1451.7™

IEEE

IEEE Standard for A Smart Transducer Interface for Sensors and Actuators—Transducers to Radio Frequency Identification (RFID) Systems Communication Protocols and Transducer Electronic Data Sheet Formats

IEEE Instrumentation and Measurement Society

Sponsored by the
TC9 Sensor Technology (IM/ST) Committee

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

26 June 2010

IEEE Std 1451.7™-2010

IEEE Std 1451.7™-2010

IEEE Standard for Smart Transducer Interface for Sensors and Actuators—Transducers to Radio Frequency Identification (RFID) Systems Communication Protocols and Transducer Electronic Data Sheet Formats

Sponsor

TC9 Sensor Technology (IM/ST) Committee

of the

IEEE Instrumentation and Measurement Society

Approved 25 March 2010

IEEE-SA Standards Board

Abstract: Data formats designed to facilitate communications between radio frequency identification (RFID) systems and smart RFID tags with integral transducers (sensors and actuators) are introduced in this standard. New transducer electronic data sheet (TEDS) formats for smart RFID tags, based on the IEEE 1451 family of standards, are defined. Also, a comprehensive command set for smart RFID tags is defined.

Keywords: network capable application processor (NCAP), radio frequency identification (RFID) tag, sensor command, sensor integration, sensor interface, sensor standard, smart RFID tags, transducer electronic data sheet (TEDS), transducer interface module (TIM)

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

PDF: ISBN 978-0-7381-6245-4 STD96058
Print: ISBN 978-0-7381-6246-1 STDPD96058

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 or the soundness of any judgments 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, or every ten years for stabilization. When a document is more than five years old and has not been reaffirmed, or more than ten years old and has not been stabilized, 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 his or her independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

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. A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal interpretation of the IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, 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. Recommendations to change the status of a stabilized standard should include a rationale as to why a revision or withdrawal is required. Comments and recommendations on standards, and requests for interpretations should be addressed to:

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

Introduction

This introduction is not part of IEEE Std 1451.7-2010, IEEE Standard for Smart Transducer Interface for Sensors and Actuators—Transducers to Radio Frequency Identification (RFID) Systems Communication Protocols and Transducer Electronic Data Sheet Formats.

This standard describes communication methods and data formats, and it provides a transducer electronic data sheet (TEDS) for sensors working in cooperation with radio frequency identification (RFID) systems. This document does not outline, recommend, or prescribe any specific air-interface protocol. This document is intended to be air-interface agnostic.

This standard is the seventh member of the IEEE 1451 family of standards. In the IEEE 1451 family, transducers (sensors or actuators) are connected to a transducer interface module (TIM), which is connected to a network capable application processor (NCAP) to allow network access of transducer data. IEEE Std 1451.0™-2007 [B1][a] defined a set of common functionality, commands, and TEDS for the family of IEEE 1451 smart transducer standards. IEEE Std 1451.1™-1999 [B2] defined a smart transducer object model and communication methods to facilitate the access of smart transducer in a network. IEEE Std 1451.2™-1997 [B3] defined serial interfaces for connecting transducer modules to a network processor. IEEE Std 1451.3 [B4] defined a transducer interface for distributed multidrop systems. IEEE Std 1451.4™-2004 [B5] defined a mixed-mode transducer interface that allows the transfer of digital transducer electronic data sheet and analog sensor signals on the same wires. IEEE Std 1451.5™-2007 [B6] defined a wireless communication interface for connecting transducers using various wireless communication protocols.

Notice to users

Laws and regulations

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

Copyrights

This document is copyrighted by the IEEE. It is made available for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making this document available for use and adoption by public authorities and private users, the IEEE does not waive any rights in copyright to this document.

[a] The numbers in brackets correspond to those of the bibliography in Annex F.

Updating of IEEE documents

Users of IEEE standards should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE Standards Association web site at http://ieeexplore.ieee.org/xpl/standards.jsp, or contact the IEEE at the address listed previously.

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

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: http://standards.ieee.org/reading/ieee/updates/errata/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 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 provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this standard was completed, the Sensor and RFID Integration (IM/ST/RFID) Working Group had the following membership:

Curtis L. Rozeboom, *Chair*
Kang Lee, *Vice Chair*

Paul Chartier
Farron Dacus
Randy Drago
Steve Fick
Craig K. Harmon
Bill Hoffman
Dan Kimball
Alfonso Rodriguez-Herrera
Darold Wobschall

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

Royce Beacom
Martin J. Bishop
Keith Chow
Ryon Coleman
Thomas Dineen
Carlo Donati
Randy Drago
David Droste
Lee Eccles
Sergiu Goma
Randall Groves
Werner Hoelzl
Piotr Karocki
James Kemerling
Chad Kiger
Jeremy Landt
Kang Lee
G. Luri
Gary Michel
Jeffrey Moore
Jay Nemeth-Johannes
Michael S. Newman
Ulrich Pohl
Josef Preishuber-Pfluegl
Robert Robinson
Alfonso Rodriguez
Curtis L. Rozeboom
Bartien Sayogo
James E. Smith
Matthew Smith
Joseph Stanco
Rene Struik
Walter Struppler
Mark Sturza
David Tepen
Jonathan Tucker
Stephen Webb
Oren Yuen
Janusz Zalewski

When the IEEE-SA Standards Board approved this amendment on 25 March 2010, it had the following membership:

Robert M. Grow, *Chair*
Richard H. Hulett, *Vice Chair*
Steve M. Mills, *Past Chair*
Judith Gorman, *Secretary*

Karen Bartleson
Victor Berman
Ted Burse
Clint Chaplin
Andy Drozd
Alexander Gelman
Jim Hughes
Young Kyun Kim
Joseph L. Koepfinger*
John Kulick
David J. Law
Hung Ling
Oleg Logvinov
Ted Olsen
Ronald C. Petersen
Thomas Prevost
Jon Walter Rosdahl
Sam Sciacca
Mike Seavey
Curtis Siller
Don Wright

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Satish K. Aggarwal, *NRC Representative*
Richard DeBlasio, *DOE Representative*
Michael Janezic, *NIST Representative*

Don Messina
IEEE Standards Program Manager, Document Development

Kathryn Bennett
IEEE Standards Program Manager, Technical Program Development

Contents

IEEE Standard for Smart Transducer Interface for Sensors and Actuators—Transducers to Radio Frequency Identification (RFID) Systems Communication Protocols and Transducer Electronic Data Sheet Formats

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

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

1. Overview

1.1 Scope

This standard defines data formats to facilitate communications between radio frequency identification (RFID) systems and smart RFID tags with integral transducers (sensors and actuators). The standard defines new transducer electronic data sheet (TEDS) formats based on the IEEE 1451 family of standards. This standard also defines a command structure and specifies the communication methods with which the command structure is designed to be compatible.

1.2 Purpose

The purpose of this standard is to provide methods for interfacing transducers and RFID tags, and for reporting transducer data within the RFID infrastructure. This standard will reduce the cost and time required to integrate transducer and RFID systems, as well as provide a means for device and equipment interoperability.

2. Normative references

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

ISO/IEC 19762-1-5:2008, Information Technology—Automatic Identification and Data Capture (AIDC) Techniques—Harmonized Vocabulary (all parts).[1]

3. Definitions, acronyms, and abbreviations

3.1 Definitions

For the purposes of this document, the following terms and definitions apply. *The IEEE Standards Dictionary: Glossary of Terms & Definitions* and the ISO/IEC 19762-1-5:2008 series should be referenced for terms not defined in this clause.[2] This clause provides definitions that are either not found in other standards, or have been modified for use with this standard. Also included in this clause are definitions of data types, string language codes, compact physical units, and a unique identification system.

3.1.1 1451.7 sensor: A device that responds to biological, chemical, or physical stimulus (such as heat, light, sound, pressure, magnetism, motion, and gas detection) and provides a measured response of the observed stimulus.

3.1.2 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 and not the state value, is the "measurement."

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

3.1.4 sensor: A device that responds to biological, chemical, or physical stimuli (such as heat, light, sound, pressure, magnetism, motion, and gas detection) and transmits the resulting signal or data for providing a measurement, operating a control, or both.

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

3.1.5 smart transducer: A smart transducer is 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.

3.1.6 time tick: The number of sample intervals that have occurred.

3.1.7 transducer electronic data sheet: A data sheet stored in some form of electrically readable memory, which describes a Transducer Channel.

3.2 Acronyms and abbreviations

ADC	analog-to-digital converter
AES	advanced encryption standard
AI	air interface
EUI	extended unique identifier
FIFO	first-in–first-out
LI	local index
LSB	least significant bit
LST	last sample taken
MSB	most significant bit
OUI	organizational unique identifier (allocated by IEEE)
RC	rollover count
RFU	reserved for future use
RN	random number
RFID	radio frequency identification
RFU	reserved for future use
RTLS	real-time locating system
SCap	sample capacity
SCOEOET	sample count of events outside either threshold
SF	scale factor
SFE	scale factor exponent
SFS	scale factor significand
SI	International System of Units, reference the International System of Units (SI)
SO	scale offset
SPI	serial peripheral interface
TEDS	transducer electronic data sheet
TI	total index
TIM	Transducer Interface Module
UTC	Coordinated Universal Time
VLSI	very large-scale integration
XOR	exclusive OR

3.3 Conformance

Conformance to the specifications within this standard requires that all nonoptional sections be implemented in the vendor device.

3.4 Word usage

Several keywords are used to differentiate among various levels of requirements and options, as follows:

Shall

The key word "shall" indicates a mandatory requirement. Designers are required to implement all such mandatory requirements to ensure interoperability with other products that conform to the specifications in this standard.

Recommended

Recommended is a key word indicating flexibility of choice with a strong preference alternative. The word "should" has the same meaning.

Should

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

May

May is a key word that indicates flexibility of choice with no implied preference.

4. Transducer and RFID system interface specification

This standard specifies a set of features that allow a smart transducer to communicate with the outside world using the techniques employed by RFID systems. The list below shows the four design elements that must be embodied in all smart transducers, which conform to this standard:

Essential elements of a conformant smart transducer:

- Communications protocol
- Command structure
- Transducer electronic data sheet (TEDS)
- Transducer data

The communications protocol provides the direct link between the outside world and the smart transducer. At the time of publication of this IEEE standard, Clause 5 identifies a number of the ISO/IEC Air Interface specifications that are candidates for supporting this standard; other air interfaces may also support this standard. Each air interface protocol that supports IEEE 1451.7 shall refer to it as a normative reference and implement the requirements specified in this document. The command structure is the language, specified in Clause 7, with which the actions of the smart transducer are controlled. The TEDS contains the capability and configuration information, as specified in Clause 6, for each particular smart transducer. The transducer data, also described in Clause 6, constitute the results of sensor measurements.

Figure 1 provides an example of the minimum components required in a system interfaced with an IEEE 1451.7-compliant transducer. The elements common to all IEEE 1451.7-compliant systems are shaded.

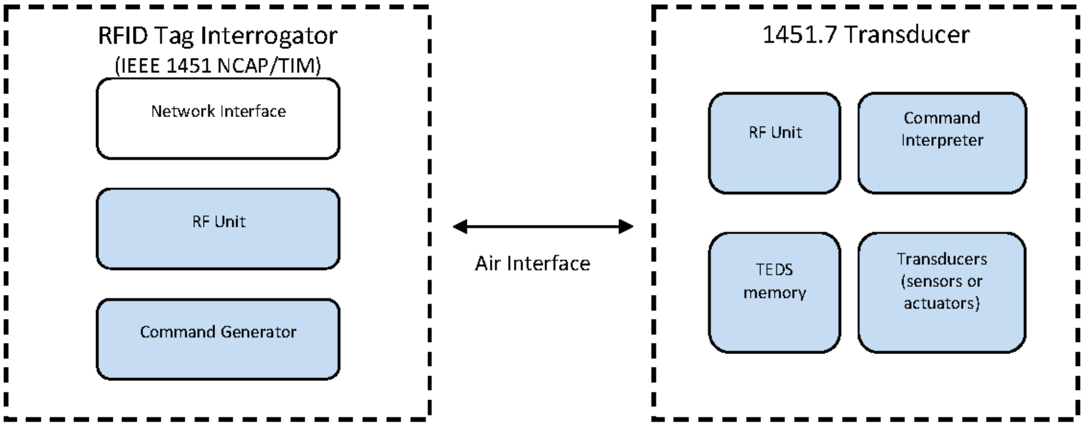


Figure 1 —RFID system interface with smart transducer

For clarity, Figure 1 does not show the numerous other hardware elements common to generic RFID tag interrogators and IEEE 1451.7 transducers.

Input from the RFID Tag Interrogator or its network interface is processed by the Command Generator, and then transmitted by its RF Unit. Within the IEEE 1451.7 Transducer, the Command Interpreter processes radio signals sent to it from the RF Unit, into low-level control signals. These signals are required for the various functions of which the particular smart transducer is capable. As a minimum requirement, every 1451.7-compliant Transducer shall be able to retrieve data from the TEDS memory, and all data records.

5. Air interface applicability [RFID and real-time locating system (RTLS)]

The IEEE 1451.7 command structure supports the air interface communications protocols described in the following ISO/IEC standards; additional air interface protocols may be added by declaring compliance with this standard:

- ISO/IEC 18000-2:2009 [B9],[3] Information Technology—Radio Frequency Identification for Item Management—Part 2: Parameters for Air Interface Communications below 135 kHz.
- ISO/IEC 18000-3:2009 [B10], Information Technology—Radio Frequency Identification for Item Management—Part 3: Parameters for Air Interface Communications at 13.56 MHz.
- ISO/IEC 18000-4:2008 [B11], Information Technology—Radio Frequency Identification for Item Management—Part 4: Parameters for Air Interface Communications at 2.45 GHz.
- ISO/IEC 18000-6:2006 [B12], Information Technology—Radio Frequency Identification for Item Management—Part 6: Parameters for Air Interface Communications at 860 MHz to 960 MHz.
- ISO/IEC 18000-7:2009 [B13], Information Technology—Radio Frequency Identification for Item Management—Part 7: Parameters for Active Air Interface Communications at 433 MHz.
- ISO/IEC 24730-2:2006 [B14], Information Technology—Real-Time Locating Systems (RTLS)—Part 2: 2.4 GHz Air Interface Protocol.
- ISO/IEC 24730-5:2000 [B15], Information Technology Automatic Identification and Data Capture Techniques—Real Time Locating Systems (RTLS)—Part 5: Chirp Spread Spectrum (CSS) at 2.4 GHz.

[3] The numbers in brackets correspond to those of the bibliography in Annex F.

6. Sensor security and data structures

IEEE 1451.7 sensors are designed on a hierarchical structure with the following two levels:

a) The sensor type (e.g., temperature or relative humidity) determined by its physical design

b) The data type defined by the measured or derived data extracted from the sensor (e.g., present sampled value, or a count of occurrences of out-of-limit samples, or a detailed data log)

Effectively, the hierarchy separates the functions of which an IEEE 1451.7 sensor is capable from the type of data that are extracted.

6.1 Overview

This standard specifies sensor types and scaling for RFID sensor data acquisition, files that control data acquisition and hold desired sensor data, and commands for accessing data files. The data available for extraction from each sensor shall consist of the following:

- The sensor identifier (6.3)
- The sensor characteristics record (6.4)
- The sample and configuration record (6.5)
- The event administration record (6.6)
- The event record (6.7)

6.2 Sensor security system basic operation

This standard specifies optional security in two forms, which are *air interface security* and *direct sensor security*. Either or both of these may be used, and there are options within the two forms.

Air Interface Security provides methods for the sensor to take advantage of the security built into a particular RFID air interface. It operates by the tag passing a security status code to the sensor informing the sensor of the security state of the tag. The sensor then appropriately limits its command execution according to a security function code programmed by the user. If the sensor is deeply embedded within the tag electronics, then this method of sensor security is as secure as the air interface security. However, air interface security has limitations in that it is not available for all air interfaces, is commonly not fully secure in a cryptographic sense, is sometimes “one way” with only the reader authenticating to the tag, and it may be physically defeated for modular sensors.

To overcome these problems, direct sensor security is also specified. This standard provides for sensor security ranging from a simple password system for reader-only authentication, to encrypted two-way authentication of reader and sensor, to authentication of reader and sensor on each command/response exchange, to encryption of data flow in the link. There is a high degree of flexibility provided, such as a choice of encryption algorithms and the opportunity to specify different algorithms and different keys for authentication and data encryption (different keys must be supported if different algorithms are allowed for authentication and data encryption because the key length can differ). Other flexibility includes reader-commanded random number (RN)/security token lengths that may be different for each link and that may change from initial authentication to continuing authentication of an established session. Authentication and encryption may also be independently commanded as applying to either or both links. Under this system, optional encryption capability is required to authenticate the tag to the reader. The reader may, if desired, authenticate itself to the tag without encryption using only a password.

The type of security supported by the sensor is reported in the TEDS. Because this standard allows both Air Interface Security and Direct Sensor Security to be implemented simultaneously, separate fields for security capability description are provided for each. If the two security function codes are programmed to different levels, the sensor will default to the security level of the more secure (it will limit command set execution for both to that specified by the most limiting). The two passwords/keys may be different, and if different, both will be required to access the sensor.

6.2.1 Air interface security system support

The air interface security may be supported through a security status reporting system (using the "Tag Security Status Code") where the tag reports to the sensor the security state it is in, and the sensor responds appropriately. Typically, the Tag Security Status Code reports to the sensor whether the tag has "authenticated" that the reader is authorized for some degree of sensor access via a password method (improved air interface security using encryption-based authorization is under development). A 3-bit Tag Security Status Code is provided for this function, as defined in Table 12. The code from this table is preappended to the payload of the air interface "transport command" that moves sensor commands as payload to the sensor controller. The sensor controller interprets how to respond to the Tag Security Status Code of the tag as defined in the Air Interface Security Function Code of Table 14. The Air Interface Security Function Code is basically a user-programmed choice of the air interface security capabilities as defined in the Air Interface Security Capabilities Code field of TEDS Table 2. It operates by specifying the classes of command that the sensor may execute, for example allowing read operations but not write operations without having passed air interface security. See 6.5.7 for specific information on the command classes.

6.2.2 Direct sensor security system support

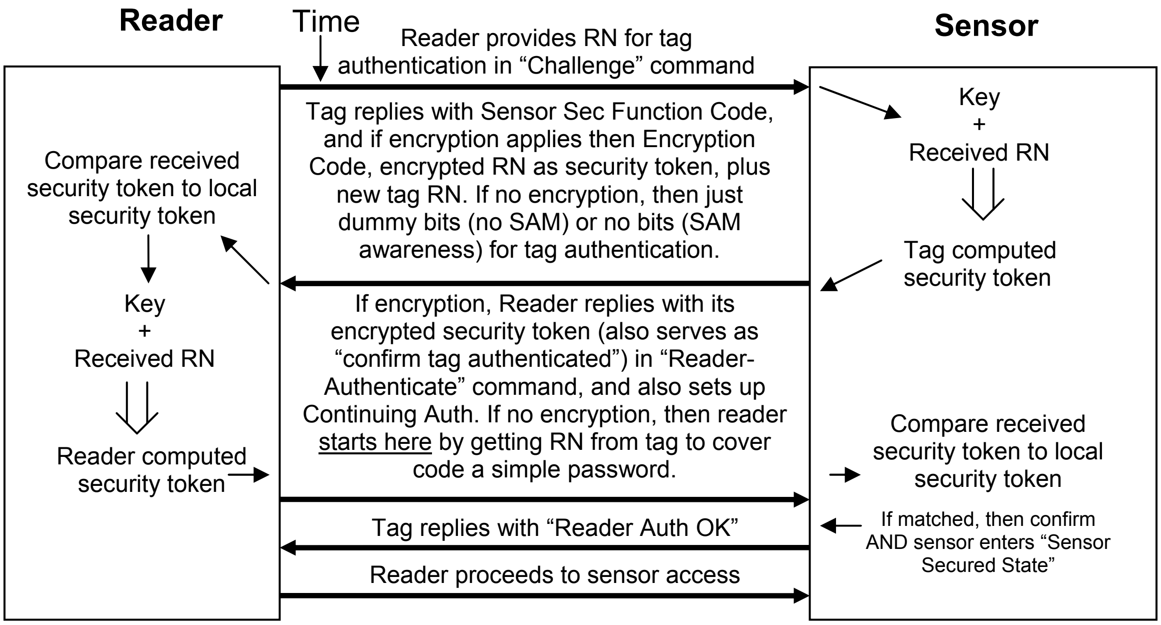
The sensor manufacturer and the user through programming options may extend or replace the air interface security with direct or embedded sensor security. Such security may share a password or key(s) with the air interface or may be different. The "password" used for one-way authentication is replaced by a "key" or keys when encrypted two-way, two-way authentication is used. If both security methods are supported, then the sensor will assume the higher security level of the two Security Function Codes (Field 6 and Field 7) programmed in the Sample and Configuration Record of Table 11. The Sensor Security Function Code (Field 7) of Table 11 is a user-selected choice from the options available under the detailed descriptions of Table 6.

The direct sensor security can be as simple as the reader authenticating itself to the sensor via a simple password. This standard provides a structure and a command set that can allow for "Two-Way Authentication," meaning that both Reader and Sensor can privately authenticate that the other has a secret key. Passwords may currently be 16, 32, 64, or 128 bits long. Keys may also be any of these lengths, and also they may be any number if a specific key length is specified as always associated with a particular encryption algorithm. An algorithm in this standard may also be left open with respect to length, and then the particular length specified in the TEDS of Table 2.

The two-way authentication is implemented with a private key encryption system. This requires not only selection of particular encryption or hashing algorithms but also random number generation on both reader and sensor. For each authentication, random numbers are generated by each side (reader or tag) to be used in combination with the secret key to generate constantly changing "security tokens" (an encryption function output that is a function of both the key and the new random number) that prove the authenticity of the side receiving the random number and generating the security token. Because the secret and untransmitted key is essential to generating the security tokens, the random numbers are transmitted in plaintext (unless the full communication is encrypted). For example, for the sensor to authenticate the reader, the sensor provides the reader with a random number. The reader encrypts or hashes that random number with the key (both are inputs to the encryption function) and sends the encrypted result (reader security token) back to the sensor. The sensor has also run the same random number and key through the encryption algorithm and obtained an internal result that it can compare to the result provided by the reader. If they match, this proves to the sensor that the reader has the key. This result was achieved without ever having to transmit the key where it could be intercepted by an eavesdropper. This process may be extended to encryption of regular message traffic, and not just authentication.

The flow of events to perform two-way authentication is shown in Figure 2. If encryption is not available or not selected for use, then reader skips the tag authentication part of the process. In that case, the first step is for the reader to request a random number (Request RN command) from the sensor to use in "exclusive OR (XOR) cover coding" a transmission of the reader authenticating password to the sensor (via the Reader-Authenticate command). Because the random number used for cover coding was transmitted in the clear where it could be intercepted, and in this case, there is no encryption (the openly transmitted RN itself serves as a kind of key to "uncover" the password), this is a very modest degree of security. The only reason it provides modest security is that the tag transmission of the RN is usually a low-power transmission, such that an eavesdropper must be within a fairly limited range to perform the intercept. It is suitable to prevent casual access to the sensor by unauthorized readers, but it is not suitable to prevent determined access. This same situation applies to air interface security where the reader provides a cover coded password to the tag.

If a sensor supports two-way authentication, it may drop back to using this limited one-way reader-only authentication where the key serves as a password. If this operation is desired, it is commanded by the reader via the Encryption Usage flag in the Reader-Authenticate command. If "Continuing Authentication" whereby each command/response exchange is independently authenticated is supported, then that authentication may be two-way or one-way in either direction as chosen by the reader. If encryption of data flow is supported, then it may also be selected as two-way or one-way in either direction. More specific information is provided in the data structures and commands, and in particular in the description of the Reader-Authenticate command.


Reader
Time
Sensor
Reader provides RN for tag authentication in "Challenge" command
Tag replies with Sensor Sec Function Code, and if encryption applies then Encryption Code, encrypted RN as security token, plus new tag RN. If no encryption, then just dummy bits (no SAM) or no bits (SAM awareness) for tag authentication.
Key + Received RN
Tag computed security token
Compare received security token to local security token
Key + Received RN
Reader computed security token
If encryption, Reader replies with its encrypted security token (also serves as "confirm tag authenticated") in "Reader-Authenticate" command, and also sets up Continuing Auth. If no encryption, then reader starts here by getting RN from tag to cover code a simple password.
Compare received security token to local security token
Tag replies with "Reader Auth OK"
If matched, then confirm AND sensor enters "Sensor Secured State"
Reader proceeds to sensor access

NOTE—SAM stands for security accounts manager.

Figure 2—Two-way secure authentication with exit points

6.3 Sensor identifier

Each sensor shall have a 64-bit (an 8-octet array) sensor identifier (ID), which is compliant with the global identifier format defined in the IEEE 64-bit extended unique identifier (EUI-64) as shown in Table 1. The most significant 3 or 4 octets of a EUI-64 are the 24-bit or 36-bit Company_ID [organizational unique identifier (OUI)], which is obtained from the IEEE registration authority. The IEEE administers the assignment of OUI-24 or OUI-36 Company_ID. The least-significant 28-bit or 40-bit extension identifier is assigned by the manufacturer. The Sensor ID is permanently encoded by the sensor manufacturer. This is in addition to any unique identifier(s) that the RFID tag might contain.

Table 1 —Sensor ID

Company_ID	**Extension identifier**
OUI-24 24 bits	40 bits
OUI-36 36 bits	28 bits

Guidelines for the 64-bit global identifier (EUI-64) registration authority can be found at http://standards.ieee.org/regauth/oui/tutorials/EUI64.html.

For example, assume that a manufacturer's IEEE-assigned OUI-24 Company_ID value is ACDE48$_{16}$ and the manufacturer-selected extension identifier for a given component is 234567ABCD$_{16}$. The EUI-64 value generated from these two numbers is ACDE48234567ABCD$_{16}$, whose byte and bit representations are illustrated in Figure 3.

Company_ID			**Extension identifier**					**field**
addr+0	addr+1	Addr+2	addr+3	addr+4	addr+5	addr+6	addr+7	order
AC	DE	48	23	45	67	AB	CD	hex
10101100	11011110	01001000	00100011	01000101	01100111	10101011	11001101	bits
most significant bit (MSB)					least significant bit (LSB)			

Figure 3—OUI-24 example

For example, assume that a manufacturer's IEEE-assigned OUI-36 Company_ID value is 8765432AB$_{16}$, and the manufacturer-selected extension identifier for a given component is 567ABCD$_{16}$. The EUI-64 value generated from these two numbers is 8765432AB567ABCD$_{16}$, whose byte and bit representations are illustrated in Figure 4.

Company_ID					**Extension identifier**				**Field**
addr+0	addr+1	Addr+2	addr+3	addr+4	addr+4	addr+5	addr+6	addr+7	order
87	65	43	2A	B	5	67	AB	CD	hex
10000111	01100101	01000011	00101010	1011	0101	01100111	10101011	11001101	bits
MSB					LSB				

Figure 4 —OUI-36 example

6.4 Sensor characteristics TEDS (Type 1)

The TEDS is the primary means of identifying the sensor's functional capability. The structure of the primary sensor characteristics is defined in Table 2,[4] with further detailed specifications in 6.4.1 through 6.4.18.

Table 2 —Primary Sensor Characteristics TEDS (Type 1)

Field	Name	Size	Reference	Example/note
1	TEDS type	3 bits	6.4.1	001_2.
2	Sensor type	7 bits	6.4.2	0001110_2 = Relative humidity.
3	Units extension	5 bits	6.4.3	Subtype, e.g., for chemical sensors.
4	Sensor map	16 bits	6.4.4	
5	Data resolution	5 bits	6.4.5	Sensor capability.
6	Scale Factor Significand	11 bits	6.4.6	Sensor capability.
7	Scale Factor Exponent	6 bits	6.4.6	Sensor capability.
8	Scale Offset Significand	11 bits	6.4.6	Sensor capability.
9	Scale Offset Exponent	6 bits	6.4.6	Sensor capability.
10	Data uncertainty	3 bits	6.4.7	Sensor capability.
11	Sensor Reconfiguration Capability	1 bit	6.4.8	0 = NO, 1 = YES.
12	Memory Rollover Capability	1 bit	6.4.9	0 = NO, 1 = YES.
13	Air Interface Security Capability Code (see NOTE 1)	3 bits	6.4.10	See Table 5 for details.
14	Sensor Security Capability Code (see NOTE 1)	3 bits	6.4.11	000_2 = No Direct Sensor Security. If greater than zero, then at least one-way password security is supported. If greater than zero and at least one authentication encryption algorithm is supported, then two-way initial encrypted authentication is also supported. See the Continuing Authentication Capability Field of this table for directions supported when continuing to authenticate each command. See Table 6 for Sensor Security Capability Code assignments.
15	Sensor Authentication Encryption Capability Map	7 bits	6.4.12	Choices of encryption algorithms for authentication that the sensor supports. If all zeroes, then encrypted authentication is not supported.
16	Sensor Data Encryption Capability Map	7 bits	6.4.13	Choices of encryption algorithms for data that the sensor supports. If all bits are zero, then data encryption is not supported. If data encryption is supported, the directions supported are detailed in the Data Encryption Capability Field of this table.
17	Sensor Authentication Password/Key Size (see NOTE 2)	3 bits	6.4.16	000_2 = 16 bits, 001_2 = 32 bits, 010_2 = 64 bits, 011_2 = 128 bits, 100_2 ~ 111_2 = reserved for future use (RFU).

[4] Notes in text, tables, and figures of a standard are given for information only and do not contain requirements needed to implement this standard.

Field	Name	Size	Reference	Example/note
18	Sensor Data Encryption Key Size (see NOTE 3)	3 bits	6.4.17	000_2 = 16 bits, 001_2 = 32 bits, 010_2 = 64 bits, 011_2 = 128 bits, 100_2 ~ 111_2 = RFU.
19	Random Number Sizes Supported (see NOTE 4)	3 bits	6.4.14	000_2 = 16 bits, 001_2 = 16 & 32 bits, 010_2 = 16, 32 & 64 bits, 011_2= 16, 32, 64, & 128 bits, 100_2 ~ 111_2 = RFU.
20	Continuing Authentication Capability Field (see NOTE 5)	2 bit	6.4.15	See Table 9 for details.
21	Data Encryption Capability Field	2 bit	6.4.18	Table 10 for details.
22	Clock Accuracy (see NOTE 6)	3 bits		00: >10% 001: 10% 010: 5% 011: 2% 100: 1%. 101: 300 ppm 110: 100 ppm 111: <100 ppm
23	RFU	17 bits		
TOTAL		128 bits		

NOTE 1—If the air interface and sensor security systems are both supported and if Security Function Codes based on the capability codes are programmed to different levels, then the more secure mode shall apply to how the sensor processes commands.

NOTE 2—For sensor authentication, the term "password/key" is used instead of simply "key" because this field functions as a key if the sensor has authentication encryption but as a password if it does not. Although sensor authentication password/key sizes are 16 bits and greater, if Sensor Security Capability Code is 000, then there is no password or key and this overrides the password/key length field. It is possible via the Sensor Authentication Encryption Capability Map to select a standardized encryption algorithm with a key length that overrides the key length field. The key length field is to allow algorithms that support multiple key lengths to specify the particular length the tag uses. For example, the advanced encryption standard (AES) can have key lengths of 128, 192, and 256 bits (although only 128 is specified for this standard version).

NOTE 3—Although sensor data encryption key sizes are 16 bits and greater, if the Sensor Security Capability Code is 000, then there is no key and this overrides the key length field. It is possible via the Sensor Data Encryption Capability Map to select a standardized encryption algorithm with a key length that overrides the key length field.

NOTE 4—A random number generator is needed to support authentication, which it does by providing a continuously changing number to encrypt into a security token that proves the key is possessed. The supported random number sizes in this version are all or a subset of 16, 32, 64, and 128. Although the random number size is 16 bits and greater, if the Sensor Encryption Capability Code is 000, then there is no random number generator, and this overrides the Random Number Size field. The actual random number sizes to be used by each side of the link are provided by the Challenge command for the initial authentication, and by the Reader-Authenticate command for Continuing Authentication. See NOTE 5.

NOTE 5—Continuing Authentication is an optional ability to authenticate all commands and responses individually, as opposed to a single authentication where it is assumed that following authentication commands are not subject to hostile action.

NOTE 6—Clock accuracy applies to logged data and, if supported, then also to the Secure Session Timer of Table 16. The range of values shown is suitable for the two main classes of reference sources, which are free running relaxation (RC) oscillators (trimmed and untrimmed) and low-cost, low-power crystal timers (such as standard 32.768 kHz watch crystal-based Real Time Clocks).

The manufacturer shall permanently lock the Primary Sensor Characteristics TEDS.

6.4.1 TEDS type

This field identifies the TEDS format. For the purpose of this standard, the only relevant code is 001_2 for the Primary Sensor Characteristics TEDS.

NOTE—The total number of TEDS formats that may be specified in this standard is nominally limited to 8 (but that may be extended using the type 2 or 3 TEDS).

6.4.2 Sensor type

The Sensor Type field defines the International System of Units (SI) unit, or derived SI unit that the sensor is capable of monitoring. The 7-bit sensor type code value is all that is required for sensor data processing to identify the associated unit and to process the data.

Annex A lists the code values that are assigned and supported by this standard.

6.4.3 Units extension

The only units extension that has been defined in this standard is for chemical substances. These qualify the sensor type with a unit of measure of *amount of substance (Moles)*, or parts per million.

The code values are defined in Annex B.

6.4.4 Sensor map of supported measurement codes

The Sensor Map field indicates the capability of the sensor to provide data of the types defined in Table 3. Each sensor shall support at least one type of data. The code value represents the position in the field of a bit set to 1 to indicate that the sensor supports the particular type of measurement. As read from left to right, the first position is designated by code value 0.

Code values 0 through 9 denote measurement types for which only one measured value is returned in response to a request for data. Code values 10 through 13 denote measurement types for which multiple data values can be returned by the sensor.

The number of bits in each measured value size of data is determined by the code value for Field 5 of the Primary Sensor Characteristic TEDS (Table 2).

The bit sequence (for Table 3) is mapped from left to right and it indicates the presence (with a 1) or absence (with a 0) of a measurement capability. Examples are listed after Table 3.

Table 3 —Supported measurement type codes

Code value	Measurement type	Single/multiple	Field size in event record
0	Present (point-of-time) value	Single	Data = 1 to 32 bits (see 6.4.5)
1	Maximum (or peak) value	Single	Data = 1 to 32 bits (see 6.4.5)
2	Minimum (or lowest) value	Single	Data = 1 to 32 bits (see 6.4.5)
3	Average value	Single	Data = 1 to 32 bits (see 6.4.5)
4	Variance	Single	Data = 1 to 32 bits (see 6.4.5)
5	Standard deviation	Single	Data = 1 to 32 bits (see 6.4.5)
6	Observed value at a predetermined sample time	Single	Data = 1 to 32 bits (see 6.4.5) Also requires specific sample count value (see 6.6.7)
7	Sample count and data value upon sample time after alarm has tripped	Single	Data = 1 to 32 bits (see 6.4.5), and specific sample count value (see 6.6.8)
8	Count of readings over maximum threshold value	Single	Data = 1 byte with the value incremented by 1 for each event
9	Count of readings below minimum threshold value	Single	Data = 1 byte with the value incremented by 1 for each event
10	Data log of observed value recorded at each sample interval	Multiple, record for each event	Data = 1 to 32 bits (see 6.4.5)
11	Data log of observed value with time tick (8-bit code) reporting outside either threshold	Multiple, record for each event	Time tick = 1 byte Data = 1 to 32 bits (see 6.4.5)
12	Data log of observed value with time tick (16-bit code) reporting outside either threshold	Multiple, record for each event	Time tick = 2 bytes Data = 1 to 32 bits (see 6.4.5)
13	Data log of all observed values after an initial alarm value has triggered	Single record for time stamp and multiple record for each event	Sample data = 1 to 32 bits (see 6.4.5) Also requires specific sample count value (see 6.6.10)
14	Temperature Time Integration	TBD[a]	TBD
15	RFU		

[a] TBD = to be determined.

EXAMPLE
Bit map 1110000000010000 indicates that the sensor supports the following measurement types:

- Present (point-of-time) value
- Maximum (or peak) value
- Minimum (or lowest) value
- Observed value with time tick (8-bit code) reporting outside either threshold

6.4.5 Data resolution

The Memory Resolution field identifies the numeric data format that is used to convey sensor data for Measurement Codes 0 to 13. The data format shall be the number of bits, from 1 to 32, of output data defined by the manufacturer. In most cases, this will be equivalent to the bit output of the analog-to-digital converter (ADC), but may also be truncated.

The 5-bit code in Field 5 of the Primary Sensor Characteristics TEDS shall have the value 00000_2 for a 1-bit output and the value 11111_2 for a 32-bit output.

6.4.6 Scale factors for transmitted data

The value of any particular sensor datum is a real number *R*, which is encoded for transmission as a binary integer N_b that is equivalent to the decimal integer N_d.

The real number *R* is calculated from N_d using the equation:

$R = N_d \times \text{SF} + \text{SO}$, where the scale factor SF is given by $\text{SF} = \text{SFS}_\text{d} \times 10^{\text{SFEd}}$ and the scale offset SO is given by $\text{SO} = \text{SOS}_\text{d} \times 10^{\text{SOEd}}$

The scale factor parameters SOS, SOE, SFS, and SFE are defined next.

The binary Scale Factor Significand (SFSb) is a signed 11-bit binary number that represents the decimal Scale Factor Significand (SFSd), which is defined to have 2048 possible values between –1.024 and +1.023, inclusive. The binary Scale Factor Significand is calculated by multiplying the decimal Scale Factor Significand by 100010, rounding to the nearest integer, and then converting to the signed 11-bit binary format in which –102410 = 100000000002, and +102310 = 011111111112. It is encoded in Field 6 of the TEDS.

NOTE 1—The 11-bit binary format representation is a two's-complement numeral system.

The binary Scale Factor Exponent (SFE_b) is a signed 6-bit binary number that represents the decimal Scale Factor Exponent (SFE_d) in the format in which $-32_{10} = 100000_2$ and $+31_{10} = 011111_2$. It is encoded in Field 7 of the TEDS.

The binary Scale Offset Significand (SOS_b) is a signed 11-bit binary number that represents the decimal Scale Offset Significand (SOS_d), which is defined to have 2048 possible values between –1.024 and +1.023, inclusive. The binary Scale Offset Significand is calculated by multiplying the decimal Scale Offset Significand by 1000_{10}, rounding to the nearest integer, and then converting to the signed 11-bit binary format in which $-1024_{10} = 10000000000_2$, and $+1023_{10} = 01111111111_2$. It is encoded in Field 8 of the TEDS.

The binary Scale Offset Exponent (SOE_b) is a signed 6-bit binary number that represents the decimal Scale Offset Exponent *(*SOE_d*)* in the format in which $-32_{10} = 100000_2$ and $+31_{10} = 011111_2$. It is encoded in Field 9 of the TEDS.

NOTE 2—This declaration holds the range of the error to 0.05% to 0.5%, with the average value of 0.13%.

> EXAMPLE 1
> Mains voltage monitor, range 216 V to 253 V, with 8-bit ADC
>
> The real number 216 is represented by decimal 0 and therefore by binary 0, and the real number 253 is represented by decimal 255 (based on the capability of the 8-bit ADC) and therefore by binary 11111111. The consequent constraints on the scale factor SF and scale offset SO are:
>
> $$216 = 0 \times \text{SF} + \text{SO}$$
> $$253 = 255 \times \text{SF} + \text{SO}$$
>
> Therefore, $\text{SO} = 216 = 0.216 \times 10^3$, and $\text{SF} = (253 - 216)/255 = 0.145 = 0.145 \times 10^0$, and
>
> $$\text{SOS}_\text{d} = 0.216_{10}, \text{ and } \text{SOS}_\text{b} = 00011011000_2$$

$\mathrm{SOE_d} = 3_{10}$, and $\mathrm{SOE_b} = 000011_2$
$\mathrm{SFS_d} = 0.145_{10}$, and $\mathrm{SFS_b} = 00010010001_2$
$\mathrm{SFE_d} = 0_{10}$, and $\mathrm{SFE_b} = 000000_2$

EXAMPLE 2
Temperature sensor, range –10 °C to 75 °C, with 12-bit ADC

The real number –10 will be represented by decimal 0 and therefore by binary 0, and the real number 75 will be represented by decimal 4095 (based on the capability of the 12-bit ADC) and therefore by binary 111111111111. The consequent constraints on the scale factor SF and scale offset SO are:

$$-10 = 0 \times \mathrm{SF} + \mathrm{SO}$$
$$75 = 4095 \times \mathrm{SF} + \mathrm{SO}$$

Therefore $\mathrm{SO} = -10 = -1 \times 10^{1}$, and $\mathrm{SF} = (75 + 10)/4095 = 0.0208 = 0.208 \times 10^{-1}$, and

$\mathrm{SOS_d} = -1.000_{10,}$ and $\mathrm{SOS_b} = 10000011000_2$
$\mathrm{SOE_d} = 1_{10}$, and $\mathrm{SOE_b} = 000001_2$
$\mathrm{SFS_d} = 0.208_{10}$, and $\mathrm{SFS_b} = 00011010000_2$
$\mathrm{SFE_d} = -1_{10}$, and $\mathrm{SFE_b} = 111111_2$

6.4.7 Data uncertainty

This field contains a code corresponding to a value that represents the accuracy of the transmitted data. The value is expressed as a percentage *X* of each transmitted value and means that the actual value is estimated, at the 95% level of statistical confidence, to differ from the transmitted value by an amount not exceeding *X*% of the transmitted value. Table 4 shows the codes and values.

Table 4 —Data uncertainty codes

Code	Value (%)
000	< 1
001	1
010	2
011	3
100	5
101	10
110	20
111	>20

EXAMPLE

Analysis conducted by the sensor manufacturer reveals that the expanded combined relative standard uncertainty due to the ADC performance characteristics (sensor manufacturing tolerances, and variations in environmental conditions within specified limits) is 7.89%. It is recommended that the manufacturer reports this value on its website and assigns the code 101.

6.4.8 Sensor reconfiguration

The sensor reconfiguration capability bit, (Field 11, Table 2) indicates whether or not the sensor supports the capability for the user to reconfigure the sensor. If this bit is set to 1, then reconfiguration is possible. If this bit is set to 0, then depending on the type of sensor and its use in an application, the sensor may be used as follows:

- To monitor only a single item.
- To monitor a different item until the end of service life of the sensor. This can be achieved by changing the unique item identifier on the RFID tag.

6.4.9 Memory rollover capability

Memory rollover applies only to Table 3—Measurement Codes Fields 10 to 13, which provide multiple data values. As successive sample data are recorded, the available memory may be filled. Without memory rollover, at the point where the memory becomes full the recording of data ceases, and a memory-full alarm is set. With memory rollover enabled, at the point where the memory becomes full the earliest record is overwritten on a first-in–first-out (FIFO) basis.

The Memory Rollover Capability Field of the Primary Sensor Characteristics TEDS (Table 2, Field 12) is a single bit that indicates whether or not the sensor supports the capability to apply memory rollover on the sensor. If this bit is set to 1, then memory rollover is supported. The Sample and Configuration Record (Table 11, Field 5) has a bit that declares whether the application administrator for the current mission has enabled memory rollover.

6.4.9.1 Memory rollover capability for measurement code 10

Each sample in memory is referred to by an index that is one less than the sample number of the particular sample. For example, S_0 taken at relative time zero [the (Coordinated Universal Time (UTC) time stamp taken upon successfully starting a mission time plus the monitor delay value that gives the time monitoring actually begins] is the first sample, and S_1 taken one sample interval later is the second sample. If sample capacity SCap is full, then the SCapth sample has index SCap – 1. If memory rollover has occurred one time such that the very first sample (index zero) has been rewritten, then that once rewritten sample may be referred to as having “total index” = SCap, and “local index” = 0 (it is the 0 index element within this “pass”). At this point, the number of samples taken equals SCap + 1. If memory rollover is supported and enabled, then the correct time stamp of each sample can be easily calculated as follows:

Table 17, Field 1, gives the total number of segments (see 6.6) that can be stored, referred to as C10SC (Code 10 sample capacity). The total number of samples that can be stored (SCap) is given by 32 × C10SC.

Letting SC be the current total sample count (tracked via Field 5 of Table 17), the number of times memory rollover has occurred is:

$$\text{Rollover Count} = \text{RC} = \text{Int}\left[\frac{\text{SC}-1}{\text{SCap}}\right]$$

where Int[*x*] is defined as the largest integer less or equal to *x*.

Next, define the variable local index (LI) as the index from 0 to SCap –1 that describes a sample position.

Let us now define the last sample taken (LST) in the most recent memory rollover period as having local index Max (LI_{max}).

LI_{max} is given by:

$$\text{LI}_{\text{max}} = (\text{SC}-1)-(\text{SCap} \times \text{RC})$$

The total index TI (the index relative to the 0th index first sample taken) for samples in the most recent memory rollover period (those with local index ≤ LI_{max}) is given by:

$$\text{TI} = (\text{SCap} \times \text{RC}) + \text{LI} \;\; (\text{for local index} \le \text{LI}_{\text{max}})$$

The TI for samples taken in the previous (next to last) roll over period (those with indexes from LI_{max} + 1 to SCap – 1) is given by:

$$\text{TI} = \left[\text{SCap} \times (\text{RC} - 1)\right] + \text{LI} \;\; (\text{for local index} > \text{LI}_{\text{max}} \text{ and RC} > 0)$$

The time stamp in seconds of the LIth indexed sample (t(LI indexed sample)$_{\text{s}}$) in memory, so long as TI is properly calculated as per the above based on the position of LI relative to LI_{max}, is always given by:

$$t(\text{LI indexed sample})_{\text{s}} = \text{UTC} + \text{MD}_{\text{s}} + (\text{TI} \times \text{SaIn}_{\text{s}})$$

where UTC is given in Field 1 of Table 11, MD_{s} is the monitor delay in Field 3 of Table 11 expressed in seconds, and SaIn_{s} is the Sample Interval in Field 2 of Table 11 expressed in seconds. Then, t(LI indexed sample)$_{\text{s}}$ can be converted to any desired time stamp format.

These equations may also be used to find the data of interest (find the proper local indexes) that corresponds to a certain time period. The pertinent manipulations of the previous equations are presented next.

First,

$$\text{TI} = \text{Round}\left[\frac{\left[t(\text{desired})_{\text{s}}\right] - \text{UTC-MD}_{\text{s}}}{\text{SaIn}_{\text{s}}}\right]$$

where Round[x] returns the nearest even integer to x.

And then

$$\text{LI} = \text{TI} - \text{RC} \times \text{SCap} \;\; (\text{if TI} \ge \text{SCap} \times \text{RC})$$

or

$$\text{LI} = \text{TI} - (\text{RC} - 1) \times \text{SCap} \;\; (\text{if TI} < \text{SCap} \times \text{RC})$$

6.4.9.2 Memory rollover capability for measurement code 11

If memory rollover is supported and enabled, then the correct time stamp of each sample is given by the time tick associated with it. Notice that when the sample count is beyond the encoding capacity, the sensor shall stop writing records to the memory assigned for this measurement code. Table 17, Field 2 (Code 11, sample capacity), gives the total number of segments that can be stored. The total number of samples that can be stored (SCap) is given by 32 × C10SC.

The number of times memory rollover has occurred is:

$$\text{Rollover Count} = \text{Int}\left[\frac{\text{SCOEOET} - 1}{\text{SCap}}\right]$$

where SCOEOET (sample count of events outside either threshold) is given in Field 9 of Table 17.

The time of the particular LI sample can be calculated as follows:

$$\text{t(LI indexed sample)}_{\text{s}} = \text{UTC} + \text{MD}_{\text{s}} + \left[\text{Sample Tick (LI indexed sample)} \times \text{SaIn}_{\text{s}}\right]$$

t(LI indexed sample)$_{\text{s}}$, MD$_{\text{s}}$, and SaIn$_{\text{s}}$ are as defined in 6.4.9.1.

6.4.9.3 Memory rollover capability for measurement code 12

If memory rollover is supported and enabled, then the correct time stamp of each sample is given by the time tick associated with it. Notice that when the sample count is beyond the encoding capacity, the sensor shall stop writing more records in the memory assigned for this measurement code. Table 17, Field 3 (Code 12 sample capacity), gives the total number of segments that can be stored. The total number of samples that can be stored (SCap) is given by 32 × C10SC. The number of times memory rollover has occurred is:

$$\text{Rollover Count} = \text{Int}\left[\frac{\text{SCOEOET} - 1}{\text{SCap}}\right]$$

where SCOEOET is given in Field 9 of Table 17.

The time of the particular LI sample can be calculated as follows:

$$t\text{(LI indexed sample)}_{\text{s}} = \text{UTC+MD}_{\text{s}} + \text{(Sample Tick(LI indexed sample)} \times \text{SaIn}_{\text{s}})$$

t(LI indexed sample)$_{\text{s}}$, MD$_{\text{s}}$, and SaIn$_{s}$ are as defined in 6.4.9.1.

6.4.9.4 Memory rollover capability for measurement code 13

These calculations are only valid if the alarms are triggered. Field 6 of Table 17 reveals the alarm status.

If memory rollover is supported and enabled, then the correct time stamp of each sample can be easily calculated as follows:

The number of times memory rollover has occurred is:

$$\text{Rollover Count} = \text{RC} = \text{Int}\left[\frac{\text{SC} - \text{SC@1TE}}{\text{SCap}}\right]$$

where SC is given in Field 5 of Table 17, SC@1TE (Sample count at 1st threshold event) is given in Field 10 of Table 17, and C13SC is the Code 13 sample capacity in segments given in Field 4 of Table 17. The total number of samples that can be stored (SCap) is given by 32 × C10SC.

$\mathrm{LI}_{\mathrm{max}}$ is defined as:

$$\mathrm{LI}_{\mathrm{max}} = (\mathrm{SC}\text{-}\mathrm{SC@1TE}) - (\mathrm{SCap} \times \mathrm{RC})$$

The TI (the index relative to the 0th index first sample taken) for samples in the most recent rollover period (those with local index ≤ $\mathrm{LI}_{\mathrm{max}}$) is given by:

$$\mathrm{TI}{=}(\mathrm{SCap} \times \mathrm{RC}) + \mathrm{LI} \;\; (\text{for local index} \le \mathrm{LI}_{\mathrm{max}})$$

The TI for samples taken in the previous (next to last) memory rollover period (those with indexes from $\mathrm{LI}_{\mathrm{max}} + 1$ to SCap – 1) is given by:

$$\mathrm{TI}{=}\left[\mathrm{SCap} \times (\mathrm{RC} - 1)\right] + \mathrm{LI} \;\; (\text{for local index} > \mathrm{LI}_{\mathrm{max}} \text{ and RC} > 0)$$

The time stamp in seconds of the LIth indexed sample ($t(\text{LI indexed sample})_{\mathrm{s}}$) in memory, so long as TI is properly calculated as per the above based on the position of LI relative to $\mathrm{LI}_{\mathrm{max}}$, is always given by:

$$t(\text{LI indexed sample})_{\mathrm{s}} = \mathrm{UTC} + \mathrm{MD}_{\mathrm{s}} + ((\mathrm{SC@1TE{+}TI} - 1) \times \mathrm{SaIn}_{\mathrm{s}})$$

where UTC is given in Field 1 of Table 11, MD_{s}, and SaIn_{s} are as defined in 6.4.9.1.

The previous equations may also be used to find the data of interest (find the proper local indexes) that corresponds to a certain time period. The pertinent manipulations of the previous equations are:

First,

$$\mathrm{TI} = \mathrm{Round}\left[\frac{t(\mathrm{desired})_{\mathrm{s}} - \mathrm{UTC}\text{-}\mathrm{MD}_{\mathrm{s}}}{\mathrm{SaIn}_{\mathrm{s}}}\right] - \mathrm{SC@1TE{+}1}$$

where Round[x] returns the nearest even integer to x. MD_{s}, and $\mathrm{SaIn}_{\mathrm{s}}$ are as defined in 6.4.9.1.

And then

$$\mathrm{LI} = \mathrm{TI} - \mathrm{RC} \times \mathrm{SCap} \;\; (\text{if TI} \ge \text{SCap} \times \text{RC})$$

or

$$\mathrm{LI} = \mathrm{TI} - (\mathrm{RC} - 1) \times \mathrm{SCap} \;\; (\text{if TI} < \text{SCap} \times \text{RC})$$

6.4.10 Air Interface Security Capability

This field reports whether the sensor supports air interface security and the allowed level of security options (command set blocking). The Air Interface Security Function Code as programmed in Table 11 Sample and Configuration Record and as detailed in Table 14 is the user selection of allowed option levels.

For this standard, air interface security is assumed to be limited to whether an air interface security check based on a simple password has been passed or not. This is referred to as Air Interface Security Level 1. More advanced air interface security is expected in the future. Table 5 presents Air Interface Security Capability Codes.

Table 5 —Air Interface Security Capability Codes

Air Interface Security Capability Code	Definition
000	This sensor does not support Air Interface Security. An attempt to program an Air Interface Security Function Code > 000 will result in an error code.
001	This sensor supports Air Interface Security Capability 1, defined as the ability to accept programming of Air Interface Security Function Codes 000 to 011 based on Air Interface Level 1 security (see Table 14). An attempt to program a Security Function Code > 001 will result in an error code.
X = 010 to 111	This sensor supports air interface security levels up to and including this value. These Air Interface Security Capability Codes are RFU under this standard.

6.4.11 Sensor Security Capability

This field reports whether the sensor supports its own internal security and the allowed level of security options. The Sensor Security Function Code as programmed in Table 11 Sample and Configuration Record and as detailed in Table 15 is the user selection of allowed option levels. Table 6 presents Sensor Security Capability Codes.

Table 6 —Sensor Security Capability Codes

Sensor Security Capability Code	Definition
000	This sensor does not support direct security. An attempt to program a Sensor Security Function Code > 000 will result in an error code. There is no authentication password/key. and the key read-lock and write-lock fields of the Event Administration Record of Table 17 are not present.
001	This sensor supports Sensor Security Capability 1, defined as the ability to accept programming of Sensor Security Function Codes 000 to 011 (see Table 15). An attempt to program a Security Function Code > 001 will result in an error code.
X = 010 to 111	This sensor supports security levels up to and including X. These Sensor Security Capability Codes are RFU under this standard.

6.4.12 Sensor authentication encryption capability map

Table 7 provides the 7-bit map of supported sensor authentication encryption algorithms.

Table 7 —Sensor authentication encryption capability map

Bit position	Encryption algorithm	Comment
0	AES	The most widespread algorithm, suitable for very-large-scale integration (VLSI) or software implementation, standard key lengths of 128, 192, and 256 bits (although only 128 supported in this standard).
1	SHA 1	Hash SHA1 with 160-bit output and 528-bit input. The key and random number make up part of the 528-bit input, and the remainder are fixed. More security is provided by choosing longer lengths for the key and random number.
2	RFU	
3	RFU	
4	RFU	
5	RFU	
6	RFU	

6.4.13 Sensor data encryption capability map

Table 8 provides the 7-bit map of supported sensor data encryption algorithms.

Table 8 —Sensor data encryption capability map

Bit position	Encryption algorithm	Comment
0	AES	The most widespread algorithm, suitable for VLSI or software implementation, standard key lengths of 128, 192, and 256 bits (although only 128 supported under this version).
1	RFU	
2	RFU	
3	RFU	
4	RFU	
5	RFU	
6	RFU	

6.4.14 Random number size

This 3-bit field is used to report the sizes of the random number supported by this sensor. This standard currently supports RN sizes of 16, 32, 64, and 128 bits, with four RN sizes reserved for future use. If the Sensor Security Capability Code is 000, then there is no random number generator and this overrides the Random Number Size field. The actual RN lengths in use are as selected by the Challenge and Reader-Authenticate commands. Supporting a given size requires supporting all the sizes less than that size. For example, if the sensor supports RN size of 128 bits, this means that it also supports RN sizes 16, 32, and 64 bits.

6.4.15 Continuing Authentication Capability Field

Continuing Authentication is an optional feature whereby the reader and/or the sensor may continue to authenticate themselves to each other on every command and response. Continuing Authentication prevents unauthorized readers and/or tags from masquerading as authentic by inserting themselves into the communication after the initial authentication-to-enter-into-a-secured-state is completed. The available options that the sensor supports are as given in Table 9.

Table 9 —Continuing Authentication Capability Field

Value	Capability
00	The sensor does not support Continuing Authentication in either link.
01	The sensor supports only Continuing Authentication of the sensor (requires new reader RNs on each reader command).
10	The sensor supports only Continuing Authentication of the reader (new sensor RNs on each sensor reply).
11	The sensor supports Continuing Authentication of both sensor and reader (new RNs and tokens supplied by both sides on every transmission).

If two-way Continuing Authentication is supported, then the reader may select it for forward communication or reverse communication or both. The Continuing Authentication modes are commanded as options within the Reader-Authenticate command (see 7.17).

6.4.16 Sensor Authentication Password/Key Size

This 3-bit field is used to report the size of the sensor authentication password/key. The term “password/key” is used instead of just “key” because it functions as a password if the sensor does not feature encryption based authentication, but as a key if the sensor does feature encryption. This standard currently supports sensor authentication password/key sizes of 16, 32, 64, and 128 bits, with four sizes reserved for future use. A particular encryption algorithm may be specified to override these key sizes to have a custom length. If the Sensor Security Capability Code is 000, then there is no sensor authentication password/key and this overrides the Sensor Authentication Password/Key Size field.

6.4.17 Sensor Data Encryption Key Size

This 3-bit field is used to report the size of the sensor data encryption key. This standard currently supports sensor data encryption key sizes of 16, 32, 64, and 128 bits, with four key sizes reserved for future use. A particular encryption algorithm may be specified to override these key sizes to have a custom length. If the Sensor Security Capability Code is 000, then there is no data encryption key and this overrides the Sensor Data Encryption Key Size field.

6.4.18 Data Encryption Capability field

Data encryption is an optional feature that is available if at least one data encryption algorithm is available under the Sensor Data Encryption Capability Map of Table 8. This 2-bit field specifies whether encryption is available for each of the forward link and reverse link, as shown in Table 10.

Table 10 —Data Encryption Capability Field

Value	Capability
00	The sensor supports no data encryption.
01	The sensor supports only encryption from sensor to reader.
10	The sensor supports only encryption from reader to sensor.
11	The sensor supports encryption in both links.

When encryption is in use, it is applied to the total command and/or response. Thus, security tokens under data encryption are encrypted twice: once for the original authentication encryption of random number into token and then again for encryption of the entire message. This double encryption of the security token allows for simpler design by not having to track token position in the encrypted stream.

6.5 Sample and Configuration Record

The Sample and Configuration Record consists of a number of fields as defined in Table 11, with further detailed specifications in the subclauses that follow Table 11.

Table 11 —Sample and Configuration Record

Field	Name	Size	Reference	Example/note
1	UTC time stamp at configuration, or upon successfully beginning a mission. See NOTE.	32 bits	6.5.1	2008-08-08 08:08:08
2	Sample interval	16 bits	6.5.2	Either in seconds or in minutes.
3	Monitor delay	16 bits	6.5.3	Either in seconds or in minutes.
4	Alarm values set	2 bits	6.5.4	00 = none, 01 = lower only, 10 = upper only, 11 = both.
5	Memory rollover enabled	1 bit	6.5.5	0 = switched OFF. 1 = switched ON.
6	Air Interface Security Function Code	3 bits	6.5.6	See Table 14 for detailed definition.
7	Sensor Security Function Code	3 bits	6.5.9	See Table 15 for detailed definition.
8	Sensor Authentication Encryption Function Code	3 bits	6.5.10	The selection among the available methods.
9	Sensor Data Encryption Function Code	3 bits	6.5.11	The selection among the available methods.
10	Security Timer Duration	3 bits	6.5.12	See Table 16 for detailed definition. To accuracy specified in TEDS Table 2.

11	Begin-End-Mission Authority	1 bit	7.15	0 = Requires user to have write-level authority per Security Function Code in order to end mission or start new mission after a mission has ended due to full memory or sampling limitation. 1 = Allows user with only read level authority to end mission or start new mission after another mission has ended.
12	Upper alarm threshold	1 to 32 bits	6.5.14	Encoded in the same format as sensor data (see 6.4.5).
13	Lower alarm threshold	1 to 32 bits	6.5.15	Encoded in the same format as sensor data (see 6.4.5) Characteristic TEDS.
NOTE—The UTC time stamp is written at configuration, and then again upon beginning a mission or a remission. A mission is begun with the Begin-End-Mission command. Monitoring begins upon expiration of the monitor delay, and the monitor delay timer begins running when the mission begins.				

6.5.1 UTC time stamp at configuration and beginning of mission

The time stamp shall be based on the UTC time epoch beginning at 1970-01-01 00:00:00. At configuration, the time stamp shall be set to the current 32-bit UTC, precise to 1 s. This time stamp is captured when the Write-Sample-and-Configuration-Record command is executed, and it is updated at the time a valid Begin-End-Mission command is successfully executed to begin a mission.

NOTE—This time stamp can be achieved by taking the most significant 32 bits from the IEEE 1588 standard synchronized time stamp. Interpretation to the UTC format of year-month-day/hour-minute-second is beyond the scope of this standard.

6.5.2 Sample interval

The sample interval is the amount of time between successive samples of sensor data.

The sample interval is expressed as a 16-bit code. When the MSB of this code is set to 0, the sample interval is expressed in seconds. When this bit is set to 1, the unit of time is minutes. The following 15 bits of this code constitute a 15-bit unsigned binary integer that indicates the number of seconds or minutes in the sample interval.

The code 0000000000000000 shall be used to signal continuous monitoring.

EXAMPLES:

The code 0111111111111111 indicates that the sample interval in seconds is equal to $111111111111111_2 = 32{,}767_{10}$. This is the maximum value possible when setting the time interval in seconds, and is equivalent to 9 h, 46 min, and 7 s.

The code 1000010110100000 indicates that the sample interval in minutes is equal to $000010110100000_2 = 1440_{10}$. This code indicates that the sample interval is exactly 24 h.

The code 1111111111111111 indicates that the sample interval in minutes is equal to $111111111111111_2 = 32{,}767_{10}$. This is the maximum value possible when setting the time interval in minutes, and is equivalent to 22 days, 18 h, and 7 min.

6.5.3 Monitor delay

The monitor delay field indicates the time between the beginning of the mission and the beginning of the monitoring. The mission begins upon successful execution of the Begin-End-Mission command. Monitoring begins immediately if the monitor delay is zero or after the expiration of monitor delay if the monitor delay is greater than zero.

No sample records are written to memory and no threshold alarms triggered until the monitor delay period has passed. For Measurement Code 10, the first recorded value is the first sample at the instant that the monitor delay ends.

The monitor delay has the same format as that of the sample interval.

6.5.4 Alarm values set

A sensor can be configured to not set alarms, to set alarms according to lower or upper thresholds, or to set alarms according to a window defined by lower and upper thresholds. The Alarm Values Set Field (Field 4, Table 11), is a 2-bit code that defines which and how many Fields 6 and 7 are present in this section of the TEDS, as follows:

00 = none
01 = lower only
10 = upper only
11 = both

6.5.5 Memory Rollover Enabled

A sensor that supports memory rollover might, or might not, be configured to enable memory rollover to occur. If a sensor supports more than one multiple value measurement type, the Memory Rollover capability setting shall apply to all the multiple value types supported.

The Memory Rollover Enabled Field (Field 5, Table 11) is a 1-bit code that indicates whether rollover has been switched ON or OFF. Code = 1 indicates that memory rollover has been switched ON; Code = 0 indicates that memory rollover has been switched OFF.

With Memory Rollover Enabled set to OFF, samples are recorded until the associated memory becomes full, at which point a Memory Full alarm is triggered and no further data are recorded. In contrast, with memory rollover switched ON, when the first instance of the memory becoming “full” is reached, the sensor will overwrite the earliest recorded sample. This process continues on an FIFO basis.

When Memory Rollover Enabled is set to ON, a memory-full alarm is triggered only when the sample time mechanism reaches its capacity. For example, with Measurement Type 11, when the 255th sample interval is reached, it is no longer possible to add to this sample data.

NOTE—The Memory Rollover Enabled Field can only be used if the processes that support this feature exist on the sensor (see 6.4.9).

6.5.6 Air Interface Tag Security Status Code

User programming of the Air Interface Security Function Code may support various security functions such as no security, sensor requires security check passed for write operations, and sensor requires security check passed for both read and write operations. The use of 3-bit fields for these codes allows room for future growth.

To implement the security function, it is the responsibility of the tag to prefix each command with the Tag Security Status Code. For example, tags that use a transport command to move the sensor command as a payload would automatically prefix the payload with the Tag Security Status Code that the tag has determined to be in effect via real-time air interface operations. The command shall be ignored by the sensor and the sensor shall reply with an Air Interface Security Failure Code if the Air Interface Security Function Code in use requires security not satisfied by the Tag Security Status Code prefixed to the command by the RFID tag. Tag Security Status Code values are as shown in Table 12.

Table 12 —Tag Security Status Codes (applies only to air interface security)

Tag Security Status Code	Interpretation	Comment
000	No security check has been passed.	Some or all commands may still be executed depending on how the sensor Air Interface Security Function Code is programmed.
001	Level 1 Air interface Security has been passed.	This level typically refers to a simple password with unencrypted one-way reader authentication, which is all that exists in common air interfaces as of the publication of this document. See NOTE.
010-111	RFU	RFU
NOTE—Development of more sophisticated RFID air interface security with encryption based authentication is underway.		

6.5.7 Sensor Command Classes

The sensor commands are broken into classes to allow the Air Interface Security Function Code and Sensor Security Function Code values to dictate which classes of commands will be executed as a function of security state. These are as outlined in Table 13.

Table 13 —Sensor Command Classes

Command Class	Commands
Read	Read-Sensor-Identifier Read-Primary-Characteristics-TEDS Read-Sample-And-Configuration Read-Alarm-Status Read-Single-Memory-Record Read-Event-Administration-Record Read-Event-Record-Segments Read-Partial-Event-Record-Segment Read-Any-Field
Write	Write-Sample-And-Configuration Write-Event-Administration-Field-7 Erase-Event-Administration Erase-Event-Records Erase Sample-And-Configuration-Record
Key Write	ReadWriteLock-Keys
Security Set Up	Challenge Reader-Authenticate Request-RN
Security Control	Encryption-On-Off Close-Secure-Session
Special Cases	Begin-End-Mission: read or write at different times, see command and Table 52

6.5.8 Air Interface Security Function Code

The Air Interface Security Function Code is the user-selected security behavior with respect to what is supported under the Air Interface Security Capability Code and the Tag Security Status Code passed to the sensor by the tag. For this standard, only Air Interface Security Level 1 or the lack of it can be reported by the Tag Security Status Code.

The only commands that can be accessed with only air interface security are Read and Write. The Key Write, Security-Set Up, and Security Control classes of sensor commands are reserved for sensors that support direct sensor security. Table 14 presents Air Interface Security Function Codes.

Table 14 —Air Interface Security Function Codes

Air Interface (AI) Security Function Code	Commands executed with tag Security Status Code = 000 (no security check passed)	Commands executed with tag Security Status Code = 001 (level 1 Air Interface Security passed)
000 See NOTE 1 through NOTE 4.	Read & Write	Read & Write
001	Read & Write	Read only
010	Read & Write	None (no access allowed)
011	Read only—the sensor is “mission-locked” and “write-locked”)	None
100-111	RFU	RFU

NOTE 1—Once programmed above AI Security Function Code 000, air interface security must be passed to allow any write, including to the AI Security Function Code itself.

NOTE 2—The security strength of AI Security Function Codes 000, 001, 010, and 011 are equal to those of the same numeric value Sensor Security Function Codes. If one is programmed higher than the other, then the higher code becomes the security function code for both modes.

NOTE 3—Because Begin-End-Mission is either read or write at different times, it can be executed with only air interface security.

NOTE 4—Only read and write commands are indicated because security commands are not pertinent to air interface security. Sensors that do not support direct sensor security will not support Security Set-Up and Security Control commands. Sensors that support both air interface and direct sensor security will have their Security Set-Up and Security Control commands authorized via Sensor Security Capability Code > 000, and their Key Write commands will be authorized by having Sensor Security Capability Code > 000 and Sensor Security Function Code of 000, 001, or 010.

6.5.9 Sensor Security Function Code

This field determines how the sensor reacts to a successful reader authentication, and the options are as described in Table 15. As the code progresses higher, the security becomes “stronger” in the sense that the sensor blocks execution of more command types. Command types currently defined are Key Write, Write, Read, Security Set-Up, and Security Control.

If both air interface and sensor security are supported, and if they are programmed via the function code to different security levels, then the more secure mode programmed will be in effect for both security systems. Security strength is increasing as the code increases.

Table 15 —Sensor Security Function Codes

Code	Commands executed with authentication password/key	Commands executed without authentication password/key
000 See NOTE 1 through NOTE 5.	All	Write, Read, Security Set Up, Security Control, but not Key Write.
001	All	Reads, Security Set Up, Security Control.
010	All	None (no access allowed without reader authentication)
011	Read, Security Set Up, Security Control (but not Write or Key Write—the sensor is "mission-locked" and "write-locked")	None
100-111	RFU	RFU

NOTE 1—Sensor Security Function Code 000 means that the sensor does not block out any commands except password/key(s) writes without current authentication password/key. Some commands serve no purpose in that case but can still be executed. For example, Reader-Authenticate can be executed and the sensor will provide the correct response, but whether the tag receives the correct security token or not, it will still execute all subsequent commands. There is, thus, no point in activating two-way continuing authentication with this Sensor Security Function code. However, there is value in the Sensor Continuing Authentication with this code because the reader is then aware of the validity of the sensor.

NOTE 2—Initial password setup may be done with Sensor Security Function Code = 000, 001, and 010. Once the Sensor Security Function Code is set to 011, no writes or execution of Key Write class commands are allowed.

NOTE 3—When password/key(s) are not write-locked and the Sensor Security Function Code authorizes password/key(s) rewriting (code 000, 001, and 010), the confirmation of the reader having the correct authentication password/key provides authority to allow rewriting of authentication password (no authentication encryption supported), authentication key (authentication encryption supported), and data encryption key (data encryption supported).

NOTE 4—Once programmed above 000, the Sensor Security Function Code may not be programmed back to 000 without the authentication password/key and write access.

NOTE 5—Password/key(s) reads and writes are never allowed after the password/key(s) are read-locked and write-locked.

6.5.10 Sensor Authentication Encryption Function Code

This 3-bit code can be all zeroes to indicate that authentication encryption shall not be used (thus, only reader authentication via password will be supported by the sensor while so programmed), or a value from 001 to 111 to point to a particular authentication encryption as described in Table 7.

6.5.11 Sensor Data Encryption Function Code

This 3-bit code can be all zeroes to indicate that data encryption shall not be used, or a value from 001 to 111 to point to a particular data encryption as described in Table 8. This data encryption then applies to the encryption links chosen by the Encryption-ON-OFF command of Table 62.

6.5.12 Security Timer duration

This field sets the duration of the Security Timer. The timer begins running upon successful reader authentication, and it shall reset for every valid command received by the sensor while in a secure state.

6.5.13 Secure Session Timer

The Secure Session Timer is supported if the sensor supports encryption and two-way authentication. The time selected by the user is programmed in the Sample and Configuration Record. The timer begins running upon successful reader authentication, and it shall reset for every valid command received by the sensor while in a secure state. If the timer expires, then the sensor departs its secure state and requires a new authentication to perform secure operations. This timer provides a means for resetting the sensor to require reauthentication if the RFID tag departs the read range of the reader while in a secure state before its secure session is closed. Timer values are as shown in Table 16 and shall be to clock accuracy specified in the TEDS Table 2.

Table 16 —Secure Timer Code values

Security Timer Codes	Times
000	50 ms or less
001	100 ms
010	200 ms
011	400 ms
100	800 ms
101	1.6 s
110	3.2 s
111	>3.2 s

6.5.14 Upper alarm threshold value

Table 11, Field 6, defines the upper alarm threshold value. This value is encoded in the Memory Resolution declared in Field 5 through Field 9 of Table 2, the Primary Sensor Characteristics TEDS. The upper alarm threshold value shall be equal to or less than the largest possible value for sensor data, and greater than the lower alarm threshold value if this is set, or greater than the smallest possible value for sensor data if a lower limit is not set.

EXAMPLE
For a temperature sensor with range –10 °C to 75 °C and 12-bit ADC (as per Example 2 of 6.4.6), the upper alarm threshold value is to set to 28 °C.

From the earlier example, values of temperature T are calculated from transmitted integers N using the equation

$$T = N_{10} \times 0.0208 - 10$$

Simple algebra reveals that $N_{10} = 1826.9$ when $T = 28$. The 12-bit binary equivalent of 1827_{10} is 011100100011_2. This is the upper alarm threshold value.

6.5.15 Lower alarm threshold value

Table 11, Field 7, defines the lower alarm threshold value. It has the same format as the upper alarm threshold value. The lower alarm threshold value shall be equal to or greater than the smallest possible value for sensor data, and less than the upper alarm threshold value, if this is set; or, the value shall be less than the largest possible value for sensor data if an upper limit is not set.

6.6 Event Administration Record

The event administration record contains information that allows the RFID interrogator or processes above the interrogator to calculate exactly how many sensor words are stored for Measurement Codes 10, 11, 12, and 13 (Table 3). Additionally, it contains the alarm triggered field and the sample counts needed for Measurement Codes 6 and 7. It also contains the current password/keys(s) read-lock and write-lock status bits (7.18), and a field to indicate whether a mission is in progress. In 6.6.1 through 6.6.11, each field is defined.

This standard defines a segment as a group of 32 sensor words. The length of each word for Measurement Codes 10 and 13 is specified by Field 5 of Table 2, and the length of each word for Measurement Codes 11 and 12 is specified by the sum of the time tick length and the Memory Resolution given in Field 5 of Table 2. In this way, memory size is given in multiples of one segment. Table 17 presents event administration record.

Table 17 —Event Administration Record

Field	Name	Size	Reference	Note
1	Code 10 Sample capacity	11 bits	6.6.1	This field applies only for Measurement Type 10 (only present if Measurement Code 10 is present in Field 4, Table 2).
2	Code 11 Sample capacity	3 bits	6.6.2	This field applies only for Measurement Type 11 (only present if Measurement Code 11 is present in Field 4, Table 2).
3	Code 12 Sample capacity	11 bits	6.6.3	This field applies only for Measurement Type 12 (only present if Measurement Code 12 is present in Field 4, Table 2).
4	Code 13 Sample capacity	11 bits	6.6.4	This field applies only for Measurement Type 13 (only present if Measurement Code 13 is present in Field 4, Table 2).
5	Sample count	16 bits	6.6.5	This field is always present.
6	Alarm triggered	4 bits	6.6.6	This field is always present.
7	Sample count at predetermined time	16 bits	6.6.7	This field applies only for Measurement Type 6 (only present if Measurement Code 6 is present in Field 4, Table 2).
8	Sample count at critical event	16 bits	6.6.8	This field applies only for Measurement Type 7 (only present if Measurement Code 7 is present in Field 4 Table 2).
9	Sample count of events Outside either threshold	16 bits	6.6.9	This field applies only for Measurement Types 11 and 12 (only present if Measurement Code 11 or 12 are present in Field 4, Table 2).
10	Sample count at the first threshold event	16 bits	6.6.10	This field applies only for Measurement Type 13 (only present if Measurement Code 13 is present in Field 4, Table 2).

Field	Name	Size	Reference	Note
11	Password/key(s) Read-Lock and Write-Lock status flags	4 bits	7.18	First bit indicates whether optional authentication password/key is read-locked, the second bit indicates whether it is write-locked. The third bit indicates whether the data encryption key is read-locked, and the fourth bit indicates whether it is write-locked. In all cases, 0 indicates unlocked and 1 indicates locked. Not present if associated functionality is not supported. For example, if data encryption is not supported, then this field reduces to two bits to cover authentication Read-Lock and Write-Lock status only. These bits are written by the ReadWriteLock-Keys command with lock function selected.
12	Mission in Progress	1 bit	6.6.11	0: No mission in progress. 1: Mission is now in progress. This means that either data are being monitored, or the monitor delay timer is running in preparation for monitoring data.

6.6.1 Code 10 sample capacity (C10SC)

This field indicates the maximum memory size assigned for Measurement Code 10. The value is indicated in segments; therefore, with 11 bits, possible sizes go from 1 segment (00000000000_2) to 2048 segments (11111111111_2). This field is only present if Measurement Code 10 is present in the sensor map (Field 4 of Table 2).

6.6.2 Code 11 sample capacity (C11SC)

This field indicates the maximum memory size assigned for Measurement Code 11. The value is indicated in segments; therefore, with 3 bits, possible sizes go from 1 segment (000_2) to 8 segments (111_2). This field is only present if Measurement Code 11 is present in the sensor map (Field 4 of Table 2).

6.6.3 Code 12 sample capacity (C12SC)

This field indicates the maximum memory size assigned for Measurement Code 12. The value is indicated in segments; therefore, with 11 bits, possible sizes go from 1 segment (00000000000_2) to 2048 segments (11111111111_2). This field is only present if Measurement Code 12 is present in the sensor map (Field 4 of Table 2).

6.6.4 Code 13 sample capacity (C13SC)

This field indicates the maximum memory size assigned for Measurement Code 13. The value is indicated in segments; therefore, with 11 bits, possible sizes go from 1 segment (00000000000_2) to 2048 segments (11111111111_2). This field is only present if Measurement Code 13 is present in the sensor map (Field 4 of Table 2).

6.6.5 Sample count

Table 17, Field 5, represents the sample count since the beginning of monitoring. The sensor is expected to make the first measurement (but not necessarily to log that measurement) right after the mission starts and the time stored in the monitor delay field (Table 11, Field 3) is exhausted, and the value of that field becomes 1_{10}. The maximum number that can be encoded by the Sample Count Field is 65536; after this Sample Count number has been reached, the sensor shall stop taking samples.

6.6.6 Alarms triggered

Table 17, Field 6, is a 4-bit field that is updated as alarm conditions are observed, with code 0 = the specific alarm is not set, and code 1 = the specific alarm is set. The inputs for the 4 bits are as follows:

- The first bit identifies whether an upper alarm has been triggered.
- The second bit identifies whether a lower alarm has been triggered.
- The third bit identifies whether, for any of the data log type of records, a memory full condition has been reached and that memory rollover has not been asserted or is not possible to assert for the memory.
- The fourth bit indicates a low battery status, to rules defined by the sensor manufacturer.

NOTE—If a battery supports more than one sensor, then any or all of the sensors can trigger this particular alarm.

6.6.6.1 Low battery issues

As the range of low battery conditions and the effects of them are so varied, exact specification of sensor response due to low battery condition is beyond the scope of this standard. However, as described in 6.6.6, this standard does specify a means by which the sensor can report low-battery events and allow the application and user an opportunity to tailor their response to the particular conditions. In general, a low battery indication follows some of the Response Codes as a warning to the user that that the battery should be replaced (if possible), and that it is possible that the data has been corrupted by the low battery. It is then up to the user and the application to determine the degree of trust to be placed in a sensor or tag with a low battery, as these conditions are highly variable. For example, a low battery indication on a write or erase operation is not a reliable indicator that logged data read back is invalid, since it takes higher energy to perform a write or erase than to perform a read operation.

6.6.7 Sample count at predetermined time

This field contains the sample count at a predetermined time in units of sample intervals (see Field 2 of Table 11).

6.6.8 Sample count and data following alarm event

This field is associated with Table 3, Field 7, and stores the sample count at the first instance of an alarm being triggered (at the next sample time following exceeding threshold, at which time the sensor value may be well past the alarm threshold), along with a measurement of the data value at the sample time.

6.6.9 Sample count of events outside either threshold

This field contains the count of the number of events that have occurred outside either threshold. Note that the alarms have to be set in order to count the events. The alarm values are as follows:

- If the alarm values set (see Field 4 of Table 11) contain the value 00_2, the sample count of events outside either threshold field will never be incremented.
- If the alarm values set contain 01_2, the Sample Count of Events outside Either Threshold field will only be increased when the data measured is below the lower threshold.
- If the alarm values set contain 10_2, the Sample Count of Events outside Either Threshold field will only be increased when the data measured are above the higher threshold.

- If the alarm values set contain 11_2, the Sample Count of Events outside Either Threshold field will be increased when the data measured are outside either threshold.

6.6.10 Sample count at the first threshold event

This field contains the sample count when the data measured are outside either threshold (assuming the alarm values set has values 01_2, 10_2, or 11_2) for the first time.

6.6.11 Mission in Progress

This field is included to aid the application to determine whether the sensor is stopped or the sensor is either observing values or the monitor delay timer has begun in preparation to observe values (i.e., there is a mission in progress). Its value is in knowing whether a previous reader has successfully begun a mission. This field is set to 1 to indicate mission in progress. The Mission in Progress field is set to 0 if a mission has not been started, or after successfully executing a Begin-End-Mission command with parameter Begin/End set to 1 (end mission).

6.7 Event records

An event record shall be maintained for each measurement code that is declared by a binary 1 being set in the Sensor map (Field 4 of Table 2). How data are handled once memory capacity is filled depends on the Measurement Code used.

The following examples refer to Table 3:

a) Single Record measurement codes allow overwriting the current record when the data recording memory becomes full (see 6.7.1 and 6.7.2).

b) Multiple Record measurement codes have two options when the data recording memory becomes full: Memory Rollover capability ON or Memory Rollover capability OFF.

   1) With *Memory Rollover capability ON*: When the memory becomes full, data continue to be recorded into memory on a FIFO basis (latest time-stamped record overwrites the earliest time-stamped record).

   2) With *Memory Rollover capability OFF*: When the memory becomes full, no more data will be recorded into memory.

The basic length and constructs of each event record are given in Table 3. The following subclauses specify features of the memory and basic process rules.

6.7.1 Single event records

This structure applies to Table 3, Measurement Codes 0 to 5, for the following:

- Present (point-of-time) value
- Maximum (or peak) value
- Minimum (or lowest) value
- Average value
- Variance

- Standard deviation

There is only one instance of each of these event records, and by the nature of continual monitoring, the value in the recorded value can be overwritten.

The length of the record is determined by the data format specified for the particular sensor, and it can therefore be from 1 bit to 32 bits long.

6.7.2 Single event with time stamp

This structure applies to Table 3 for the following:

- 6: Observed value at a predetermined sample time
- 7: Continual monitoring to record a single threshold defined event

There is only one instance of each of these event records, and the value in the record cannot be overwritten.

6.7.3 Event counts

This structure applies to Table 3 for the following:

- 8: Count of readings over maximum threshold value
- 9: Count of readings below minimum threshold value

If the total number of counts reaches 255, then the sensor shall cease to generate additional records for these codes.

The length of each of these records is fixed at a single byte.

6.7.4 Data log of all sampled events

This structure applies to Table 3, Measurement Code 10, for a data log of observed values recorded at each sample interval.

The size of this multiple record is determined by the number of logical records factored by the size of the data transmission (of between 1 bit and 32 bits). The record size is clearly defined in Table 3.

Subclause 6.5.5 defines the process when the memory capacity is “full” with and without memory rollover being switched ON.

6.7.5 Data log plus time tick

This structure applies to Table 3 as follows:

- 11: Data log of observed value with time tick (8-bit code) reporting outside either threshold
- 12: Data log of observed value with time tick (16-bit code) reporting outside either threshold

The time tick is the sample ordinal number since the beginning of the monitoring process. Because these event records only retain out-of-limit sample readings, the number of records can be relatively small, even nonexistent, compared with the number of samples.

The size of this multiple record is determined by the number of records, factored by the size of the data transmission (of between 1 bit and 32 bits) and the requirement to encode an 8-bit time tick (code 11) or a 16-bit time tick (Measurement Code 12). The record size is clearly defined in Table 3.

Subclause 6.5.5 defines the process when the memory capacity is "full" with and without memory rollover being switched ON.

In addition to the Memory Full condition, which indicates a greater number of out-of-limit sample readings than anticipated in the design, another "overflow" condition can occur. If the total number of samples is greater than the encoding capacity of the time tick component of the record, for example if sample number 280 is out of limit but for Measurement Code 11 record with only an 8-bit time tick, then the sensor process shall cease to generate additional records.

6.7.6 Data log of all observations after initial alarm

This structure applies to Table 3, Measurement Code 13, for a data log of the observed values recorded at each sample interval after an initial upper or lower threshold has been crossed for the first time. When this occurs, the sample count at the time of the first alarm is stored in Field 10 of the Event Administration Record (see 6.6.10). The associated observed value, and all subsequent observed values, is written to this record. It is not necessary to record subsequent time values because the time of each sample can be calculated from the time stamp by adding increments equivalent to the sample interval. Therefore, each additional record consists only of an observed value.

Subclause 6.5.5 defines the process when the memory capacity is "full" with and without memory rollover being switched ON.

7. Command overview

The commands described in the following list, together with their associated responses, provide user-based instructions to the sensors. The structure of each command and response does not include any additional overhead bits in terms of the preamble and trailer necessary to implement the commands across the air interface. Effectively, the commands and responses described below are the "payload" for specific air interface commands and responses. In the event of multiple sensors, the air interface command structure must be able to retrieve data concerning multiple sensors. Communications via the RFID air interface are discussed in Clause 8.

Most of the commands and responses have optional random number fields and security tokens attached. They may be of any length as allowed in the TEDS (currently 16, 32, 64, and 128 bits) and as selected in the Challenge and Reader-Authenticate commands. These are used in the case of "Continuing Authentication" where each command and reply is independently authenticated. This may be applied one way (reader only) or two way (reader and sensor). The Reader Continuing Authentication is supported by the sensor supplying a random number appended to its replies, which the reader then encrypts into a one-time-use security token to be attached to the next reader command. The length of the RN supplied by the tag to be encrypted into a token by the reader is given in the Reader-Authenticate command. The Sensor Continuing Authentication is supported by the reader appending a random number to each command, which is then encrypted with the secret key by the sensor to use as a security token in the reply to the command. The length of the RN supplied by the reader to be encrypted into a token by the tag is given in the Challenge command. The allowed lengths of RNs are as described in the TEDS.

The commands support one of four means of addressing a sensor, as follows:

a) Tag-level addressing such as port numbers, of which the sensor does not need to be aware

b) Using a subaddress number declared by the sensor

c) Using the 64-bit unique sensor identifier as a subaddress

d) Using the first three fields of the sensor characteristics TEDS, uniquely identifying the type of sensor, as a subaddress

The three "subaddressing" methods described may be used either at the top level of a sensor-addressing system or as a subaddress below a tag-level address, such as a physical port on the tag. The system in use would normally either be unique (the reader is preaware of sensor addressing) or be described to the reader via a sensor directory on the tag. A typical operation would be for the reader to read a sensor directory off the tag that describes basic sensor functions and the sensor addresses, so that transport commands carrying sensor commands may be efficiently pointed to the desired sensors.

A sensor shall support at least one of the addressing mechanisms and may support others as defined in this document. The choice can be affected by the choice of RFID air interface protocol and tag architecture with which the sensor integrates. The addressing mechanisms shall be declared on the sensor manufacturer's data sheet.

The sensor shall support each of the commands and responses if the inherent functionality is supported. For example, if no detailed event logs are capable of being recorded by the sensor, then the Read-Event-Record-Segments command does not have to be supported. Depending on the design of the sensor, this might also apply to other commands.

The use of security that varies over the previously described parameters requires the sensor to reply to commands that fail the security requirements with a Response Code. Improved design results from using a large enough Response Code field to describe a variety of possible reasons for failure of the sensor to obey the command. There are variations in the nature of the Response Codes, particularly in the four classes of commands that perform data reads, data writes, data erasures, and security operations. These four cases are in general as given by Table 18 through Table 21 but are given in exact detail in the specified responses to each command. Table 18 through Table 21 are informational only and do not specify actual command codes within this standard.

Table 18 —General form of Read Commands Response Codes

000	Sensor not properly addressed, reply truncated following last bit of response code.
001	Command not recognized, reply truncated following last bit of response code. This includes failure to decrypt if forward link encryption is in effect.
010	Unspecified failure, reply truncated following last bit of battery code.
011	Air Interface Security Failure, reply truncated following last bit of response code.
100	Sensor Security Failure (bad token), reply truncated following last bit of response code.
101	Failure due to length mismatch of either security token or RN.
110	RFU or Failure due to command details not being supported by the particular sensor, reply truncated following last bit of battery code.
111	Success.

Table 19 —General form of Write Commands Response Codes

000	Sensor not properly addressed, reply truncated following last bit of response code.
001	Command not recognized, reply truncated following last bit of response code. This includes failure to decrypt if forward link encryption is in effect.
010	Unspecified failure, reply truncated following last bit of battery code.
011	Air Interface Security Failure, reply truncated following last bit of response code.
100	Sensor Security Failure (bad token), reply truncated following last bit of response code.
101	Various, such as security cannot be so programmed or length mismatch of either security token or RN.
110	Failure due to command details not being supported by the particular sensor, reply truncated following last bit of battery code.
111	Success.

Table 20 —General form of Erase Commands Response Codes

000	Sensor not properly addressed, reply truncated following last bit of response code.
001	Command not recognized, reply truncated following last bit of response code.
010	Unspecified failure, reply truncated following last bit of battery code.
011	Air Interface Security Failure, reply truncated following last bit of response code.
100	Sensor Security Failure, reply truncated following last bit of response code.
101	Erase failed to complete, reply truncated following last bit of battery code.
110	Failure due to RN or security token length mismatch.
111	Success.

Table 21 —General form of Security Commands Response Codes

000	Sensor not properly addressed, reply truncated following last bit of response code.
001	Command not recognized, reply truncated following last bit of response code.
010	Unspecified failure, reply truncated following last bit of battery code.
011	Air Interface Security Failure, reply truncated following last bit of response code.
100	Sensor Security Failure, reply truncated following last bit of response code.
101	RFU for specific command, such as RN or Security Token length mismatch.
110	RFU for specific command, such as Failure due to command details not being supported by the particular sensor, reply truncated following last bit of battery code.
111	Success.

The command response codes 000 to 100 are the same for all commands. This covers the common cases of sensor not correctly addressed, command not recognized (it is essential for this response to be identical for all commands because some commands may not be supported), unspecified failure (often a result of a low battery), air interface security failure, and sensor security failure. Since one of the three remaining codes (101, 110, 111) is for command success indication (usually 111), that leaves two codes for other command specific responses. A common case for at least one of those two codes is for command details to not be supported.

NOTE—If multiple errors occur in parallel, the highest priority error shall be Sensor Security Failure, the next highest shall be Air Interface Security Failure, and the next highest shall be Unspecified (which reports the battery state).

7.1 Read-Sensor-Identifier

This command provides the linkage between the RFID means of addressing the sensor and a sensor- related method of identifying the sensor. The command is intended for use in RFID tag architectures where the sensor component is interchangeable. The command is not required if the RFID tag has its own method of providing a mapping among a port number, sensor identifier, and the Primary TEDS Fields 1, 2, and 3.

For security purposes, this command is considered a “Read” command. Table 22 presents the Read-Sensor-Identifier command.

Table 22 —Read-Sensor-Identifier command

	Command	Parameter
# bits	5	1
Description	00001	0 = respond with the subaddress number and identifier 1 = respond with the subaddress and Primary TEDS Fields 1, 2, and 3

This command differs from the Read-Primary-Characteristics—TEDS (see 7.2) in that its sole purpose is to provide a means of addressing a particular sensor in subsequent rounds of enquiries. Table 23 presents the Read-Sensor-Identifier response.

Table 23 —Read-Sensor-Identifier response

	Response	Response code	Battery status code	7-Bit subaddress	Sensor ID (conditional)	Sensor type (conditional)
# bits	5	3	1	7	64	15
Description	00001	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = RFU. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low		See 6.3	Primary TEDS Fields 1, 2, and 3

7.2 Read-Primary-Characteristics—TEDS

The purpose of this command is to provide information about the sensor's basic function. It has the following two options for the response: 1) to return only the primary characteristics TEDS or 2) to return the primary characteristics TEDS as well as the unique sensor identifier.

The length of the Sensor Communications ID (the third element in the command) is determined by the value of the Sensor Address Type. If the Sensor Address Type = 00, then the type code forms part of the command, but there are no additional bits for the Sensor Communications ID. This is because the addressing mechanism is included on the RFID tag and incorporated within the RFID commands.

If a command contains a Sensor Address Type that is not supported by the sensor, then an error shall be returned. For security purposes, this command is considered a "Read" command. Table 24 presents the Read-Primary-Characteristics—TEDS command, and Table 25 presents the Read-Primary-Characteristics—TEDS response.

NOTE—These options for addressing apply to all the subsequent commands specified in this clause.

Table 24 —Read-Primary-Characteristics—TEDS command

	Command	Sensor address type	Sensor Comms ID	Parameter	Reader security token (see NOTE 1 and NOTE 3)	New reader RN (see NOTE 2 and NOTE 3)
# bits	5	2	0 or 7 or 15 or 64	1	0 or Sensor RN length	0 or Reader RN length
Description	00010	If = 00, then no subaddressing.	No data encoded for this parameter	0 = respond only with Primary TEDS 1 = Primary TEDS + Unique Identifier	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits			
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits			
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits			
NOTE 1—The Reader Security Token is a reader-encrypted version of the last tag provided RN. It is only included if Reader Continuing Authentication is in effect, which is an option under the Reader-Authenticate command. The length is provided in the Reader-Authenticate command as the length of the Tag Continuing RN, which instructs the tag to keep providing RNs of the desired length on every tag reply.						
NOTE 2—The new reader-supplied random number is only supplied if Sensor Continuing Authentication is in effect as commanded in the Reader-Authenticate command. In that case, the Reader-Authenticate command also informs the sensor of the Continuing Reader RN Length of the new Reader RN that the reader will supply with each command.						
NOTE 3—If data encryption is in effect, it provides a form of Continuing Authentication in which a possibly weaker encryption may apply. Therefore, the option of both Continuing Authentication and data encryption to apply simultaneously is preserved.						

Table 25 —Read-Primary-Characteristics—TEDS response

	Response	Response code	Battery status code	Sensor ID (conditional)	Characteristics TEDS	Sensor security token (see NOTE 1)	New sensor RN (see NOTE 2)
# bits	5	3	1	64	128	0 or Reader RN length	0 or Sensor RN length
Description	00010	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. This includes failure to decrypt if forward encryption is in effect. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure (bad token), reply truncated after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low	See 6.3	All Primary TEDS Fields	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

NOTE 1—The Sensor Security Token is a sensor-provided encryption of the random number last provided by the reader. It is only included if Sensor Continuing Authentication is in effect, which is an option under the Reader-Authenticate command.

NOTE 2—The new sensor-supplied random number is included only if Reader Continuing Authentication is in effect as commanded in the Reader-Authenticate command.

7.3 Write-Sample-and-Configuration

This command delivers the complete Sample and Configuration Record as a bit string from the application to the sensor. It is the responsibility of the application to ensure that various fields are logically structured in accordance with the capabilities of the sensor, as defined by the Primary Sensor Characteristics TEDS. The Erase Event-Administration Record (see 6.6) shall be the first command that is invoked before any attempt is made to reconfigure the sensor.

If either a lower alarm threshold or an upper alarm threshold is supported by the sensor but does not require to be set by the application, a string of bits value 0_2 is used to indicate that no threshold is set. This string is the same length as the data transmission value (Field 5 of the Primary Sensor Characteristics TEDS; see 6.4).

NOTE—This is possible because this all-zero value defines the absolute lowest reading capability of the sensor, therefore making it impossible to record a lower value.

For security purposes, this command is considered a "Write" command (Table 26). Table 27 presents the Write-Sample-and-Configuration response.

Table 26 —Write-Sample-and-Configuration command

	Command	**Sensor address type**	**Sensor Comms ID**	**Mandatory parameters**	**Upper alarm threshold (mandatory)**	**Lower alarm threshold (mandatory)**	**Reader security token**	**New reader RN**
# bits	5	2	0 or 7 or 15 or 64	83	1 ~ 32	1 ~ 32	0 or Sensor RN length	0 or Reader RN length.
Description	00011	If = 00, then no subaddressing.	No data encoded for this parameter	Sample & Configuration Fields 1 – 11.	See NOTE	See NOTE	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else new RN of Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits					
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits					
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits					

NOTE—Field 5 of the Primary Sensor Characteristics TEDS (Data Resolution in Table 2) and the "Alarm Values Set" Field 4 of Table 11 determine the number of bits in the upper and lower alarm threshold fields that are part of the total payload of this command.

Table 27 —Write-Sample-and-Configuration response

	Response	**Response code**	**Battery status code**	**Sensor security token**	**New sensor RN**
# bits	5	3	1	0 or Reader RN length	0 or Sensor RN length
Description	00011	000 = Sensor not properly addressed, reply truncated following last bit of response code. 001 = Command not recognized, reply truncated following last bit of response code. 010 = Unspecified failure, reply truncated following last bit of battery code. For this command, this includes the case of security token or Sensor RN length mismatch. 011 = Air Interface Security Failure, reply truncated following last bit of response code. 100 = Sensor Security Failure, reply truncated following last bit of response code. 101 = Security cannot be programmed as given in the air interface or sensor security function code. 110 = Failure due to command details not being supported by the particular sensor, reply truncated following last bit of battery code. 111 = Success.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

This command must be prevented from unauthorized reductions in the security levels of the Air Interface Function Code and Sensor Security Function Code. Thus, once these fields are programmed above 000, this command will return an Air Interface Security Failure 011 or Sensor Security Failure 100 without reprogramming the Sample and Configuration if the user cannot authenticate having write access.

The response code 110 indicates the following:

- An error in the structure of the command (e.g., an alarm being set for Fields 12 or 13 being inconsistent with the value in Field 4, or a length in Field 12 or 13 being inconsistent with Field 5 of the Primary Sensor Characteristics TEDS Record).
- A command request to set an alarm that is not supported.

7.4 Read-Sample-and-Configuration

This command reads the values previously set in the Sample and Configuration record, including the threshold value of the alarm levels. For security purposes, this command is considered a “Read” command. Table 28 presents the Read-Sample-and-Configuration command. Table 29 presents the Read-Sample-and-Configuration response.

Table 28 —Read-Sample-and-Configuration command

	Command	Sensor address type	Sensor Comms ID	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	0 or Sensor RN length.	0 or Reader RN length.
Description	00100	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits		
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits		
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits		

Table 29 —Read-Sample-and-Configuration response

	Response	Response code	Battery status code	Mandatory parameters	Upper alarm threshold (mandatory)	Lower alarm threshold (mandatory)	Sensor security token	New sensor RN
# bits	5	3	1	83	1 to 32	1 to 32	0 or Reader RN length.	0 or Sensor RN length.
Description	00100	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated	0: Battery OK 1: Battery low	Sample & Configuration Fields 1 through 11.	See NOTE	See NOTE	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else new RN of Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

		after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.						
NOTE—Field 5 of the Primary Sensor Characteristics TEDS (Data Resolution in Table 2) and the “Alarm Values Set” Field 4 of Table 11 determine the number of bits in the upper and lower alarm threshold fields that are part of the total response of this command. Some sensors have the capability to support both upper and lower alarm thresholds, but only one might be configured for the application. To distinguish which alarm thresholds are intended to be set, an alarm value with a string of 0_2 is used to indicate that the alarm level is not set. This is possible because this value defines the absolute lowest reading capability of the sensor, therefore making it impossible to record a lower value.								

The upper and lower alarm threshold values are returned if set, or with a 0 value if not set.

7.5 Read-Alarm-Status

This command provides the application with information as to whether any alarm has been set for the sensor. If an alarm has been triggered, then the application may use the Read-Data-Logged-Record (see 7.6), command to access more detailed data.

For security purposes, this command is considered a “Read” command. Table 30 presents the Read-Alarm Status command. Table 31 presents the Read-Alarm-Status response.

Table 30 —Read-Alarm-Status command

	Command	**Sensor address type**	**Sensor Comms ID**	**Reader security token**	**New reader RN**
# bits	5	2	0 or 7 or 15 or 64	0 or Sensor RN length.	0 or Reader RN length.
Description	00101	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits		
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits		
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits		

Table 31 —Read-Alarm-Status response

	Response	**Response code**	**Battery status code**	**Alarms set**	**Alarms triggered**	**Sensor map for measurement types**	**Sensor security token**	**New sensor RN**
# bits	5	3	1	2	4	16	0 or Reader RN length	0 or Sensor RN length
Description	00101	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low	See 6.5.4	See 6.6.6	Primary TEDS Field 4	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

7.6 Read-Single-Memory-Record

This command is used to provide the application with values from single value measurement types. The command addresses a single memory record.

For security purposes, this command is considered a “Read” command. Table 32 presents the Read-Single-Memory-Record command.

Table 32 —Read-Single-Memory-Record command

	Command	**Sensor address type**	**Sensor Comms ID**	**Measurement type**	**Reader security token**	**New reader RN**
# bits	5	2	0 or 7 or 15 or 64	4	0 or Sensor RN length.	0 or Reader RN length.
Description	00110	If = 00, then no subaddressing.	No data encoded for this parameter	Permitted values: 0000 to 1001 RFU: 1110 and 1111. Other values are for data logged records.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits			
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits			
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits			

The response for Types 6 and 7 require the appropriate specific sample count value to be obtained from the Event Administration Record.

Table 33 shows responses for different types of single-memory records that are supported by the standard. The last four rows of the table indicate the structure of the response for each of the measurement types. The same response structure can be used because responses are synchronous with commands.

Table 33 —Read-Single-Memory-Record response

	Response	**Response code**	**Battery status code**	**Specific sample count value (conditional)**	**Sampled data (conditional)**	**Sample count (conditional)**	**Sensor security token**	**New sensor RN**
# bits	5	3	1	16	1 bit to 32 bits	8	0 or Reader RN length.	0 or Sensor RN length.
Description	00110	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low		Size based on memory resolution (see 6.4.5)	Count up to 255	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
Types 0 to 5	YES			No	Yes	No		
Type 6	YES			See 6.6.7	Yes	No		
Type 7	YES			See 6.6.8	Yes	No		
Types 8 and 9	YES			No	No	Yes		

7.7 Read-Event-Administration-Record

This command is used to provide the application with all the essential parameters to be able to process event records and other critical records selectively.

For security purposes, this command is considered a “Read” command. Table 34 presents the Read-Event-Administration-Record command.

Table 34 —Read-Event-Administration-Record command

	Command	Sensor address type	Sensor Comms ID	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	0 or Sensor RN length.	0 or Reader RN length.
Description	00111	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits		
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits		
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits		

Fields 1 through 4 of the Event Administration Record determine the size, respectively, of measurement types 10, 11, 12, and 13 if they are present. If a measurement type is not present, then the field is left empty in the response because the presence of the measurement type is declared as part of the primary TEDS.

The presence of Fields 7 through 10 is also associated with the presence of the measurement type as declared as part of the primary TEDS. If the field is present, then it is 16 bits wide but has a zero value until an appropriate event takes place. Some of these (Fields) can only exist if the threshold has been crossed.

Table 35 presents the Read-Event-Administration-Record response.

Table 35 —Read-Event-Administration-Record Response

	Response	Response code	Battery status code	Memory capacity	Sample count	Alarms triggered	Sample count values	Password Read-Lock and Write-Lock status flags	MIP	Sensor security token	New sensor RN
# bits	5	3	1	0 to 36	16	4	0, 16, 32, 48, or 64	4	1	0 or Reader RN length	0 or Sensor RN length
Description	00111	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low	Fields 1 to 4 of this record, with data only if field is present	Field 5 of this record	Field 6 of this record See 6.6.6	Fields 7 through 10	Present only if direct sensor security is supported.	0: Sensor is stopped 1: Mission in Progress	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

7.8 Read-Event-Record-Segments

This command is designed to read a specified number of segments of a particular event record, as defined by the measurement type. Although the command is highly flexible and allows a very large number of segments to be transferred across the air interface, the practical experience of the radio environment will determine a realistic number of segments that can be transferred reliably.

For security purposes, this command is considered a “Read” command. Table 36 presents Read-Event-Record-Segments command.

Table 36 —Read-Event-Record-Segments command

	Command	Sensor address type	Sensor Comms ID	Measurement type	First segment number	Number of segments	Last segment number	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	4	3 or 11	6	3 or 11	0 or Sensor RN length	0 or Reader RN length
Description	01000	If = 00, then no subaddressing.	No data encoded for this parameter	Permitted values: 1010 to 1101 RFU: 1110 and 1111 Other values are for single records	3 bits for type 1011 11 bits for types 1010, 1100, 1101 Also see NOTE 1	See NOTE 2	3 bits for type 1011 11 bits for types 1010, 1100, 1101 Also see NOTE 3	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits						
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits						
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits						

NOTE 1—The value of the first segment shall not be greater than the number of segments supported by the sensor.

NOTE 2—The number of segments shall be no greater than (Last Segment Number) – (First Segment Number) + 1. The number may be smaller as discussed below.

NOTE 3—The value of the last segment shall not be greater than the number of segments supported by the sensor.

The value of the total number of segments may have one of two logical values as follows:

a) If the number covers the span from first to last segment in the command, then the sensor shall respond with the entire packet of all the required segments.

b) If the number calls for fewer segments, the intention is for the sensor to respond with that number of segments encapsulated in an RFID response, then to respond immediately with another similar sized packet and to continue until all the segments that were requested have been transmitted. This command and response asynchronous cycle shall only be possible under the following conditions:

1) The sensor shall have the facility to support this type of response. If it does not, then it shall ignore the number of segments and deliver all the requested segments as a data packet.

2) The value of the Number of Segments parameter in the command shall be an integer fraction of the total number of segments requested in the command. If this is not the case, then the sensor shall ignore the number of segments and deliver all the requested segments as a data packet.

Table 37 presents Read-Event-Record-Segments response.

Table 37 —Read-Event-Record-Segments Response

	Response	Response code	Battery status code	Segment data bits	CRC-16	Sensor security token	New sensor RN
# bits	5	1	1	32n	16	0 or Reader RN length .	0 or Sensor RN length.
Description	01000	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low	The value n is defined by the size of the data resolution, pre-pended by the sample count	See below	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
				These two fields are repeated the number of times that equals the number of segments requested in the command.			

The number of the segment data bits per segment is 32*n*, where *n* is determined as follows:

The value of *n* for measurement types 10 and 13 is equal to the size of the data resolution. The value of *n* for measurement type 11 is 8 bits for the sample count plus the size of the data resolution. The value of *n* for measurement type 12 is 16 bits for the sample count plus the size of the data resolution.

A CRC-16 (CRC-CCITT and ISO/IEC 13239) using the polynomial $x^{16} + x^{12} + x^5 + 1$ shall be generated by the sensor for each segment transmitted, with the following exception: If the command only requests one segment, then the CRC-16 shall not be generated.

NOTE—This is because the RFID process will generate a CRC-16 for the entire message that is being transmitted across the air interface.

If multiple segments are included in the response, then they shall be presented in packets from the lowest segment requested through to the highest segment requested.

7.9 Read-Partial-Event-Record-Segment

If there is a requirement to read less than a full segment, either because of the RF conditions or simply to get fewer records, the following command can be used.

For security purposes, this command is considered a "Read" command. Table 38 presents Read-Partial-Event-Record-Segment command.

Table 38 —Read-Partial-Event-Record-Segment command

	Command	**Sensor address type**	**Sensor Comms ID**	**Measurement type**	**Segment number**	**First sample number**	**Number of samples**	**Reader security token**	**New reader RN**
# bits	5	2	0 or 7 or 15 or 64	4	3 or11	5	5	0 or Sensor RN length	0 or Reader RN length
Description	01001	If = 00, then no subaddressing.	No data encoded for this parameter	Permitted values: 1010 to 1101 RFU: 1110 and 1111 Other values are for single records not covered by this command.	3 bits for type 1011 11 bits for types 1010, 1100, 1101 Also see NOTE 1.	00000 = 1 11111 = 32	00000 = 1 11111 = 32 Also see NOTE 2.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits						
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits						
		If = 11 use sensor ID subaddressing.	Sensor ID 64 bits						
NOTE 1—The value of the segment number shall not be greater than the number of segments supported by the sensor.									
NOTE 2—The sum of (First Sample Number) + (Number of Samples) shall not be greater than the number of sample values (32) in a segment.									

This command allows any number of sample values within the segment to be targeted. The sensor requires processing capability to identify the start position of the first sample value requested by the command. There is a high probability that this is not aligned with the boundary of any physical memory unit.

Table 39 presents the Read-Partial-Event-Record-Segment response.

Table 39 —Read-Partial-Event-Record-Segment response

	Response	Response code	Battery status code	Subsegment data bits	Sensor security token	New sensor RN
# bits	5	3	1	s times n	0 or Reader RN length.	0 or Sensor RN length.
Description	01001	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low	The value n is defined by the size of the data resolution, pre-pended by the sample count (if appropriate). The value of s is defined by the number of sample values requested.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

No CRC-16 generation is necessary for this response because of its relatively small size. The CRC-16 generated by the RFID tag is the only requirement.

7.10 Write-Event-Administration-Field-7

This command is mandatory only if the sensor supports measure code 6 of Table 3, which sample count at predetermined time. Field 7 of the Event Administration Record contains a 16-bit value that equates to this sample count at a predetermined time. This therefore predetermined sample count is calculated by the application and written to the sensor using this command. As the sample counter increments its values from the point of initialization, the sample count will eventually equal the 16-bit value in Field 7. At this point, the reading from the sensor is written to the appropriate record.

For security purposes, this command is considered a “Write” command. Table 40 presents the Write-Event-Administration-Field-7 command. Table 41 presents the Write-Event-Administration-Field-7 response.

Table 40 —Write-Event-Administration-Field-7 command

	Command	Sensor address type	Sensor Comms ID	Field 7 sample count	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	16	0 or Sensor RN length.	0 or Reader RN length.
Description	01010	If = 00, then no subaddressing.	No data encoded for this parameter	Event Administration Field 7	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits			
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits			
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits			

Table 41 —Write-Event-Administration-Field-7 response

	Response	Response code	Battery status code	Sensor security token	New Sensor RN
# bits	5	3	1	0 or Reader RN length	0 or sensor RN length
Description	01010	000 = Sensor not properly addressed, reply truncated following last bit of response code. 001 = Command not recognized, reply truncated following last bit of response code. 010 = Unspecified failure, reply truncated following last bit of battery code. 011 = Air Interface Security Failure, reply truncated following last bit of response code. 100 = Sensor Security Failure, reply truncated following last bit of response code. 101 = Failure due to length mismatch of either security token or RN. 110 = Failure due to command details not being supported by the particular sensor, reply truncated following last bit of battery code. 111 = Success.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

The response code 110 indicates the following:

- An error in the structure of the command (e.g., the field or the measurement code is not supported by the sensor).

7.11 Read-Any-Field

This command enables the application to read data from any single field on any record of the sensor. The field is addressed by using a code to identify the record and then the field number within the record.

Data logs cannot be read using this command. For security purposes, this command is considered a "Read" command. Table 42 presents the Read-Any-Field command. Table 43 presents the Read-Any-Field response.

Table 42 —Read-Any-Field command

	Command	**Sensor address type**	**Sensor Comms ID**	**Record ID**	**Field number**	**Reader security token**	**New reader RN**
# bits	5	2	0 or 7 or 15 or 64	2	5	0 or Sensor RN length.	0 or Reader RN length.
Description	01011	If = 00, then no subaddressing.	No data encoded for this parameter	00 = Primary Sensor Characteristics TEDS 01 = Sample & Configuration 10 = Event record 11 = Event Administration		0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits				
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits				
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits				
NOTE—The following combinations of Record ID and Field Number are not permitted because they identify complex event logs: 10 01010 10 01011 10 01100 10 01101							

Table 43 —Read-Any-Field response

	Response	**Response code**	**Battery status code**	**Selected field data**	**Sensor security token**	**New sensor RN**
# bits	5	3	1	*n*	0 or Reader RN length.	0 or Sensor RN length.
Description	01011	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Failure due to length mismatch of either security token or RN. 110 = RFU. 111 = Success.	0: Battery OK 1: Battery low		0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

7.12 Erase-Event-Administration Record

This command sets to 0_2 Fields 5, 6, 7, 8, 9, and 10 of the Event Administration Record. The sensor manufacturer shall lock Fields 1, 2, 3, and 4. The Erase-Event-Administration Record command shall be the first command that is invoked before any attempt is made to reconfigure the sensor.

For security purposes, this command is considered a “Write” command. Table 44 presents the Erase-Event-Administration-Record command. Table 45 presents the Erase-Event-Administration-Record response.

Table 44 —Erase-Event-Administration-Record command

	Command	Sensor address type	Sensor Comms ID	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	0 or Sensor RN length.	0 or Reader RN length.
Description	01100	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits		
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits		
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits		

Table 45 —Erase-Event-Administration-Record response

	Response	Response code	Battery status code	Sensor security token	New sensor RN
# bits	5	3	1	0 or Reader RN length.	0 or Sensor RN length.
Description	01100	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Erase failed to complete, battery code provided 110 = Failure due to length mismatch of either security token or RN. 111 = Success.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

7.13 Erase-Event-Records

This command erases the complete set of event records. It shall be invoked before any attempt is made to reconfigure the sensor. For security purposes, this command is considered a "Write" command. Table 46 presents the Erase-Event-Records command.

Table 46 —Erase-Event-Records command

	Command	Sensor address type	Sensor Comms ID	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	0 or Sensor RN length.	0 or Reader RN length.
Description	01101	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits		
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits		
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits		

If the sensor supports any form of data logged record(s), then the process to erase the event records might take some time. Table 47 presents the Erase Event-Records response.

Table 47 —Erase-Event-Records response

	Response	Response code	Battery status code	Sensor security token	New sensor RN
# bits	5	3	1	0 or Reader RN length.	0 or Sensor RN length.
Description	01101	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Erase failed to complete, battery code provided 110 = Failure due to length mismatch of either security token or RN. 111 = Success.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

7.14 Erase Sample-and-Configuration Record

This command erases the complete Sample and Configuration Record. It may be used as a housekeeping step before the sensor is reconfigured.

For security purposes, this command is considered a "Write" command. Table 48 presents Erase Sample-and-Configuration-Record command. Table 49 presents Erase Sample-and-Configuration-Record response.

NOTE—The Write-Sample-and-Configuration command may be used instead to simply overwrite a preexisting sample and configuration set of conditions that no longer apply.

Table 48 —Erase Sample-and-Configuration-Record command

	Command	Sensor address type	Sensor Comms ID	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	0 or Sensor RN length.	0 or Reader RN length.
Description	01110	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Port Number 7 bits		
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits		
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits		

Table 49 —Erase Sample-and-Configuration-Record response

	Response	Response code	Battery status code	Sensor security token	New sensor RN
# bits	5	3	1	0 or Reader RN length.	0 or Sensor RN length.
Description	01110	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. 011 = Air Interface Security Failure, reply truncated after response code. 100 = Sensor Security Failure, reply truncated after response code. 101 = Erase failed to complete, battery code provided 110 = Failure due to length mismatch of either security token or RN. 111 = Success.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

7.15 Begin-End-Mission

This command allows the sensor to be configured at one time and begin its mission at another time. This process saves both memory and battery life via effectively implementing an ON–OFF control. The definition of "begin mission" is to begin the monitor delay timer in preparation of taking data in the monitoring process or go straight to monitoring if the monitor delay is zero.

To use this command to end a mission in progress or to begin a new mission after the end of a mission, the user must have appropriate authority as determined by the Begin-End-Mission Authority bit in the Sample and Configuration Record and the Sensor Security Function Code in use. For example, if the user is required to have write authority to end a mission or to begin a new mission if a mission is in progress or has ended by virtue of full memory or end of sampling, then if a user with only read privileges attempts to use this command to perform these actions a security failure response will be the result.

The security level of this command cannot be specified as simply as Read or Write, as it is a function of the Begin-End-Mission Authority bit in the Sampling and Configuration Record (Table 17), what is being commanded (begin or end), and if beginning whether it is a first mission or a re-mission. See Table 50 through Table 52 for details.

Table 50 —Begin-End-Mission command

	Command	**Sensor address type**	**Sensor Comms ID**	**Begin/end** **NOTE 1**	**Reader security token**	**New reader RN**
# bits	5	2	0 or 7 or 15 or 64	1	0 or Sensor RN length.	0 or Reader RN length.
Description	01111	If = 00, then no subaddressing.	No data encoded for this parameter	0 = Begin Mission 1 = End Mission	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits			
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits			
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits			
NOTE 1—Begin Mission means start monitor delay timer. If monitor delay is zero, then the sensor goes straight to monitoring. End Mission means stop monitor delay timer and reset it to the TEDS specified value if the monitor delay timer is running. If monitoring is in progress, then End Mission means end monitoring.						

Table 51 —Begin-End-Mission response

	Response	**Response code (see NOTE 1)**	**Battery status code**	**Sensor security token**	**New sensor RN**
# bits	5	3	1	0 or Reader RN length.	0 or Sensor RN length.
Description	01111	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, still send battery code. 011 = Air interface security failure, reply truncated after response code. 100 = Sensor security failure, reply truncated after response code. See NOTE 2. 101 = Sensor unable to start or end mission, still send battery code. 110 = Failure due to length mismatch of either security token or RN. 111 = Successful.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

NOTE 1—The special nature of this command requires a unique set of response codes.

NOTE 2—Code 101 will be provided if the user attempts to end a mission or begin a new mission after a mission has ended when the user only has read privileges and the Begin-End-Mission Authority bit in the Sample and Configuration Record is set to require write privileges for ending a mission or beginning a new mission.

Table 52 —Required sensor security function code authorization for various cases of the Begin-End-Mission command

Begin/End parameter in command	Begin-End-Mission Authority bit in Sample and Configuration record	Re-mission or First Mission status	Security needed in Sensor Security Function Code to execute command	Comment
0 = Begin 1= End	0 = Need write authority to end or to re-mission 1 = Read authority is allowed to end or re-mission	0 = Re-mission 1= First Mission		
0	0	0	Write	Need Write authority to begin because it is a remission.
0	0	1	Read	Only need Read authority to begin because it is a first mission.
n/a	1	n/a	Read	Only need Read authority because Begin-End-Mission Authority bit in Sample and Configuration Record so authorizes.
1	0	n/a	Write	Must have Write authority to end any mission if Begin-End-Mission Authority bit in Sample and Configuration Record requires Write.

7.16 Challenge

This command is only supported by sensors that support direct sensor security, and only then for those that support authentication encryption for two-way authentication using secure token exchange. This command begins the two-way, two-way authorization procedure by “challenging” the sensor to authenticate itself to the RFID reader. It, thus, provides a 16-, 32-, 64-, or 128-bit random number to the tag to use with the key for encrypting the security token for the tag reply. It also informs the sensor of the length of the RN the tag should return for the reader to use subsequently for generating a token for reader authentication. If encryption is not supported (the reader should know this from the TEDS or from a sensor directory on the tag), then the “Provide Encryption Responses” flag in the command provides a means of limiting the response of the command to that appropriate for one-way (reader only) authentication in the absence of encryption. That limited response is for the sensor to reply only with the Sensor Security Function Code in use in the sensor.

For security purposes, this command is classified as “Security Set Up.” It is seeking to establish a secure session by authenticating the tag, following which the reader will authenticate (provide a security token using the key and a tag provided random number), so neither the sensor security nor air interface security should ever reject it (refuse to process it), unless the sensor does not support the command. A noteworthy point is that the sensor shall respond to this command even when Continuing Authentication is in force (it overrides Continuing Authentication). This provides a recovery method should the sensor and reader fail to maintain Continuing Authentication due to bit errors in the RF communications. Table 53 and Table 54 present the Challenge command and the Challenge response, respectively.

Table 53 —Challenge command

	Command	**Sensor address type**	**Provide encryption responses**	**Sensor Comms ID**	**Temp sensor random number length code**	**Temp reader random number length code**	**Reader provided random number**
# bits	5	2	1 (see NOTE 1)	0 or 7 or 15 or 64	0 or 3 (see NOTE 2)	0 or 3 (see NOTE 3)	0, 16, 32, 64, or 128
Description	10000	If = 00, then no subaddressing.	0: No (reply only with Sensor Security Function Code) 1: Yes (Also reply with Encryption code, encrypted security token, and tag generated RN for subsequent reader authentication.	No data encoded for this parameter	Not used if "Provide Encryption Responses" = 0/No If "Provide Encryption Responses" = 1/Yes, then: 000: 16 bits 001: 32 bits 010: 64 bits 011: 128 bits 100-111: RFU	Not used if "Provide Encryption Responses" = 0/No If "Provide Encryption Responses" = 1/Yes, then: 000: 16 bits 001: 32 bits 010: 64 bits 011: 128 bits 100-111: RFU	0: Only zero if "Provide Encryption Responses" = 0/No If "Provide Encryption Responses" = 1/Yes, then according to temp reader random number length code.
		If = 01, use 7 bit subaddressing.		Subaddress 7 bits			
		If = 10, use sensor type subaddressing.		Primary TEDS Fields 1, 2, and 3 15 bits			
		If = 11, use sensor ID subaddressing.		Sensor ID 64 bits			

NOTE 1—The reader usually knows from a sensor directory on the tag whether the sensor supports encryption and which methods, so it knows in advance how to set the "Provide Encryption Responses."

NOTE 2—The temporary sensor random number length code instructs the sensor what random number length to reply with to perform initial reader authentication. The reader uses this to generate a reader security token. This random number length or smaller must be indicated as allowed by the TEDS, or an error response will result. If continuing reader authentication is authorized in the Reader-Authentication command, then this sensor random number length may be changed by the Reader-Authentication command.

NOTE 3—The temporary reader random number length code informs the sensor what reader random number length immediately follows. The sensor uses this RN to generate a sensor security token for initial sensor authentication. This random number length or smaller must be indicated as allowed by the TEDS, or an error response will result. If continuing tag authentication is authorized in the Reader-Authentication command, then the reader random number length may be changed by the Reader-Authentication command. This would normally be to a smaller RN length to maintain authentication of an already authenticated session.

Table 54 —Challenge response

	Response	Response code (see NOTE)	Battery status code	Sensor security function code	Authentication encryption function code	Encrypted sensor security token	Sensor random number
# bits	5	3	1	3	3	16, 32, 64, or 128	0, 16, 32, 64, or 128
Description	10000	000 = Sensor not properly addressed, reply truncated following last bit of response code. 001 = Command not recognized, reply truncated following last bit of response code. 010 = Unspecified failure, reply truncated following last bit of battery code. 011 = RFU. 100 = Requested Tag Continuing. Authentication not supported, reply truncated following last bit of battery code. 101 = Requested encryption not supported, truncate reply after Authentication Encryption Function Code. 110 = At least one requested RN length not supported, truncate reply after Sensor Security Function Code. 111 = Success, full reply.	0: Battery OK 1: Battery low	As allowed by TEDS and as programmed in Sample and Configuration Record.	As allowed by TEDS and as programmed in Sampling and Configuration Record. Only provided if encryption is supported.	Same length as Temp Reader Random Number Length as provided in Challenge command. Only provided if encryption is supported.	Per Temp Sensor Random Number Length in Challenge command. Only provided if authentication encryption is supported. Used in remaining part of authentication (initial reader authentication).
NOTE—The special nature of this command requires a unique set of response codes.							

7.17 Reader-Authenticate

This command is only supported if direct sensor security is supported. It is used to provide the security token (two-way, two-way authentication) or covered password (reader only authentication) from reader to tag so that the tag can authenticate the reader. If encryption is supported, then this command follows the Challenge command in the mutual authentication procedure. If encryption is not supported (or is not desired in a given situation), then the authentication reduces to one-way reader authentication only, where the reader uses the Request-RN command to get a random number to cover code the password/key in this command.

This command also serves the purpose of setting up Continuing Authentication if it is to be used to complete a communications session with a particular sensor. Continuing Authentication can be commanded for either or both of the forward and return links. It also provides an opportunity to change the random number lengths used to perform initial authentication, typically to allow shorter RN lengths for a communications that is entering a secure session. This saves noticeable air time, as the RN length shows up twice for each transmission, once for the token of the transmit side and once to provide a new RN for the receive side to use when it transmits.

For security purposes, this command is classified as a "Security Set Up" command. The reader is attempting to authenticate itself, so neither air interface or sensor security should reject the command (decline to process it). However, in processing the command the tag will decline to authenticate the reader if the security token fails or the password does not match its internal password and will, thus, decline to process any commands for which the Sensor Security Function Code in use require a security token or password check. Table 55 presents Reader-Authenticate command. Table 56 presents Reader-Authenticate response.

Table 55 —Reader-Authenticate command

	Command	Sensor address type	Encryption usage (EU)	Reader Cont Auth (see NOTE 1)	Sensor Cont Auth (see NOTE 2)	Sensor Comms ID	Reader token/password (see NOTE 1)	Sensor Cont RN Length (see NOTE 1)	Reader Cont RN Length (see NOTE 2)	New Reader RN (see NOTE 2)
# bits	5	2	1	1	1	0 or 7 or 15 or 64	16, 32, 64, or 128	0 or 3	0 or 3	0, 16, 32, 64, or 128
Description	10001	If = 00, then no subaddressing.	0: Authentication encryption is not in use (a cover coded password is used). 1: Authentication encryption is in use (a key and encrypted RN token is used).	0: No 1: Yes, req each sensor reply to append a new RN of length Sensor RN Length per this command. Each subsequent reader command then provides either cover coded password (EU = 0) or secure token (EU = 1).	0: No 1: Yes, req each sensor reply to append a new security token	No data encoded for this parameter	If encryption based auth in use, this is a token of sensor RN length given in Challenge or Request-RN command. If nonencryption forward link Reader Auth is in use, then this is a password XOR covered using a tag provided RN obtained by the Request-RN command.	Not included if Reader Continuing Authentication = 0 000: 16 001: 32 010: 64 011: 128 100-111: RFU Used for READER cont authentication.	Not included if Sensor Continuing Authentication = 0 000: 16 001: 32 010: 64 011: 128 100-111: RFU Used for SENSOR cont Authentication.	Not included if Sensor Cont Auth = 0. Else of length given by Reader Continuing RN length.
		If = 01, use 7 bit subaddressing.				Subaddress 7 bits				
		If = 10, use sensor type subaddressing.				Primary TEDS Fields 1, 2, and 3 15 bits				
		If = 11, use sensor ID subaddressing.				Sensor ID 64 bits				

NOTE 1—The reader authenticating token or password included in this command is of the sensor RN length that was temporarily commanded in the Challenge command. This Sensor RN length may change (usually to become shorter) per this command when Reader Continuing Authentication is enabled, as a shorter reader security token may be desired for continuing authentication in order to save air time.

NOTE 2—If sensor continuing authentication is ordered, then subsequent sensor security tokens use the Reader Continuing Random Number length provided here. The very next tag reply token uses the particular Reader RN provided here.

Table 56 —Reader-Authenticate response

	Response	Response code (see NOTE 1)	Battery status code	Sensor Cont RN
# bits	5	3	1	0, 16, 32, 64, 128
Description	10001	000 = Sensor not properly addressed, reply truncated following last bit of response code. 001 = Command not recognized, reply truncated following last bit of response code. 010 = Unspecified failure, battery code still provided. See NOTE 2. 011 = Requested encryption not supported (cannot decrypt token), battery code still provided. 100 = At least one requested Continuing Auth not supported, battery code still provided. See NOTE 2. 101 = At least one requested RN length not supported, battery code still provided. See NOTE 2. 110 = Password or token fails, reader not authenticated, battery code still provided. 111 = Success, reader authenticated, battery code provided, tag enters secure state.	0: Battery OK 1: Battery low	
NOTE 1—The special nature of this command requires a unique set of response codes. NOTE 2—If a Reader-Authenticate command requests both a Continuing Authentication Mode and a Random Number Length not supported by the sensor, this shall be reported as 010 unspecified failure.				

7.18 ReadWriteLock-Keys

This command is only supported if direct sensor security is supported. For sensors that support security via sensor keys, this command is used to write, read, write-lock, and read-lock the keys (one each for authentication and data encryption, although they could be the same).

Before the password/key(s) are written, they shall be all zeroes. The first time the password/key(s) are written, they are written in the clear (Encryption Use Flag of this command = 0) and with Sensor Security Function Codes = 000, 001, or 010 (the ReadWriteLock-Keys command is not allowed in 011). It may then be read back in the clear for confirmation, as reading the key before read-lock is applied does not require the key. Following confirmation, password/key(s) should be read-locked so that they may never be read out directly again (it is pointless to allow reading the key under key protection).

Rewriting the key shall always require the current key (even if all zeroes), and it may only be done while the key is not write-locked. If the keys are rewritten, then any read-lock of the old is are cleared to allow confirmation of the new keys via readback in the clear, but the new keys should then be read-locked to prevent unauthorized access. After sensor password/key(s) is set-up, the password/keys may, if desired, be write-locked so that they may not be changed.

For security purposes, this command is classified as a “Key Write” command (it is the only such command in this version), and it always requires the authorization password/key (even if all zeroes) to change the password/key(s) or lock the password/key(s). Even if the password/key(s) are not write-locked, the Security Function Codes must be set to allow writing with the current key if that key is to be rewritten (Sensor Security Function Codes of 000, 001, and 010). See Table 15 for more information.

Details of the ability to read and write keys and to conduct the Read-Lock and Write-Lock operations are as shown in Table 57 through Table 59. In these tables, it is also required that the Sensor Security Function Code be 000, 001, or 010 to perform the Key Read, Write, and Lock operations.

Table 57 —Key requirements in the ReadWriteLock-Keys command

Key Value	Key Write operation ability	Key Read operation ability	Write-Lock operation ability	Read-Lock operation ability
All zeroes	Yes, via all zero key in command if Write-Lock Status = 0/No	Yes, if Read-Lock Status = 0/No (key not required)	Yes, via use of all zero key in command	Yes, via use of all zero key in command
> 0	Yes, via correct key in command if Write-Lock Status = 0/No.	Yes, if Read-Lock Status = 0/No (key not required)	Yes, via use of correct key in command	Yes , via use of correct key in command

Table 58 —ReadWriteLock-Keys command

	Command	Action (see NOTE)	Key switch	Sensor address type	Sensor Comms ID	Key
# bits	5	2	1	2	0 or 7 or 15 or 64	0, or per TEDS length or algorithm.
Description	10010	00: Read 01: Write 10: Write-Lock 11: Read-Lock	0: This command is for authentication password/ key 1: This command is for data encryption key	If = 00, then no subaddressing	No data encoded for this parameter	0 bits for key read or lock operations (cannot be read following read-locking). Key is required to read-lock, write-lock, or write the key. For write, length is as indicated by TEDS length or as overridden by a particular TEDS algorithm definition.
				If = 01, use 7 bit subaddressing.	Subaddress 7 bits	
				If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits	
				If = 11, use sensor ID subaddressing.	Sensor ID 64 bits	

NOTE—The current read-lock and write-lock status flags are contained in the Event Administration Record of Table 17. Except for Field 7, this record is externally read-only.

Table 59 — ReadWriteLock-Key response

	Response	Response code (see NOTE)	Battery status code	Key (see NOTE)
# bits	5	3	1	0, or as TEDS specified
Description	10010	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, reply truncated after battery code. This includes inability to write or read key due to write-lock or read-lock. 011 = Air interface security failure, reply truncated after response code. 100 = Sensor security failure (bad token), reply truncated after response code. 101 = Requested encryption not supported (cannot decrypt), reply truncated after response code. 110 = Command details not supported (such as write data encryption key when sensor does not support data encryption), reply truncated after battery code. 111 = Successful read, write, or lock. If read, key follows.	0: Battery OK 1: Battery low	0 bits if the command was a write or lock operation For allowed read operations, length as indicated in TEDS.
NOTE—The special nature of this command requires a unique set of response codes.				

7.19 Request-RN

This command requests the sensor to provide back a random number of length specified in TEDS to use for either cover coded one-way authentication or for authentication/encryption. For security purposes, this command is classified as a “Security Set Up” command (always executed despite Sensor Security Function Code and authentication status).

Note that this command does not have a Reader Security Token, and the tag shall reply to the command even when Reader Continuing Authentication is in effect. That is because a common use for it is as part of the recovery to the reader detecting a bit error(s) in the last sensor reply. If that bit error is in the sensor RN supplied as part of Reader Continuing Authentication the resulting Reader Security Token would not be correct and would cause the command to be ignored. Recovering from this error without resorting to a new Challenge command and authentication sequence can be performed by getting a new RN from the tag with the Request-RN command, and then repeating the previous command that had previously suffered a bit error in its reply. For security purposes, this is a Security Set-Up class of command.

Although this command does not support or require Reader Continuing Authentication, it does support Sensor Continuing Authentication. Table 60 and Table 61 present the Request-RN command and the Request-RN response, respectively.

Table 60 —Request-RN command

	Command	Sensor address type	Sensor Comms ID	New reader RN
# bits	5	2	0 or 7 or 15 or 64	0 or Reader RN length
Description	10011	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits	
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits	
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits	

Table 61 —Request-RN response

	Response	Response code (see NOTE)	Battery status code	Random number	Sensor security token	New sensor RN
# bits	5	3	1	16, 32, 64, or 128 as per TEDS	0 or Reader RN length	0 or Sensor RN length
Description	10011	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, still send battery code. 011 = Air interface security failure, reply truncated after response code. 100 = Sensor security failure, reply truncated after response code. 101 = RFU. 110 = Failure due to length mismatch of Reader RN, reply truncated after response code. 111 = Successful.	0: Battery OK 1: Battery low		0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
NOTE—The special nature of this command requires a unique set of response codes.						

7.20 Encryption-ON-OFF

This command (Table 62 and Table 63) turns encryption ON or OFF, and it can be specified to apply to either or both of the links as specified in Table 10. For security purposes, this command is classified as a

“Security Control” command that the sensor will only execute if the reader has successfully authenticated and the sensor is in a secured state.

Table 62 —Encryption-ON-OFF command

	Command	**Sensor address type**	**Sensor Comms ID**	**Encryption ON/OFF**	**Reader security token**	**New reader RN**
# bits	5	2	0 or 7 or 15 or 64	2	0 or Sensor RN length.	0 or Reader RN length.
Description	10100	If = 00, then no subaddressing.	No data encoded for this parameter	00: Set encryption off for both links. 01: Set encryption on for the sensor to reader link and off for the reader to sensor link. 10: Set encryption on for the reader to sensor link and off for the sensor to reader link. 11: Set encryption on for both links.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits			
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits			
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits			

Table 63 —Encryption-ON-OFF response

	Response	Response code	Battery status code	Sensor security token	New sensor RN
# bits	5	3	1	0 or Reader RN length.	0 or Sensor RN length.
Description	10100	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, still send battery code. 011 = Air interface security failure, reply truncated after response code. 100 = Sensor security failure, reply truncated after response code. 101 = Requested encryption not supported. 110 = Failure due to length mismatch of either security token or RN, reply truncated after response code. 111 = Successful.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.

7.21 Close-Secure-Session

This command (Table 64 and Table 65) is optional and only provided if direct sensor security is supported. For sensors that support security via sensor key, this command is used to terminate a secure session.

For security purposes, this command is classified as a “Security Control” command, which requires that the reader has authenticated itself to the sensor and that the sensor is in a secured state.

Table 64 —Close-Secure-Session command

	Command	Sensor address type	Sensor Comms ID	Reader security token	New reader RN
# bits	5	2	0 or 7 or 15 or 64	0 or Sensor RN length.	0 or Reader RN length.
Description	10101	If = 00, then no subaddressing.	No data encoded for this parameter	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
		If = 01, use 7 bit subaddressing.	Subaddress 7 bits		
		If = 10, use sensor type subaddressing.	Primary TEDS Fields 1, 2, and 3 15 bits		
		If = 11, use sensor ID subaddressing.	Sensor ID 64 bits		

Table 65 —Close-Secure-Session response

	Response	Response code (see NOTE)	Battery status code	Sensor security token	New sensor RN
# bits	5	3	1	0 or Reader RN length.	0 or Sensor RN length.
Description	10101	000 = Sensor not properly addressed, reply truncated after response code. 001 = Command not recognized, reply truncated after response code. 010 = Unspecified failure, still send battery code. 011 = Air interface security failure, reply truncated after response code. 100 = Sensor security failure, reply truncated after response code. 101 = RFU. 110 = Failure due to length mismatch of either security token or RN, reply truncated after response code. 111 = Successful.	0: Battery OK 1: Battery low	0: If Sensor Continuing Auth is not in effect. Else Reader RN length. Reader RN length: As given in Reader-Authenticate command if Sensor Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.	0: If Reader Continuing Auth is not in effect. Else Sensor RN length. Sensor RN length: As given in Reader-Authenticate command if Reader Continuing Auth is in effect. Currently 16, 32, 64, or 128 bits.
NOTE—The special nature of this command requires a unique set of response codes.					

8. RFID communications

8.1 Support for commands

The commands and responses that are specified in this standard are designed to be encapsulated in an air interface command and response assigned to support this standard. Such air interface transport commands and responses shall be specified within the air interface protocol standard. Once the RFID tag receives the transport command, the encapsulated IEEE 1451.7 command is passed to the sensor component for processing. The sensor response when received by the RFID component is encapsulated in the air interface transport response.

This basic procedure enables different air interface protocols to support sensors in a manner compatible with other RFID data-related processes, but leave the core IEEE 1451.7 commands, responses, and processes understandable for processing by the RFID interrogator, and other processes between the RFID tag and the application

8.2 Addressing and subaddressing of sensors

Sensors as described under this standard are most commonly addressed by either of the following:

- Physical addressing at the tag level, whereby the sensor resides on an actual tag port such as a serial peripheral interface (SPI) port. The port number is provided to the RFID interrogator by a directory structure such as a “Sensor Address Map” that resides in tag memory and documents which sensors are at which port addresses.
- Logical addressing, whereby the sensor is integrated with other tag electronics. The tag may still use a port number provided to the interrogator by a directory structure in tag memory, but in this case, there is no physical port and the port number is merely an abstraction used to distinguish between sensors.

In either of the previously discussed methods, the RFID interrogator need not know whether the sensor is physically or logically ported. It need know only what port number to point the RFID transport command to that carries the sensor commands as payload. The tag strips out the payload and then sends it to the indicated port, whether physical or logical.

The subsequent options include subaddressing the sensors. This process is performed at the sensor command level, via the codes in the Sensor Address Type field of each command. These are repeated in Table 66. These codes require further explanation when used as subaddress indicators, which is provided in the second column.

Table 66 —Sensor address types in all commands

Sensor address type	Normal interpretation
2 bits	
If = 00 then no subaddressing	This setting normally means that there is no further subaddressing, as there is only one sensor on the physical or logical port.
If = 01 use 7 bit subaddressing	This setting normally means that in *addition* to the tag level port, the sensor is subaddressed by an additional "sub-port" number or "subaddress" number. This field is 7 bits.
If = 10 use sensor type subaddressing	This setting normally means that in *addition* to the tag level port, the sensor is subaddressed by a sensor type (there is more than one sensor on the tag level port, but they may be distinguished by being of different types).
If = 11 use sensor ID subaddressing	This setting normally means that in *addition* to the tag level port, the sensor is subaddressed by a tag ID (there is more than one sensor on the tag level port, but they may be distinguished by their different 64 bit ID codes).

Note that it is possible to use Sensor Address Type codes 01, 10, and 11 without there being a formal tag level port. In that case, the RFID transport command does not carry a formal port designation, but this is logically identical to there being a *single* tag level port that need not be specified because it need not be distinguished from any other port.

Annex A

(normative)

Sensor types

Table A.1 provides the list of sensor types currently supported by this standard.

Table A.1—Sensor types

Code	Base or derived value	Special name	Special symbol	Expression in terms of other SI units	Expression in terms of SI base units
0	RFU				
1	Acceleration, linear				$m\ s^{-2}$
2	Angle, degrees		Degrees	0.0174533 rad	
3	Angle, plane	Radian	rad		
4	Concentration, amount of substance	Mole per cubic meter			$mol\ m^{-3}$
5	Concentration, relative		ppm	Parts per million	
6	Conductance, electrical	Siemens	S	$A\ V^{-1}$	$m^{-2}\ kg^{-1}\ s^{3}\ A^{2}$
7	Current, electrical	Ampere	A		A
8	Density, magnetic flux	Tesla	T	$Wb\ m^{-2}$	$kg\ s^{-2}\ A^{-1}$
9	Field strength, magnetic				$A\ m^{-1}$
10	Flux, magnetic	Weber	Wb	V s	$m^{2}\ kg\ s^{-2}\ A^{-1}$
11	Force, electromotive	Volt	V	$W\ A^{-1}$	$m^{3}\ kg\ s^{-3}\ A^{-1}$
12	Force, mechanical	Newton	N		$m\ kg\ s^{-2}$
13	Frequency	Hertz	Hz		s^{-1}
14	Humidity, relative		RH		
15	Intensity, luminous	Candela	cd		cd
16	Length	Meter	m		m
17	Mass	Kilogram	kg		kg
18	power, radiant flux	Watt	W	$J\ s^{-1}$	$m^{2}\ kg\ s^{-3}$
19	Pressure	Pascal	Pa	$N\ m^{-2}$	$m^{-1}kg\ s^{-2}$
20	Status, battery			(1 = low)	
21	Strain				m/m
22	Temperature, absolute	Kelvin	K		K
23	Temperature, Celsius	Degree Celsius	°C		T(K)–273.15
24	Time	Second	s		s
25	Time, days		Days	86 400 s	
26	Time, milliseconds			0.001 s	
27	Velocity, linear				ms^{-1}
28	Volume				m^{3}
29–127	RFU				

Annex B

(normative)

Extension codes

Table B.1 provides the list of chemical substances currently supported by this standard.

Table B.1—Chemical substances

Code	Chemical substance
0	Manufacturer defined
1	Acetone
2	Ammonia
3	Carbon dioxide
4	Carbon monoxide
5	Chlorine
6	Hydrogen
7	Hydrogen chloride
8	Hydrogen cyanide
9	Hydrogen peroxide vapor
10	Hydrogen sulfide
11	Nitrogen oxide/dioxide
12	Oxygen
13	Ozone
14–31	Reserved

Annex C

(informative)

Physical interfaces

This standard currently supports the physical connections listed in this annex. This is not an exhaustive or restrictive listing. The following are typical physical interfaces.

C.1 Interface—serial bus

For more information, see IEEE Std 1451.4-2004 [B5].

General description

Smart sensors typically use a small (embedded) microcomputer, which inputs the sensor analog signal via an ADC and outputs the sensor digital data in serial form. The most popular of the microcomputer serial data buses are described in the following sections. IEEE 1451.7 RFID sensors use a microcomputer or the equivalent and therefore will be expected to connect to the RFID Air Interface via one of these buses. All these have relatively short range and are intended to physically connect a microcomputer to another device.

C.2 1-Wire

For more information, see IEEE Std 1451.4-2004 [B5] and the publications to which it refers.

C.3 SPI

The data are transmitted out of the SO of one device into the SI (serial input) of the other with one bit sent each clock cycle (SCK) for a total of 8 bits. In effect, there is an exchange of bytes between one side and the other. A clock rate of 10 kHz to 100 MHz is common, and this standard recommends 2 MHz. There is an optional chip enable ($\overline{\text{SS}}$), which allows individually enabled multiple slaves to be connected to one master.

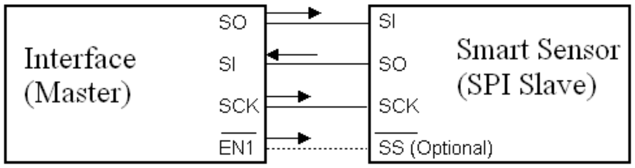


Figure C.1—SPI data and control bus

This is the simplest and most widely implemented of the serial buses. Both the master and the slave can be implemented in software (and hardware) with little difficulty.

C.4 I^{2}C (I squared C)

The I^{2}C (inter-integrated circuit) bus has one clock line and a combined (sequential) input/output data line. The master controls the clock (about 10 kHz in low-speed mode). The data transmitted from the master include the slave address (7-bit), so multiple slaves may be connected to the same line.

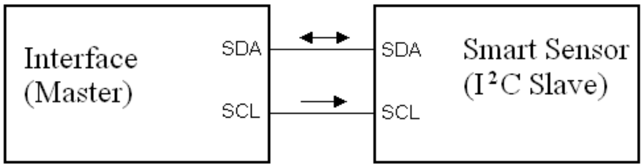


Figure C.2—I^{2}C data and control bus

This is also a popular bus. An advantage is that it uses only two wires. However, the slave is difficult to implement in software, so microcomputers are normally used with hardware implementations of I^{2}C.

Annex D

(informative)

Integration of IEEE 1451.7 transducers with other IEEE 1451 devices

The IEEE 1451 family of standards is designed to allow the interconnection of a variety of transducers (sensors and actuators) in a network, wired or wireless, with different physical configurations, such as point to point, distributed multidrop, and mixed mode. A transducer is connected to a transducer interface module (TIM), which is connected to a network capable application processor (NCAP) for network access. The TIM and NCAP can be implemented as an integrated unit (see Figure D.1) or as two separate units (see Figure D.2). A typical network has multiple NCAPs and clients. This is a suggested implementation method and is not a requirement for meeting the IEEE 1451.7 standard.

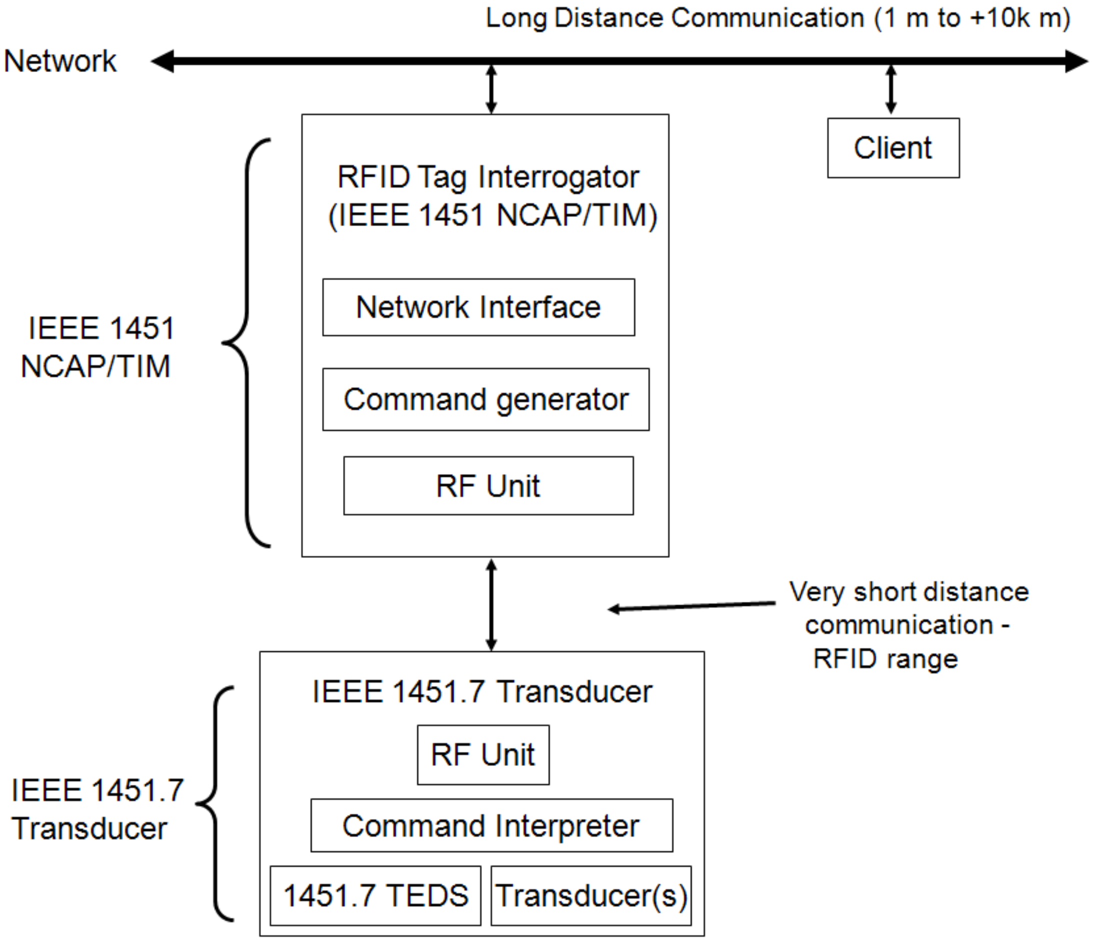


Figure D.1—Example of an integrated IEEE 1451 NCAP/TIM module with an IEEE 1451.7 transducer in a network

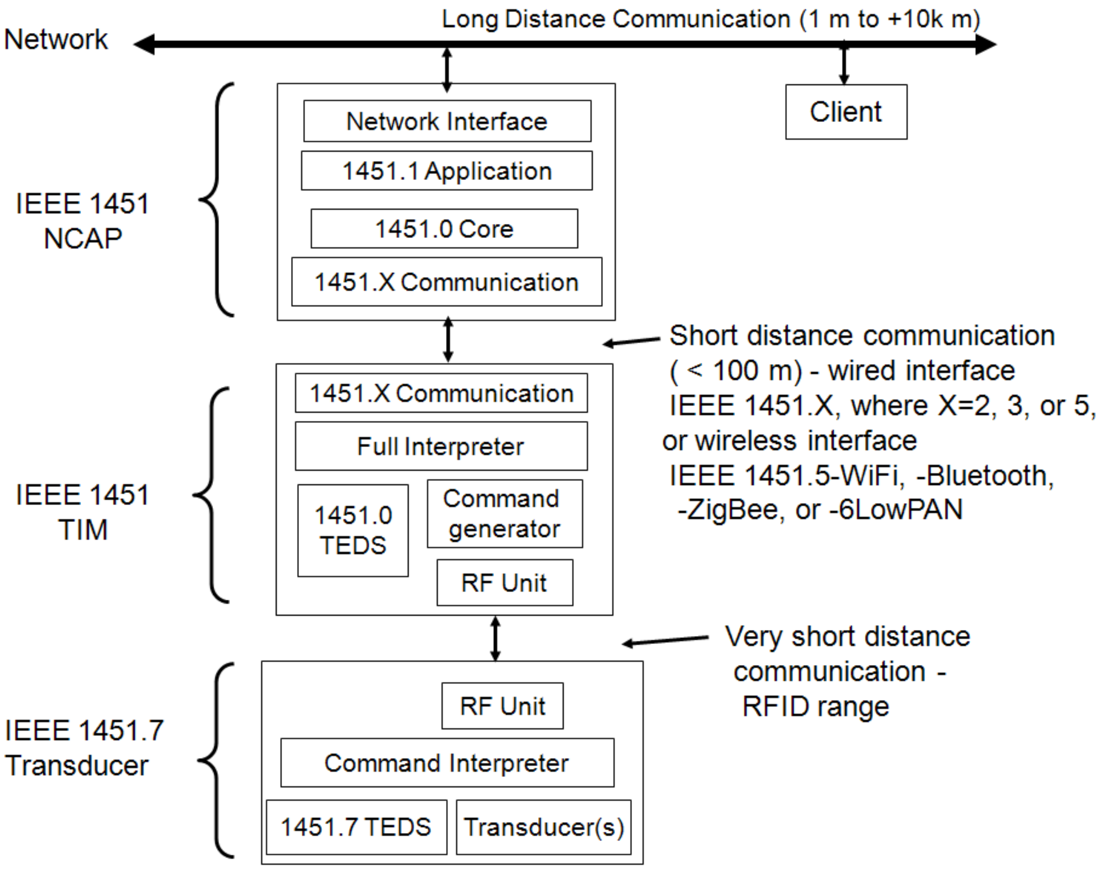

Network
Long Distance Communication (1 m to +10k m)
Client
IEEE 1451 NCAP
Network Interface
1451.1 Application
1451.0 Core
1451.X Communication
Short distance communication (< 100 m) - wired interface IEEE 1451.X, where X=2, 3, or 5, or wireless interface IEEE 1451.5-WiFi, -Bluetooth, -ZigBee, or -6LowPAN
IEEE 1451 TIM
1451.X Communication
Full Interpreter
1451.0 TEDS
Command generator
RF Unit
Very short distance communication - RFID range
IEEE 1451.7 Transducer
RF Unit
Command Interpreter
1451.7 TEDS
Transducer(s)

Figure D.2—Example of separate IEEE 1451 NCAP and TIM modules with an IEEE 1451.7 transducer in a network

Annex E

(informative)

Sensor authentication and encryption

E.1 Need for authentication and encryption

Sensors require two different types of security functions, namely authentication and encryption.

Authentication is needed to ensure authorized readers communicate only with authorized tags. Authentication of both sensors and readers are required. For example, on the one hand, if a reader attempts to make important changes to a sensor, such as erasing the log, the sensor would require the reader to prove that it is authorized to do so. On the other hand, when a reader extracts information from a sensor, it needs to ensure that the sensor information is authentic. From the previously mentioned examples, we see a clear need for two-way authentication.

Encryption is required because when sensor data are transmitted in the clear, a third party can gain unauthorized access to the information by either tapping the wire link or intercepting the radio link.

E.2 Use of a stream cipher for encryption

Sensor data may vary in size and are often transmitted over bandwidth-restricted channels. Therefore, stream ciphers are often preferred over block ciphers because block ciphers only encrypt information in fixed block sizes. For example, if a data packet is $N + 1$ bits long and the encryption block size is N bits, then two encryptions of N bits each and consequent transmission of $2N$ bits will be required. As a result, valuable bandwidth of $N - 1$ bits is wasted.

Refer to Figure E.1 for an example of using a stream cipher for encryption. Two identical random bit stream generators, one in the sender and one in the receiver, generate two identical, random, bit streams. When a stream of clear text information bits is XOR-ed with the cipher stream, the output is meaningless to observers on the public channel. However, on the receive side, the encrypted information is XOR-ed with the cipher stream for a second time. The second XOR recovers the original information. The information can only be recovered if the same shared secret and random number is used for both encryption and decryption. The random number is typically transmitted over the channel just before encryption starts to synchronize the two identical bit-stream generators.

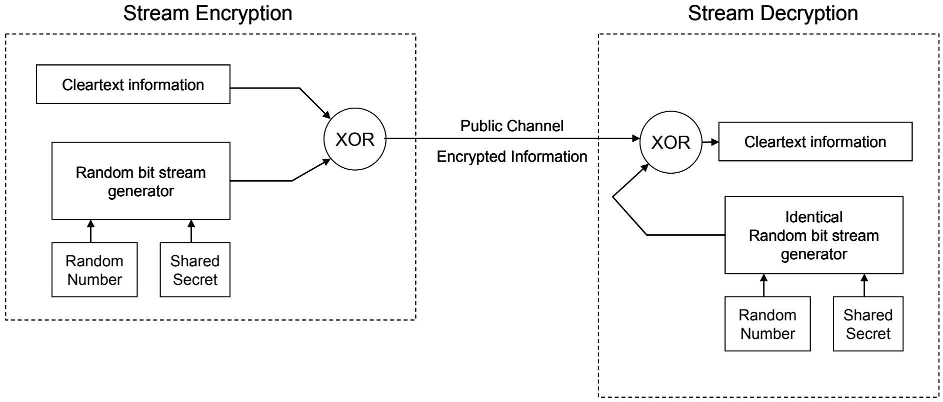


Figure E.1—Encryption and decryption in a stream cipher

E.3 Authentication using a stream cipher

Authentication is the process of confirming that a device is part of a group of trusted devices and can be performed by using a stream cipher to generate a reliable “security token” that proves the tested device has a correct secret code or “key.” At its root, all trusted devices contain a hidden, secret number, which is never revealed and which is known only to that group. A device proves itself to be an authentic member of that group by proving that it knows the secret number. The proof has to be performed without revealing the number itself.

Authentication using a stream cipher is described below. Refer to Figure E.2.

The challenging unit issues a “challenge,” which is simply the random number that initializes the random bit-stream generator.

The responding unit initializes its random bit-stream generator with the received random number and the shared secret. It then extracts the first *N* bits from the stream cipher and directly transmits them as a response. Note that the response bits are not XOR-ed in any way with any information.

The challenging unit also initializes its stream cipher with the random number and the shared secret. The challenger also calculates *N* bits. These *N* bits are then compared to the received response. If they are identical, the responding unit is considered to be authenticated.

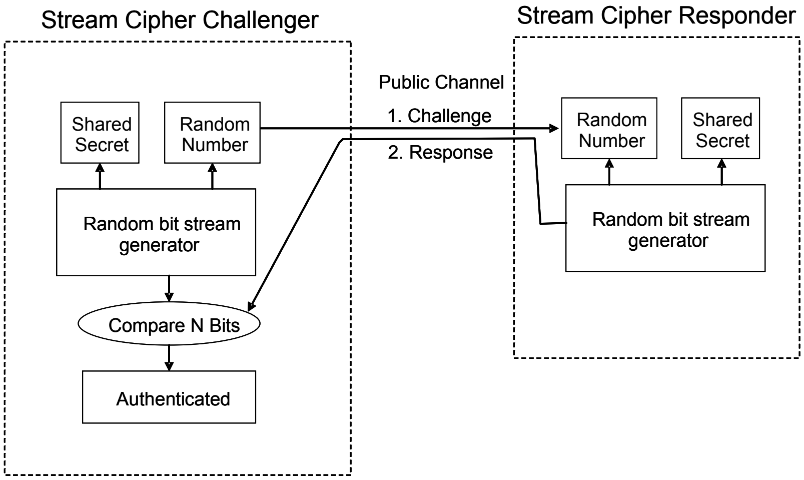


Figure E.2—Authentication using a stream cipher

E.4 Recommendations

The random number used to initialize the stream cipher should be at least 32 bits or larger. This is important to reduce the possibility of an attacker observing the channel and hoping that a previous challenge is reused. In that case, an attacker would know the valid response to the challenge because he has previously observed it on the channel.

Annex F

(informative)

Bibliography

[B1] IEEE Std 1451.0™-2007, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Functions, Communications Protocols and Transducer Electronic Data Sheet (TEDS) Formats.[5,6]

[B2] IEEE Std 1451.1™-1999, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Network Capable Application Processor (NCAP) Information Model.

[B3] 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.

[B4] 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.

[B5] 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.

[B6] IEEE Std 1451.5™-2007, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators Wireless Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats.

[B7] IEEE Std 1588™-2008, IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems.

[B8] ISO/IEC 15961-4 (in press), Information Technology—Radio Frequency Identification (RFID) for Item Management—Data Protocol—Part 4: Application Interface Commands for Battery Assist and Sensor Functionality.[7]

[B9] ISO/IEC 18000-2:2009, Information Technology—Radio Frequency Identification for Item Management—Part 2: Parameters for Air Interface Communications below 135 kHz.

[B10] ISO/IEC 18000-3:2009, Information Technology—Radio Frequency Identification for Item Management—Part 3: Parameters for Air Interface Communications at 13,56 MHz.

[B11] ISO/IEC 18000-4:2008, Information Technology—Radio Frequency Identification for Item Management—Part 4: Parameters for Air Interface Communications at 2,45 GHz.

[B12] ISO/IEC 18000-6:2006, Information Technology—Radio Frequency Identification for Item Management—Part 6: Parameters for air interface communications at 860 MHz to 960 MHz.

[B13] ISO/IEC 18000-7:2009, Information Technology—Radio Frequency Identification for Item Management—Part 7: Parameters for Active Air Interface Communications at 433 MHz.

[5] The IEEE standards or products referred to in this clause are trademarks owned by the Institute of Electrical and Electronics Engineers, Incorporated.

[6] IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards.ieee.org/).

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

[B14] ISO/IEC 24730-2:2006, Information Technology—Real-Time Locating Systems (RTLS)—Part 2: 2,4 GHz Air Interface Protocol.

[B15] ISO/IEC 24730-5:2000, Information Technology Automatic Identification and Data Capture Techniques—Real Time Locating Systems (RTLS)—Part 5: Chirp Spread Spectrum (CSS) at 2.4 GHz.

[B16] ISO/IEC 24753 (in press), Automatic Identification and Data Capture Techniques—Radio Frequency Identification (RFID) for Item Management—Application Protocol: Encoding and Processing Rules for Sensors and Batteries.