procedure, a manager selecting a specific protocol version shall select the corresponding nomenclature version.

To indicate support for multiple protocol versions, the bit values are combined. For example, if the agent supports protocol-version2, protocol-version3, and protocol-version4, it shall use protocol version bits 0x70000000 and nomenclature-version bits 0xE0000000.

Further valid combinations from future versions of this specification may be used by implementations that comply with those future versions.

Table 8—Valid combinations of protocol and nomenclature version

Protocol version	Bit value	Corresponding nomenclature version	Bit value
1	0x80000000	1	0x80000000
2	0x40000000	1	0x80000000
3	0x20000000	2	0x40000000
4	0x10000000	3	0x20000000

8.3.2 Agent procedure—association request

In the association request sent by the agent to the manager:

a) The version of the association procedure used by the agent shall be set to assoc-version1 (i.e., *assoc-version* = 0x80000000).

b) The DataProtoList structure element of the data protocol identifier shall be set to data-proto-id-20601 (i.e., *data-proto-id* = 0x5079).

c) The *data-proto-info* field shall contain a PhdAssociationInformation structure that shall contain the following parameter values:

1) The version of the data exchange protocol shall be set to protocol-version4 (i.e., *protocol-version* = 0x10000000). Support for any other version may be indicated by setting additional bits. When protocols lower than protocol-version4 are used, the agent shall use only features in that protocol.

2) At least the MDER encoding rules shall be supported (i.e., *encoding-rules* = 0x8000).

3) The protocol version bits and nomenclature version bits shall consist of valid combinations of bits as defined in Table 8.

4) The field *functional-units* may have the test association bits set but shall not have any other bits set.

5) The field *system-type* shall be set to sys-type-agent (i.e., *system-type* = 0x00800000).

6) The *system-id* field shall be set to the value of the System-Id attribute of the MDS object of the agent. The manager may use this field to determine the identity of the thermometer with which it is associating and, optionally, to implement a simple access restriction policy.

7) The *dev-config-id* field shall be set to the value of the Dev-Configuration-Id attribute of the MDS object of the agent.

8) If the agent supports only the thermometer specialization, then the *data-req-mode-capab* field shall be set to *data-req-supp-init-agent*, the *data-req-init-manager-count* shall be set to zero, and the *data-req-init-agent-count* shall be set to 1.

8.3.3 Manager procedure—association response

In the association response message sent by the manager:

a) The *result* field shall be set to an appropriate response from those defined in IEEE Std 11073-20601. For example, if all other conditions of the association protocol are satisfied, accepted is returned when the manager recognizes the *dev-config-id* of the agent and the accepted-unknown-config otherwise.

b) In the DataProtoList structure element, the data protocol identifier shall be set to data-proto-id-20601 (i.e., *data-proto-id* = 0x5079).

c) The *data-proto-info* field shall be filled in with a PhdAssociationInformation structure that shall contain the following parameter values:

1) The manager following this specialization shall support protocol-version4. The manager may support additional protocol versions and select them if the agent offers them. When protocols lower than protocol-version4 are used, the manager shall use only features in that protocol.

2) The manager shall respond with a single selected encoding rule that is supported by both agent and manager. The manager shall support at least the MDER.

3) The manager shall select a valid combination of protocol version and nomenclature version as defined in Table 8.

4) The field *functional-units* shall have all bits reset except for those relating to a test association.

5) The field *system-type* shall be set to sys-type-manager (i.e., system-type = 0x80000000).

6) The *system-id* field shall contain the unique system ID of the manager device, which shall be a valid EUI-64 type identifier.

7) The field *dev-config-id* shall be manager-config-response (0).

8) The fields *data-req-mode-capab*, *data-req-init-agent*, and *data-req-init-manager-count* shall be 0. If the agent supports only the thermometer specialization, *data-req-init-agent-count* shall be 1.

8.4 Configuring procedure

8.4.1 General

The agent enters the Configuring state if it receives an association response of accepted-unknown-config. In this case, the configuration procedure as specified in IEEE Std 11073-20601 shall be followed. Subclause 8.4.2 specifies the configuration notification and response messages for a thermometer agent with standard configuration ID 801 (0x0321). Normally, a manager would already know the standard configuration. However, for the purposes of this example, it does not.

8.4.2 Thermometer—standard configuration

8.4.2.1 Agent procedure

The agent performs the configuration procedure using a "Remote Operation Invoke | Confirmed Event Report" message with an MDC_NOTI_CONFIG event to send its configuration to the manager (see IEEE Std 11073-20601). The ConfigReport structure is used for the *event-info* field (see Table 3). For a thermometer agent with standard configuration ID 801 (0x0321), the format and contents of the configuration notification message are as follows:

0xE7 0x00	APDU CHOICE Type (PrstApdu)
0x00 0x44	CHOICE.length = 68
0x00 0x42	OCTET STRING.length = 66
0xXX 0xXX	invoke-id (differentiates this message from any other outstanding)
0x01 0x01	CHOICE(Remote Operation Invoke | Confirmed Event Report)
0x00 0x3C	CHOICE.length = 60
0x00 0x00	obj-handle = 0 (MDS object)
0xXX 0xXX 0xXX 0xXX	event-time (set to 0xFFFFFFFF if RelativeTime is not supported)
0x0D 0x1C	event-type = MDC_NOTI_CONFIG
0x00 0x32	event-info.length = 50 (start of ConfigReport)
0x03 0x21	config-report-id (Dev-Configuration-Id value)
0x00 0x01	config-obj-list.count = 1 Measurement object will be "announced"
0x00 0x2C	config-obj-list.length = 44
0x00 0x06	obj-class = MDC_MOC_VMO_METRIC_NU
0x00 0x01	obj-handle = 1 (→ 1st measurement)
0x00 0x04	attributes.count = 4
0x00 0x24	attributes.length = 36
0x09 0x2F	attribute-id = MDC_ATTR_ID_TYPE
0x00 0x04	attribute-value.length = 4
0x00 0x02 0x4B 0x5C	MDC_PART_SCADA, MDC_TEMP_BODY
0x0A 0x46	attribute-id = MDC_ATTR_METRIC_SPEC_SMALL
0x00 0x02	attribute-value.length = 2
0xF0 0x40	intermittent, stored data, upd & msmt aperiodic, agent init, measured
0x09 0x96	attribute-id = MDC_ATTR_UNIT_CODE
0x00 0x02	attribute-value.length = 2
0x17 0xA0	MDC_DIM_DEGC
0x0A 0x55	attribute-id = MDC_ATTR_ATTRIBUTE_VAL_MAP
0x00 0x0C	attribute-value.length = 12
0x00 0x02	AttrValMap.count = 2
0x00 0x08	AttrValMap.length = 8
0x0A 0x4C 0x00 0x02	MDC_ATTR_NU_VAL_OBS_BASIC | value length = 2
0x0A 0x82 0x00 0x08	MDC_ATTR_TIME_STAMP_BO | value length = 8

Note, at the locations of the message, where the content is not fixed, the value "0xXX" denotes a placeholder and depends on the implementation or on the preceding messaging of the agent.

8.4.2.2 Manager procedure

The manager shall respond to a configuration notification message using a "Remote Operation Response | Confirmed Event Report" data message with an MDC_NOTI_CONFIG event using the ConfigReportRsp structure for the *event-info* field (see Table 3). As a response to the standard configuration notification message in 8.4.2.1, the format and contents of the manager's configuration notification response message are as follows:

0xE7 0x00	APDU CHOICE Type (PrstApdu)
0x00 0x16	CHOICE.length = 22
0x00 0x14	OCTET STRING.length = 20
0xXX 0xXX	invoke-id (differentiates this message from any other outstanding)
0x02 0x01	CHOICE (Remote Operation Response | Confirmed Event Report)
0x00 0x0E	CHOICE.length = 14
0x00 0x00	obj-handle = 0 (MDS object)
0xXX 0xXX 0xXX 0xXX	currentTime
0x0D 0x1C	event-type = MDC_NOTI_CONFIG
0x00 0x04	event-reply-info.length = 4
0x03 0x21	ConfigReportRsp.config-report-id = 0x0321
0x00 0x00	ConfigReportRsp.config-result = accepted-config.

Again, the value "0xXX" denotes a placeholder and refers to a fixed location, varying content parts of the message.

8.5 Operating procedure

8.5.1 General

Measurement data and status information are communicated from the thermometer agent during the Operating state. If not stated otherwise, the operating procedure for a thermometer agent of this standard shall be as specified in IEEE Std 11073-20601.

8.5.2 GET thermometer MDS attributes

See Table 4 for a summary of the GET service.

If the attribute-id-list field in the roiv-cmip-get service message is empty, the thermometer agent shall respond with a rors-cmip-get service message in which the attribute-list contains a list of all implemented attributes of the MDS object.

If the manager requests specific MDS object attributes, indicated by the elements in attribute-id-list, then the thermometer agent shall respond with a rors-cmip-get service message in which the attribute-list contains a list of the requested attributes of the MDS object that are implemented.

8.5.3 Measurement data transmission

See Table 3 for a summary of the event report services available for measurement data transfer.

Measurement data transfer for a thermometer agent of this standard shall always be initiated by the thermometer (see agent-initiated measurement data transmission in IEEE Std 11073-20601). To limit the amount of data being transported within an APDU, the thermometer agent shall not include more than 25 temporarily stored measurements in a single event report. If more than 25 pending measurements are available for transmission, they shall be sent using multiple event reports. If multiple temperature measurements are available, up to 25 measurements should be transmitted within a single event report. Alternatively, they may be transmitted using a single event report for each temperature measurement. However, the former strategy is recommended to reduce overall message size and power consumption.

8.6 Time synchronization

Time synchronization between a thermometer agent and a manager may be used to coordinate the clocks used when reporting physiological events. Note that the mechanism for synchronizing an agent to a manager is outside the scope of this standard. If time synchronization is used, then this shall be reported in the Mds-Time-Info attribute of the MDS object.

9. Test associations

9.1 General

The test association provides a manufacturer the mechanism to test or demonstrate features of a product in a comprehensive manner. This clause defines the behavior of the standard thermometer agent during a test association. Support for test association is optional.

9.2 Behavior with standard configuration

An agent or manager entering a test association using the configuration ID for the standard thermometer device of this standard shall enter the Operating state in test mode. When in test mode, where possible, this should be indicated visually to any user. Normal functionality shall be suspended, and any test data generated shall not be processed by the device as physiological data.

The thermometer agent shall send a single simulated temperature value of 59.99 °C (a value never seen in normal usage and outside normal range) within 30 s of entering the Operating state. If the measurement-status attribute of the numeric object is implemented, then the test-data bit shall be set.

The test association is terminated in a manner consistent with the agent's normal behavior for terminating an association.

9.3 Behavior with extended configurations

This specification does not define a test association that uses an extended configuration.

10. Conformance

10.1 Applicability

This standard shall be used in conjunction with IEEE Std 11073-20601.

An implementation or a system can conform to the following elements of this standard:

— Domain information model class hierarchy and object definitions (object attributes, notifications, methods, and data type definitions)

— Nomenclature code values

— Protocol and service models

— Communication service model (association and configuration)

10.2 Conformance specification

This standard offers levels of conformance with respect to strict adherence to the standard device and the use of extensions for the following:

— Information model of a specific device

— Use of attributes, value ranges, and access methods

A vendor shall specify the level of conformance for an implementation based on this standard and provide details of the way in which the definitions of this standard and any extensions are applied.

Specifications shall be provided in the form of a set of implementation conformance statements (ICSs) as detailed in 10.4.

Since this standard is used in conjunction with IEEE Std 11073-20601, the ICS should be created for this standard first. The ICS created for IEEE Std 11073-20601 may then refer to the ICS for this standard where applicable.

10.3 Levels of conformance

10.3.1 General

This standard defines the following levels of conformance.

10.3.2 Conformance level 1: Base conformance

The application uses elements of the information, service, and communication models (object hierarchy, actions, event reports, and data type definitions) and the nomenclature scheme defined in IEEE Std 11073-20601 and IEEE Std 11073-104zz. All mandatory features defined in the object definition tables and in the ICS tables are implemented. Furthermore, any conditional, recommended, or optional features that are implemented shall follow the requirements in IEEE Std 11073-20601 and IEEE Std 11073-104zz.

10.3.3 Conformance level 2: Extended nomenclature (ASN.1 and/or IEEE Std 11073-10101)

Conformance level 2 meets conformance level 1 but also uses or adds extensions in at least one of the information, service, or nomenclature models. Extensions to nomenclature codes shall conform to the IEEE 11073-10101 framework and lie within the private nomenclature extension range (0xF000–0xFFFF).

Extensions to the information or service models shall be fully defined using ASN.1 where appropriate and have their behavior fully described following the framework of IEEE Std 11073-20601 and/or ISO/IEEE 11073-20101. All extensions shall be specified and include reference to the definition for the extension, or where no publicly available reference is available, the definition of the extension should be appended to the conformance statement.

10.4 Implementation conformance statements

10.4.1 General format

The ICSs are provided as an overall conformance statement document that comprises a set of tables in the form given by the templates in the following clauses.

Each ICS table has the following columns:

Index	Feature	Reference	Req/Status	Support	Comment

The table column headings have the following meaning:

— Index: an identifier (e.g., a tag) of a specific feature.

— Feature: briefly describes the characteristic for which a conformance statement is being made.

— Reference: to the clause/paragraph within this document or an external source for the definition of the feature (may be empty).

— Req/Status: specifies the conformance requirement (e.g., mandatory or recommended)—in some cases, this standard does not specify conformance requirements but requests the status of a particular feature be provided.

— Support: specifies the presence or absence of a feature and any description of the characteristics of the feature in the implementation. This column is to be filled out by the implementer.

— Comment: contains any additional information on the feature. This column is to be filled out by the implementer.

Subclauses 10.4.2 to 10.4.6 specify the format of the specific ICS tables.

10.4.2 General implementation conformance statement

The general ICS specifies the versions/revisions that are supported by the implementation and high-level system behavior.

Table 9 shows the general ICSs.

Table 9—IEEE 11073-10408 general ICSs table

Index[a]	Feature	Reference	Req./Status	Support	Comment
GEN11073-10408-1	Implementation Description	—	Identification of the device/ application. Description of functionality.		
GEN11073-10408-2	Standards followed and their revisions	(standard documents)	(set of existing revisions)	(set of supported revision)	
GEN11073-10408-3	Nomenclature document used and revision	(standard documents)	(set of existing revisions)	(set of supported revisions)	

Table 9—IEEE 11073-10408 general ICSs table *(continued)*

Index[a]	Feature	Reference	Req./Status	Support	Comment
GEN11073-10408-4	Conformance Adherence - Level 1 -	See 10.3.2	Base conformance declaration that device meets the following IEEE 11073-10408 conformance requirements: a) All mandatory requirements shall be implemented. b) If implemented, conditional, recommended, and optional requirements shall conform to this standard.	Yes/No (No is not expected as No implies that the implementation is non-conformant)	
GEN11073-10408-5	Conformance Adherence - Level 2 -	See 10.3.3	In addition to GEN 11073-10408-4, if the device implements extensions and/or additions, they shall conform to nomenclature codes from ASN.1 and/or IEEE 11073-10101 framework. These extensions should also be defined in ICS tables pointing toward their reference.	Yes/No	
GEN11073-10408-6	Object Containment Tree	See 6.3	Provide Object Containment Diagram showing relations between object instances used by the application. A conforming implementation uses only object relations as defined in the DIM.		
GEN11073-10408-7	Nomenclature document used and revision	(standard documents)	(set of existing revisions)	(set of supported revision)	
GEN11073-10408-8	Data Structure Encoding	—	—	description of encoding method(s) for ASN.1 data structures	
GEN11073-10408-9	Use of Private Objects	—	Does the implementation use objects that are not defined in the DIM?	Yes/No (If yes: explain in Table 10)	

Table 9—IEEE 11073-10408 general ICSs table *(continued)*

Index[a]	Feature	Reference	Req./Status	Support	Comment
GEN11073-10408-10	Use of Private Nomenclature Extensions	—	Does the implementation use private extensions to the nomenclature (i.e., 0xF000–0xFFFF codes from IEEE Std 11073-10101)? Private Nomenclature extensions are allowed *only* if the standard nomenclature does not include the specific terms required by the application.	Yes/No (If yes: explain in Table 13)	
GEN11073-10408-11	11073-20601 Conformance	—	Provide the conformance report required by IEEE Std 11073-20601.		

[a] The prefix *GEN11073-10408-* is used for the index in the general ICSs table.

10.4.3 DIM MOC implementation conformance statement

The DIM MOC ICS defines which objects are implemented. Information on each object shall be provided as a separate row in the template of Table 10.

Table 10—Template for DIM MOC ICS table

Index	Feature	Reference	Req./Status	Support	Comment
MOC-n	Object description	Reference to the clause in the standard or other location where the object is defined.	Implemented	Specify restrictions (e.g., maximum number of supported instances)	

The n in the Index column should be the object handle for implementations that have predefined objects. Otherwise the Index column shall simply be a unique number (1..m).

All private objects shall be specified and include either a reference to the definition for the object or, where no publicly available reference is available, the definition of the object should be appended to the conformance statement.

The Support column should indicate any restrictions for the object implementation.

An object containment diagram (class instance diagram) should be provided as part of the DIM MOC ICS.

10.4.4 MOC attribute ICS

The MOC attribute ICS defines which attributes, including any inherited attributes, are used/supported in each object of an implementation. Information on each attribute of an object shall be provided as a separate row in the template of Table 11. A separate MOC attribute ICS shall be provided for each object.

Table 11—Template for MOC attribute ICS table

Index	Feature	Reference	Req./Status	Support	Comment
ATTR-n-x	Attribute Name. Extended attributes shall include the attribute ID also.	Fill in the reference to the ASN.1 structure if the attribute is not defined in this standard.	M = Mandatory / C = Conditional / R = Recommended / O = Optional (as per definition in Attribute Definition Tables)	Implemented? Yes/No Static/Dynamic Specify restrictions (e.g., value ranges). Describe how attribute is accessed (e.g., Get, Set, sent in config event report or sent in a data event report). Describe any specific restrictions.	

The Support column shall specify whether the attribute is implemented; for extension attributes, whether the attribute value is static or dynamic; any value ranges; restrictions on attribute access or availability; and any other information.

The n in the Index column refers to the ID of the managed object for which the table is supplied (i.e., the index of the managed object as specified in the MOC ICS). There is one separate table for each supported managed object.

The x in the Index column is a unique serial number (1..m).

10.4.5 MOC notification implementation conformance statement

The MOC notification ICS specifies all implemented notifications (typically in the form of the event report service) that are emitted by the agent. Table 12 provides a template for use. One table has to be provided for each object that supports special object notifications. One row of the table shall be used for each notification.

Table 12—Template for MOC notification ICS table

Index	Feature	Reference	Req./Status	Support	Comment
NOTI-n-x	Notification Name and Notification ID	Reference to the clause in the standard or other location where the event is defined.		The Support column shall specify how the notification is sent and any restrictions.	

The n in the Index column refers to the ID of the managed object for which the table is supplied (i.e., the index of the managed object as specified in the POC ICS). There is one separate table for each managed object that supports specific object notifications (i.e., events).

The x in the Index column is a unique serial number (1..m).

All private notifications should be specified and include reference to the definition for the notification. Where no publicly available reference is available, the definition of the notification should be appended to the conformance statement.

10.4.6 MOC nomenclature conformance statement

The MOC nomenclature ICS specifies all nonstandard nomenclature codes that are utilized by the agent. Table 13 provides a template for use. One row of the table is to be used for each nomenclature element.

Table 13—Template for MOC nomenclature ICS table

Index	Feature	Reference	Req./Status	Support	Comment
NOME-n	Nomenclature Name and Nomenclature value	Reference to the clause in the standard or other location where the nomenclature is defined or used.		Describe how the nomenclature is used. Describe any specific restrictions.	

The n in the Index column is a unique serial number (1..m).

Annex A

(informative)

Bibliography

Bibliographical references are resources that provide additional or helpful material but do not need to be understood or used to implement this standard. Reference to these resources is made for informational use only.

[B1] IEC 60601-1, Ed. 3, Medical electrical equipment—Part 1: General requirements for basic safety and essential performance.[9]

[B2] IEC 60601-2, Medical electrical equipment—Part 2: Particular requirements for the basic safety and essential performance for specific device. (See the entire series of standards, Part 2-1 through Part 2-51.)

[B3] IEC 62304, Medical device software—Software life-cycle processes.

[B4] IEC 80001-1, Application of risk management for IT-networks incorporating medical devices—Part 1: Roles, responsibilities, and activities.

[B5] ISO 14971, Medical devices—Application of risk management to medical devices.[10]

[B6] ISO/IEEE 11073-10201:2004, Health informatics — Point-of-care medical device communication — Part 10201: Domain information model.[11]

[B7] ISO/IEEE 11073-20101:2004, Health informatics — Point-of-care medical device communication — Part 20101: Application profile — Base standard.

[B8] ITU-T Rec. X.680, Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation.[12]

[9] IEC publication are available from the International Electrotechnical Commission (http://www.iec.ch).
[10] ISO publication are available from the International Organization for Standardization (http://www.iso.ch/).
[11] ISO/IEEE publications are available from the International Organization for Standardization (http://www.iso.ch/). ISO/IEC publications are also available in the United States from Global Engineering Documents (http://global.ihs.com/). Electronic copies are available in the United States from the American National Standards Institute (http://www.ansi.org/).
[12] ITU publications are available from the International Telecommunications Union (http://www.itu.in/).

Annex B

(normative)

Any additional ASN.1 definitions

No additional ASN.1 definitions are defined.

Annex C

(normative)

Allocation of identifiers

This annex contains the nomenclature codes used in this document and not found in IEEE Std 11073-20601. For those not contained in this annex, the normative definition is found in IEEE Std 11073-20601.

The format used here follows that of IEEE Std 11073-10101.

```
/*****************************************************************************
* From Communication Infrastructure (MDC_PART_INFRA)
*****************************************************************************/
#define MDC_DEV_SPEC_PROFILE_TEMP          4104  /* Thermometer MDS          */

/*****************************************************************************
* From Medical supervisory control and data acquisition (MDC_PART_SCADA)
*****************************************************************************/

/* Copy of the nomenclature codes already defined in IEEE Std 11073-10101.   */

#define MDC_TEMP_TYMP                      19320 /* Tympanum (ear drum)      */
#define MDC_TEMP_RECT                      57348 /* Rectum                   */
#define MDC_TEMP_ORAL                      57352 /* Mouth                    */
#define MDC_TEMP_EAR                       57356 /* Ear (usually earlobe)    */
#define MDC_TEMP_FINGER                    57360 /* Finger                   */
#define MDC_TEMP_TOE                       57376 /* Toe                      */
#define MDC_TEMP_BODY                      19292 /* General body temperature
                                                    measurement              */

/* New nomenclature codes introduced in the present document (IEEE Std 11073-10408). */

#define MDC_TEMP_AXILLA                    57380 /* Axillary (armpit)        */
#define MDC_TEMP_GIT                       57384 /* Gastro-intestinal tract  */

/*****************************************************************************
* From Dimensions (MDC_PART_DIM)
*****************************************************************************/
#define MDC_DIM_FAHR                       4416  /* °F                       */
```

Annex D

(informative)

Message sequence examples

Figure D.1 shows a sequence diagram of the messaging procedure corresponding to the following use case. The user of a thermometer agent device intends to connect it to a manager device for the first time. The thermometer is capable of performing temperature measurements.

a) When the user connects the thermometer, the manager does not yet know the configuration of the agent and sends a response to the agent's association request with the result *accepted-unknown-config*. See E.2.2.2 and E.2.2.3 for the corresponding PDU examples.

b) As a consequence of this, the agent sends its configuration information to the manager. After getting confirmation from the manager accepting the configuration of the agent, the agent device is ready to send measurements. Both devices enter the Operating state. See E.3.2.2 and E.3.2.3 for the corresponding PDU examples.

c) Subsequently, the manager may request the MDS object attributes of the agent by sending a data message with the "Remote Operation Invoke | Get" command. As a response, the agent reports its MDS object attributes to the manager using a data message with the "Remote Operation Response | Get" command. See E.4.2 and E.4.3 for the corresponding PDU examples.

d) As a next step, the user of the agent device takes a single measurement. The measurement data is transmitted to the manager using a confirmed event report. After having successfully received the measurement data, the manager sends a confirmation to the agent. See E.5.1 and E.5.2 for the corresponding PDU examples.

e) The user ends the measurement session (e.g., by pushing a proper button on the device or just by not using the device for a duration longer than a certain time period). As a consequence, the agent disassociates from the manager by sending an association release request. The manager responds with an association release response. See E.6.1 and E.6.2 for the corresponding PDU examples.

f) When the agent requests to associate to the manager for the next measurement session (e.g., the next day), the result in the manager's response is *accepted*, as it already knows the configuration of the agent from the previous measurement session. Both devices transition directly to the Operating state.

g) Finally, the last two steps shown are similar as in item d) and item e). The user takes a single confirmed measurement followed by releasing the association.