

BS EN 61158-5-19:2014



BSI Standards Publication

Industrial communication networks — Fieldbus specifications

Part 5-19: Application layer service definition — Type 19 elements

bsi.

...making excellence a habit.™

National foreword

This British Standard is the UK implementation of EN 61158-5-19:2014. It is identical to IEC 61158-5-19:2014. It supersedes BS EN 61158-5-19:2012 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee AMT/7, Industrial communications: process measurement and control, including fieldbus.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2014.
Published by BSI Standards Limited 2014

ISBN 978 0 580 79461 2
ICS 25.040.40; 35.100.70; 35.110

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 October 2014.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

EUROPEAN STANDARD

EN 61158-5-19

NORME EUROPÉENNE

EUROPÄISCHE NORM

October 2014

ICS 25.040.40; 35.100.70; 35.110

Supersedes EN 61158-5-19:2012

English Version

**Industrial communication networks - Fieldbus specifications -
Part 5-19: Application layer service definition - Type 19 elements
(IEC 61158-5-19:2014)**

Réseaux de communication industriels - Spécifications des
bus de terrain - Partie 5-19: Définition des services de la
couche application - Eléments de type 19
(CEI 61158-5-19:2014)

Industrielle Kommunikationsnetze - Feldbusse -
Teil 5-19: Dienstfestlegungen des Application Layer
(Anwendungsschicht) - Typ 19-Elemente
(IEC 61158-5-19:2014)

This European Standard was approved by CENELEC on 2014-09-22. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.



European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Foreword

The text of document 65C/763/FDIS, future edition 3 of IEC 61158-5-19, prepared by SC 65C "Industrial networks" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 61158-5-19:2014.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2015-06-22
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2017-09-22

This document supersedes EN 61158-5-19:2012.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association.

Endorsement notice

The text of the International Standard IEC 61158-5-19:2014 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

IEC 61131-1	NOTE	Harmonized as EN 61131-1.
IEC 61784-1	NOTE	Harmonized as EN 61784-1.
IEC 61784-2	NOTE	Harmonized as EN 61784-2.
IEC 61800-1	NOTE	Harmonized as EN 61800-1.
IEC 61800-7 Series	NOTE	Harmonized as EN 61800-7 Series (not modified).

Annex ZA (normative)

Normative references to international publications with their corresponding European publications

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

NOTE 1 When an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cenelec.eu

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IEC 61131-3	-	Programmable controllers - Part 3: Programming languages	EN 61131-3	-
IEC 61158-1	2014	Industrial communication networks - Fieldbus specifications - Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series	EN 61158-1	2014
IEC 61158-3-16	2007	Industrial communication networks - Fieldbus specifications - Part 3-16: Data-link layer service definition - Type 16 elements	EN 61158-3-16	2008
ISO/IEC 7498-1	-	Information technology - Open Systems Interconnection - Basic reference model: The basic model	-	-
ISO/IEC 8822	-	Information technology - Open Systems Interconnection - Presentation service definition	-	-
ISO/IEC 8824-1	-	Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation	-	-
ISO/IEC 9545	-	Information technology - Open Systems Interconnection - Application layer structure	-	-
ISO/IEC 10646	-	Information technology - Universal Coded Character Set (UCS)	-	-
ISO/IEC 10731	-	Information technology - Open Systems Interconnection - Basic Reference Model - Conventions for the definition of OSI services	-	-
ISO/IEC/IEEE 60559	-	Information technology - Microprocessor Systems - Floating-Point arithmetic	-	-

CONTENTS

INTRODUCTION.....	6
1 Scope.....	7
1.1 General.....	7
1.2 Specifications.....	8
1.3 Conformance.....	8
2 Normative references.....	8
3 Terms, definitions, abbreviations, symbols and conventions.....	9
3.1 ISO/IEC 7498-1 terms.....	9
3.2 ISO/IEC 8822 terms.....	9
3.3 ISO/IEC 9545 terms.....	9
3.4 ISO/IEC 8824-1 terms.....	9
3.5 Fieldbus application-layer specific definitions.....	10
3.6 Abbreviations and symbols.....	11
3.7 Conventions.....	12
4 Concepts.....	14
5 Data type ASE.....	14
5.1 Bitstring types.....	15
5.2 Unsigned types.....	16
5.3 Integer types.....	16
5.4 Floating Point types.....	17
5.5 Structure types.....	17
6 Communication model specification.....	18
6.1 Concepts.....	18
6.2 ASEs.....	18
6.3 ARs.....	30
6.4 Summary of AR classes.....	32
6.5 Permitted FAL services by AREP role.....	32
Bibliography.....	34
Table 1 – Read service parameters.....	20
Table 2 – Write service parameters.....	21
Table 3 – Read service parameters.....	22
Table 4 – Write service parameters.....	23
Table 5 – Notify service parameters.....	23
Table 6 – Get network status service parameters.....	25
Table 7 – Get device status service parameters.....	25
Table 8 – Network status change report service parameters.....	26
Table 9 – Station status change report service parameters.....	26
Table 10 – Set device status service parameters.....	27
Table 11 – Enable RTC service parameters.....	28
Table 12 – Enable hotplug service parameters.....	29
Table 13 – Notify RTC service parameters.....	29
Table 14 – Disable RTC service parameters.....	30

Table 15 – AREP (SVC) class summary..... 32

Table 16 – AREP (RTC-MS) class summary 32

Table 17 – AREP (RTC-CC) class summary..... 32

Table 18 – FAL services by AR type 33

INTRODUCTION

This part of IEC 61158 is one of a series produced to facilitate the interconnection of automation system components. It is related to other standards in the set as defined by the “three-layer” fieldbus reference model described in IEC 61158-1.

The application service is provided by the application protocol making use of the services available from the data-link or other immediately lower layer. This standard defines the application service characteristics that fieldbus applications and/or system management may exploit.

Throughout the set of fieldbus standards, the term “service” refers to the abstract capability provided by one layer of the OSI Basic Reference Model to the layer immediately above. Thus, the application layer service defined in this standard is a conceptual architectural service, independent of administrative and implementation divisions.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 5-19: Application layer service definition – Type 19 elements

1 Scope

1.1 General

The fieldbus application layer (FAL) provides user programs with a means to access the fieldbus communication environment. In this respect, the FAL can be viewed as a “window between corresponding application programs.”

This standard provides common elements for basic time-critical and non-time-critical messaging communications between application programs in an automation environment and material specific to Type 19 fieldbus. The term “time-critical” is used to represent the presence of a time-window, within which one or more specified actions are required to be completed with some defined level of certainty. Failure to complete specified actions within the time window risks failure of the applications requesting the actions, with attendant risk to equipment, plant and possibly human life.

This standard defines in an abstract way the externally visible service provided by the fieldbus application layer in terms of

- a) an abstract model for defining application resources (objects) capable of being manipulated by users via the use of the FAL service,
- b) the primitive actions and events of the service;
- c) the parameters associated with each primitive action and event, and the form which they take; and
- d) the interrelationship between these actions and events, and their valid sequences.

The purpose of this standard is to define the services provided to

- a) the FAL user at the boundary between the user and the application layer of the fieldbus reference model, and
- b) Systems Management at the boundary between the application layer and Systems Management of the fieldbus reference model.

This standard specifies the structure and services of the fieldbus application layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498-1) and the OSI application layer structure (ISO/IEC 9545).

FAL services and protocols are provided by FAL application-entities (AE) contained within the application processes. The FAL AE is composed of a set of object-oriented application service elements (ASEs) and a layer management entity (LME) that manages the AE. The ASEs provide communication services that operate on a set of related application process object (APO) classes. One of the FAL ASEs is a management ASE that provides a common set of services for the management of the instances of FAL classes.

Although these services specify, from the perspective of applications, how request and responses are issued and delivered, they do not include a specification of what the requesting and responding applications are to do with them. That is, the behavioral aspects of the applications are not specified; only a definition of what requests and responses they can

send/receive is specified. This permits greater flexibility to the FAL users in standardizing such object behavior. In addition to these services, some supporting services are also defined in this standard to provide access to the FAL to control certain aspects of its operation.

1.2 Specifications

The principal objective of this standard is to specify the characteristics of conceptual application layer services suitable for time-critical communications, and thus supplement the OSI Basic Reference Model in guiding the development of application layer protocols for time-critical communications.

A secondary objective is to provide migration paths from previously-existing industrial communications protocols. It is this latter objective which gives rise to the diversity of services standardized as the various Types of IEC 61158, and the corresponding protocols standardized in subparts of IEC 61158-6.

This specification may be used as the basis for formal application programming interfaces. Nevertheless, it is not a formal programming interface, and any such interface will need to address implementation issues not covered by this specification, including

- a) the sizes and octet ordering of various multi-octet service parameters, and
- b) the correlation of paired request and confirm, or indication and response, primitives.

1.3 Conformance

This standard does not specify individual implementations or products, nor does it constrain the implementations of application layer entities within industrial automation systems.

There is no conformance of equipment to this application layer service definition standard. Instead, conformance is achieved through implementation of conforming application layer protocols that fulfill the application layer services as defined in this standard.

2 Normative references

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

NOTE All parts of the IEC 61158 series, as well as IEC 61784-1 and IEC 61784-2 are maintained simultaneously. Cross-references to these documents within the text therefore refer to the editions as dated in this list of normative references.

IEC 61131-3, *Programmable controllers – Part 3: Programming languages*

IEC 61158-1:2014, *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-3-16:2007, *Industrial communication networks – Fieldbus specifications – Part 3-16: Data-link layer service definition – Type 16 elements*

ISO/IEC 7498-1, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

ISO/IEC 8822, *Information technology – Open Systems Interconnection – Presentation service definition*

ISO/IEC 8824-1, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*

ISO/IEC 9545, *Information technology – Open Systems Interconnection – Application Layer structure*

ISO/IEC 10646, *Information technology – Universal Coded Character Set (UCS)*

ISO/IEC 10731, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services*

ISO/IEC/IEEE 60559, *Information technology – Microprocessor Systems – Floating-Point arithmetic*

3 Terms, definitions, abbreviations, symbols and conventions

For the purposes of this document, the following terms, definitions, symbols, abbreviations and conventions as defined in these publications apply:

3.1 ISO/IEC 7498-1 terms

- a) application entity
- b) application process
- c) application protocol data unit
- d) application service element
- e) application entity invocation
- f) application process invocation
- g) application transaction
- h) real open system
- i) transfer syntax

3.2 ISO/IEC 8822 terms

- a) abstract syntax
- b) presentation context

3.3 ISO/IEC 9545 terms

- a) application-association
- b) application-context
- c) application context name
- d) application-entity-invocation
- e) application-entity-type
- f) application-process-invocation
- g) application-process-type
- h) application-service-element
- i) application control service element

3.4 ISO/IEC 8824-1 terms

- a) object identifier
- b) type

3.5 Fieldbus application-layer specific definitions

3.5.1

coded character set

code

set of unambiguous rules that establish a character set and one-to-one relationship between the characters of the set and their representation by one or more bit combinations

3.5.2

cross communication

direct data transfer between slave devices (without active involvement of master)

3.5.3

cycle time

duration of a communication cycle

3.5.4

cyclic data

part of a telegram, which does not change its meaning during cyclic operation of the network

3.5.5

device

slave in the communication network

Note 1 to entry: Examples are a power drive system as defined in the IEC 61800 standard family, I/O stations as defined in the IEC 61131 standard family, etc.

3.5.6

device status

four adjacent octets inside the acknowledge telegram containing status information for each device

3.5.7

element

part of IDNs

Note 1 to entry: Each IDN has 7 elements, whereas each one has a specific meaning (e.g., number, name, data).

3.5.8

hot plug

possibility to open the communication network and insert or remove slaves while the network is still in real-time operation

3.5.9

identification number

designation of operating data under which a data block is preserved with its attribute, name, unit, minimum and maximum input values, and the data

3.5.10

loopback

mode by which a device passes on a received telegram to the same port and to the other port, either changed or unchanged

3.5.11

master

node, which assigns the other nodes (i.e., slaves) the right to transmit

3.5.12**physical layer**

first layer of the ISO-OSI reference model

3.5.13**protocol**

convention about the data formats, time sequences, and error correction in the data exchange of communication systems

3.5.14**service channel****SVC**

non real-time transmission of information upon master request during RT channel

3.5.15**slave**

node, which is assigned the right to transmit by the master

3.5.16**station**

node

3.5.17**topology**

physical network architecture with respect to the connection between the stations of the communication system

3.6 Abbreviations and symbols

AHS	Service transport handshake of the device (acknowledge HS)
AP	Application Process
APO	Application Object
AR	Application Relationship
AREP	Application Relationship End Point
ASE	Application Service Element
CC-data	Cross Communication
Cnf	Confirmation
DA	Destination address
DAT	Duration of acknowledge telegram
FAL	Fieldbus Application Layer
ID	Identification Number
IDN	Identification Number
Ind	Indication
MS	Master Slave
NRC	Non Real Time Channel
NA	Not applicable
Req	Request
Rsp	Response
RTC	Real Time Channel
RTE	Real Time Ethernet

3.7 Conventions

3.7.1 Overview

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of two parts, its class specification, and its service specification.

The class specification defines the attributes of the class. The attributes are accessible from instances of the class using the Object Management ASE services specified in Clause 5 of this standard. The service specification defines the services that are provided by the ASE.

3.7.2 General conventions

This standard uses the descriptive conventions given in ISO/IEC 10731.

3.7.3 Conventions for class definitions

Class definitions are described using templates. Each template consists of a list of attributes for the class. The general form of the template is shown below:

FAL ASE:		ASE Name
CLASS:		Class name
CLASS ID:		#
PARENT CLASS:		Parent class name
ATTRIBUTES:		
1	(o)	Key Attribute: numeric identifier
2	(o)	Key Attribute: name
3	(m)	Attribute: attribute name(values)
4	(m)	Attribute: attribute name(values)
4.1	(s)	Attribute: attribute name(values)
4.2	(s)	Attribute: attribute name(values)
4.3	(s)	Attribute: attribute name(values)
5.	(c)	Constraint: constraint expression
5.1	(m)	Attribute: attribute name(values)
5.2	(o)	Attribute: attribute name(values)
6	(m)	Attribute: attribute name(values)
6.1	(s)	Attribute: attribute name(values)
6.2	(s)	Attribute: attribute name(values)
SERVICES:		
1	(o)	OpsService: service name
2.	(c)	Constraint: constraint expression
2.1	(o)	OpsService: service name
3	(m)	MgtService: service name

- (1) The "FAL ASE:" entry is the name of the FAL ASE that provides the services for the class being specified.
- (2) The "CLASS:" entry is the name of the class being specified. All objects defined using this template will be an instance of this class. The class may be specified by this standard, or by a user of this standard.
- (3) The "CLASS ID:" entry is a number that identifies the class being specified. This number is unique within the FAL ASE that will provide the services for this class. When qualified by the identity of its FAL ASE, it unambiguously identifies the class within the scope of the FAL. The value "NULL" indicates that the class cannot be instantiated. Class IDs between 1 and 255 are reserved by this standard to identify standardized classes. They have been

assigned to maintain compatibility with existing national standards. CLASS IDs between 256 and 2048 are allocated for identifying user defined classes.

- (4) The "PARENT CLASS:" entry is the name of the parent class for the class being specified. All attributes defined for the parent class and inherited by it are inherited for the class being defined, and therefore do not have to be redefined in the template for this class.

NOTE The parent-class "TOP" indicates that the class being defined is an initial class definition. The parent class TOP is used as a starting point from which all other classes are defined. The use of TOP is reserved for classes defined by this standard.

- (5) The "ATTRIBUTES" label indicate that the following entries are attributes defined for the class.

- a) Each of the attribute entries contains a line number in column 1, a mandatory (m) / optional (o) / conditional (c) / selector (s) indicator in column 2, an attribute type label in column 3, a name or a conditional expression in column 4, and optionally a list of enumerated values in column 5. In the column following the list of values, the default value for the attribute may be specified.
- b) Objects are normally identified by a numeric identifier or by an object name, or by both. In the class templates, these key attributes are defined under the key attribute.
- c) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting is used to specify
 - i) fields of a structured attribute (4.1, 4.2, 4.3),
 - ii) attributes conditional on a constraint statement (Clause 5). Attributes may be mandatory (5.1) or optional (5.2) if the constraint is true. Not all optional attributes require constraint statements as does the attribute defined in (5.2).
 - iii) the selection fields of a choice type attribute (6.1 and 6.2).

- (6) The "SERVICES" label indicates that the following entries are services defined for the class.

- a) An (m) in column 2 indicates that the service is mandatory for the class, while an (o) indicates that it is optional. A (c) in this column indicates that the service is conditional. When all services defined for a class are defined as optional, at least one has to be selected when an instance of the class is defined.
- b) The label "OpsService" designates an operational service (1).
- c) The label "MgtService" designates an management service (2).
- d) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting within the list of services is used to specify services conditional on a constraint statement.

3.7.4 Conventions for service definitions

3.7.4.1 General

The service model, service primitives, and time-sequence diagrams used are entirely abstract descriptions; they do not represent a specification for implementation.

3.7.4.2 Service parameters

Service primitives are used to represent service user/service provider interactions (ISO/IEC 10731). They convey parameters which indicate information available in the user/provider interaction. In any particular interface, not all parameters need be explicitly stated.

The service specifications of this standard uses a tabular format to describe the component parameters of the ASE service primitives. The parameters which apply to each group of service primitives are set out in tables. Each table consists of up to five columns for the

- 1) Parameter name,

- 2) request primitive,
- 3) indication primitive,
- 4) response primitive, and
- 5) confirm primitive.

One parameter (or component of it) is listed in each row of each table. Under the appropriate service primitive columns, a code is used to specify the type of usage of the parameter on the primitive specified in the column:

- M parameter is mandatory for the primitive
- U parameter is a User option, and may or may not be provided depending on dynamic usage of the service user. When not provided, a default value for the parameter is assumed.
- C parameter is conditional upon other parameters or upon the environment of the service user.
- (blank) parameter is never present.
- S parameter is a selected item.

Some entries are further qualified by items in brackets. These may be

- a) a parameter-specific constraint:
 - “(=)” indicates that the parameter is semantically equivalent to the parameter in the service primitive to its immediate left in the table.
- b) an indication that some note applies to the entry:
 - “(n)” indicates that the following note "n" contains additional information pertaining to the parameter and its use.

3.7.4.3 Service procedures

The procedures are defined in terms of

- the interactions between application entities through the exchange of fieldbus Application Protocol Data Units, and
- the interactions between an application layer service provider and an application layer service user in the same system through the invocation of application layer service primitives.

These procedures are applicable to instances of communication between systems which support time-constrained communications services within the fieldbus application layer.

4 Concepts

The common concepts and templates used to describe the application layer service in this standard are detailed in IEC 61158-1, Clause 9.

5 Data type ASE

Data types as specified in IEC 61158-1, Clause 9 is applied with the following restrictions:

Only nesting level of 1 is supported.

Only the following basic data types are supported:

- BitString8
- BitString16
- BitString32

BitString64
 Unsigned16
 Unsigned32
 Unsigned64
 Integer16
 Integer32
 Integer64
 VisibleString1
 Float32
 Float64

5.1 Bitstring types

5.1.1 BitString8

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 22
2	Data type Name = Bitstring8
3	Format = FIXED LENGTH
5.1	Octet Length = 1

This type contains 1 element of type BitString.

5.1.2 BitString16

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 23
2	Data type Name = Bitstring16
3	Format = FIXED LENGTH
5.1	Octet Length = 2

This type is a BitString16 and has a length of two octets.

5.1.3 BitString32

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 24
2	Data type Name = Bitstring32
3	Format = FIXED LENGTH
5.1	Octet Length = 4

This type is a BitString16 and has a length of four octets.

5.1.4 BitString64

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 57
2	Data type Name = Bitstring64
3	Format = FIXED LENGTH
5.1	Octet Length = 8

This type is a BitString16 and has a length of eight octets.

5.2 Unsigned types

5.2.1 Unsigned16

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 6
2	Data type Name = Unsigned16
3	Format = FIXED LENGTH
4.1	Octet Length = 2

This type is a binary number. The most significant bit of the most significant octet is always used as the most significant bit of the binary number; no sign bit is included. This unsigned type has a length of two octets.

5.2.2 Unsigned32

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 7
2	Data type Name = Unsigned32
3	Format = FIXED LENGTH
4.1	Octet Length = 4

This type is a binary number. The most significant bit of the most significant octet is always used as the most significant bit of the binary number; no sign bit is included. This unsigned type has a length of four octets.

5.2.3 Unsigned64

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 56
2	Data type Name = Unsigned64
3	Format = FIXED LENGTH
4.1	Octet Length = 8

This type is a binary number. The most significant bit of the most significant octet is always used as the most significant bit of the binary number; no sign bit is included. This unsigned type has a length of eight octets.

5.3 Integer types

5.3.1 Integer16

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 3
2	Data type Name = Integer16
3	Format = FIXED LENGTH
4.1	Octet Length = 2

This integer type is a two's complement binary number with a length of two octets.

5.3.2 Integer32

CLASS:	Data type
ATTRIBUTES:	
1	Data type Numeric Identifier = 4
2	Data type Name = Integer32
3	Format = FIXED LENGTH

4.1 Octet Length = 4

This integer type is a two's complement binary number with a length of four octets.

5.3.3 Integer64

CLASS: Data type

ATTRIBUTES:

1	Data type Numeric Identifier	=	55
2	Data type Name	=	Integer64
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	8

This integer type is a two's complement binary number with a length of eight octets.

5.4 Floating Point types

5.4.1 Float32

CLASS: Data type

ATTRIBUTES:

1	Data type Numeric Identifier	=	8
2	Data type Name	=	Float32
4	Format	=	FIXED LENGTH
4.1	Octet Length	=	4

This type has a length of four octets. The format for Float32 is that defined by ISO/IEC/IEEE 60559 as single precision.

5.4.2 Float64

CLASS: Data type

ATTRIBUTES:

1	Data type Numeric Identifier	=	15
2	Data type Name	=	Float64
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	8

This type has a length of eight octets. The format for Float64 is that defined by ISO/IEC/IEEE 60559 as double precision.

5.5 Structure types

5.5.1 STRING2

CLASS: Data type

ATTRIBUTES:

1	Data type Numeric Identifier	=	not used
2	Data type Name	=	STRING2
3	Format	=	STRUCTURE
5.1	Number of Fields	=	2
5.2.1	Field Name	=	Charcount_Element
5.2.2	Field Data type	=	UINT
5.3.1	Field Name	=	String2contents_Element
5.3.2	Field Data type	=	OctetString

This IEC 61131-3 data type extension is composed of two elements. Charcount_Element gives the current number of characters in the String2contents_Element (one UINT per character). Characters are as specified in ISO/IEC 10646.

6 Communication model specification

6.1 Concepts

6.1.1 Communication mechanisms

Two communication mechanisms are supported by devices within the network:

- cyclic transmission of data in a high efficient manner using a publisher subscriber model,
- non-cyclic transmission of data using a client server communication model.

AREPs, which act as push publisher or push subscriber, are used for cyclic transmission.

AREPs, which act as a client or server, are used for non-cyclic data transmission.

6.1.2 IDN concept

The application data which is transmitted cyclically and non-cyclically between FAL users is mapped on so-called identification numbers (IDNs). These IDNs correspond to the APOs as defined and are described in Annex A of IEC 61158-3-16:2007.

6.2 ASEs

6.2.1 Identification number (IDN) ASE

6.2.1.1 Overview

The IDN ASE provides read and write access to the attributes of IDNs provided by a device.

6.2.1.2 IDN class specification

6.2.1.2.1 Formal model

FAL ASE:		IDN ASE
CLASS:		IDN
CLASS ID:		not used
PARENT CLASS:		TOP
ATTRIBUTES:		
1	(m)	Key Attribute: Identification
1.1	(m)	Key Attribute: Identification number
1.2	(o)	Key Attribute: Structure element
1.3	(o)	Key Attribute: Structure index
2	(o)	Attribute: Name
3	(m)	Attribute: Data Attribute
4	(o)	Attribute: Unit
5	(o)	Attribute: Minimum value
6	(o)	Attribute: Maximum value
7	(m)	Attribute: Operation Data
SERVICES:		
1	(m)	OpsService: Read
2	(m)	OpsService: Write

6.2.1.2.2 Attributes

Identification

This key attribute identifies an instance of this object class. The addressing can be simple or extended.

Identification number

This mandatory attribute is a numerical identifier to address an instance of this object class. In case of a simple addressing scheme, this is the only attribute required for identification.

Structure element

This optional attribute is used in case of an extended addressing.

Structure index

This optional attribute is used in case of an extended addressing.

Name

This optional attribute specifies a symbolic name of this object class.

Data attribute

This mandatory attribute specifies all information which is needed to display or convert the data intelligibly. This includes data type, data length, conversion factor, read/write permission depending on the communication phase and whether the data is associated to a command.

Unit

This optional attribute specifies the unit of the operation data.

Minimum value

This optional attribute specifies the minimum input value for the operation data.

Maximum value

This optional attribute specifies the maximum input value for the operation data.

Operation data

This mandatory attribute specifies the operation data of this instance of the object class.

6.2.1.3 IDN ASE service specification

6.2.1.3.1 Supported services

Subclause 6.2.1.3 specifies the definition of the services that are unique to this ASE. The services defined for this ASE are:

- Read
- Write

6.2.1.3.2 Read service

6.2.1.3.2.1 Service overview

This confirmed service is used to read an element of an IDN on demand.

6.2.1.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 1.

Table 1 – Read service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Device Address	M	M (=)		
Identification	M	M (=)		
Attribute	M	M (=)		
Result(+)			S	S (=)
Value			M	M (=)
Result(-)			S	S (=)
Error Info			M	M (=)

NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.

AREP

This parameter is the local identifier for the desired AR.

Device address

This parameter identifies the address of the device.

Identification

This parameter specifies an IDN object to be read by the key attribute.

Attribute

This parameter specifies the element of an IDN object to be read by the key attribute.

Value

This parameter specifies the value read.

Error info

This parameter provides error information for service errors.

6.2.1.3.3 Write service**6.2.1.3.3.1 Service overview**

This confirmed service is used to write an element of an IDN.

6.2.1.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 2.

Table 2 – Write service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Device Address	M	M (=)		
Identification	M	M (=)		
Attribute	M	M (=)		
Value	M	M (=)		
Result(+)			S	S (=)
Result(-)			S	S (=)
Error Info			M	M (=)

NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.

AREP

This parameter is the local identifier for the desired AR.

Device address

This parameter identifies the address of the device.

Identification

This parameter specifies an IDN object to be written by the key attribute.

Attribute

This parameter specifies the element of an IDN object to be written by the key attribute.

Value

This parameter specifies the value to be written.

Error info

This parameter provides error information for service errors.

6.2.2 CYCIDN ASE**6.2.2.1 Overview**

The CYCIDN ASE provides cyclic read and write access to the operation data of IDNs provided by a device.

6.2.2.2 Cyclic Identification Number (CYCIDN) class specification**6.2.2.2.1 Formal model**

FAL ASE:		CYCIDN ASE
CLASS:		CYCIDN
CLASS ID:		not used
PARENT CLASS:		TOP
ATTRIBUTES:		
1	(m)	Key Attribute: Identification
1.1	(m)	Key Attribute: Identification Number
1.2	(o)	Key Attribute: Resource Element
1.3	(o)	Key Attribute: Subindex
2	(m)	Attribute: Data
SERVICES:		
1	(m)	OpsService: Read
2	(o)	OpsService: Write

3 (m) OpsService: Notify

6.2.2.2.2 Attributes

AREP

This parameter is the local identifier for the desired AR.

Identification

This parameter specifies an IDN object to be read by the key attribute.

Data

This attribute specifies the data which is read or written.

6.2.2.3 CYCIDN ASE service specification

6.2.2.3.1 Supported services

Subclause 6.2.2.3 specifies the definition of services that are unique to this ASE. The services defined for this ASE are:

- Read
- Write
- Notify

6.2.2.3.2 Read service

6.2.2.3.2.1 Service overview

This unconfirmed service is used to cyclically read operation data of an IDN.

6.2.2.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 3.

Table 3 – Read service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M			
Device Address	M			
Identification	M			
Operation Data	M			
Error Info	M			

AREP

This parameter is the local identifier for the desired AR.

Device address

This parameter identifies the address of the device.

Identification

This parameter specifies an IDN object to be read by the key attribute.

Operation data

This parameter specifies the operation data which is being read.

Error info

This parameter provides error information for service errors.

6.2.2.3.3 Write service**6.2.2.3.3.1 Service overview**

This confirmed service is used to write cyclically an element of an IDN.

6.2.2.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 4.

Table 4 – Write service parameters

Parameter name	Req	Ind	Cnf
Argument			
AREP	M		
Device Address	M		
Identification	M		
Operation Data	M		
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.			

AREP

This parameter is the local identifier for the desired AR.

Device address

This parameter identifies the address of the device.

Identification

This parameter specifies an IDN object to be written by the key attribute.

Operation data

This parameter specifies the operation data which is being written.

Error info

This parameter provides error information for service errors.

6.2.2.3.4 Notify service**6.2.2.3.4.1 Service overview**

This service is used to notify that new cyclic data has been received. This service may be used by an application to trigger the cyclic reading and writing of data depending on the device implementation.

6.2.2.3.4.2 Service primitives

The service parameters for each primitive are shown in Table 5.

Table 5 – Notify service parameters

Parameter name	Ind
Argument	
AREP	M

AREP

This parameter is the local identifier for the desired AR.

6.2.3 Management (MGT) ASE

6.2.3.1 Overview

The MGT ASE provides management services for a Type 19 network and Type 19 devices.

6.2.3.2 MGT class specification

6.2.3.2.1 Formal model

FAL ASE:		MGT ASE
CLASS:		MGT ASE
CLASS ID:		not used
PARENT CLASS:		TOP
ATTRIBUTES:		
1	(m)	Attribute: Topology
SERVICES:		
1	(m)	OpsService: Get Network Status
2	(m)	OpsService: Get Device Status
3	(m)	OpsService: Network Status Change Report
4	(m)	OpsService: Device Status Change Report
5	(m)	OpsService: Set Device Status
6	(m)	OpsService: Enable RTC
7	(m)	OpsService: Notify RTC
8	(m)	OpsService: Disable RTC

6.2.3.2.2 Attributes

6.2.3.2.2.1 Topology

This attribute specifies the current topology of the network.

6.2.3.3 MGT ASE service specification

6.2.3.3.1 Supported services

Subclause 6.2.3.3 specifies the definition of the services that are unique to this ASE.

The services defined for this ASE are:

- Get Network Status
- Get Device Status
- Network Status Change Report
- Device Status Change Report
- Set Device Status
- Enable RTC
- Enable Hotplug
- Notify RTC
- Disable RTC

6.2.3.3.2 Get network status service

6.2.3.3.2.1 Service overview

This local service is used to obtain the status of the network.

6.2.3.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 6.

Table 6 – Get network status service parameters

Parameter name	Req	Cnf
Argument AREP	M	
Result Network Status		M M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

AREP

This parameter is the local identifier for the desired AR.

Network status

This parameter indicates consistency of the primary and secondary networks. Possible values are:

- RING
- LINE
- ERROR

6.2.3.3.3 Get device status service**6.2.3.3.3.1 Service overview**

This local service is used to obtain the status of the specified device.

6.2.3.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 7.

Table 7 – Get device status service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument AREP Device Identifier	M M			
Result Device Status				M M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.				

AREP

This parameter is the local identifier for the desired AR.

Device identifier

This parameter defines the device for which the status is requested.

Device status

This parameter indicates the status of the device specified by the device identifier, and includes the following information.

- a) existence

TRUE	the device exists
FALSE	the device does not exist

b) mode

LB	the device is in loopback mode
FF	the device is in fast forward mode

6.2.3.3.4 Network status change report service

6.2.3.3.4.1 Service overview

This local service is used to inform of changes in network status.

6.2.3.3.4.2 Service primitives

The service parameters for each primitive are shown in Table 8.

Table 8 – Network status change report service parameters

Parameter name	Ind
Argument	
AREP	M
Network Status	M

AREP

This parameter is the local identifier for the desired AR.

Network status

This parameter indicates the current status of the network, and has the following possible values:

- RING
- LINE
- ERROR

6.2.3.3.5 Device status change report service

6.2.3.3.5.1 Service overview

This local service is used to inform of changes in the device status.

6.2.3.3.5.2 Service primitives

The service parameters for each primitive are shown in Table 9.

Table 9 – Station status change report service parameters

Parameter name	Ind
Argument	
AREP	M
Device Identifier	M
Device Status	M

AREP

This parameter is the local identifier for the desired AR.

Device identifier

This parameter indicates the status of the device, of which the status has been changed.

Device status

This parameter indicates the status of the device specified by the request primitive and includes the following information.

a) existence

TRUE	the device exists
FALSE	the device does not exist

b) mode

LB	the device is in loopback mode
FF	the device is in fast forward mode

6.2.3.3.6 Set device status service**6.2.3.3.6.1 Service overview**

This local service is used by the master to set the status of the specified device.

6.2.3.3.6.2 Service primitives

The service parameters for each primitive are shown in Table 10.

Table 10 – Set device status service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
DeviceIdentifier	M	M (=)		
DeviceStatus	M	M (=)		

AREP

This parameter is the local identifier for the desired AR.

Device identifier

This parameter indicates the remote device for which the status is to be set.

Device status

This parameter specifies the mode of the device specified by the device identifier, and includes the following information:

LB	the device is in loopback mode
FF	the device is in fast forward mode

6.2.3.3.7 Enable RTC**6.2.3.3.7.1 Service overview**

This service is used by the master to enable the cyclic communication (RTC) in the network.

6.2.3.3.7.2 Service primitives

The service parameters for each primitive are shown in Table 11.

Table 11 – Enable RTC service parameters

Parameter name	Req	Cnf
Argument		
AREP	M	
Cycle Time	M	
List of Device Addresses	M	
List of incoming connections for each device	M	
List of outgoing connections for each device	M	
Topology	O	
Result(+)		S
List of Present Device Addresses		M
Result(-)		S
Error Info		M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

AREP

This parameter is the local identifier for the desired AR.

Cycle time

This parameter defines the cycle time for a network.

List of device addresses

This parameter specifies a list of devices that are part of a network.

List of incoming connections for each device

This parameter specifies a list of incoming connections for each device that is part of a network.

List of outgoing connections for each device

This parameter specifies a list of outgoing connections for each device that is part of a network.

Topology

This parameter defines the expected topology of a network.

List of present device addresses

This parameter specifies a list of present devices in a network.

Error info

This parameter provides error information for service errors.

6.2.3.3.8 Enable hotplug**6.2.3.3.8.1 Service overview**

This service is used by the master to enable the hotplug capability in a network.

6.2.3.3.8.2 Service primitives

The service parameters for each primitive are shown in Table 12.

Table 12 – Enable hotplug service parameters

Parameter name	Req	Cnf
Argument		
AREP	M	
List of Device Addresses	M	
Result(+)		S
Result(-)		S
Error Info		M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

AREP

This parameter is the local identifier for the desired AR.

List of device addresses

This parameter specifies a list of devices that are to be hot plugged to a network.

Error info

This parameter provides error information for service errors.

6.2.3.3.9 Notify RTC**6.2.3.3.9.1 Service overview**

This service is used to notify a slave device that the network has been initialized.

6.2.3.3.9.2 Service primitives

The service parameters for each primitive are shown in Table 13.

Table 13 – Notify RTC service parameters

Parameter name	Ind
Argument	
AREP	M
Cycle Times	M
List of incoming connections	M
List of outgoing connections	M
Topology	O

AREP

This parameter is the local identifier for the desired AR.

Cycle time

This parameter defines the cycle time for a network which is set by the master device.

List of incoming connections

This parameter specifies a list of incoming connections which are configured by the master device.

List of outgoing connections

This parameter specifies a list of outgoing connections which are configured by the master device.

Topology

This optional parameter defines the expected topology as set by the master device.

6.2.3.3.10 Disable RTC

6.2.3.3.10.1 Service overview

This service is used by the master to disable the cyclic communication (RTC) in a network.

6.2.3.3.10.2 Service primitives

The service parameters for each primitive are shown in Table 14.

Table 14 – Disable RTC service parameters

Parameter name	Req	Cnf
Argument AREP	M	
Result(+)		S
Result(-) Error Info		S M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

AREP

This parameter is the local identifier for the desired AR.

Error Info

This parameter provides error information for service errors.

6.3 ARs

6.3.1 General

Application Relationships provided are

- point-to-point user-Triggered Confirmed client/server AREP (SVC-AR);
- point-to-point network-Scheduled Unconfirmed publisher/subscriber AREP (RTC-MS-AR);
- multipoint network-Scheduled Unconfirmed publisher/subscriber AREP (RTC-CC-AR).

6.3.2 Point-to-point user-triggered confirmed client/server AREP (SVC)

This class is defined to support the on-demand exchange of confirmed services between a master and a slave application process. Unconfirmed services are not supported by this type of AR. The behavior of this class is described as follows. An IDN ASE user wishing to convey a request APDU submits it as an IDN ASE Service Data Unit to its AREP and the AREP sending the request APDU queues it to its underlying layer for transfer at the next available opportunity.

The AREP receiving the request APDU from its underlying layer queues it for delivery to its IDN ASE user in the order in which it was received.

The AREP receiving the request APDU accepts the corresponding response APDU from its IDN ASE user and queues it to the underlying layer for transfer.

The AREP that issued the request APDU receives the response APDU from its underlying layer and queues it for delivery to its IDN ASE user in the order in which it was received. It also stops its associated service response timer.

6.3.3 Point-to-point network-scheduled unconfirmed publisher/subscriber AREP (RTC-MS)

This class is defined to support the “push” model for scheduled unbuffered distribution of unconfirmed services between a master and a slave (MS) application process.

The behavior of this type of AR can be described as follows.

An AR ASE user wishing to convey a request APDU submits it as an AR ASE Service Data Unit to its AREP for distribution. Sending AREP writes the APDU into the internal buffer, completely replacing the existing contents of the buffer. The AR ASE transfers the buffer contents at the next scheduled transfer opportunity.

If the AREP receives another APDU before the buffer contents are transmitted, the buffer contents will be replaced with the new APDU, and the previous APDU will be lost. When the buffer contents are transmitted, the AR ASE notifies the user of transmission.

At the receiving endpoint, the APDU is received from the network and is written immediately into the buffer, completely overwriting the existing contents of the buffer. The endpoint notifies the user that the APDU has arrived and delivers it to the user according to the local user interface. If the APDU has not been delivered before the next APDU arrives, it will be overwritten by the next APDU and lost.

An FAL user receiving the buffered transmission may request to receive the currently buffered APDU later.

6.3.4 Point-to-multipoint network-scheduled unconfirmed publisher/subscriber AREP (RTC-CC)

This class is defined to support the “push” model for scheduled and buffered distribution of unconfirmed services to one or more application processes. The services provided by this AREP may be used for a direct communication between devices.

The behaviour of this type of AR can be described as follows.

An AR ASE user wishing to convey a request APDU submits it as an AR ASE Service Data Unit to its AREP for distribution. The sending AREP writes the APDU into the internal buffer, completely replacing the existing contents of the buffer.

The AREP transfers the buffer contents at the next scheduled transfer opportunity.

If the AREP receives another APDU before the buffer contents are transmitted, the buffer contents will be replaced with the new APDU, and the previous APDU will be lost. When the buffer contents are transmitted, the AR ASE notifies the user of transmission.

At the receiving endpoint, the APDU is received from the network and is written immediately into the buffer, completely overwriting the existing contents of the buffer. The endpoint notifies the user that the APDU has arrived and delivers it to the user according to the local user interface. If the APDU has not been delivered before the next APDU arrives, it will be overwritten by the next APDU and lost.

An FAL user receiving the buffered transmission may request to receive the currently buffered APDU later.

6.4 Summary of AR classes

Table 15 defines the characteristics of point-to-point user-triggered confirmed client/server AREP (SVC) class. The Class ID column have been assigned to be compatible with existing standards.

Table 15 – AREP (SVC) class summary

FAL ASE	Class	Class ID
Roles	Client	NA
	Server	NA
Cardinality	one-to-one	NA
Conveyance paths	Bi-directional	NA
Trigger policy	User-triggered	NA
Conveyance policy	Queued	NA

Table 16 defines the characteristics of point-to-point network-scheduled unconfirmed publisher/subscriber AREP (RTC-MS) class. The Class ID column have been assigned to be compatible with existing standards.

Table 16 – AREP (RTC-MS) class summary

FAL ASE	Class	Class ID
Roles	Publisher	NA
	Subscriber	NA
Cardinality	One-to-one	NA
Conveyance paths	Unidirectional	NA
Trigger policy	Network-scheduled	NA
Conveyance policy	Buffered	NA

Table 17 defines the characteristics of point-to-point network-scheduled unconfirmed publisher/subscriber AREP (RTC-CC) class. The Class ID column have been assigned to be compatible with existing standards.

Table 17 – AREP (RTC-CC) class summary

FAL ASE	Class	Class ID
Roles	Publisher	NA
	Subscriber	NA
Cardinality	One-to-many	NA
Conveyance paths	Unidirectional	NA
Trigger policy	Network-scheduled	NA
Conveyance policy	Non buffered	NA

6.5 Permitted FAL services by AREP role

Table 18 below defines the valid combinations of services and AR types (which service APDUs) and can be sent or received by AR with the specified type. “Unc” and “Cnf” columns indicate whether the service listed in the left-hand column is unconfirmed or confirmed respectively.

Table 18 – FAL services by AR type

FAL Services	Used AREPs		Client		Server		Publisher	Subscriber
	SVC	RTC-MS	req	cnf	ind	res	req	req
IDN ASE								
Read	X		X	X	X	X		
Write	X		X	X	X	X		
CYCIDN ASE								
Read		X						X
Write		X					X	

Bibliography

IEC 61131-1, *Programmable controllers – Part 1: General information*

IEC 61784-1, *Industrial communication networks – Profiles – Part 1: Fieldbus profiles*

IEC 61784-2, *Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3*

IEC 61800-1, *Adjustable speed electrical power drive systems – Part 1: General requirements – Rating specifications for low voltage adjustable speed d.c. power drive systems*

IEC 61800-7 (all parts), *Adjustable speed electrical power drive systems – Generic interface and use of profiles for power drive systems*

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards-based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Copyright

All the data, software and documentation set out in all British Standards and other BSI publications are the property of and copyrighted by BSI, or some person or entity that owns copyright in the information used (such as the international standardization bodies) and has formally licensed such information to BSI for commercial publication and use. Except as permitted under the Copyright, Designs and Patents Act 1988 no extract may be reproduced, stored in a retrieval system or transmitted in any form or by any means – electronic, photocopying, recording or otherwise – without prior written permission from BSI. Details and advice can be obtained from the Copyright & Licensing Department.

Useful Contacts:

Customer Services

Tel: +44 845 086 9001

Email (orders): orders@bsigroup.com

Email (enquiries): cservices@bsigroup.com

Subscriptions

Tel: +44 845 086 9001

Email: subscriptions@bsigroup.com

Knowledge Centre

Tel: +44 20 8996 7004

Email: knowledgecentre@bsigroup.com

Copyright & Licensing

Tel: +44 20 8996 7070

Email: copyright@bsigroup.com



...making excellence a habit.™