INTERNATIONAL STANDARD

ISO/IEC/IEEE 21450

First edition
2010-05-15

Information technology — Smart transducer interface for sensors and actuators — Common functions, communication protocols, and Transducer Electronic Data Sheet (TEDS) formats

Technologies de l'information — Interface de transducteurs intelligente pour capteurs et actuateurs — Fonctions communes, protocoles de communication et formats des feuilles de données électroniques du transducteur (TEDS)

Reference number
ISO/IEC/IEEE 21450:2010(E)

© IEEE 2007

Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 21450-2010

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat, the IEC Central Office and IEEE do not accept any liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies and IEEE members. In the unlikely event that a problem relating to it is found, please inform the ISO Central Secretariat or IEEE at the address given below.

COPYRIGHT PROTECTED DOCUMENT

© IEEE 2007

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO or IEEE at the respective address below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York • NY 10016-5997, USA
E-mail stds.ipr@ieee.org
Web www.ieee.org

ISO version published 2010

Published in Switzerland

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub

Full document details: ISO/IEC/IEEE 21450-2010

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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.

The main task of ISO/IEC JTC 1 is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is called to the possibility that implementation of this standard may require the use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. ISO/IEEE is not responsible for identifying essential patents or patent claims for which a license may be required, for conducting inquiries into the legal validity or scope of patents or patent claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance or a Patent Statement and Licensing Declaration Form, 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 ISO or the IEEE Standards Association.

ISO/IEC/IEEE 21450 was prepared by the Technical Committee on Sensor Technology of the IEEE Instrumentation and Measurement Society of the IEEE (as IEEE 1451.0™-2007). It was adopted by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 31, *Automatic identification and data capture techniques*, in parallel with its approval by the ISO/IEC national bodies, under the “fast-track procedure” defined in the Partner Standards Development Organization cooperation agreement between ISO and IEEE. IEEE is responsible for the maintenance of this document with participation and input from ISO/IEC national bodies.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

(blank page)

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

1451.0™

IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats

IEEE Instrumentation and Measurement Society

Sponsored by the
Technical Committee on Sensor Technology (TC-9)

IEEE
3 Park Avenue
New York, NY 10016-5997, USA

21 September 2007

IEEE Std 1451.0™-2007

Documents
Document Sharing Hub

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 21450-2010

(blank page)

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats

Sponsor

Technical Committee on Sensor Technology (TC-9)
of the
IEEE Instrumentation and Measurement Society

Approved 9 August 2007

American National Standards Institute

Approved 22 March 2007

IEEE-SA Standards Board

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

Abstract: This standard provides a common basis for members of the IEEE 1451 family of standards to be interoperable. It defines the functions that are to be performed by a transducer interface module (TIM) and the common characteristics for all devices that implement the TIM. It specifies the formats for Transducer Electronic Data Sheets (TEDS). It defines a set of commands to facilitate the setup and control of the TIM as well as reading and writing the data used by the system. Application programming interfaces (APIs) are defined to facilitate communications with the TIM and with applications.

Keywords: actuator, application programming interface, communication protocol, network-capable application processor, sensor, smart transducer, transducer electronic data sheet, transducer interface module

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

Copyright © 2007 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 21 September 2007. Printed in the United States of America.
2nd Printing 2 November 2007. A printing error for Equation (15) has been corrected.

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

Print: ISBN 0-7381-5597-7 SH95684
PDF: ISBN 0-7381-5598-5 SS95684

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

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

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

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

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

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

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

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

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

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

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

Documents
Document Sharing Hub

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

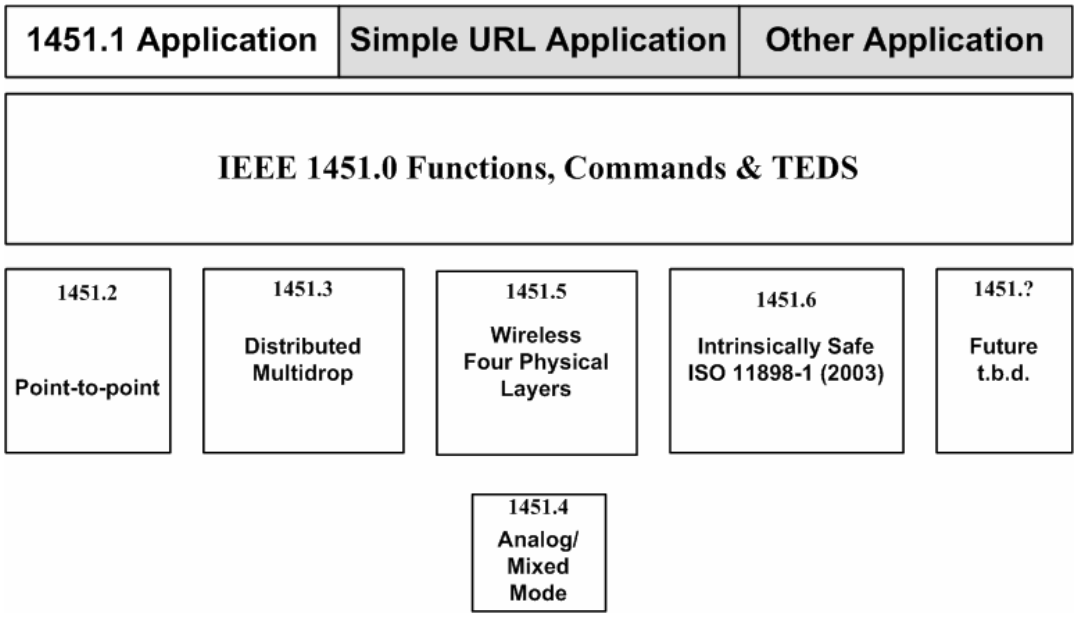
Full document details: ISO/IEC/IEEE 21450-2010

Introduction

This introduction is not part of IEEE Std 1451.0-2007, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats.

This standard is intended to provide a basis for all future members of the IEEE 1451 family of standards that use digital interfaces. It should also be adopted by the existing members of the IEEE 1451 family of standards as they are revised in the future in order to provide the highest degree of compatibility among the members of the family. This standard does not apply to IEEE Std 1451.4™-2004, which only provides a size-constrained TEDS and an analog interface.

The relationships between this standard and the other members of the family are shown in the following diagram. Three of these standards were complete before this standard was started and do not comply with this standard but will in the future as they are revised. They are IEEE Std 1451.1™-1999, IEEE Std 1451.2™-1997, and IEEE Std 1451.3™-2003. IEEE Std 1451.1 is an application that, in the future, will fit between the user's network and this standard. IEEE Std 1451.2 and IEEE Std 1451.3 will also be modified to interface with this standard. When these changes are made, the functions of an IEEE 1451 transducer will be as defined in this standard as will be the commands and TEDS. IEEE 1451.5™-2007, which uses any of several different wireless communications media, and IEEE P1451.6™ have been written around the functions, commands, and TEDS as described in this standard. IEEE Std 1451.4 uses an analog signal interface and a TEDS that is not the same as that used by other members of the family. It may be used as the input to any of the other standards in the family but does not comply with the functions, commands, and TEDS defined in this standard. Items shown with a gray background are items that are not covered by any of the IEEE 1451 family of standards but that may be used.



The underlying purpose of this family of standards is to allow manufacturers to build elements of a system that are interoperable. To accomplish this goal, the IEEE 1451 family of standards divides the parts of a system into two general categories of devices. One is the network capable application processor (NCAP) that functions as a gateway between the users' network and the transducer interface modules (TIMs). The NCAP is a processor-based device that has two interfaces. The physical interface to the users' network is not specified in any of this family of standards. IEEE Std 1451.1 provides a logical object model for this

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

interface. Other applications may also be used at the manufacturer's discretion. The communications interface between the NCAP and the TIMs is defined in the remaining members of the family of standards. Different manufacturers may build the NCAPs and TIMs, and if both comply with this standard, they should be interoperable.

This standard provides a description of the functions that are to be performed by a transducer interface module or TIM. Provisions are made for a high level of addressing that is independent of the physical medium-level and low-level protocols that are used to implement the communications. It defines the common characteristics for all devices that implement the transducer modules. The timing of the acquiring or processing of the data samples is described. Methods of grouping the outputs from multiple transducers within one TIM are defined. Common status words are also defined.

A standard set of commands are defined to facilitate the setup and control of the transducer modules as well as to read and write the data used by the system. Commands are also provided for reading and writing the TEDS that supply the system with the operating characteristics that are needed to use the transducer modules. A method of adding manufacturer unique commands is included.

In addition, this standard provides formats for the TEDS. Several TEDS are defined in the standard. Four of these TEDS are required, and the remaining TEDS are optional. Some TEDS are provided to allow the user to define information and to store it in the TEDS.

This standard provides areas that are "open to manufacturers." It should be noted that any use of these areas may compromise the "plug-and-play" potential of controllers and TIMs.

Notice to users

Errata

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

Interpretations

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

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims or determining whether any licensing terms or conditions are reasonable or non-discriminatory. Further information may be obtained from the IEEE Standards Association.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

Contents

1 Overview 1

1.1 Scope 3
1.2 Purpose 3
1.3 Conformance 3

2 Normative references 5

3 Definitions, acronyms, and abbreviations 6

3.1 Definitions 6
3.2 Acronyms and abbreviations 9

4 Data types 10

4.1 Unsigned octet integer 10
4.2 Unsigned 16 bit integer 10
4.3 Signed 32 bit integer 10
4.4 Unsigned 32 bit integer 10
4.5 Single-precision real 11
4.6 Double-precision real 11
4.7 String 11
4.8 Boolean 11
4.9 IEEE1451Dot0::Args::TimeRepresentation 12
4.10 Data types for associated applications 13
4.11 Physical Units 13
4.12 Universal unique identification 15
4.13 Arbitrary octet array 15
4.14 String array 16
4.15 Boolean array 16
4.16 Array of 8 bit signed integers 16
4.17 Array of 16 bit signed integers 16
4.18 Array of 32 bit signed integers 16
4.19 Array of 8 bit unsigned integers 17
4.20 Array of 16 bit unsigned integers 18
4.21 Array of 32 bit unsigned integers 18
4.22 Array of single-precision real numbers 18
4.23 Array of double-precision real numbers 18
4.24 Array of TimeDuration data types 18
4.25 Array of TimeInstance data types 19

5 Smart transducer functional specification 19

5.1 IEEE 1451 family reference model 19
5.2 Plug-and-play capability 23
5.3 Addresses 23
5.4 Common characteristics 25
5.5 Transducer Electronic Data Sheets 27
5.6 TransducerChannel type descriptions 31
5.7 Embedded TransducerChannels 34
5.8 TransducerChannel groups 34

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.9 TransducerChannel proxy 35
5.10 Attributes and operating modes 36
5.11 Triggering 41
5.12 Synchronization 48
5.13 Status 49
5.14 Service request logic 55
5.15 Hot-swap capability 56

6 Message structures 56

6.1 Data transmission order and bit significance 56
6.2 Command message structure 57
6.3 Reply messages 58
6.4 TIM initiated message structure 58

7 Commands 59

7.1 Standard commands 60
7.2 Manufacturer-defined commands 81

8 TEDS specification 81

8.1 General format for TEDS 81
8.2 Order of octets in numeric fields 83
8.3 TEDS identification header 83
8.4 Meta-TEDS 84
8.5 TransducerChannel TEDS 94
8.6 Calibration TEDS 119
8.7 Frequency Response TEDS 136
8.8 Transfer Function TEDS 139
8.9 Text-based TEDS 149
8.10 End User Application Specific TEDS 154
8.11 User's Transducer Name TEDS 155
8.12 Manufacturer-defined TEDS 157
8.13 PHY TEDS 158

9 Introduction to the IEEE 1451.0 API 158

9.1 API goals 159
9.2 API design decisions 160
9.3 IEEE1451Dot0 162

10 Transducer services API 173

10.1 IEEE1451Dot0::TransducerServices::TimDiscovery 173
10.2 IEEE1451Dot0::TransducerServices::TransducerAccess 175
10.3 IEEE1451Dot0::TransducerServices::TransducerManager 181
10.4 IEEE1451Dot0::TransducerServices::TedsManager 187
10.5 IEEE1451Dot0::TransducerServices::CommManager 190
10.6 IEEE1451Dot0::TransducerServices::AppCallback 191

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

11 Module Communications API 193

11.1 IEEE1451Dot0::ModuleCommunication::Comm 193
11.2 IEEE1451Dot0::ModuleCommunication::P2PComm 197
11.3 IEEE1451Dot0::ModuleCommunication::NetComm 201
11.4 IEEE1451Dot0::ModuleCommunication::Registration 210
11.5 IEEE1451Dot0::ModuleCommunication::P2PRegistration 212
11.6 IEEE1451Dot0::ModuleCommunication::NetRegistration 214
11.7 IEEE1451Dot0::ModuleCommunication::Receive 217
11.8 IEEE1451Dot0::ModuleCommunication::P2PReceive 217
11.9 IEEE1451Dot0::ModuleCommunication::NetReceive 218

12 HTTP protocol 220

12.1 IEEE 1451.0 HTTP API 221
12.2 Discovery API 224
12.3 Transducer access API 226
12.4 TEDS Manager API 232
12.5 Transducer Manager API 238

Annex A (informative) Bibliography 244

Annex B (informative) Guidance to Transducer Services Interface 246

Annex C (informative) Guidance to Module Communication Interface 251

Annex D (informative) XML Schema for Text-based TEDS 261

Annex E (informative) Example Meta-Identification TEDS 278

Annex F (informative) Example TransducerChannel Identification TEDS 280

Annex G (informative) Example Calibration Identification TEDS 282

Annex H (informative) Example Commands TEDS 284

Annex I (informative) Example Location and Title TEDS 287

Annex J (infomative) Example Units Extension TEDS 289

Annex K (informative) Examples of Physical Units 290

Annex L (informative) TEDS read and write protocols 296

Annex M (informative) Trigger logic configurations 298

Annex N (informative) Notation summary for IDL 303

Annex O (informative) TEDS implementation of a simple sensor 307

Annex P (informative) IEEE list of participants 324

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats

1. Overview

This standard introduces the concept of a transducer interface module (TIM) and a network capable application processor (NCAP) connected by a media specified by another member of the IEEE 1451 family of standards. A TIM is a module that contains the interface, signal conditioning, analog-to-digital and/or digital-to-analog conversion and, in many cases, the transducer. A TIM may range in complexity from a single sensor or actuator to units containing many transducers (sensors and actuators). An NCAP is the hardware and software that provides the gateway function between the TIMs and the user network or host processor. Another member of the standards family provides the communications interface between an NCAP or host processor and one or more TIMs. Three types of transducers are recognized by this standard. They are sensors, event sensors, and actuators.

A transducer is denoted "smart" in this context because of three features:

— It is described by a machine-readable Transducer Electronic Data Sheet (TEDS).

— The control and data associated with the transducer are digital.

— Triggering, status, and control are provided to support the proper functioning of the transducer.

An NCAP or a host processor controls a TIM by means of a digital interface medium. The NCAP mediates between the TIM and a higher level digital network and may provide local intelligence.

This standard defines an application program interface (API) for applications that provide communications between the users' network and the IEEE 1451.0 layer. An API is also provided between the IEEE 1451.0 layer and the underlying physical communications layers usually referred to in this standard as IEEE 1451.X. These API definitions are provided for systems that have visible interfaces. For TIMs and

Documents
Document Sharing Hub

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 21450-2010

NCAPs with a single set of hardware and software without regard to distinguishing separate interfaces between IEEE 1451.0 functionality and IEEE 1451.X functionality, the API is optional as long as the messages at visible interfaces conform to the rest of the standard. The definition of these APIs is to facilitate modular design to the extent that multiple suppliers can provide different functionality and yet have the various parts integrate seamlessly.

This standard defines TIMs that can be plugged into a system and can be used without having to add special drivers, profiles, or make any other changes to the system. This process is referred to as a "plug-and-play" operation. The primary features that enable a plug-and-play operation are the TEDS and the command set. A TIM may be added to or removed from an active IEEE 1451 physical layer with no more than a momentary impact on the data being transferred over the bus. "Hot swap" is the term used to refer to this feature.

This standard is organized as follows:

Clause 1: "Overview" provides the scope of this standard.

Clause 2: "Normative references" lists references to other standards and documents that are useful in applying this standard.

Clause 3: "Definitions, acronyms, and abbreviations" provides definitions that are either not found in other standards or have been modified for use with this standard.

Clause 4: "Data types" defines the data types used in the standard.

Clause 5: "Smart transducer functional specification" specifies the functions required of a TIM and of each TransducerChannel it comprises.

Clause 6: "Message structures" specifies the message structures that are used to encapsulate the information being passed between an NCAP and TIMs.

Clause 7: "Commands" provides the command syntax and the expected replies.

Clause 8: "TEDS specification" specifies the transducer electronic data sheet structure and content.

Clause 9: "Introduction to the IEEE 1451.0 API" gives the common features of the two APIs.

Clause 10: "Transducer services API" gives the API that an application would use to utilize this standard.

Clause 11: "Module communications API" gives the API that this standard would use to communicate to the TIMs using the communications features of a physical interface defined by another member of the IEEE 1451 family of standards.

Clause 12: "HTTP protocol" is used to transfer or convey information on the World Wide Web. It is intended to provide a simpler protocol than is currently supplied by IEEE Std 1451.1-1999.

Annex: A: "Bibliography" provides references for additional information about topics referred to in this document.

Annex B: "Guidance to Transducer Services Interface" gives examples of the use of this interface for measurement and control applications that interact with the IEEE 1451.0 layer using the Transducer Services interface.

Annex C: "Guidance to Module Communication Interface" gives additional guidance for the logical communication between the NCAP and TIMs or between TIMs using the Module Communication API.

Annex D: "XML Schema for Text-based TEDS" gives the basic schemas for the Text-based TEDS defined in this standard.

Annex E: "Example Meta-Identification TEDS" gives an example of a possible Meta-Identification TEDS.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

Annex F: “Example TransducerChannel Identification TEDS” gives an example of a possible TransducerChannel Identification TEDS.

Annex G: “Example Calibration Identification TEDS” gives an example of a possible Calibration Identification TEDS.

Annex H: “Example Commands TEDS” gives an example of a possible Commands TEDS.

Annex I: “Example Location and Title TEDS” gives an example of a possible Location and Title TEDS.

Annex J: “Example Units Extension TEDS” gives an example of a possible Units Extension TEDS.

Annex K: “Examples of Physical Units” gives a series of examples of implementations of Physical Units using the representation specified in this standard.

Annex L: “TEDS read and write protocols” describes processes that may be used to write or read the TEDS.

Annex M: “Trigger logic configurations” shows some possible configurations of the trigger logic allowed in this standard.

Annex N: “Notation summary for IDL” is intended to give guidance on the use of IDL notation in this standard.

Annex O: “TEDS implementation of a simple sensor” is an example of a simple sensor implemented using the structures defined in this standard.

1.1 Scope

This project develops a set of common functionality for the family of IEEE 1451 smart transducer interface standards. This functionality is independent of the physical communications media. It includes the basic functions required to control and manage smart transducers, common communications protocols, and media-independent TEDS formats. It defines a set of implementation-independent APIs. This project does not specify signal conditioning and conversion, physical media, or how the TEDS data are used in applications.

1.2 Purpose

There are currently three approved and three proposed smart transducer interface standards in the IEEE 1451 family of standards. They all share certain characteristics, but no common set of functions, communications protocols, and TEDS formats provides interoperability among these standards. This standard will provide that commonality and will simplify the creation of future standards for different physical layers that are interoperable within the family.

1.3 Conformance

The philosophy underlying the conformance requirements of this subclause is to provide the structure necessary to raise the level of interoperability of transducers and systems built to this standard, while leaving open opportunity for continued technical improvement and differentiation.

TIM implementation shall be deemed in conformance with this standard provided the following requirements are met:

a) The TIM supports the required functional specifications identified in Clause 5.

Documents
Document Sharing Hub

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Full document details: ISO/IEC/IEEE 21450-2010

b) The TIM supports the message structures specified in Clause 6.

c) The TIM supports the required commands specified in Clause 7.

d) The TIM supports required TEDS that have the format and content specified in Clause 8.

e) The TIM supports one of the communications protocols and physical media defined by another member of the IEEE 1451 family of standards.

NOTE—Several features are highly desirable and are supported by this standard, but they are not practical to make into hard requirements. It is desirable that the sense element for a sensor be an integral part of the TIM, but for sensing elements like structural strain gages and thermocouples, this is not practical, so it has not been made into a hard requirement. In addition, it is very desirable that the TEDS be located within the TIM, but there are systems where the environment and/or the physical size make this impractical so the standard allows the TEDS to be located remote from the TIM..[1]

An NCAP implementation shall be deemed in conformance with this standard provided the following requirements are met:

— The NCAP supports the required functional specifications identified in Clause 5.

— The NCAP supports the message structures specified in Clause 6.

— The NCAP supports the required commands specified in Clause 7.

— The NCAP supports one of the communications protocols and physical media defined by another member of the IEEE 1451 family of standards.

1.3.1 Conformance keywords

Several keywords are used to differentiate among various levels of requirements and optionality, as follows.

1.3.1.1 Shall

The keyword “shall” indicates a mandatory requirement. Designers are required to implement all such mandatory requirements to ensure interoperability with other products that conform to this standard.

1.3.1.2 Shall not

The keyword “shall not” indicates a mandatory exclusion. Designers are required NOT to implement all such exclusions to ensure interoperability with other products that conform to this standard.

1.3.1.3 Recommended

“Recommended” is a keyword indicating flexibility of choice with a strong preference alternative. The word “should” has the same meaning.

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

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

1.3.1.4 Should

"Should" is a keyword indicating flexibility of choice with a strong preference alternative. The phrase *it is recommended* has the same meaning.

1.3.1.5 Should not

"Should not" is a keyword indicating flexibility of choice with a strong preference that a given alternative not be implemented.

1.3.1.6 May

"May" is a keyword that indicates flexibility of choice with no implied preference.

2. Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

ANSI X3.4-1986 (Reaff 1992), Coded Character Sets—7-bit American National Standard Code For Information Interchange.[2]

Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation, 6 October 2000.[3]

HTTP URL syntax (RFC 2616), HyperText Transfer Protocol (W3C).[4]

IEEE Std 754™-1985 (Reaff 1990), IEEE Standard for Binary Floating-Point Arithmetic.[5, 6]

IEEE Std 802.3™-2002, IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access method and physical layer specifications.

IEEE Std 1451.1™-1999, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Network Capable Application Processor (NCAP) Information Model.

IEEE Std 1451.2™-1997, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Transducer to Microprocessor Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats.

[2] ANSI publications are available from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (http://www.ansi.org/).
[3] Documents on the eXtensible Markup language can be downloaded from http://www.w3.org/TR/2000/REC-xml-20001006 or ordered from the World Wide Web Consortium, c/o MIT, 32 Vassar Street, Room 32-G515, Cambridge, MA 02139 USA.
[4] Documents describing the HTTP 1.1 Protocol can be downloaded from http://www.w3.org/Protocols/ or ordered from the World Wide Web Consortium, c/o MIT, 32 Vassar Street, Room 32-G515, Cambridge, MA 02139 USA.
[5] IEEE publications are available from the Institute of Electrical and Electronics Engineers, Inc., 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards.ieee.org/).
[6] The IEEE standards or products referred to in this clause are trademarks of the Institute of Electrical and Electronics Engineers, Inc.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

IEEE Std 1451.3™-2003, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Digital Communication and Transducer Electronic Data Sheet (TEDS) Formats for Distributed Multidrop Systems.

IEEE Std 1451.4™-2004, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Mixed-Mode Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats.

IEEE Std 1588™-2002, IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems.

ISO 639: 1988-04-01 (E/F), Codes for the Representation of Names of Languages.[7]

ISO 19136, Geographic information—Geography Markup Language (GML).

ISO/IEC 14750: 1999-03-15, Information technology—Open Distributed Processing—Interface Definition Language.[8]

3. Definitions, acronyms, and abbreviations

3.1 Definitions

For the purposes of this standard, the following terms and definitions apply. *The Authoritative Dictionary of IEEE Standards Terms* [B2][9] should be referenced for terms not defined in this clause.

3.1.1 actuator: A transducer that accepts a data sample or samples and converts them into an action. The action may be completely contained within the TIM or may change something outside of the TIM.

3.1.2 address: A character or group of characters that identifies a register, a particular part of storage, or some other data source or destination.

3.1.3 AddressGroup: A collection of TransducerChannels that respond to a single address.

3.1.4 buffer: An intermediate data storage location used to compensate for the difference in rate of flow of data or time of occurrence of events when transmitting information from one device to another.

3.1.5 calibration: The process used to determine the information that resides in the Calibration TEDS to support correction.

3.1.6 ControlGroup: Manufacturer specifications that define the inherent relationships between the TransducerChannels of a multichannel TIM. This ControlGrouping information is not normally used by the TIM itself. This information is used to identify TransducerChannels that are used to control some characteristic of another TransducerChannel. For example, a ControlGroup could be used to identify an actuator that is used to set the threshold of an analog event sensor.

[7] ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, CH-1211, Genève 20, Switzerland/ Suisse (http://www.iso.ch/). ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (http://www.ansi.org/).

[8] ISO/IEC publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, CH-1211, Genève 20, Switzerland/Suisse (http://www.iso.ch/). ISO/IEC publications are also available in the United States from Global Engineering Documents, 15 Inverness Way East, Englewood, CO 80112, USA (http://global.ihs.com/). Electronic copies are available in the United States from the American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (http://www.ansi.org/).

[9] The numbers in brackets correspond to those in the bibliography in Annex A.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

3.1.7 correction: The evaluation of a multinomial function using information from the Calibration TEDS together with data from one or more transducers.

3.1.8 data model: The numeric format in which the TIM shall output or accept data.

3.1.9 data set: The collection of samples acquired by a sensor (or applied by an actuator) in response to a trigger command.

3.1.10 data sheet: A set of information on a device that defines the parameters of operation and conditions of usage (usually produced by the device's manufacturer).

3.1.11 digital interface: A communications media and a protocol for transferring information by binary means only.

3.1.12 electronic data sheet: A data sheet stored in some form of electrically readable memory (as opposed to a piece of paper).

3.1.13 embedded transducer: A device that behaves as a transducer from the point of view of the NCAP even though nothing outside of the TIM is sensed or changed. Embedded transducers are useful for setting or reading operating parameters of other transducers.

3.1.14 enumeration: The assignment of a numeric value to a specific meaning within the context of a specific data field. Binary numbers are usually expressed in decimal terms for human convenience. Not all possible numeric values need to have a specific meaning. Values without meaning are declared to be unused or reserved for future use. Enumeration is the process of declaring the encoding of human interpretable information in a manner convenient for digital electronic machine storage and interchange. Any subclause that defines a TEDS data field to be enumerated shall contain a table that defines the meaning of the data field for each numeric value possible. The meanings encoded in each data field shall be specific and unique to that data field and only to that data field. The value becomes meaningless if not associated with the data field and its defining table.

3.1.15 event sensor: A sensor that detects a change of state in the physical world. The fact that a change of state has occurred and/or instant in time of the change of state, not the state value, is the “measurement.”

3.1.16 hot swap: The act of connecting or disconnecting a TIM from a transducer interface medium without first turning off the power that is supplied to the TIM over the medum.

3.1.17 least significant bit (lsb): The bit in the binary notation of a number that is the coefficient of the lowest exponent possible.

3.1.18 message: Information that is to be passed between devices as a single logical entity. A message may occupy one or more packets.

3.1.19 meta-: A Greek prefix meaning that which pertains to the whole or overall entity or that which is in common or shared with all member entities comprising the whole.

3.1.20 meta-TEDS: The collection of those TEDS data fields that pertain to the whole or overall entity or those that are in common or shared with all member entities (TransducerChannels) comprising the whole product.

3.1.21 multinomial: A linear sum of terms involving powers of more than one variable.

$$\sum_{i_1=0}^{N_1}\sum_{i_2=0}^{N_2}\cdots\sum_{i_m=0}^{N_m} A(i_1, i_2, \cdots i_m)\, x_1^{i_1} x_2^{i_2} \cdots x_m^{i_m}$$

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

3.1.22 network-capable application processor (NCAP): A device between the transducer modules and the network. The NCAP performs network communications, TIM communications, and data conversion or other processing functions.

3.1.23 not-a-number (NaN): As defined in IEEE Std 754-1985,[10] a bit pattern of a single-precision or double-precision real number data type that is a result of an invalid floating point operation. For single-precision real numbers, the bit pattern shall have an exponent of 255 (decimal) and a nonzero mantissa. The sign is not considered in determining whether a value is not a number.

3.1.24 octet: A group of 8 bits. (In the United States, an octet is usually referred to as a byte.)

3.1.25 packet: A block of information that is to be passed by the physical layer between devices in a single transmission.

3.1.26 sample latched: The term is used in a sensor to signal that the sample has been acquired. This may be when a sample and hold circuit switches to the hold mode or other similar operation. For an actuator, it signals that the sample has been moved to the output logic.

3.1.27 sensor: A transducer that converts a physical, biological, or chemical parameter into an electrical signal.

3.12.28 setup time: The time between the initial request for a function to be performed and when the task is actually initiated.

3.1.29 signal conditioning: The transducer signal processing that involves operations such as amplification, compensation, filtering, and normalization.

3.1.30 smart transducer: A transducer that provides functions beyond those necessary for generating a correct representation of a sensed or controlled quantity. This functionality typically simplifies the integration of the transducer into applications in a networked environment.

3.1.31 synchronization signal: For the purposes of this standard, a signal transmitted by the NCAP to provide clock or time synchronization signals to a TIM or group of TIMs. Synchronization signals are not available with all physical layer standards. Low-cost and performance TIMs may not implement a receiver for the synchronization signal.

3.1.32 transducer: A device that converts energy from one domain into another. The device may be either a sensor or an actuator.

3.1.33 transducer interface module (TIM): A module that contains the TEDS, logic to implement the transducer interface, the transducer(s) or connection to the transducer(s), and any signal conversion or signal conditioning.

3.1.34 TransducerChannel: A transducer and all of the signal conditioning and conversion components associated with that transducer.

3.1.35 TransducerChannel number: A 16 bit number assigned to an individual TransducerChannel within a TIM by the manufacturer.

3.1.36 TransducerChannel proxy: A device that is created to allow a collection of TransducerChannels to be treated as a single entity. A TransducerChannel proxy is similar to a normal TransducerChannel except

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

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 21450-2010

that it does not require a TransducerChannel TEDS, it cannot have a Calibration TEDS, Transfer Function TEDS, or Frequency Response TEDS. It may support other TEDS. A TransducerChannel proxy may respond to commands.

3.1.37 Transducer Electronic Data Sheet (TEDS): An electronic data sheet describing a TIM or a TransducerChannel. The structures of multiple TEDS are described in this standard.

3.1.38 transfer: The act or process of moving information from one digital device to another.

3.1.39 transfer function: The function that defines the response of a TransducerChannel in both amplitude and frequency.

3.1.40 trigger: A signal or message that is used to start an action.

3.1.41 VectorGroup: Manufacturer specifications that define the inherent relationships between the TransducerChannels of a multichannel TIM. This VectorGrouping information is not normally used by the TIM itself. This information is normally used by NCAP applications to properly compose human readable displays or in formulating other computations. For example, VectorGroupings may be used to indicate which TransducerChannels represent each of the three vector axes of a three-axis vector measurement.

3.1.42 virtual TEDS: A TEDS that is stored permanently in a location other than the TIM.

3.2 Acronyms and abbreviations

AD	anno domini
CRS	Coordinate Reference System
DTD	document type declaration, ref. Extensible Markup Language (XML)
HTTP	Hypertext Transfer Protocol
MJD	modified Julian date
NaN	not a number
NCAP	network-capable application processor
OMG	Object Management Group, a consortium committed to developing technically excellent, commercially viable, and vendor-independent specifications for the software industry
PHY	physical layer
SI	International System of Units, reference *The International System of Units (SI)* [B9]
TAI	International Atomic Time
TIM	transducer interface module
TEDS	transducer electronic data sheet
TSI	Transducer Service Interface
USASCII	U.S. American standard code for information interchange (ANSI X3.4-1986)
UTC	universal coordinated time
UUID	universal unique identifier
Xdcr	transducer
XML	eXtensible Markup Language

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

4. Data types

All data types used throughout the remainder of this standard are defined in subordinate subclauses.

4.1 Unsigned octet integer

Symbol: UInt8

Size: 1 octet

IDL: `typedef octet UInt8;`

This data type represents positive integers from 0 to 255.

4.2 Unsigned 16 bit integer

Symbol: UInt16

Size: 2 octets

IDL: `typedef unsigned short UInt16;`

A UInt16 may take any value between 0 and 65 535.

4.3 Signed 32 bit integer

Symbol: Int32

Size: 4 octets

IDL: `typedef long Int32;`

This data type is used to represent a signed integer from –2 147 483 648 to 2 147 483 647.

4.4 Unsigned 32 bit integer

Symbol: UInt32

Size: 4 octets

IDL: `typedef unsigned long UInt32;`

This data type is used to represent positive integers from 0 to 4 294 967 295.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 21450-2010

4.5 Single-precision real

Symbol: Float32

Size: 4 octets

```
IDL: typedef float Float32;
```

A single-precision real number is a 32 bit binary sequence that encodes real numbers as specified in IEEE Std 754-1985.

4.5.1 Floating-point NaN in TEDS

According to IEEE Std 754-1985, a single precision number with an exponent of 255 and a fractional portion (mantissa) that is nonzero is NaN regardless of the sign bit. The recommended value for use in a TEDS field for NaN is 0x7FFFFFFF (hex).

4.6 Double-precision real

Symbol: Float64

Size: 8 octets

```
IDL: typedef double Float64;
```

A double-precision real number is a 64 bit binary sequence that encodes real numbers as specified in IEEE Std 754-1985.

4.7 String

Symbol: _String

Size: variable

```
IDL: typedef string _String; // Leading '_' to escape reserved IDL
keyword
```

The use of text strings, as type string as a basic type that is defined in IEEE Std 1451.2-1997, has been replaced with the use of OMG standards for IDL.

The XML used in 8.9 is the text-based TEDS. The controlling document for XML is the W3C Recommendation *Extensible Markup Language (XML) 1.0 (Second Edition)*. All text strings in the TEDS shall conform to this recommendation or to a future update to it.

4.8 Boolean

Symbol: _Boolean

Size: 1 octet

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

```
IDL: typedef boolean _Boolean; // Leading '_' to escape reserved IDL
keyword
```

The type Boolean is a basic type that is defined in the OMG standards for IDL.

For the purposes of this standard, a bit or octet with a nonzero value is considered True. A zero value represents the False state of the variable.

4.9 IEEE1451Dot0::Args::TimeRepresentation

This abstract class defines the time representation. It is subclassed into TimeInstance and TimeDuration. The definition of the two arguments is given in Table 1.

```
IDL: struct TimeRepresentation {
   UInt32 secs;
   UInt32 nsecs;
};
```

NOTE—This format is corrected for the flaw in negative time representation discovered in IEEE Std 1588-2002.

Table 1—TimeRepresentation data structure

Parameter	Type	Description
secs	UInt32	**Seconds**—This unsigned 32 bit number represents the number of seconds from the beginning of the epoch (normally 0 hours on 1 January 1970).
nsecs	UInt32	**Sign, Nanoseconds**—This unsigned 32 bit integer comprises two smaller fields. The most significant bit will be interpreted as the sign bit of the entire time value. The least significant 31 bits represent the number of nanoseconds to be added to the value identified by the **Seconds** field before the sign is applied. The value specified in the **Nanoseconds** field is constrained to the domain 0 to 999 999 999, inclusive.

This standard uses the time representation specified in IEEE Std 1588-2002. One consequence of this choice is that time is measured in TAI seconds instead of UTC. TAI is the international standard for time based on the SI second as realized on the rotating geoid. TAI is implemented by a suite of atomic clocks and forms the timekeeping basis for other time scales in common use. Of these scales, UTC is the time scale of most engineering and commercial interest. The UTC representation is specified by ISO 8601 [B5] as YYYY-MM-DD for the date and hh:mm:ss for the time in each day.

The duration of a second in UTC time is identical to the duration of a second of TAI. UTC time differs from the TAI time by a constant offset. This offset is modified on occasion by adding or subtracting leap seconds.

Starting on 0 hours of 1 January 1972 (MJD 41 317.0), the world's standard time systems began the implementation of leap seconds to allow only integral second correction between UTC Seconds and TAI. Time in both UTC and TAI is expressed in days, hours, minutes, and seconds. Leap second corrections, which are applied to UTC but not to TAI, are made preferably following second 23:59:59 of the last day of June or December. The first such correction, a single positive leap second correction, was made following 23:59:59 on 30 June 1972, and UTC was 11 seconds behind TAI following that instant.

4.9.1 IEEE1451Dot0::Args::TimeDuration

This subclass of time representation is used to specify a time interval rather than a time value. The definition of the two arguments is the same as shown in 4.9.

```
IDL: struct TimeDuration {
```


© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

```
   UInt32 secs;
   UInt32 nsecs;
};
```

4.9.2 IEEE1451Dot0::Args::TimeInstance

Time values are represented by this struct in the IEEE 1451.0 layer. This struct is appropriate when trying to represent a time value and not time duration. A time value that occurs before the epoch is specified by a negative "nsecs" field.

```
IDL: struct TimeInstance {
UInt32   secs;
UInt32   nsecs;
};
```

The TimeInstance is based on an epoch that is defined as starting at midnight 1 January 1970. The definition of the two arguments is the same as shown in 4.9.

4.10 Data types for associated applications

The data types in this clause are not used in this standard but are provided to allow the applications using this standard to pass data in these formats.

4.10.1 Eight bit signed integer

Symbol: Int8

Size: 1 octet

```
IDL: typedef char Int8;
```

This datatype represents integers from –128 to 127.

4.10.2 Sixteen bit signed integer

Symbol: Int16

Size: 2 octets

```
IDL: typedef short   Int16;
```

This data type represents integers from –32 768 to 32 767.

4.11 Physical Units

Symbol: UNITS

Size: 11 octets

```
IDL: struct Units {
```

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

```
    UInt8 interpretation;
    UInt8 radians;
    UInt8 steradians;
    UInt8 meters;
    UInt8 kilograms;
    UInt8 seconds;
    UInt8 amperes;
    UInt8 kelvins;
    UInt8 moles;
    UInt8 candelas
    UInt8 Units Extension TEDS Access Code
};
```

Physical Units are a binary sequence of 10 octets that encode Physical Units as described in Table 2 and Table 3. Each field shall be interpreted as an unsigned integer. A unit shall be represented as a product of the SI base units, plus radians and steradians, each raised to a rational power. This structure encodes only the exponents; the product is implicit. For examples of Physical Units, see Annex J. For further information, see Hamilton [B1].

Table 2—Physical Units data type structure

Field	Description	Data type	Number of octets
1	Physical Units interpretation—see Table 3	UInt8	1
2	(2 * <exponent of radians>) + 128	UInt8	1
3	(2 * <exponent of steradians>) + 128	UInt8	1
4	(2 * <exponent of meters>) + 128	UInt8	1
5	(2 * <exponent of kilograms>) + 128	UInt8	1
6	(2 * <exponent of seconds>) + 128	UInt8	1
7	(2 * <exponent of amperes>) + 128	UInt8	1
8	(2 * <exponent of kelvins>) + 128	UInt8	1
9	(2 * <exponent of moles>) + 128	UInt8	1
10	(2 * <exponent of candelas>) + 128	UInt8	1

The U/U forms (enumerations one and three) are for expressing “dimensionless” units such as strain (meters per meter) and concentration (moles per mole). The numerator and denominator units are identical, each being specified by subfields 2 through 10.

Boolean data (values in {0, 1} or {False, True}) shall be represented as digital data (enumeration 4).

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

Table 3—Physical Units interpretation

Value	Manifest constant	Definition
0	PUI_SI_UNITS	Unit is described by the product of SI base units, plus radians and steradians, raised to the powers recorded in fields 2 through 10. Units for some quantities, such as the number of people through a turnstile, cannot be represented using these units. Enumeration zero, with fields 2–10 set to 128, is the appropriate enumeration for these cases when a quantity is being defined.
1	PUI_RATIO_SI_UNITS	Unit is U/U, where U is described by the product of SI base units, plus radians and steradians, raised to the powers recorded in fields 2 through 10.
2	PUI_LOG10_SI_UNITS	Unit is log10(U), where U is described by the product of SI base units, plus radians and steradians, raised to the powers recorded in fields 2 through 10.
3	PUI_LOG10_RATIO_SI_UNITS	Unit is log10(U/U), where U is described by the product of SI base units, plus radians and steradians, raised to the powers recorded in fields 2 through 10.
4	PUI_DIGITAL_DATA	The associated quantity is digital data (for example, a bit vector) and has no unit. Fields 2–10 shall be set to 128. The "digital data" type applies to data that do not represent a quantity, such as the current positions of a bank of switches.
5	PUI_ARBITRARY	The associated physical quantity is represented by values on an arbitrary scale (for example, hardness). Fields 2–10 are reserved and shall be set to 128.
6–255	—	Reserved.

4.12 Universal unique identification

Symbol: UUID

Size: 10 octets

```
IDL: typedef UUID Short [5];
```

The UUID is an identification field associated with the TIM whose value is unique on the planet. The UUID field shall be 10 octets long and consists of four subfields (from most to least significant order: location, manufacturer's, year, and time) defined in Table 4.

There shall be no requirement that the interpretation of the UUID reflect the actual place or time of manufacture of the TIM. The use of time and location in the algorithm shall be used only to ensure uniqueness. Thus, a manufacturer shall be allowed to use the fields defined by the algorithm as desired, provided that when interpreted according to Table 4, there shall be no other possible manufacturer that could claim use of the same UUID for a TIM.

4.13 Arbitrary octet array

Symbol: OctetArray

Size: varies

This data type comprises an arbitrary number of octets, treated as an aggregate entity that may or may not be interpreted as a number. An OctetArray can be a structure comrpising one or more primitive data types, arrays of primitive data types, or smaller OctetArrays.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

4.14 String array

Symbol: StringArray

Size: varies

This data type comprises an arbitrary number of string data types (see 4.7), treated as an aggregate entity.

4.15 Boolean array

Symbol: BooleanArray

Size: varies

This data type comprises an arbitrary number of _boolean data types (see 4.7), treated as an aggregate entity.

4.16 Array of 8 bit signed integers

Symbol: Int8Array;

Size: varies

This data type comprises an arbitrary number of octets, treated as an aggregate entity made up of 8 bit signed integers (Int8).

4.17 Array of 16 bit signed integers

Symbol: Int16Array;

Size: varies

This data type comprises an arbitrary number of 16 bit signed integers (Int16), treated as an aggregate entity.

4.18 Array of 32 bit signed integers

Symbol: Int32Array;

Size: varies

This data type comprises an arbitrary number of 32 bit signed integers (Int16), treated as an aggregate entity.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 21450-2010

Table 4—Universal unique identification datatype structure

Field	Description	Number of bits
1	Location field: The value of this field shall be chosen by the TIM manufacturer to identify a particular location on the earth, the "location,'" over which the manufacturer has physical control. This value may represent the actual location of a TIM manufacturer. A manufacturer may use different values of this field in his operation as long as all values meet the requirements of this subclause. The representation of the location field shall be 42 bits. The msb shall indicate North (asserted) or South (not asserted) latitude. The next 20 most significant bits of this field represent the magnitude of the "location" latitude as an integer number of arc seconds. The next most significant bit shall indicate East (asserted) or West (not asserted) longitude. The remaining 20 bits shall represent the magnitude of the "location" longitude as an integer number of arc seconds. Latitude magnitude values greater than 90 degrees are reserved. Longitude magnitude values of greater than 180 degrees are reserved. NOTE—One arc second at the equator is about 30 m. Thus, the range represented by each 20-bit field is 0 to 1 048 575 arcseconds or 0 to 291 degrees, which is sufficient to represent latitude and longitude on the earth's surface.	42
2	Manufacturer's field: The value of this field may be chosen by the TIM manufacturer for any purpose provided there is no interference in the use of the location field. Location field interference occurs when there is more than one manufacturer that could claim physical control over the location defined by the location field. If such interference exists, all manufacturers affected shall negotiate the use of the manufacturer's field values to resolve any interference. That is, the combination of the location field and the manufacturer's fields shall unambiguously define a specific TIM manufacturer. This negotiation shall be reopened every time there is a change in the interference.	4
3	Year field: The value of this field shall be the value of the current year. The representation of the year field shall be a 12 bit integer value. The range of this field shall be the years 0 to 4095 A.D. The beginning of the year shall be deemed to start on January 1, 00:00:00, TAI.	12
4	Time field: This value shall be chosen by the TIM manufacturer such that in combination with the location, manufacturer's, and year fields, the resulting UUID is unique for all TIMs made under the manufacturer's control. The choice of values for the time field shall be further restricted such that the values when interpreted as the time since the beginning of the year do not represent times prior to the manufacturer obtaining physical control over the "location" nor values in the future. The representation of the time field shall be a 22 bit integer. The range shall be 0 to 4 194 303. When it is necessary to interpret this field as time since the beginning of the year, the representation shall be an integer number of 10 s intervals. In this case, time values greater than one year are reserved. The beginning of the year shall be deemed to start on January 1, 00:00:00, TAI. NOTE—There are approximately 31 536 000 s per year.	22

4.19 Array of 8 bit unsigned integers

Symbol: UInt8Array;

Size: varies

This data type comprises an arbitrary number of octets, treated as an aggregate entity made up of 8 bit unsigned integers (Int8).

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

4.20 Array of 16 bit unsigned integers

Symbol: UInt16Array;

Size: varies

This data type comprises an arbitrary number of 16 bit unsigned integers (Int16), treated as an aggregate entity.

4.21 Array of 32 bit unsigned integers

Symbol: UInt32Array;

Size: varies

This data type comprises an arbitrary number of 32 bit signed integers (Int16), treated as an aggregate entity.

4.22 Array of single-precision real numbers

Symbol: Float32Array;

Size: varies

This data type comprises an arbitrary number of single-precision real numbers as specified in IEEE Std 754-1985, treated as an aggregate entity.

4.23 Array of double-precision real numbers

Symbol: Float64Array;

Size: varies

This data type comprises an arbitrary number of double-precision real numbers as specified in IEEE Std 754-1985, treated as an aggregate entity.

4.24 Array of TimeDuration data types

Symbol: TimeDurationArray;

Size: varies

This data type comprises an arbitrary number of TimeDuration data types as specified in 4.9.1, treated as an aggregate entity.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

4.25 Array of TimeInstance data types

Symbol: TimeInstanceArray;

Size: varies

This data type comprises an arbitrary number of TimeInstance data types as specified in 4.9.2, treated as an aggregate entity.

5. Smart transducer functional specification

This clause of the standard provides a description of the functions expected to be in an IEEE 1451 smart transducer. The details of these functions are provided in Clause 6 through Clause 11.

5.1 IEEE 1451 family reference model

The reference model shown in Figure 1 and Figure 2 shows the relationships among the various IEEE 1451 standards family members. They are to be used to help define what is in each of the standards in the family. It is not intended to imply a required implementation of devices that comply with the standard. Subclauses 5.1.1 through 5.1.15 give a further description of the elements shown in Figure 1 and Figure 2.

5.1.1 User's network

The user's network is outside the scope of the IEEE 1451 family of standards. It may be anything that the user desires. The only requirement that this places on the NCAP is that the NCAP have the appropriate network interface hardware and software. Different NCAPs are required for networks using different physical communications media or protocols.

5.1.2 Network access module

The network access module is the part of the NCAP that provides the interface to the user's communications network. This communications network is not defined as part of any of the IEEE 1451 family of standards. Since the communications network is not defined, the functions of this module are not fully defined by any IEEE 1451 standard. IEEE Std 1451.1-1999 provides a logical model for this module, but the use of IEEE Std 1451.1-1999 is not required for an NCAP to be in compliance with this standard or other members of the IEEE 1451 family based on this standard.

5.1.3 Transducer Services Interface

This software interface is defined in this standard in terms of an application program interface or API. See Clause 10 for the details of this interface.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.1.4 NCAP IEEE 1451.0 services

The block defines the functions and services provided to the NCAP communications module and to the NCAP applications. The functions provided by this module are defined in this standard. These functions include the command set and the TEDS. Features are included in the standard to support capabilities such as triggering and synchronous sampling of sensors, but the detailed implementation is left up to the other standards in the family.

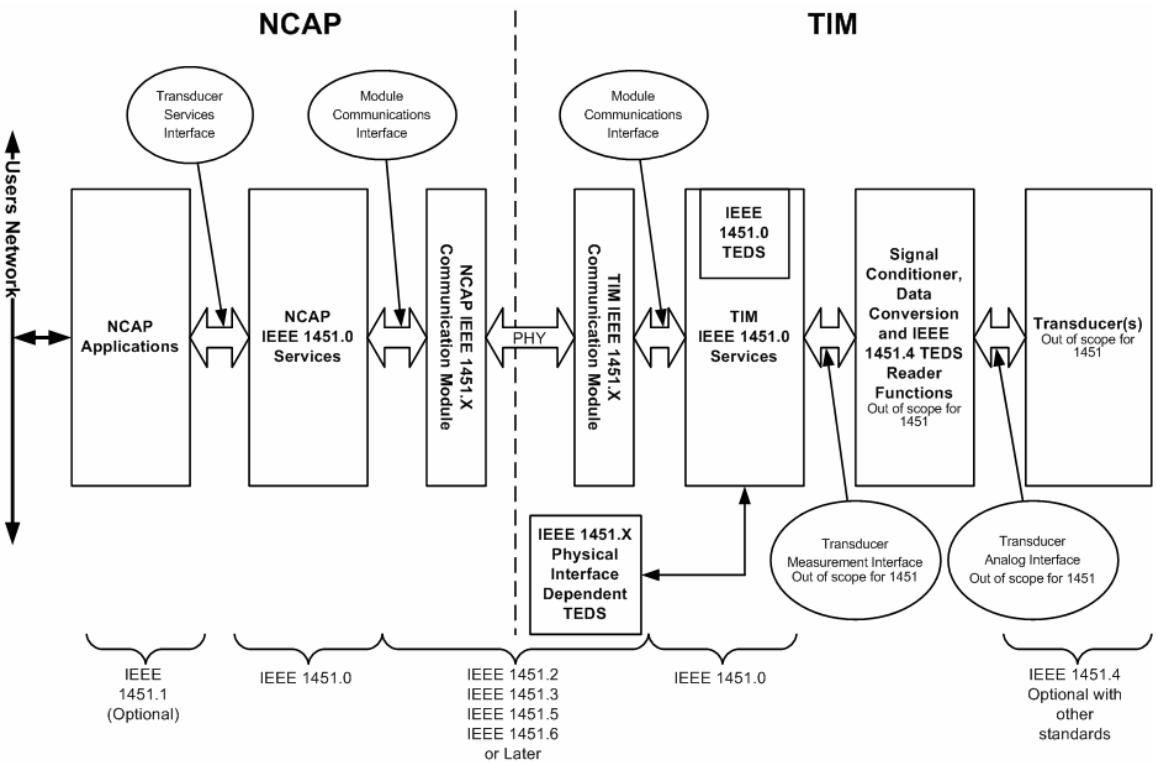


Figure 1—Reference model

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 21450-2010

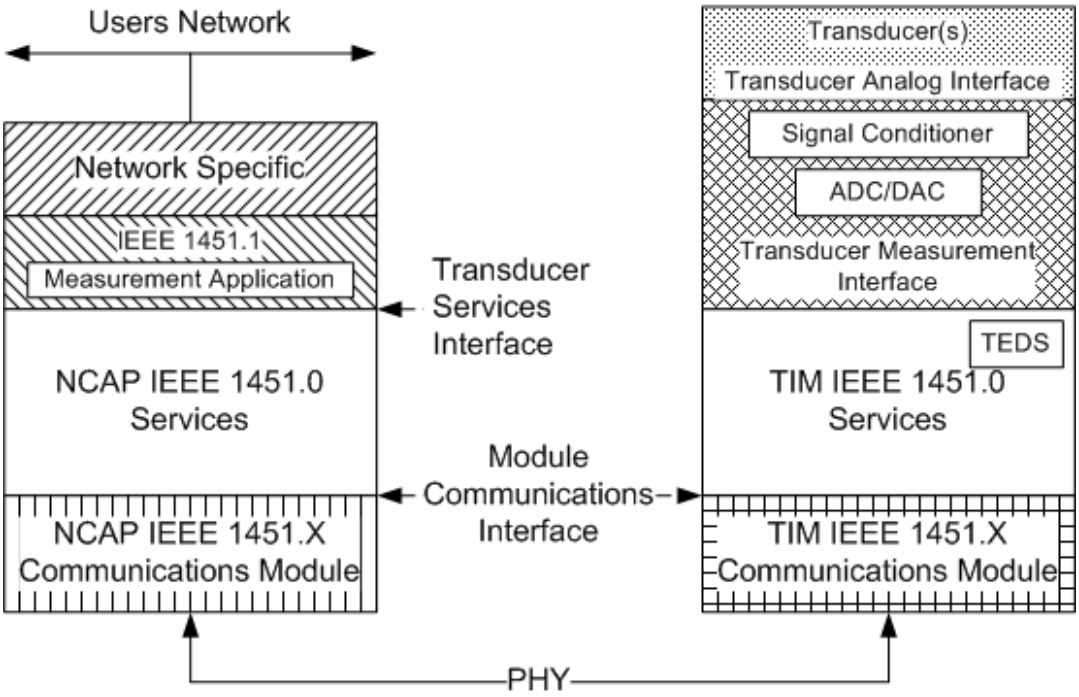


Figure 2—Alternative view of the reference model

5.1.5 Module Communications Interface

The Module Communications Interface is defined in this standard in terms of an API that is used to pass information across the interface between the NCAP IEEE 1451.0 services and the NCAP IEEE 1451.X Communication Module. The physical aspects of this interface are not described in the standard. That is left up to the manufacturers. See Clause 11 for details of this interface.

5.1.6 NCAP IEEE 1451.0/X Communication Module

Other members of the IEEE 1451 family of standards cover the NCAP communications module. IEEE Std 1451.2-1997, IEEE Std 1451.3-2003, and IEEE 1451.5™-2007 [B4] should all be examples of these standards. This module provides the low levels of the communications protocol stack and any other services that the NCAP requires to function in the system.

5.1.7 Physical interface (PHY)

The physical interface between the NCAP and the TIM is left up to the other members of the standards family to define. It is not described in this standard. Functions needed across this interface such as synchronization, if applicable, are described but not their implementation.

5.1.8 TIM IEEE 1451.X Communication Module

The TIM IEEE P1451.0/X Communication Module is logically the complement of the NCAP IEEE 1451.0/X Communication Module but not identical since there are services required in the TIM that are different than the services required in the NCAP. As with the NCAP IEEE 1451.0/X Communication

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

Module, this module is not defined in this standard but is defined by other members of the family of standards.

5.1.9 TIM IEEE 1451.0 services

This standard describes the services that are performed in this function. Most services performed by the NCAP services have counterparts within this block.

5.1.10 Transducer measurement API

This interface is left up to the manufacturer and is outside of the scope of this standard.

5.1.11 Signal conditioner and data conversion functions

The details of the signal conditioning and data conversion functions are outside of the scope of this standard.

5.1.12 Transducer analog interface

This interface is the physical interface between the signal conditioning and data conversion functions and is outside of the scope for all members of the IEEE 1451 family of standards except IEEE Std 1451.4-2004.

5.1.13 Transducer

The transducer is considered outside of the scope by all members of the IEEE 1451 family of standards.

5.1.14 TEDS

The basic TEDS are defined in this standard along with the access methods for them. However, some information that relates to the physical interface is defined in other standards in the IEEE 1451 family. These TEDS are defined in the other standards. See Clause 8 for detailed descriptions of the TEDS.

5.1.15 The use of IEEE Std 1451.4-2004 with this reference model

The IEEE 1451.0 layer is defined for the transfer of digital information between modules in a system. The IEEE 1451.0 TEDS are used to describe the entire TIM, including the transducer, signal conditioner, and data converters. The analog information that is being described as transported using these interfaces is not covered by any of this family of standards. However, IEEE Std 1451.4-2004 adds a TEDS in parallel with the analog output from a transducer. These TEDS give the characteristics of the analog information that may be obtained using an IEEE 1451.4 sensor. The use of IEEE 1451.4 transducers is not required by any other member of the IEEE 1451 family of standards. However, it is allowed as shown in this reference model. To use an IEEE 1451.4 device with an IEEE 1451.0-compatible device requires the appropriate signal conditioner for the transducer and an IEEE 1451.4 TEDS Translator. The TEDS Translator is required to combine the contents of the IEEE 1451.4 TEDS with the characteristics of the signal conditioner to allow the IEEE 1451.0 TEDS to describe the overall TIM. Both sets of TEDS are described in the standards for the family members, but the details of how to combine the TEDS are not covered in any of the standards.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

IEEE 1451.4 transducers are commonly used with interfaces that do not comply with the other members of the IEEE 1451 family of standards. In these cases this reference model does not apply.

5.2 Plug-and-play capability

A TIM(s) and NCAP that are built to this standard shall be able to be connected using a compatible physical communications media and shall be able to operate without any changes to the system software. There shall be no need for different drivers, profiles, or other system software changes to provide basic operation of the transducer. Applications that use, produce, or display the data associated with a transducer are not included in this statement. Features that go beyond the requirements of this standard are allowed, but they impact the plug-and-play capability of the TIM or NCAP.

5.3 Addresses

Two levels of addressing are used in this standard. One level of addressing is associated with the physical layer implementation. The details of this address are covered in the members of the IEEE 1451 family that define the communications media. The Module Communications API links this address to the "destId" with the discoverDestination call described in 11.3.13. This level of addressing allows messages (see Clause 6) to be sent between a TIM and an NCAP or between TIMs. For more details about this address, see the standard for the physical layer communications.

The second level of addressing is the TransducerChannel number for a given TransducerChannel within the TIM. These 16 bit numbers are used in command messages as described in 6.1.2 as the Destination TransducerChannel number or in reply messages (see 6.1.4) as a source TransducerChannel number. This level is used to specify to the TIM how the message should be directed or to tell the NCAP where it came from within the TIM. Rules are associated with the interpretation of a TransducerChannel number that are given in Table 5. Each row in the table defines a name for a class of addresses and that name is used throughout the standard.

Not all commands may be honored by all address classes. The address class that honors a particular command is defined in Clause 7 along with the definition of the commands.

Table 5—Rules for TransducerChannel number usage

Address class	Value	Description
Global	0xFFFF	Global addressing is a special GroupAddress that pertains to all TransducerChannels on the TIM.
AddressGroup	0x8000<A≤0xFFFE	There are two types of AddressGroups as defined in Table 6. The Bit Mapped AddressGroups are used when a few AddressGroups are needed and it is desirable to send a command to multiple AddressGroups at the same time. Binary AddressGroup are used when a large number of different AddressGroups are required. Address 0x8000 is a nonfunctional address.
TransducerChannel	1≤A≤0x7FFF	An address with the most significant bit set to zero. The remaining 15 bits identify the TransducerChannel for which the message is intended.
TIM	0	An address within the message of zero indicates that the message is intended for the TIM and not an individual TransducerChannel.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

Table 6—AddressGroups

Address class	Value	Description
Bit Mapped	0x8000<A≤0xBFFF	The most significant bit is set to one, the next most significant bit is zero, and at least one other bit is set to a one. The pattern in the two most significant bit is set to indicate that this is a group address. The other bit or bits identify the group(s).
Binary	0xC000≤A≤0xFFFE	The two most significant bits are set to one, and the remaining 14 bits form a binary pattern identifying a single address group.

5.3.1 Global addresses

Commands addressed to the Global address shall be received and honored by all TransducerChannels on the addressed TIM(s). If the received command is not implemented by a TIM or TransducerChannel, the command shall be ignored and no error shall be generated.

5.3.2 AddressGroup addresses

Commands issued to an AddressGroup shall be honored by all TransducerChannels that have been initialized as members of that AddressGroup. See the AddressGroup definition command in 7.1.2.3 for how to assign or clear an AddressGroup assignment. If the received command is not implemented by a TIM or TransducerChannel, the command shall be ignored and the command rejected bit shall be set in the TIM or TransducerChannel.

The use of AddressGroups across multiple TIMs requires that a method of addressing multiple TIMs be established before TransducerChannels can be assigned to AddressGroups. See 10.1.3 and 10.2.4 for the details of the method required to make these assignments.

5.3.2.1 Bit-mapped AddressGroup addresses

Each AddressGroup is represented by having the most significant bit in the AddressGroup assignment set to one, the next most significant bit set to zero, and one other bit set in the TransducerChannel number field of the address class. The one, zero pattern in the two most significant bits of the TransducerChannel number field indicates that the address is a bit-mapped group address. Using bitwise OR logic, one command may be issued to multiple AddressGroups simultaneously. This allows for 14 different AddressGroups.

5.3.2.2 Binary AddressGroup addresses

Each AddressGroup is represented by having the two most significant bits in the AddressGroup assignment set to one. The remaining 14 bits in the TransducerChannel number field of the address class identify the AddressGroup. This allows for 16 383 different AddressGroups.

5.3.3 TransducerChannel number

Command and reply messages that have a number between one and 0x7FFF, inclusive, in the TransducerChannel number field are addressed to a TransducerChannel. They shall be received and honored by the TransducerChannel to which they are addressed. If the received command is not implemented by the TransducerChannel, the command shall be ignored and an error shall be generated as specified for the individual command.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.3.4 TIM addresses

Command and reply messages that have zero in the TransducerChannel number field are addressed to a TIM. They shall be received and honored by the TIM to which it is addressed by the "destId." They are intended for the interface module and not for any particular TransducerChannel within the TIM. If the received command is not implemented by a TIM, the command shall be ignored and an error shall be generated as specified for the individual command.

5.4 Common characteristics

Subclauses 5.4.1 through 5.4.4 define the operating characteristics that are common to all TransducerChannel types.

5.4.1 Operating states

A high-level state transition diagram for a TransducerChannel is provided as Figure 3. There are two basic operating states for a TransducerChannel after it comes out of initialization. The TransducerChannel enters the "Transducer Idle" state when it comes out of initialization. A TransducerChannel accepts most commands when it is in the Transducer Idle state. A TransducerChannel enters the Transducer Operating state upon receipt of a TransducerChannel Operate (see 7.1.4.1) command and remains in this state until the TransducerChannel Idle (see 7.1.4.2) command is received.

As shown in Figure 4, a TIM has three states. A TIM is placed in the TIM Initialization state by a reset (see 7.1.7.1) command or by a power-up event. Once it completes the initialization process, it transitions to the TIM Active state. A TIM may be placed in the TIM Sleep state by the TIM Sleep (see 7.1.6.2) command. The only command that a TIM shall accept when in the TIM Sleep state is the Wake-up (see 7.1.5.1) command that returns it to the TIM Active state.

For more details about what commands may be accepted by a TransducerChannel in each state, see the detailed description of each command in Clause 7.

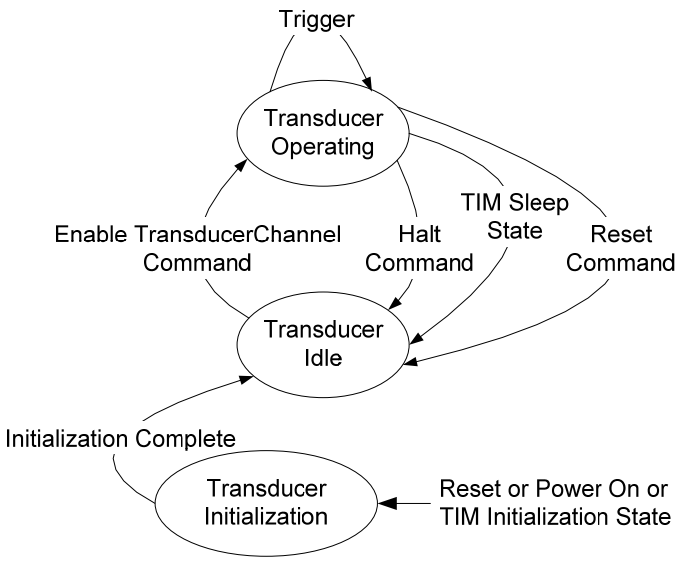


Figure 3—TransducerChannel operating states

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

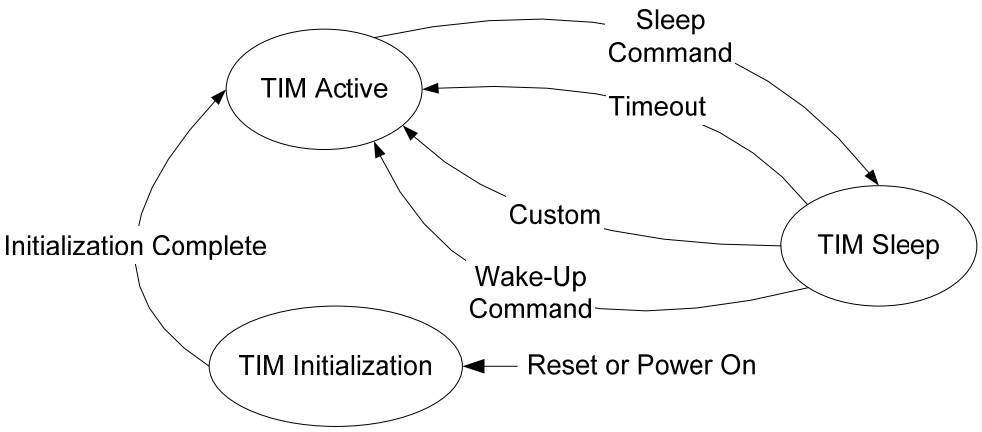


Figure 4—TIM operating states

5.4.2 Enabling and disabling triggers

A TransducerChannel may have its ability to be triggered enabled or disabled by means of commands. Unless otherwise noted, all operations described in this document are for TransducerChannels with triggers enabled.

5.4.3 Diagnostics

One prime reason for using Smart Transducers is to be able to have devices that can perform self-check or diagnostics. The standard provides a mechanism to engage diagnostics, but no requirements are imposed on the scope of the diagnostic logic.

5.4.4 Structures used to store and transmit data

Three structures are used to store and transmit data in this standard. They are the data set, the message, and the packet. The following subclauses describe each of these structures. The applications above the protocol stack deal with data sets. The upper layers of the protocol stack deal with messages. If a data set contains more octets than can be sent with a single message, it shall be broken up by the application into multiple messages.

5.4.4.1 Data sets

All TransducerChannels operate with “data sets.” Three fields within the TransducerChannel TEDS define a data set. The Maximum data repetitions field (see 8.5.2.28) defines the maximum number of individual data samples in a data set. The actual number of samples may be made lower than the number in the Maximum data repetitions field by the optional Set TransducerChannel data repetition count (see 7.1.2.1) command. When reset or powered up, the unit uses the default value. If the data repetition count is programmable, the default value shall be zero. If the data repetition count is not programmable, the default value shall be the same as the number in the Maximum data repetitions field of the TEDS. The second field is the Series increment field (see 8.5.2.30). This field is used to determine the interval between samples, and it may be overridden by a manufacturer-defined command or an embedded actuator. The third field is the Series units field (see 8.5.2.31). This field defines the units for the Series increment field. The implication of the Series units field is that the units of the Series increment field do not need to be time and that if this is the case, the time interval between samples may not be uniform.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 21450-2010

Example: The Series units are Kelvins, and the Series increment is 0.5. This combination would cause a sensor to acquire data every 0.5 degrees. The samples would be at uniform temperature intervals instead of at uniform time intervals.

5.4.4.2 Messages and packets

Messages may contain up to 65 535 octets plus the octets in headers. However, the data link and physical layers of the protocol stack transmit packets. The maximum packet length is defined in the standard defining the physical and data link layers. If a message is too long to fit within a single packet, it is the responsibility of the data link layer in the protocol stack to break messages down into multiple packets for transmission.

5.5 Transducer Electronic Data Sheets

TEDS are blocks of information in one of the formats defined in one of the subclauses in Clause 8. They are intended to be stored in nonvolatile memory within a TIM. However, there are applications where this is not practical, so the standard allows them to be stored in other places in the users system. When stored in some location other than the TIM, they are referred to as "virtual" TEDS. The manufacturer of the TransducerChannel is expected to provide the virtual TEDS but not to store them in the TIM. It is the responsibility of the user's system to provide a link between the only information that is guaranteed to be available from the TIM, that is the UUID, and the file that contains the virtual TEDS in the system. The NCAP or host processor provides this service if it is used. A TIM may provide a mixture of TEDS stored in the TIM and virtual TEDS. The response to the Query TEDS command (see 7.1.1.1) contains flags that indicate whether the TEDS is supported and whether it is virtual. A virtual TEDS is considered "supported" if the manufacturer provided a file on some suitable media containing the information, regardless of whether the user loaded it into the system.

As a general rule a TEDS is not changed once the manufacturer or the user establishes the contents of a TEDS. However, it is possible to design TransducerChannels that can change the contents of a TEDS during operation. This change can be caused by the use of an embedded actuator to change what would normally be a constant in the TEDS or by logic within the TIM or TransducerChannel that makes a change within the device requiring a change to the contents of a TEDS. To be able to handle this case, the TEDS attributes contain a bit, the "Adaptive" bit, that identifies whether the TEDS can be changed by logic internal to the TIM or TransducerChannel. A status bit is set if the contents of a TEDS are changed without writing the TEDS.

Four TEDS are required for all TIMs, and others are optional. The required TEDS are as follows:

Meta-TEDS

TransducerChannel TEDS

User's Transducer Name TEDS

PHY TEDS

See Clause 8 for the detailed specifications for all TEDS defined by this standard.

5.5.1 Required TEDS

A high-level description of the required TEDS along with the purpose of these TEDS is given in 5.5.1.1 through 5.5.1.4.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.5.1.1 Meta-TEDS

The meta-TEDS give some worst-case timing parameters that are available to be used by the NCAP to set time-out values in the communications software to determine when the TIM is not responding. The remainder of this TEDS describes the relationships between the TransducerChannels that exist within the TIM. See 8.4 for the specifications for this TEDS.

5.5.1.2 TransducerChannel TEDS

The TransducerChannel TEDS gives detailed information about a specific transducer. It gives what physical parameter is being measured or controlled, the range over which the TransducerChannel operates, the characteristics of the digital I/O, the operational mode(s) of the unit, and the timing information. See 8.5 for the specifications for this TEDS.

5.5.1.3 User's Transducer Name TEDS

The User's Transducer Name TEDS is intended to be used to provide a place for the user of the transducer to store the name by which the system will know the transducer. The structure of this TEDS is recommended in the standard, but since the contents are user defined, it cannot be required. All the manufacturer of the TIM needs to do is to provide the nonvolatile memory that the user may write using the standard TEDS access methods. See 5.5.1.3 for further details of this TEDS.

NOTE—The User's Transducer Name TEDS is intended to support "Object Tags," as defined in IEEE Std 1451.1-1999 or other similar uses.

5.5.1.4 PHY TEDS

The PHY TEDS is dependent on the physical communications media used to connect the TIM to the NCAP and is not defined in the standard although the method of accessing it is defined.

5.5.2 Optional TEDS

A high-level description of the optional TEDS along with the purpose of these TEDS is given in 5.5.2.1 through 5.5.2.10.

5.5.2.1 Calibration TEDS

The Calibration TEDS provides the calibration constants necessary to convert the output of a sensor into engineering units or to convert an engineering units value into the form required by an actuator. Two methods are supported. One is the common linear method that uses the general form of $y = mx + b$. The other method is a completely general form that can do almost anything. See 8.5.2.55 for the specifications for this TEDS.

5.5.2.2 Frequency Response TEDS

The Frequency Response TEDS uses a table to provide the frequency response of the TransducerChannel. See 8.7 for the specifications for this TEDS.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.5.2.3 Transfer Function TEDS

The Transfer Function TEDS describes a way to link a series of individual transfer functions together to describe the frequency response of a TransducerChannel in an algorithmic form. See 8.8 for the specifications for this TEDS.

5.5.2.4 Text-based TEDS

Text-based TEDS are a family of TEDS that provide text-based information about a TIM or TransducerChannel. These TEDS can be in one or more languages. They consist of a directory that facilitates access to a particular language sub-block within the TEDS followed by the blocks of TEDS that are defined using XML. See 8.9 for the specifications for this TEDS.

5.5.2.5 Commands TEDS

The commands TEDS is a Text-based TEDS that provides a way for the manufacturer to specify additional commands beyond those included in the standard. These commands are intended primarily to allow the commands necessary to set up a specific transducer and signal conditioner. Annex D contains an example schema for this TEDS, and Annex H gives an example of this TEDS.

5.5.2.6 Identification TEDS

IEEE Std 1451.2-1997 identified a Meta-Identification TEDS, a Channel Identification TEDS, and a Calibration Identification TEDS. These TEDS were intended to provide textual information about the STIM, the TransducerChannel, and the calibration. In this standard, these TEDS have all been lumped into the category of Text-based TEDS and are not individually defined. The Schema in Annex D and examples in Annex E, Annex F, and Annex G may be used to duplicate the intent of these TEDS as defined in IEEE Std 1451.2-1997. Provisions are made in this standard to access TEDS with these names.

5.5.2.7 Geographic location TEDS

The Geographic location TEDS is a Text-based TEDS that contain static geographic location information. It is expected to be written by the user to indicate the location at which the TIM is installed. The contents of this TEDS shall conform to the definition of a Text-based TEDS as specified in 8.9. The data block shall be in the Geography Markup Language (GML) developed by the Open Geospatial Consortium (OGC) and specified in ISO/DIS 19136.

The Geographic Location TEDS specifies the location of the sensor. The sensor location shall be defined in accordance with the Geography Markup Language (GML) schema for a point: “gml:point”.

Gml:point is an instance of gml:PointType that is included in the GML Schema: geometryBasic0d1d.xsd

The GML schema geometryBasic0d1d.xsd is identified by this URI:

urn:opengis:specification:gml:schema-xsd:geometryBasic0d1d:v3.1.0

Use of gml:point in this specification shall be restricted in the following ways:

gml:point shall use the DirectPositionType, i.e., gml:pos

gml:point shall not use gml:coordinates or gml:coord

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

gml:pos shall include the attribute gml:SRSReferenceGroup

Note that values of gml:pos are constrained by the Coordinate Reference System (CRS) identified in SRSReferenceGroup

gml:SRSReferenceGroup shall include srsName

gml:SRSReferenceGroup may include srsDimension

gml:SRSReferenceGroup may include gml:SRSInformationGroup

In this specification, srsName shall be given a namespace Uniform Resource Name (URN). An example for the World Geodetic System 1984 (WGS 84) CRS is as follows:

urn:ogc:def:crs:OGC:1.3:CRS84

The WGS 84 CRS contains geographic latitude, and then longitude; that is, the X axis corresponds to latitude, and the Y axis corresponds to longitude.

It is expected that the manufacturer will provide space for this TEDS and the user will provide the actual information that goes into it. The amount of memory required is a function of the actual GML location type. This standard does not specify the amount of memory to allot to this TEDS, but a typical single language block for this TEDS is expected to occupy 60 to 80 octets of memory if using a single octet for each character.

For the GeoLocTEDS, a simple approach is to specify use of "GML Point." This point specifies the location of the sensor once it has been installed. In the GML Point, a Coordinate Reference System (CRS) needs to be identified. Without a CRS, the location of the point may be inaccurate with the possibility of a 100 m difference between two geographic CRS origins. It is recommended that the writer of this TEDS restricts the specification to be an identifier of the CRS as specified on GML Point.

5.5.2.8 Units extension TEDS

There are cases, especially in chemical sensors, where it is not possible to completly express the Physical Units in SI units. For example, a sensor sensing the percentage of a certain chemical in a sample would use a ratio of units, enumeration 1 in the units type field, and the units would Moles. However, this does not identify Moles of what chemical. To solve this issue, this Text-based TEDS is provided to give a place to include text that would extend the SI units.

5.5.2.9 End User Application Specific TEDS

The End User Application Specific TEDS is much like the User's Transducer Name TEDS in that it is a block of memory that the users may use to store anything that they desire in any format that they want to use. See 8.10 for the specifications for this TEDS.

5.5.2.10 Manufacturer defined TEDS

Provisions are included in the standard to allow a manufacturer to define TEDS that are not included in the standard. The use and structure of manufacturer defined TEDS are left entirely up to the manufacturer. The manufacturer is not required to make these TEDS accessible by the user. See 8.11.2.3 for the specifications for this TEDS.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.6 TransducerChannel type descriptions

This subclause specifies the general behavior of three TransducerChannel types. Attributes in the TransducerChannel TEDS describe the capabilities of the various TransducerChannels to the system. The detailed timing and control of these TransducerChannel types are specified in later subclauses of this standard. The TransducerChannel types are as follows:

— Sensor

— Event sensor

— Actuator

The term "TransducerChannel" as used in this subclause refers to the physical transducer and all electronics in the path that connects the transducer to the communications functions.

5.6.1 Sensor

A sensor shall measure some physical parameter and return digital data representing that parameter. On the receipt of a trigger, if triggering is enabled, the sensor shall start the collection and storing of a data set within the TIM. The timing of the individual samples in the data set shall be controlled by the TIM and is a function of the operating mode of the sensor. A sensor, in the transducer operating state, shall respond to a Read TransducerChannel data-set segment command (see 7.1.3.1) by returning the appropriate data set. If a new data set is not available, the TransducerChannel shall respond with the same data that were returned on the previous Read TransducerChannel data-set segment command.

5.6.2 Event sensor

An event sensor differs from a sensor in that it does not determine the magnitude of some physical phenomena but determines when a change of state has occurred. This change of state may be an analog signal crossing some threshold or it is a set of discrete bits that match or fail to match a given bit pattern. Thus, the output of an event sensor indicates the state of its input. The two allowable states are one and zero. Two pieces of information may be determined from the output of an event sensor. One piece is the current state of the input, and the other is the time that a change in state occurred.

The TEDS definition for an event sensor is the same as for any other transducer.

5.6.2.1 Event sensor output

The data model of an event sensor output is defined in the TransducerChannel TEDS in the same way that the output of any other sensor is defined. However, the magnitude of the output shall be either zero or one.

5.6.2.2 Event time

The event sensor only reports the fact that an event has occurred. Other elements in the system are needed to determine when the event occurred. There are multiple ways that this process may be accomplished, and just which method is chosen depends on the requirements of the application. If polling is used to determine when the event occurred, the time of the event can only be known to within the polling interval. In the streaming data transmission mode, the NCAP can determine when the event occurred by the time of arrival of the message. The physical layer is a major factor in determining how much lag there is between the event and the NCAP recognition of the event. On the other end of the spectrum, the time of occurrence of

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

the event may be determined to within a microsecond or less by the use of an embedded time interval sensor or a TimeInstance sensor in the TIM with the event sensor. See M.2 and M.3 for a discussion of time interval and TimeInstance sensors.

5.6.2.3 Analog event sensors

Analog event sensors have an input signal that is in an analog form. An event is determined when the input crosses a threshold. For an analog event sensor, upper and lower thresholds are defined. The lower threshold differs from the upper by a quantity called hysteresis. The hysteresis value shall be zero or greater. During steady-state operation, a rising transition (i.e., the condition of the input that causes the output state to transition from zero to one) shall be when the output state is zero and the analog input value passes through the upper threshold. A falling transition (i.e., the condition of the input that causes the output state to transition from one to zero) shall be when the output state is one and the analog input value passes through the lower threshold. Figure 5 shows this behavior graphically. Changes of the output state shall only be reported after the device has been armed by receiving a trigger. The upper threshold and the hysteresis may be fixed at the time of manufacturing the event sensor, or they may be made programmable. The recommended method of making them programmable is to use embedded actuators to set the upper threshold and hysteresis.

Initialization of an event sensor will depend on the design. However, the following process is recommended. For Event Sensors that report only falling transitions, the output should be initialized to the one state if the analog input value is greater than or equal to the upper threshold. If the input is less than the upper threshold, the output should be initialized to the zero state. For Event Sensors that report only rising transitions, the output state should be initialized to the zero state if the analog input is less than or equal to the lower threshold. If the input is greater than the lower threshold, it should be initialized to the one state. For Event Sensors that can report both transitions, pick one of these methods and recognize that if the input is within the hysteresis band, the first transition may be missed. More elaborate schemes such as controlling the analog input value during initialization are required to avoid this problem.

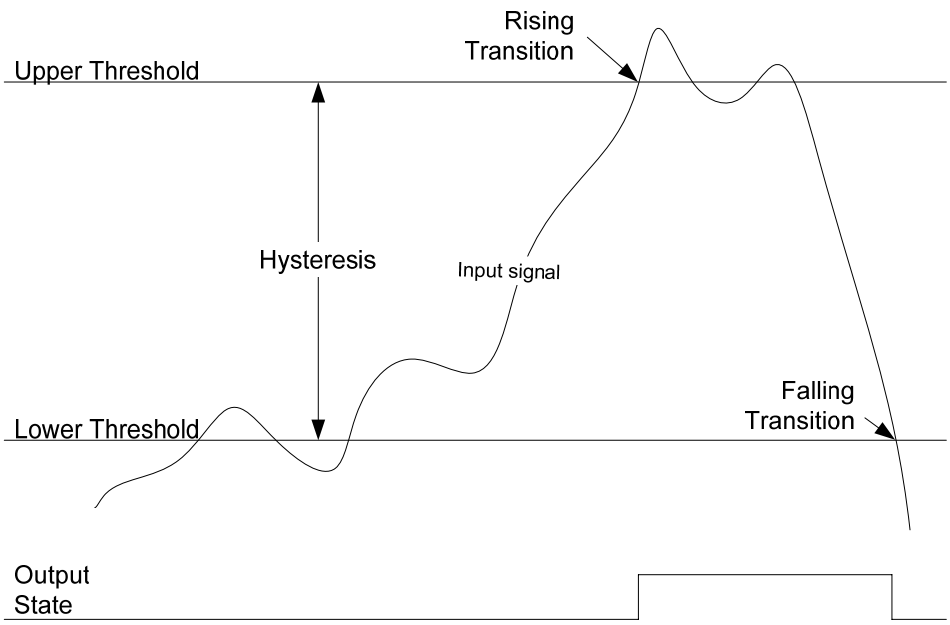


Figure 5—Analog event sensor behavior

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 21450-2010

5.6.2.4 Digital event sensors

Digital event sensors have one or more discrete input signals. An event is determined when the input matches a defined pattern for a manufacturer-defined fixed period of time. A rising transition shall be when the inputs match a specific digital pattern without change for a fixed period of time. The period of time should be determined by the manufacturer to match the characteristics of the event but may be controlled by manufacturer unique commands or an embedded actuator. A falling transition shall be when the inputs cease to match the pattern for some fixed time.

A digital event sensor may have several associated embedded TransducerChannels. It may include a sensor(s) to return the digital input on command and actuators to set up a complex event sensor. Embedded actuators may be used to define patterns, masks, time delays, and combinations of several discrete inputs that are combined into one event. The TransducerChannel TEDS for these embedded TransducerChannels shall be appropriate to their function. The function of each embedded TransducerChannel is defined by ControlGroups in the Meta-TEDS that are defined in 8.4.2.8.

5.6.2.5 Transitions reported

An event sensor may be designed to signal an event on falling transitions or rising transitions, or both. The Edge-to-Report command (see 7.1.2.9) allows the system to select which transitions to report as events. The edge-to-report capabilities of the event sensor along with the default condition are defined in the TransducerChannel TEDS (see 8.5). If the event sensor is in the transducer operating state when an event occurs, an event sensor shall set the data/event bit in the status register and shall issue a service request message if the status-event protocol is enabled (see 7.1.1.11). If status-event protocol is disabled, the data/event bit in the status register shall be set, but no message shall be transmitted.

Whether an event has occurred since the last status read may be determined by examining the data/event status bit as described in 5.13.6.

5.6.2.6 Event sensor status

An event sensor shall set the TransducerChannel data/event status bit whenever an event is detected in either the transducer operating or the transducer idle states. If the event sensor is in the transducer operating state but has not been triggered, it shall set the TransducerChannel missed data or event status bit coincident with any event that occurs. The TransducerChannel missed data or event status bit shall not be set if the event sensor is in the transducer idle state.

Consistency checks may be made by the TIM if the bit pattern is changeable for a digital event sensor, or either the upper threshold or hysteresis is changeable for an analog event sensor. Inconsistent values are signaled by setting the hardware error status bit (see 5.13.7). If inconsistent values are checked, the check shall be made immediately following a change in any of these parameters. The check shall consist of verifying the following relationships:

For analog values:

Maximum sensed input > upper threshold ≥ (upper threshold - hysteresis) > minimum sensed input value

For digital patterns: The pattern is a legal input.

NOTE—Consistency checking by the TIM shall only use data in the data model specified by the TransducerChannel TEDS of the embedded TransducerChannels.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.6.2.7 Continuous sampling event sensors

Like sensors and actuators, event sensors may be operated in the continuous sampling mode (see 5.10.1.6). In the continuous sampling mode, an event sensor in the transducer operating state shall detect and place in a buffer the fact that an event occurred without subsequent triggers.

5.6.2.8 Event sensors in the streaming data transmission mode

Like sensors and actuators, event sensors may be operated in the streaming data transmission mode. In the Streaming when a buffer is full mode (see 5.10.2.2), an event sensor in the transducer operating state shall transmit a message each time an event occurs. In the Streaming at a fixed interval data transmission mode (see 5.10.2.3), the event sensor shall store in a buffer the fact that an event or events has occurred and shall transmit the buffer at the appropriate time.

5.6.2.9 Time between events

The minimum time between events that an event sensor can detect is limited by the design of the event sensor. The minimum time between events that can be detected by the event sensor shall be specified in the TransducerChannel TEDS field, TransducerChannel sampling period (8.5.2.36).

5.6.3 Actuator

An actuator shall cause a physical or embedded output action to occur. The actuator output state changes to match the appropriate data set when a triggering event occurs. If there is more than one data point in a data set, the interval between issuing the individual data points shall be under the control of the TIM.

An actuator may be built that does not require a data set be written to it before it performs an action. This type of device always uses a default data set or no data at all and performs a predefined action upon the receipt of a trigger.

5.7 Embedded TransducerChannels

Embedded TransducerChannels are TransducerChannels whose functionality is completely contained within a TIM. An embedded TransducerChannel does not sense or control any function outside of the TIM. For example, embedded actuators may be used to set the threshold and hysteresis of an event sensor. An embedded digital event sensor may be set up to detect and report any change in the status within a TIM. Embedded TransducerChannels appear to the system as normal TransducerChannels. They respond to commands, have TEDS, and are counted when determining the number of TransducerChannels in a TIM.

NOTE—The advantage of using “embedded” TransducerChannels over control commands is that “embedded” TransducerChannels are supported by TEDS, which gives the overall system a standard way to obtain the details of operation of the TIM. They should be used when it is difficult to specify the resolution of a control command in a standard way.

5.8 TransducerChannel groups

Two types of TransducerChannel groups are defined in this standard. They are ControlGroups and VectorGroups. They are similar in implementation but are used for two different functions.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

5.8.1 ControlGroups

ControlGroups are used to define collections of TransducerChannels when one TransducerChannel is the primary channel, and the remaining TransducerChannels in the group provide either additional information about the primary TransducerChannel or are used to control some aspect of the primary TransducerChannel. For example, a ControlGroup may be used to define up to three additional TransducerChannels associated with an analog event sensor. One is a sensor that is used to measure the analog input to the event sensor. The second is an actuator that is used to set the threshold for the event sensor. The third is an actuator that may be used to set the hysteresis for the event sensor.

5.8.2 VectorGroups

VectorGroups are used to define relationships between TransducerChannels within a single multichannel TIM that imply a display or mathematical relationship between the TransducerChannels. For example, they may be used to identify the relationships between the components of a three-axis accelerometer. VectorGroups are used by the software in the NCAP or host processor to group the outputs of the individual TransducerChannels into vectors for display or computational purposes.

NOTE—VectorGroups are in some ways similar to TransducerChannel proxies. TransducerChannel proxies are used to identify TransducerChannels that are grouped together for reasons such as efficient transmission of the data and/or simultaneous triggering. Proxies may or may not represent a vector for display or computational purposes. However, it is recommended that all vectors be implemented as proxies, especially in spatial vector applications (like velocity or acceleration), where the measurements of the three components at different points in time could lead to misinterpretation.

5.9 TransducerChannel proxy

A TransducerChannel proxy is an artificial construct used to combine the outputs of multiple sensors or the input to multiple actuators into a single structure. A TransducerChannel proxy has a TransducerChannel number and may be read or written, but it does not have the other characteristics of a TransducerChannel. This means that a proxy does not have a TransducerChannel TEDS, a Calibration TEDS, a Transfer Function TEDS, or a Frequency Response TEDS. The TransducerChannel proxies that exist in a TIM are defined in the Meta-TEDS (see 8.4.2.14).

Like any other transducer, a TransducerChannel proxy is permanently assigned a TransducerChannel number by the manufacturer. Thus, a proxy may be addressed in a manner consistent with all other TransducerChannels implemented on the TIM.

A proxy shall not represent a collection of incompatible transducer types. In other words, a proxy represents a group of sensors or a group of actuators but shall not represent a group containing both transducer types.

Reading or writing the data from the individual members of a proxy may be allowed or disallowed at the manufacturer's discretion. If the manufacturer chooses to disallow reading or writing the individual members of a proxy, then the receipt of one of those commands shall set the appropriate TransducerChannel command rejected bit (see 5.13.4) in the status word.

Two methods may be used to combine the data-sets for the proxy. They are the "block" and "interleave" methods. These two methods are shown graphically in Figure 6. The block method allows the data sets to be of different lengths. With the interleave method, all data sets shall have the same size.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.

Documents
Document Sharing Hub
Full document details: ISO/IEC/IEEE 21450-2010

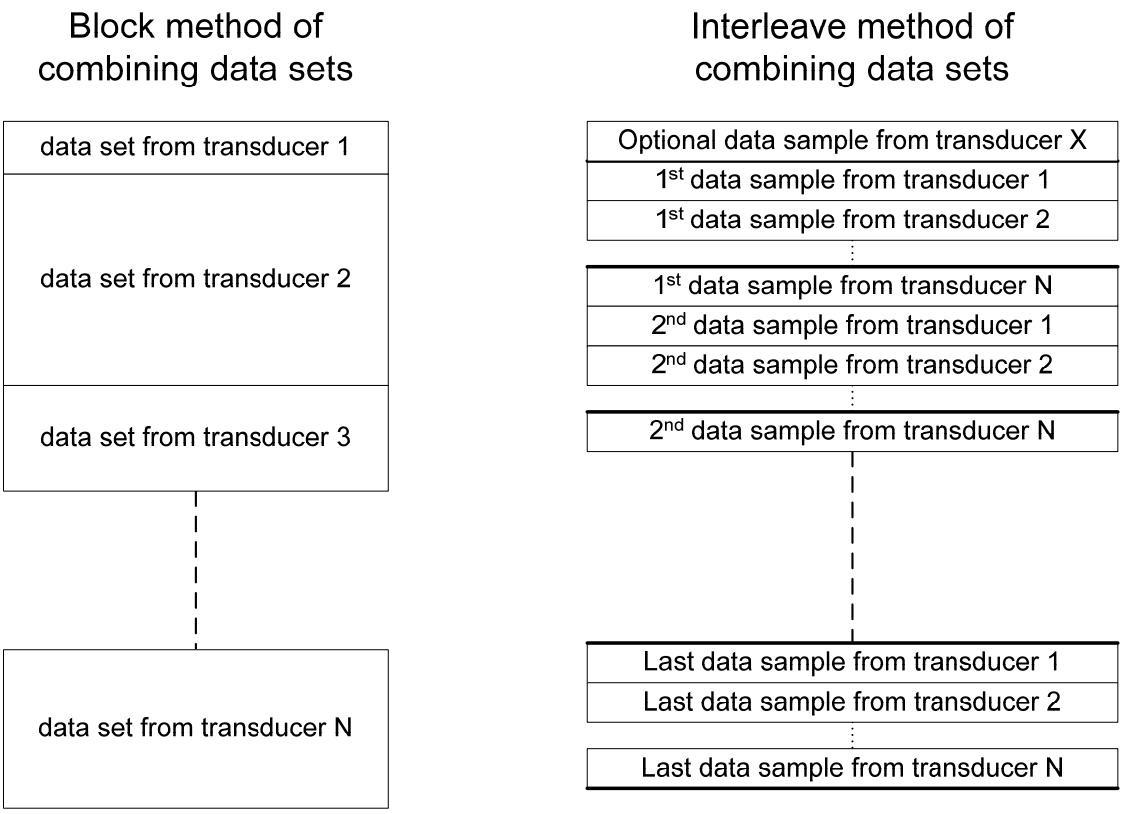


Figure 6—Methods of combining data sets

5.10 Attributes and operating modes

Subclauses 5.10.1 through 5.10.7.3 define the operating modes that are described by the attributes found in the TransducerChannel TEDS. Each attribute indicates whether a TransducerChannel supports the associated operating mode. Some of these modes of operation are mutually exclusive, and others may be enabled at the same time.

5.10.1 Sampling modes

A sensor or actuator may be operated in one of five sampling modes. The sampling modes have different relationships between the trigger and the sampling of the data by a sensor or the application of a sample to the output of an actuator. The mode or modes that a TransducerChannel is capable of operating in is defined by the attributes given in the TransducerChannel TEDS. For the possible values of these attributes, see 8.5.2.44.

The trigger-initiated and free-running sampling modes are mutually exclusive. The TransducerChannel shall be operated in one of these two modes. The other three modes are variations on these two basic operating modes.

© IEEE 2007 – All rights reserved
Authorized licensed use limited to: Tsinghua University. Downloaded on February 25,2026 at 04:34:22 UTC from IEEE Xplore. Restrictions apply.
Full document details: ISO/IEC/IEEE 21450-2010