



BSI Standards Publication

Industrial communication networks – Fieldbus specifications

Part 5-2: Application layer service definition –
Type 2 elements (IEC 61158-5-2:2019)

National foreword

This British Standard is the UK implementation of EN IEC 61158-5-2:2019. It is identical to IEC 61158-5-2:2019. It supersedes BS EN 61158-5-2:2014, which will be withdrawn on 15 May 2022.

The UK participation in its preparation was entrusted to Technical Committee GEL/65/3, 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 2019
Published by BSI Standards Limited 2019

ISBN 978 0 539 04841 4

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 August 2019.

Amendments/corrigenda issued since publication

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

English Version

Industrial communication networks - Fieldbus specifications -
Part 5-2: Application layer service definition - Type 2 elements
(IEC 61158-5-2:2019)

Réseaux de communication industriels - Spécifications des
bus de terrain - Partie 5-2 : Définition des services de la
couche application - Éléments de type 2
(IEC 61158-5-2:2019)

Industrielle Kommunikationsnetze - Feldbusse - Teil 5-2:
Dienstfestlegungen des Application Layer
(Anwendungsschicht) - Typ 2-Elemente
(IEC 61158-5-2:2019)

This European Standard was approved by CENELEC on 2019-05-15. 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, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, 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: Rue de la Science 23, B-1040 Brussels

European foreword

The text of document 65C/947/FDIS, future edition 4 of IEC 61158-5-2, 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 IEC 61158-5-2:2019.

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) 2020-02-15
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2022-05-15

This document supersedes EN 61158-5-2:2014.

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

Endorsement notice

The text of the International Standard IEC 61158-5-2:2019 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 61158-2:2014	NOTE	Harmonized as EN 61158-2:2014 (not modified)
IEC 61784-1:2019	NOTE	Harmonized as EN IEC 61784-1:2019 (not modified)
IEC 61784-2:2019	NOTE	Harmonized as EN IEC 61784-2:2019 (not modified)
IEC 62026-3	NOTE	Harmonized as EN 62026-3

Annex ZA

(normative)

Normative references to international publications with their corresponding European publications

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE 1 Where 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	2003	Programmable controllers -- Part 3: - Programming languages	-	-
IEC 61158-1	2019	Industrial communication networks - EN IEC 61158-1 Fieldbus specifications - Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series	-	2019
IEC 61158-3-2	2014	Industrial communication networks - EN 61158-3-2 Fieldbus specifications - Part 3-2: Data- link layer service definition - Type 2 elements	-	2014
IEC 61158-3- 2:2014/Amd 1	2019		-	-
IEC 61158-4-2	2019	Industrial communication networks - - Fieldbus specifications - Part 4-2: Data-link layer protocol specification - Type 2 elements	-	-
IEC 61158-6-2	2019	Industrial communication networks - EN 61158-6-2 Fieldbus specifications - Part 6-2: Application layer protocol specification - Type 2 elements	-	2019
IEC 61588	2009	Precision clock synchronization protocol for - networked measurement and control systems	-	-
IEC 61784-3-2	-	Industrial communication networks - - Profiles - Part 3-2: Functional safety fieldbuses - Additional specifications for CPF 2	-	-
ISO 639-2	-	Codes for the representation of names of - languages - Part-2: Alpha-3 code	-	-
ISO 8859-1	1987	Information processing - 8-bit single-byte - coded graphic character sets - Part 1: Latin alphabet No. 1	-	-

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
ISO 8859-2	1987	Information processing - 8-bit single byte - coded graphic character sets - Part 2: Latin alphabet No. 2		-
ISO 8859-3	1988	Information processing - 8-bit single-byte - coded graphic character sets - Part-3: Latin alphabet no. 3		-
ISO 8859-4	1988	Information processing - 8-bit single-byte - coded graphic character sets - Part-4: Latin alphabet no. 4		-
ISO/IEC 8859-5	1988	Information processing - 8-bit single-byte - coded graphic character sets - Part 5: Latin/Cyrillic alphabet		-
ISO 8859-6	1987	Information processing - 8-Bit single-byte - coded graphic character sets - Part 6: Latin/Arabic alphabet		-
ISO 8859-7	1987	Information processing - 8-bit single-byte - coded graphic character sets - Part 7: Latin/Greek alphabet		-
ISO 8859-8	1988	Information processing; 8-bit single-byte - coded graphic character sets; Part 8: Latin/hebrew alphabet		-
ISO/IEC 8859-9	1989	Information processing - 8-bit single-byte - coded graphic character sets - Part 9: Latin alphabet No. 5		-
ISO 11898	1993	Road vehicles - Interchange of digital - information - Controller area network (CAN) for high-speed communication		-
ISO/IEC 646	-	Information technology; ISO 7-bit coded - character set for information interchange		-
ISO/IEC 7498-1	-	Information technology - Open Systems - Interconnection - Basic reference model: The basic model		-
ISO/IEC 8859-1	-	Information technology - 8-bit single-byte - coded graphic character sets - Part-1: Latin alphabet No. 1		-
ISO/IEC 9545	-	Information technology - Open Systems - Interconnection - Application layer structure		-
ISO/IEC 10646	-	Information technology - Universal - Multiple-Octet 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		-
IETF RFC 1759	-	Printer MIB	-	-

CONTENTS

FOREWORD.....	6
INTRODUCTION.....	8
1 Scope.....	9
1.1 General.....	9
1.2 Specifications	10
1.3 Conformance	10
2 Normative references	10
3 Terms, definitions, symbols, abbreviated terms and conventions	12
3.1 ISO/IEC 7498-1 terms.....	12
3.2 ISO/IEC 8822 terms.....	13
3.3 ISO/IEC 9545 terms.....	13
3.4 ISO/IEC 8824-1 terms.....	13
3.5 Type 2 fieldbus data-link layer terms.....	13
3.6 Type 2 fieldbus application-layer specific definitions	13
3.7 Type 2 abbreviated terms and symbols	21
3.8 Conventions.....	22
3.8.1 Overview	22
3.8.2 General conventions.....	23
3.8.3 Conventions for class definitions	23
3.8.4 Conventions for service definitions	24
4 Common concepts	25
5 Data type ASE.....	25
5.1 General.....	25
5.2 Formal definition of data type objects.....	25
5.3 FAL defined data types	26
5.3.1 Fixed length types	26
5.3.2 String types	31
5.3.3 Structure types	32
5.4 Data type ASE service specification.....	36
6 Communication model specification	36
6.1 Concepts	36
6.1.1 General	36
6.1.2 General concepts	36
6.1.3 Relationships between ASEs	37
6.1.4 Naming and addressing	38
6.1.5 Data types	39
6.2 ASEs	46
6.2.1 Object management ASE.....	46
6.2.2 Connection manager ASE.....	154
6.2.3 Connection ASE	172
6.3 ARs	186
6.3.1 Overview	186
6.3.2 UCMM AR formal model	197
6.3.3 Transport AR formal model.....	199
6.3.4 AR ASE services	209
6.4 Summary of FAL classes	217

6.5 Permitted FAL services by AR type	218
Bibliography.....	220
Figure 1 – Overview of ASEs and object classes	38
Figure 2 – Addressing format using MAC, class, instance and attribute IDs	39
Figure 3 – Identity object state transition diagram	60
Figure 4 – Explicit and Implicit Setting interaction	63
Figure 5 – Static Assembly state transition diagram	67
Figure 6 – Dynamic Assembly state transition diagram	68
Figure 7 – Typical timing relationships for acknowledged data production.....	79
Figure 8 – Example of a COS system with two acking devices	80
Figure 9 – Message flow in COS connection – one Connection object, one consumer.....	80
Figure 10 – Message flow in COS connection – multiple consumers	81
Figure 11 – Path Reconfiguration in a ring topology	93
Figure 12 – CPF2 time synchronization offset clock model.....	94
Figure 13 – CPF2 time synchronization system with offset clock model	95
Figure 14 – CPF2 time synchronization group startup sequence	97
Figure 15 – Parameter object state transition diagram	104
Figure 16 – Example of Find_Next_Object_Instance service	130
Figure 17 – Transmission Trigger Timer behavior	180
Figure 18 – Inactivity watchdog timer	181
Figure 19 – Using tools for configuration.....	181
Figure 20 – Production Inhibit Timer behavior.....	182
Figure 21 – Context of transport services within the connection model.....	189
Figure 22 – Application-to-application view of data transfer	189
Figure 23 – Data flow diagram for a link producer	190
Figure 24 – Data flow diagram for a link consumer.....	191
Figure 25 – Triggers	192
Figure 26 – Binding transport instances to the producer and consumer of a transport connection that does not have a reverse data path	193
Figure 27 – Binding transport instances to the producers and consumers of a transport connection that does have a reverse data path	193
Figure 28 – Binding transport instances to the producer and consumers of a multipoint connection when the transport connection does not have a reverse data path	194
Figure 29 – Binding transport instances to the producers and consumers of a multipoint connection when the transport connection does have reverse data paths	194
Table 1 – Valid IANA MIB printer codes for character set selection	35
Table 2 – Common elements	41
Table 3 – ST language elements.....	42
Table 4 – Type conversion operations.....	43
Table 5 – Values of implementation-dependent parameters	45
Table 6 – Extensions to IEC 61131-3:2003	45
Table 7 – Identity object state event matrix	61

Table 8 – Static Assembly state event matrix	67
Table 9 – Static Assembly instance attribute access	68
Table 10 – Dynamic Assembly state event matrix	69
Table 11 – Dynamic Assembly instance attribute access.....	69
Table 12 – Message Router object Forward_Open parameters	72
Table 13 – Acknowledge Handler object state event matrix.....	76
Table 14 – Producing I/O application object state event matrix	77
Table 15 – PTPEnable attribute default values.....	84
Table 16 – Profile identification.....	90
Table 17 – Profile default settings and ranges	91
Table 18 – Profile transports.....	91
Table 19 – Default PTP clock settings.....	92
Table 20 – Hand_Set clock quality management.....	92
Table 21 – Path Reconfiguration Signalling message.....	93
Table 22 – Parameter object state event matrix	104
Table 23 – Status codes	107
Table 24 – Get_Attributes_All service parameters.....	109
Table 25 – Set_Attributes_All service parameters	111
Table 26 – Get_Attribute_List service parameters.....	113
Table 27 – Set_Attribute_List service parameters	115
Table 28 – Reset service parameters.....	117
Table 29 – Start service parameters	119
Table 30 – Stop service parameters.....	120
Table 31 – Create service parameters	122
Table 32 – Delete service parameters.....	124
Table 33 – Get_Attribute_Single service parameters.....	125
Table 34 – Set_Attribute_Single service parameters	127
Table 35 – Find_Next_Object_Instance service parameters	129
Table 36 – NOP service parameters	131
Table 37 – Apply_Attributes service parameters	132
Table 38 – Save service parameters	134
Table 39 – Restore service parameters.....	135
Table 40 – Get_Member service parameters.....	137
Table 41 – Set_Member service parameters	139
Table 42 – Insert_Member service parameters.....	141
Table 43 – Remove_Member service parameters.....	143
Table 44 – Group_Sync service parameters.....	144
Table 45 – Add_AckData_Path service parameters.....	146
Table 46 – Remove_AckData_Path service parameters	147
Table 47 – Get_Enum_String service parameters	148
Table 48 – Symbolic_Translation service parameters.....	150
Table 49 – Flash_LEDs service parameters	151
Table 50 – Multiple_Service_Packet service parameters.....	153

Table 51 – CM_Open service parameters	163
Table 52 – CM_Close service parameters	165
Table 53 – CM_Unconnected_Send service parameters	167
Table 54 – CM_Get_Connection_Data service parameters	168
Table 55 – CM_Search_Connection_Data service parameters	170
Table 56 – CM_Get_Connection_Data service parameters	171
Table 57 – I/O Connection object attribute access	176
Table 58 – Bridged Connection object attribute access	177
Table 59 – Explicit messaging object attribute access	178
Table 60 – Connection_Bind service parameters	184
Table 61 – Service_Name service parameters	185
Table 62 – How production trigger, transport class, and CM_RPI determine when data is produced	188
Table 63 – Transport classes	199
Table 64 – UCMM_Create service parameters	210
Table 65 – UCMM_Delete service parameters	211
Table 66 – UCMM_Write service parameters	212
Table 67 – UCMM_Abort service parameters	213
Table 68 – TR_Write service parameters	214
Table 69 – TR_Trigger service parameters	215
Table 70 – TR_Packet_arrived service parameters	215
Table 71 – TR_Ack_received service parameters	216
Table 72 – TR_Verify service parameters	216
Table 73 – TR_Status_updated service parameters	217
Table 74 – FAL class summary	218
Table 75 – FAL services by AR type	219

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**INDUSTRIAL COMMUNICATION NETWORKS –
FIELDBUS SPECIFICATIONS –****Part 5-2: Application layer service definition –
Type 2 elements**

FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

Attention is drawn to the fact that the use of the associated protocol type is restricted by its intellectual-property-right holders. In all cases, the commitment to limited release of intellectual-property-rights made by the holders of those rights permits a layer protocol type to be used with other layer protocols of the same type, or in other type combinations explicitly authorized by its intellectual-property-right holders.

NOTE Combinations of protocol types are specified in IEC 61784-1 and IEC 61784-2.

International Standard IEC 61158-5-2 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation.

This fourth edition cancels and replaces the third edition published in 2014. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

- addition of a data type in 5.3.2;
- clarifications of Object management ASE in 6.2.1;
- extensions of General ASE in 6.2.1.2.1;
- extensions/clarifications of Identity ASE in 6.2.1.2.2;
- update of Message Router ASE in 6.2.1.2.4;
- extensions/clarifications of Time Sync ASE in 6.2.1.2.6;
- updates of Parameter ASE in 6.2.1.2.7;
- updates of FAL ASE service specification in 6.2.1.3;
- extensions/clarifications of Connection manager ASE in 6.2.2;
- extensions/clarifications of Connection ASE in 6.2.3;
- extensions/clarifications of Application type in 6.3.1.4.5.
- miscellaneous editorial corrections.

The text of this International Standard is based on the following documents:

FDIS	Report on voting
65C/947/FDIS	65C/950/RVD

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

This publication has been drafted in accordance with ISO/IEC Directives, Part 2.

A list of all parts of the IEC 61158 series, published under the general title *Industrial communication networks – Fieldbus specifications*, can be found on the IEC web site.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

A bilingual version of this publication may be issued at a later date.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

This document 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 document 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 document is a conceptual architectural service, independent of administrative and implementation divisions.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 5-2: Application layer service definition – Type 2 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 part of IEC 61158 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 2 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 International Standard defines in an abstract way the externally visible service provided by the Type 2 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 document 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 document specifies the structure and services of the Type 2 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 document to provide access to the FAL to control certain aspects of its operation.

1.2 Specifications

The principal objective of this document 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 document 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 Type 2 application layer services as defined in this document.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. 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 61131-3:2003¹, *Programmable controllers – Part 3: Programming languages*

IEC 61158-1:2019, *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-3-2:2014, *Industrial communication networks – Fieldbus specifications – Part 3-2: Data-link layer service definition – Type 2 elements*

IEC 61158-3-2:2014/AMD1:2019

IEC 61158-4-2:2019, *Industrial communication networks – Fieldbus specifications – Part 4-2: Data-link layer protocol specification – Type 2 elements*

¹ A newer edition of this standard has been published, but only the cited edition applies.

IEC 61158-6-2:2019, *Industrial communication networks – Fieldbus specifications – Part 6-2: Application layer protocol specification – Type 2 elements*

IEC 61588:2009, *Precision clock synchronization protocol for networked measurement and control systems*

IEC 61784-3-2, *Industrial communication networks – Profiles – Part 3-2: Functional safety fieldbuses – Additional specifications for CPF 2*

ISO/IEC 646, *Information technology – ISO 7-bit coded character set for information interchange*

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

ISO/IEC 8859-1, *Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*

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*

ISO 639-2, *Codes for the representation of names of languages – Part 2: Alpha-3 code*

ISO 8859-1²:1987, *Information processing – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*

ISO 8859-2³:1987, *Information processing – 8-bit single-byte coded graphic character sets – Part 2: Latin alphabet No. 2*

ISO 8859-3⁴:1988, *Information processing – 8-bit single-byte coded graphic character sets – Part 3: Latin alphabet No. 3*

ISO 8859-4⁵:1988, *Information processing – 8-bit single-byte coded graphic character sets – Part 4: Latin alphabet No. 4*

ISO 8859-5⁶:1988, *Information processing – 8-bit single-byte coded graphic character sets – Part 5: Latin/Cyrillic alphabet*

² A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

³ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

⁴ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

⁵ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

ISO 8859-6⁷:1987, *Information processing – 8-bit single-byte coded graphic character sets – Part 6: Latin/Arabic alphabet*

ISO 8859-7⁸:1987, *Information processing – 8-bit single-byte coded graphic character sets – Part 7: Latin/Greek alphabet*

ISO 8859-8⁹:1988, *Information processing – 8-bit single-byte coded graphic character sets – Part 8: Latin/Hebrew alphabet*

ISO 8859-9¹⁰:1989, *Information processing – 8-bit single-byte coded graphic character sets – Part 9: Latin alphabet No. 5*

ISO 11898:1993¹¹, *Road vehicles – Interchange of digital information – Controller area network (CAN) for high-speed communication*

IETF RFC 1759, *Printer MIB*, available at <<http://www.ietf.org>> [viewed 2018-09-04]

3 Terms, definitions, symbols, abbreviated terms and conventions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

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

⁶ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

⁷ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

⁸ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

⁹ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

¹⁰ A newer edition of this standard has been published by ISO/IEC, but the cited edition is the one used in the referenced IETF standards.

¹¹ A newer edition of this standard has been published, but only the cited edition applies.

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 Type 2 fieldbus data-link layer terms

The following terms, defined in IEC 61158-3-2 and IEC 61158-4-2, apply.

- a) DL-time
- b) DL-scheduling-policy
- c) DLCEP
- d) DLC
- e) DL-connection-oriented mode
- f) DLPDU
- g) DLSDU
- h) DLSAP
- i) fixed tag
- j) generic tag
- k) link
- l) MAC ID
- m) network address
- n) node address
- o) node
- p) tag
- q) scheduled
- r) unscheduled

3.6 Type 2 fieldbus application-layer specific definitions

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

3.6.1

allocate

take a resource from a common area and assign that resource for the exclusive use of a specific entity

3.6.2

application

function or data structure for which data is consumed or produced

3.6.3

application objects

multiple object classes that manage and provide a run time exchange of messages across the network and within the network device

3.6.4

application process

part of a distributed application on a network, which is located on one device and unambiguously addressed

3.6.5

application process object

component of an application process that is identifiable and accessible through an FAL application relationship

3.6.6

application process object class

class of application process objects defined in terms of the set of their network-accessible attributes and services

3.6.7

application relationship

cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation

Note 1 to entry: This relationship is activated either by the exchange of application-protocol-data-units or as a result of preconfiguration activities.

3.6.8

application relationship application service element

application-service-element that provides the exclusive means for establishing and terminating all application relationships

3.6.9

application relationship endpoint

context and behavior of an application relationship as seen and maintained by one of the application processes involved in the application relationship

Note 1 to entry: Each application process involved in the application relationship maintains its own application relationship endpoint.

3.6.10

attribute

description of an externally visible characteristic or feature of an object

Note 1 to entry: The attributes of an object contain information about variable portions of an object. Typically, they provide status information or govern the operation of an object. Attributes may also affect the behavior of an object. Attributes are divided into class attributes and instance attributes.

3.6.11**behavior**

indication of how an object responds to particular events

3.6.12**Best Master Clock Algorithm****BMCA**

algorithm performed by each node to determine the clock that will become the master clock on a subnet and the grandmaster clock for the domain

Note 1 to entry: The algorithm primarily compares priority1, clock quality, priority2, and source identity to determine the best master among available candidates.

3.6.13**boundary clock**

clock that has multiple Precision Time Protocol (PTP) ports in a domain and maintains the timescale used in the domain

Note 1 to entry: It may serve as the source of time, i.e., be a master clock, and may synchronize to another clock, i.e., be a slave clock.

[SOURCE: IEC 61588:2009, 3.1.3, modified – second sentence changed to a Note]

3.6.14**class**

set of objects, all of which represent the same kind of system component

Note 1 to entry: A class is a generalization of an object; a template for defining variables and methods. All objects in a class are identical in form and behavior, but usually contain different data in their attributes.

3.6.15**class attribute**

attribute that is shared by all objects within the same class

3.6.16**class code**

unique identifier assigned to each object class

3.6.17**class specific service**

service defined by a particular object class to perform a required function which is not performed by a common service

Note 1 to entry: A class specific object is unique to the object class which defines it.

3.6.18**client**

a) object which uses the services of another (server) object to perform a task

b) initiator of a message to which a server reacts

3.6.19**clock**

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

Note 1 to entry: There are three types of clocks in IEC 61588:2009, boundary, transparent and ordinary clocks.

[SOURCE: IEC 61588:2009, 3.1.4, modified – different Note]

3.6.20

communication objects

components that manage and provide a run time exchange of messages across the network

EXAMPLES Connection Manager object, Unconnected Message Manager (UCMM) object, and Message Router object.

3.6.21

connection

logical binding between application objects that may be within the same or different devices

Note 1 to entry: Connections may be either point-to-point or multipoint.

3.6.22

connection ID

CID

identifier assigned to a transmission that is associated with a particular connection between producers and consumers, providing a name for a specific piece of application information

3.6.23

connection path

octet stream that defines the application object to which a connection instance applies

3.6.24

connection point

buffer which is represented as a subinstance of an Assembly object

3.6.25

consume

act of receiving data from a producer

3.6.26

consumer

node or sink that is receiving data from a producer

3.6.27

consuming application

application that consumes data

3.6.28

cyclic

repetitive in a regular manner

3.6.29

device

physical hardware connected to the link

Note 1 to entry: A device may contain more than one node.

3.6.30

device profile

collection of device dependent information and functionality providing consistency between similar devices of the same device type

3.6.31

domain

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

[SOURCE: IEC 61588:2009, 3.1.7]

3.6.32**end node**

producing or consuming node

3.6.33**endpoint**

one of the communicating entities involved in a connection

3.6.34**epoch**

origin of a time scale

[SOURCE: IEC 61588:2009, 3.1.9]

3.6.35**error**

discrepancy between a computed, observed or measured value or condition and the specified or theoretically correct value or condition

3.6.36**event**

instance of a change of conditions

3.6.37**frame**

denigrated synonym for DLPDU

3.6.38**grandmaster clock**

within a domain, clock that is the ultimate source of time for clock synchronization using the PTP protocol

[SOURCE: IEC 61588:2009, 3.1.13]

3.6.39**group**

a) <general> a general term for a collection of objects. Specific uses:

b) <addressing> when describing an address, an address that identifies more than one entity

3.6.40**interface**

(a) shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics as appropriate

(b) collection of FAL class attributes and services that represents a specific view on the FAL class

3.6.41**invocation**

act of using a service or other resource of an application process

Note 1 to entry: Each invocation represents a separate thread of control that may be described by its context. Once the service completes, or use of the resource is released, the invocation ceases to exist. For service invocations, a service that has been initiated but not yet completed is referred to as an outstanding service invocation. Also for service invocations, an Invoke ID may be used to unambiguously identify the service invocation and differentiate it from other outstanding service invocations.

3.6.42

instance

actual physical occurrence of an object within a class that identifies one of many objects within the same object class

EXAMPLE California is an instance of the object class state.

Note 1 to entry: The terms object, instance, and object instance are used to refer to a specific instance.

3.6.43

instance attribute

attribute whose value is unique to an object instance and whose definition is shared by all instances of an object

3.6.44

instantiated

object that has been created in a device

3.6.45

logical device

a certain FAL class that abstracts a software component or a firmware component as an autonomous self-contained facility of an automation device

3.6.46

Lpacket

Link packet

piece of application information that contains a size, control octet, tag, and link data

Note 1 to entry: Peer data-link layers use Lpackets to send and receive service data units from higher layers in the OSI stack.

3.6.47

management information

network-accessible information that supports managing the operation of the fieldbus system, including the application layer

Note 1 to entry: Managing includes functions such as controlling, monitoring, and diagnosing.

3.6.48

master clock

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

[SOURCE: IEC 61588:2009, 3.1.17]

3.6.49

member

piece of an attribute that is structured as an element of an array

3.6.50

Message Router

object within a node that distributes messaging requests to appropriate application objects

3.6.51

method

<object> synonym for an operational service which is provided by the server ASE and invoked by a client

3.6.52**module**

hardware or logical component of a physical device

3.6.53**multipoint connection**

connection from one node to many

Note 1 to entry: Multipoint connections allow messages from a single producer to be received by many consumer nodes.

3.6.54**network**

set of nodes connected by some type of communication medium, including any intervening repeaters, bridges, routers and lower-layer gateways

3.6.55**object**

abstract representation of a particular component within a device, usually a collection of related data (in the form of variables) and methods (procedures) for operating on that data that have clearly defined interface and behavior

3.6.56**object specific service**

service unique to the object class which defines it

3.6.57**ordinary clock**

clock that has a single Precision Time Protocol (PTP) port in a domain and maintains the timescale used in the domain

Note 1 to entry: It may serve as a source of time, i.e., be a master clock, or may synchronize to another clock, i.e., be a slave clock.

[SOURCE: IEC 61588:2009, 3.1.22, modified – second sentence changed to a Note]

3.6.58**originator**

client responsible for establishing a connection path to the target

3.6.59**parent clock**

master clock to which a clock is synchronized

[SOURCE: IEC 61588:2009, 3.1.23]

3.6.60**peer**

role of an AR endpoint in which it is capable of acting as both client and server

3.6.61**physical device**

automation or other network device

3.6.62**point-to-point connection**

connection that exists between exactly two application objects

3.6.63

Precision Time Protocol

PTP

protocol defined by IEC 61588:2009

Note 1 to entry: As an adjective, it indicates that the modified noun is specified in or interpreted in the context of IEC 61588:2009.

[SOURCE: IEC 61588:2009, 3.1.28, modified – second sentence changed to a Note]

3.6.64

produce

act of sending data to be received by a consumer

3.6.65

producer

node that is responsible for sending data

3.6.66

property

general term for descriptive information about an object

3.6.67

PTP message

one of the message types defined in IEC 61588:2009

[SOURCE: IEC 61588:2009, 3.1.33]

3.6.68

PTP port

logical access point of a clock for PTP communications to the communications network

[SOURCE: IEC 61588:2009, 3.1.35]

3.6.69

resource

processing or information capability of a subsystem

3.6.70

serial number

unique 32-bit integer value assigned by each manufacturer/vendor to every device having Type 2 communication capabilities

Note 1 to entry: The Vendor ID and serial number jointly form a unique identifier for each device.

3.6.71

server

- a) role of an AREP in which it returns a confirmed service response APDU to the client that initiated the request
- b) object which provides services to another (client) object

3.6.72

service

operation or function than an object and/or object class performs upon request from another object and/or object class

3.6.73**synchronized clocks**

(to a specified uncertainty) two clocks which have the same epoch and for which measurements of the time of a single event at an arbitrary time differ by no more than the specified uncertainty

[SOURCE: IEC 61588:2009, 3.1.40, modified – reworded]

3.6.74**System Time**

absolute time value as defined by CPF2 time synchronization in the context of a distributed time system where all devices have a local clock that is synchronized with a common master clock

Note 1 to entry: In the context of CPF2, System Time is a 64-bit integer value in units of nanoseconds with a value of 0 corresponding to the date 1970-01-01.

3.6.75**target**

end-node to which a connection is established

3.6.76**transparent clock**

device that measures the time taken for a Precision Time Protocol (PTP) event message to transit the device and provides this information to clocks receiving this PTP event message

[SOURCE: IEC 61588:2009, 3.1.46, modified – truncated]

3.6.77**Unconnected Message Manager****UCMM**

component within a node that transmits and receives unconnected explicit messages and sends them directly to the Message Router object

3.6.78**unconnected service**

messaging service which does not rely on the set up of a connection between devices before allowing information exchanges

3.6.79**vendor ID**

identification of each product manufacturer/vendor by a unique number

Note 1 to entry: Vendor IDs are assigned by ODVA, Inc.

3.7 Type 2 abbreviated terms and symbols

AE	Application Entity
AL	Application layer
APO	Application object
AP	Application process
APDU	Application protocol data unit
AR	Application relationship
AREP	Application relationship end point
ASCII	American standard code for information interchange
ASE	Application service element

CID	Connection ID
CM_API	Actual packet interval
CM_RPI	Requested packet interval
Cnf	Confirmation
CR	Communication relationship
DL-	(as a prefix) data-link-
DLC	Data-link connection
DLCEP	Data-link connection end point
DLL	Data-link layer
DLM	Data-link-management
DLSAP	Data-link service access point
DLSDU	DL-service-data-unit
FAL	Fieldbus application layer
FIFO	First-in first-out
ID	Identifier
IEC	International Electrotechnical Commission
Ind	Indication
IP	Internet protocol
ISO	International Organization for Standardization
I/O	Input/output
LME	Layer management entity
O2T	Originator to target (connection characteristics)
O⇒T	Originator to target (connection characteristics)
OSI	Open systems interconnect
PDU	Protocol data unit
PL	Physical layer
PTP	Precision Time Protocol [IEC 61588:2009]
QoS	Quality of service
REP	Route endpoint
Req	Request
Rsp	Response
SAP	Service access point
SDU	Service data unit
SEM	State event matrix
STD	State transition diagram, used to describe object behavior
T2O	Target to originator (connection characteristics)
T⇒O	Target to originator (connection characteristics)

3.8 Conventions

3.8.1 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 document. The service specification defines the services that are provided by the ASE.

3.8.2 General conventions

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

Bold font is used in this document to highlight parameter names or important requirement elements from surrounding text.

3.8.3 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 document, or by a user of this document.
- (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 99, 240 and 767 are reserved by this document to identify standardized classes. CLASS IDs between 100 and 199, 768 and 1 279 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 document.

(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.
If an attribute is optional, its default value (specified in IEC 61158-6-2) shall provide the same behavior as if the attribute was not implemented.
- 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 (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.

3.8.4 Conventions for service definitions

3.8.4.1 General

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

3.8.4.2 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 document 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.

3.8.4.3 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 Common concepts

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

5 Data type ASE

5.1 General

An overview of the data type ASE and the relationships between data types is provided in IEC 61158-1, 10.2.

5.2 Formal definition of data type objects

The template used to describe the data type class in Clause 5 is detailed in IEC 61158-1, 10.2. This includes the specific ASE structure and the definition of its attributes.

5.3 FAL defined data types

5.3.1 Fixed length types

5.3.1.1 Boolean types

5.3.1.1.1 Boolean

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

This data type expresses a Boolean data type with the values TRUE and FALSE.

5.3.1.1.2 BOOL

This IEC 61131-3 type is the same as Boolean.

5.3.1.2 Bitstring types

5.3.1.2.1 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.

5.3.1.2.2 SWORD

This IEC 61131-3 type is the same as Bitstring8.

5.3.1.2.3 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

5.3.1.2.4 WORD

This IEC 61131-3 type is the same as Bitstring16.

5.3.1.2.5 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

5.3.1.2.6 DWORD

This IEC 61131-3 type is the same as Bitstring32.

5.3.1.2.7 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

5.3.1.2.8 LWORD

This IEC 61131-3 type is the same as Bitstring64.

5.3.1.3 Date types**5.3.1.3.1 DATE**

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= DATE
3	Format	= FIXED LENGTH
4.1	Octet Length	= 2

This IEC 61131-3 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. It expresses the date as a number of days, starting from 1972-01-01 (January 1st, 1972), the start of the Coordinated Universal Time (UTC) era, until 2151-06-06 (June 6th, 2151), i.e. a total range of 65 536 days.

5.3.1.3.2 TimeOfDay

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= 12
2	Data type Name	= TimeOfDay
4	Format	= FIXED LENGTH
4.1	Octet Length	= 6

This data type is composed of two elements of unsigned values and expresses the time of day and the date. The first element is an Unsigned32 data type and gives the time after the midnight in milliseconds. The second element is an Unsigned16 data type and gives the date counting the days from 1984-01-01 (January 1st, 1984).

5.3.1.3.3 TimeOfDay without date indication

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= 52
2	Data type Name	= TimeOfDay without date indication
4	Format	= FIXED LENGTH
4.1	Octet Length	= 4

This data type is composed of one element of an unsigned value and expresses the time of day. The element is an Unsigned32 data type and gives the time after the midnight in milliseconds.

5.3.1.3.4 TIME_OF_DAY

This IEC 61131-3 type is the same as TimeofDay without date indication.

5.3.1.4 Numeric types

5.3.1.4.1 Floating Point types

5.3.1.4.1.1 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.

5.3.1.4.1.2 REAL

This IEC 61131-3 type is the same as Float32.

5.3.1.4.1.3 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.

5.3.1.4.1.4 LREAL

This IEC 61131-3 type is the same as Float64.

5.3.1.4.2 Integer types

5.3.1.4.2.1 Integer8

CLASS: Data type

ATTRIBUTES:

1	Data type Numeric Identifier	=	2
2	Data type Name	=	Integer8
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	1

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

5.3.1.4.2.2 SINT

This IEC 61131-3 type is the same as Integer8.

5.3.1.4.2.3 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.

5.3.1.4.2.4 INT

This IEC 61131-3 type is the same as Integer16.

5.3.1.4.2.5 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.

5.3.1.4.2.6 DINT

This IEC 61131-3 type is the same as Integer32.

5.3.1.4.2.7 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.

5.3.1.4.2.8 LINT

This IEC 61131-3 type is the same as Integer64.

5.3.1.4.3 Unsigned types**5.3.1.4.3.1 Unsigned8****CLASS:** Data type**ATTRIBUTES:**

1	Data type Numeric Identifier	=	5
2	Data type Name	=	Unsigned8
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	1

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 type has a length of one octet.

5.3.1.4.3.2 USINT

This IEC 61131-3 type is the same as Unsigned8.

5.3.1.4.3.3 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.

5.3.1.4.3.4 UINT

This IEC 61131-3 type is the same as Unsigned16.

5.3.1.4.3.5 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.

5.3.1.4.3.6 UDINT

This IEC 61131-3 type is the same as Unsigned32.

5.3.1.4.3.7 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.

5.3.1.4.3.8 ULINT

This IEC 61131-3 type is the same as Unsigned64.

5.3.1.5 Time types

5.3.1.5.1 TIME

CLASS:	Data type
ATTRIBUTES:	
1 Data type Numeric Identifier	= not used
2 Data type Name	= TIME
3 Format	= FIXED LENGTH
4.1 Octet Length	= 4

This IEC 61131-3 type is a two's complement binary number with a length of four octets. The unit of time for this type is 1 ms.

5.3.1.5.2 ITIME

CLASS:	Data type
ATTRIBUTES:	
1 Data type Numeric Identifier	= not used
2 Data type Name	= ITIME
3 Format	= FIXED LENGTH
4.1 Octet Length	= 2

This IEC 61131-3 type extension is a two's complement binary number with a length of two octets. The unit of time for this type is 1 ms.

5.3.1.5.3 FTIME

CLASS:	Data type
ATTRIBUTES:	
1 Data type Numeric Identifier	= not used
2 Data type Name	= FTIME
3 Format	= FIXED LENGTH
4.1 Octet Length	= 4

This IEC 61131-3 type extension is a two's complement binary number with a length of four octets. The unit of time for this type is 1 µs.

5.3.1.5.4 LTIME

CLASS:	Data type
ATTRIBUTES:	
1 Data type Numeric Identifier	= not used
2 Data type Name	= LTIME
3 Format	= FIXED LENGTH
4.1 Octet Length	= 8

This IEC 61131-3 type extension is a two's complement binary number with a length of eight octets. The unit of time for this type is 1 µs.

5.3.2 String types

5.3.2.1 BitString

CLASS:	Data type
ATTRIBUTES:	
1 Data type Numeric Identifier	= 14
2 Data type Name	= Bitstring
3 Format	= STRING
5.1 Octet Length	= 1 to n

This string type is defined as a series of BitString8 elements.

5.3.2.2 OctetString

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= 10
2	Data type Name	= OctetString
3	Format	= STRING
4.1	Octet Length	= 1 to n

An OctetString is an ordered sequence of octets, numbered from 1 to n. For the purposes of discussion, octet 1 of the sequence is referred to as the first octet. IEC 61158-6-2 defines the order of transmission.

5.3.2.3 EPATH

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= EPATH
3	Format	= STRING
4.1	Octet Length	= 1 to n

An EPATH is an ordered sequence of octets, numbered from 1 to n. Its format is further specified in IEC 61158-6-2, 4.1.9.

5.3.2.4 ETH_MAC_ADDR

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= ETH_MAC_ADDR
3	Format	= ARRAY
6.1	Number of array elements	= 6
6.2	Array element data type	= UINT

The Ethernet MAC Address is an array of 6 octets. The recommended display format is “XX-XX-XX-XX-XX-XX”, starting with the first octet. The Ethernet MAC Address is assigned by the manufacturer and shall be unique per ISO/IEC/IEEE 8802-3 requirements. The general requirement is that the value of this type shall be Ethernet MAC addresses used in Ethernet frames.

5.3.3 Structure types

5.3.3.1 DATE_AND_TIME

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= DATE_AND_TIME
3	Format	= STRUCTURE
5.1	Number of Fields	= 2
5.2.1	Field Name	= Time_Of_Day_Element
5.2.2	Field Data type	= TIME_OF_DAY
5.3.1	Field Name	= Date_Element
5.3.2	Field Data type	= DATE

This IEC 61131-3 type extension is a structure which expresses both the date (as a number of days starting from 1972-01-01 until 2151-06-06), and the time of day as a number of ms starting from midnight.

5.3.3.2 SHORT_STRING

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= SHORT_STRING
3	Format	= STRUCTURE
5.1	Number of Fields	= 2
5.2.1	Field Name	= Charcount_Element
5.2.2	Field Data type	= USINT
5.3.1	Field Name	= Stringcontents_Element
5.3.2	Field Data type	= OctetString

This IEC 61131-3 type extension is composed of two elements. Charcount_Element gives the current number of characters in the Stringcontents_Element (one octet per character). Characters shall be as specified in ISO/IEC 8859-1.

5.3.3.3 STRING

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= STRING
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	= Stringcontents_Element
5.3.2	Field Data type	= OctetString

This IEC 61131-3 type is composed of two elements. Charcount_Element gives the current number of characters in the Stringcontents_Element (one USINT per character). Characters shall be as specified in ISO/IEC 8859-1.

5.3.3.4 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 shall be as specified in ISO/IEC 10646.

5.3.3.5 STRINGN

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= STRINGN
3	Format	= STRUCTURE
5.1	Number of Fields	= 3
5.2.1	Field Name	= Charsize_Element
5.2.2	Field Data type	= UINT
5.3.1	Field Name	= Charcount_Element
5.3.2	Field Data type	= UINT
5.4.1	Field Name	= StringNcontents_Element
5.4.2	Field Data type	= OctetString

This IEC 61131-3 type extension is composed of three elements. Charsize_Element gives the size of a character in StringNcontents_Element (N = number of USINT). Charcount_Element gives the current number of characters in the StringNcontents_Element (N USINT per character). Characters shall be as specified in ISO/IEC 10646.

5.3.3.6 STRINGI

CLASS:		Data type
ATTRIBUTES:		
1	Data type Numeric Identifier	= not used
2	Data type Name	= STRINGI
3	Format	= STRUCTURE
5.1	Number of Fields	= 2
5.2.1	Field Name	= Stringnum_Element
5.2.2	Field Data type	= USINT
5.3.1	Field Name	= International_String_Array
5.3.2	Field Data type	= STRINGI_ARRAY

This IEC 61131-3 type extension allows for multiple language representation for a single string. It is a structured data type which allocates a USINT variable (Stringnum_Element) containing the number of internationalized character strings and an array of structures (International_String_Array) containing the internationalized character strings.

The international character string structure (STRINGI_ELEMENT) is defined as follows:

- a USINT (Language1_Element) indicating the first ASCII character of the ISO 639-2/T language;
- a USINT (Language2_Element) indicating the second character of the ISO 639-2/T language;
- a USINT (Language3_Element) indicating the third character of the ISO 639-2/T language;
- a USINT (Datatype_Element), limited to the values 0xD0 (STRING), 0xD5 (STRING2), 0xD9 (STRINGN), and 0xDA (SHORT_STRING) indicating the structure of the character string;
- a UINT (Charset_Element) indicating the character set which the character string is based on;
- an array of octet elements which is the actual international character string (which data type is specified in Datatype_Element).

The three characters for the language come from ISO 639-2/T (Alpha-3 Terminology Code), and the character set values come from IANA MIB printer codes (IETF RFC 1759). The character set values are limited to those values that are provided in Table 1.

Table 1 – Valid IANA MIB printer codes for character set selection

Character Set	Value
ISO 8859-1:1987	4
ISO 8859-2:1987	5
ISO 8859-3:1988	6
ISO 8859-4:1988	7
ISO 8859-5:1988	8
ISO 8859-6:1987	9
ISO 8859-7:1987	10
ISO 8859-8:1989	11
ISO 8859-9:1989	12
ISO/IEC 10646-UCS-2	1 000
ISO/IEC 10646-UCS-4	1 001

5.3.3.7 SHORT_STRING**CLASS:** Data type**ATTRIBUTES:**

- 1 Data type Numeric Identifier = not used
- 2 Data type Name = SHORT_STRING
- 3 Format = STRUCTURE
- 5.1 Number of Fields = 2
- 5.2.1 Field Name = Charcount_Element
- 5.2.2 Field Data type = USINT
- 5.3.1 Field Name = Stringcontents_Element
- 5.3.2 Field Data type = OctetString

This IEC 61131-3 type extension is composed of a single elements. Charcount_Element gives the current number of characters in the Stringcontents_Element (one octet per character). Characters shall be as specified in ISO/IEC 8859-1.

5.3.3.8 STRINGI_ELEMENT**CLASS:** Data type**ATTRIBUTES:**

- 1 Data type Numeric Identifier = not used
- 2 Data type Name = STRINGI_ELEMENT
- 3 Format = STRUCTURE
- 5.1 Number of Fields = 6
- 5.2.1 Field Name = Language1_Element
- 5.2.2 Field Data type = USINT
- 5.3.1 Field Name = Language2_Element
- 5.3.2 Field Data type = USINT
- 5.4.1 Field Name = Language3_Element
- 5.4.2 Field Data type = USINT
- 5.5.1 Field Name = Datatype_Element
- 5.5.2 Field Data type = EPATH
- 5.6.1 Field Name = Charset_Element
- 5.6.2 Field Data type = UINT
- 5.7.1 Field Name = Stringcontents_Element
- 5.7.2 Field Data type = SHORT_STRING | STRING | STRING2 | STRINGN

5.4 Data type ASE service specification

There are no operational services defined for the type object.

6 Communication model specification

6.1 Concepts

6.1.1 General

This is a serial communication system for communication between devices that wish to exchange both time critical and messaging type application information. These devices include simple I/O devices, such as sensors/actuators as well as complex control devices such as robots, programmable logic controllers, welders, process controllers.

Unlike some general purpose communication systems that rely on the destination delivery model, this network uses a variant of the publisher/subscriber push model, called the producer/consumer model. The producer/consumer model allows the exchange of time critical application information between a sending device (i.e. the producer) and many receiving devices (i.e. the consumers) without the need to send the data separately to each destination. This is accomplished by attaching a unique identifier to each piece of application information that is being produced onto the network medium. Any device that requires a specific piece of application information simply filters the data on the network medium for the appropriate identifier. Many devices can receive the same piece of application information from a single producing device.

The Type 2 application layer can be associated with different data-link layers, depending on the selected communication profile.

The Type 2 specific data-link layer provides a high degree of protocol efficiency by utilizing an implied token passing mechanism. This mechanism allows all devices equal access to the network without the network overhead associated with passing a “token” to each device granting it permission to send data. The protocol utilizes a time based scheduling mechanism which provides network devices with deterministic and predictable access to the medium while preventing network collisions. This scheduling mechanism allows time critical data, which is required on a periodic, repeatable and predictable basis, to be produced on a predefined schedule without the loss of efficiency associated with continuously requesting or “polling” for the required data. The protocol supports an additional mechanism which allows data that is not time critical in nature or which is only required on an occasional basis to utilize any available network time. This unscheduled data is transmitted after the production of the time critical data has been completed and before the beginning of the next scheduled production of time critical data.

6.1.2 General concepts

Most of the general concepts described in Clause 4 apply, with some restrictions or additions as noted:

- the application layer includes those functionalities from the intermediate layers which are necessary for proper mapping onto the data-link layer,
- Client/Server relationships can be one-to-one, but also one-to-many (see AR model in 6.3.1),
- a variant of the Publisher/Subscriber Push model, the Producer/Consumer model, is used as the base of all AR's (see AR model in 6.3.1),
- an overview of the ASE/APO types and their relationships is provided in 6.1.3,
- AR follow a model which is detailed in 6.3.1,
- specific naming and addressing is specified in 6.1.4,

- common FAL attributes and parameters listed in IEC 61158-1, 9.7 are not used; instead they are specified in relevant Type-specific subclauses,
- there are no Abort or Reject services, only negative result confirmations, so the error procedure described in Annex A does not apply.

6.1.3 Relationships between ASEs

Every node shall contain as a minimum instance one of the following ASE object classes in its application layer.

Object Management ASE:

- Identity (identification and general information about the device)
- Message Router (messaging connection point for communications within the device)

Connection Manager ASE:

- Connection Manager (establishment and maintenance of connections)

or Connection ASE:

- Connection (messaging connection point for communications within the device)

AR ASE:

- Unconnected Message Manager (queued messaging services on a single link, optional for CP 2/3)

Other application layer objects may be implemented in some nodes only:

Object Management ASE:

- Assembly object (binds data from multiple objects to be sent through one connection)
- Acknowledge Handler object (handles acknowledge messages from the application objects)
- Time Sync object (interface to the IEC 61588:2009 precision clock synchronization protocol)
- Parameter object (public interface to device configuration data)
- additional application-specific objects constructed according to the general Type 2 formal model

AR ASE:

- Transports (connected messaging and data services).

Yet more objects belong to system management, primarily concerned with the data-link layer:

- ControlNet object (station management interfaces to lower layers, required in all devices using Type 2 data-link layer),
- Keeper object (holds and distributes attributes of all devices on the link, one required per link, with optional backup(s) – used in conjunction with Type 2 data-link layer),
- Scheduling object (holds information on link schedules, one required in each connection originator – used in conjunction with Type 2 data-link layer),
- TCP/IP Interface object (provides the mechanism to configure the TCP/IP interface of a device, required in all devices with a TCP/IP interface),
- Ethernet Link object (maintains link-specific counters and status information for an Ethernet port, required in all devices with an Ethernet port),

- DeviceNet object (station management interfaces to lower layers, required in all devices using ISO 11898:1993 data-link layer),
- Connection Configuration object (interface to create, configure and control connections in a device),
- DLR object (configuration and status information interface for the DLR protocol),
- QoS object (configuration interface for QoS-related behaviors in devices),
- Port object (describes Type 2 ports present on the device),
- PRP/HSR Protocol object (configuration and diagnostic interface for the PRP and HSR protocols),
- PRP/HSR Nodes Table object (record of all PRP or HSR capable nodes detected on the network).

NOTE These lower layer management objects are described in IEC 61158-4-2.

The ASE and object interactions for a device using both application and data-link Type 2 protocols are illustrated in Figure 1:

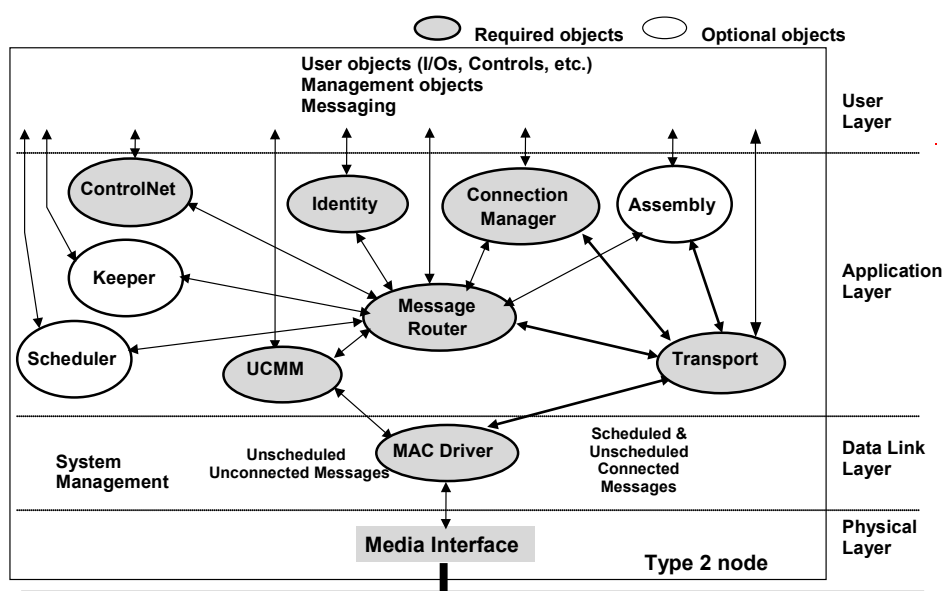


Figure 1 – Overview of ASEs and object classes

6.1.4 Naming and addressing

The information presented here provides a common basis for logically addressing separate physical components across the network. These addressing terms are used in the specifications. Figure 2 should be referenced in the following discussion.

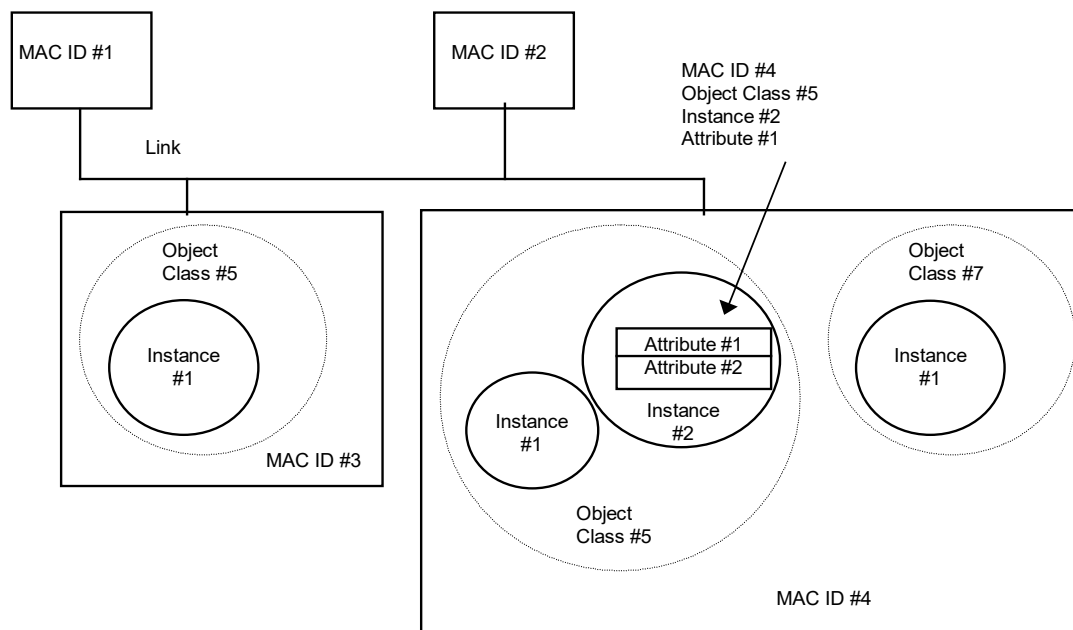


Figure 2 – Addressing format using MAC, class, instance and attribute IDs

The term “node” is used to refer to that portion of a device which contains a link interface and responds to a single MAC ID. The term “device” refers to a complete physical device connecting to the network. A device is able to contain multiple nodes.

NOTE When the Type 2 application layer is used in conjunction with Ethernet, the referenced MAC ID is actually the IP address and not the Ethernet MAC ID.

The Class ID is a unique integer identification value assigned to each object class accessible from the network. The object class may be referenced with this Class ID. In all IEC 61158 specifications for Type 2, class code is synonymous with Class ID.

The Instance ID is an integer identification value assigned to an object instance when it is created that identifies it among all Instances of the same Class. This integer is unique within the <Node:Class> in which it resides.

The Attribute ID is an integer identification value that is unique among all other attributes of that object.

The address of attribute number 1 of instance 2 in class 5 in the node with a MAC ID of 4 is MAC ID#4: Object Class #5: Instance #2: Attribute #1 as shown in Figure 2. This nomenclature is referred to as Class/Instance/Attribute addressing.

Information on how to address an object through multiple links or using a name is provided in IEC 61158-6-2 (Path definition).

6.1.5 Data types

6.1.5.1 Data type general specification

The specification of a data type is comprised of the range of values that variables of the type may take on and the operations performed on these variables.

Data is made up of elementary (primitive) data types. These elementary data types are used to construct derived (constructed) data types.

NOTE 1 Elementary and derived data type specifications correspond to the notation of IEC 61131-3. In addition, since function blocks as defined in IEC 61131-3 have both an associated data structure and a set of defined operations, these elements are also specified below as additional data types.

The derived data types are the following:

- directly derived;
- enumerated;
- subrange;
- structured;
- array.

NOTE 2 These derived data types are defined in IEC 61131-3:2003, 1.3 and 2.3.3. The means of specifying these data types and their default initial values is defined in IEC 61131-3:2003, 2.3.3.1 and 2.3.3.2. The usage of variables of these data types is defined in IEC 61131-3:2003, 2.3.3.3.

6.1.5.2 Supported elementary data types

The following basic data types defined in Clause 5 are supported:

- BOOL
- SINT
- INT
- DINT
- LINT
- USINT
- UINT
- UDINT
- ULINT
- REAL
- LREAL
- ITIME
- TIME
- FTIME
- LTIME
- DATE
- TIME_OF_DAY or TOD
- DATE_AND_TIME or DT
- STRING
- STRING2
- STRINGN
- STRINGI
- SHORT_STRING
- SWORD
- WORD
- DWORD
- LWORD
- EPATH

6.1.5.3 Compliance with IEC 61131-3

6.1.5.3.1 Compliance statement

NOTE 1 Subclause 6.1.5.3 provides information with respect to only the data types and associated operations defined in this part of IEC 61158.

NOTE 2 IEC 61131-3:2003, 1.5.1, defines the requirements which are met by programmable controller systems claiming compliance to IEC 61131-3. This provides the data type-related information to be included in the documentation of functional units which support the data types defined in this part of IEC 61158. Subsets or extensions of this documentation are provided as appropriate to the specific compliant functional unit.

An implementation shall comply with the requirements of IEC 61131-3 for language features as specified in Table 2 through Table 4.

Table 2 – Common elements

IEC 61131-3:2003 Table/feature	Feature description
10/1	BOOL data type
10/2	SINT data type
10/3	INT data type
10/4	DINT data type
10/5	LINT data type
10/6	USINT data type
10/7	UINT data type
10/8	UDINT data type
10/9	ULINT data type
10/10	REAL data type
10/11	LREAL data type
10/12	TIME data type
10/13	DATE data type
10/14	TIME_OF_DAY or TOD data type
10/15	DATE_AND_TIME or DT data type
10/16	STRING data type
10/17	SWORD data type
10/18	WORD data type
10/19	DWORD data type
10/20	LWORD data type
12/1	Directly derived data types
12/2	Enumerated data types
12/3	Subrange data types
12/4	Array data types
12/5	Structured data types
13	Standard default initial values
Subclause 2.5.1.3	User-declared functions (no table entry)
22/1	Type conversions (see Table 4)
22/2	TRUNC function
22/3	BCD_TO_** functions
22/4	*_TO_BCD functions
23/1-11	Standard functions of one numeric variable: ABS, SQRT, LN, LOG, EXP, SIN, COS, TAN, ASIN, ACOS, ATAN

IEC 61131-3:2003 Table/feature	Feature description
24/12n-18n	Standard named arithmetic functions: ADD, MUL, SUB, DIV, MOD, EXPT, MOVE
24/ 12s-15s, 17s,18s	Standard symbolic arithmetic functions: +, *, -, /, **, :=
25/1-4	Standard bit string functions: SHL, SHR, ROR, ROL
26/5s-8s	Standard named bitwise Boolean functions: AND, OR, XOR, NOT
26/5s-7s	Standard symbolic bitwise Boolean functions: &, >=1, =2k+1
27/1-4	Standard selection functions: SEL, MAX, MIN, LIMIT, MUX
28/5n-10n	Standard named comparison functions: GT, GE, EQ, LE, LT, NE
28/5s-10s	Standard symbolic comparison functions: >, >=, =, <=, <, <>
29/1-9	Standard character string functions: LEN, LEFT, RIGHT, MID, CONCAT, INSERT, DELETE, REPLACE, FIND
30/1-14	Standard functions of time data types: (see NOTE) ADD, SUB, MUL, DIV, CONCAT, DATE_AND_TIME_TO_TIME_OF_DAY, TIME_OF_DAY_TO_DATE_AND_TIME
31/1-4	Standard functions of enumerated data types: SEL, MUX, EQ, NE
32	Standard access mechanisms to function block I/O parameters
33/8a,8b,9a,9b	User-defined function blocks per IEC 61131-3:2003, 2.5.2.2, with graphical or textual rising or falling edge input options
34/1-3	Standard bistable function blocks: SR, RS, SEMA
35/1,2	Standard edge detection function blocks: R_EDGE, F_EDGE
36/1-3	Standard counter function blocks: CTU, CTD, CTUD
37/1,2a,3a, 4	Standard timer function blocks: TP, TON, TOF, RTC
55/1-17	Standard operators: (), function evaluation, **, -, NOT, *, /, MOD, +, -, <, >, <=, >=, =, <>, &, AND, XOR, OR
NOTE IEC 61131-3:2003, Table 30, limits the data types to which these operations apply.	

Table 3 – ST language elements

IEC 61131-3:2003 Table/feature	Feature description
55/1-17	Standard operators: (), function evaluation, **, -, NOT, *, /, MOD, +, -, <, >, <=, >=, =, <>, &, AND, XOR, OR
56/2	Function block invocation and output usage

Table 4 – Type conversion operations

Operation	Result	Error conditions
ANY_BIT_TO_ANY_BIT	See note 4	None
ANY_BIT_TO_ANY_INT	OUTmin + Sbk2k (note 5)	Result > OUTmax
ANY_BIT_TO_BOOL	FALSE if IN = 0 TRUE otherwise	None
ANY_BIT_TO_STRING	See note 6	None
ANY_DATE_TO_STRING	See note 7	None
ANY_INT_TO_BOOL	FALSE if IN = 0 TRUE otherwise	None
ANY_NUM_TO_ANY_INT	IN (note 8)	(IN > OUTmax) or (IN < OUTmin)
ANY_NUM_TO_ANY_REAL	IN	See note 9
ANY_NUM_TO_DATE	See note 10	
ANY_NUM_TO_STRING	See note 11	
ANY_NUM_TO_TIME	See note 12	
ANY_NUM_TO_TOD	See note 13	
ANY_REAL_TO_BOOL	FALSE if IN = 0,0 TRUE otherwise	None
BOOL_TO_ANY_BIT BOOL_TO_ANY_INT	0 if IN = FALSE 1 if IN = TRUE	None
BOOL_TO_ANY_REAL	0,0 if IN = FALSE 1,0 if IN = TRUE	None
BOOL_TO_STRING	'FALSE' if IN = FALSE 'TRUE' if IN = TRUE	None
DATE_TO_ANY_NUM	See note 14	Result > OUTmax
STRING_TO_ANY	Converted data	See note 15
TIME_TO_ANY_NUM	See note 16	Result > OUTmax
TOD_TO_ANY_NUM	See note 17	Result > OUTmax

NOTE 1 Use of the generic data types ANY_NUM, ANY_REAL, ANY_INT, and ANY_BIT defined in IEC 61131-3:2003, 2.3.2, is intended to imply a family of conversions. For instance, the conversion BOOL_TO_ANY_REAL is intended to imply BOOL_TO_REAL and BOOL_TO_LREAL.

NOTE 2 IN refers to the value of the input variable of the type conversion function.

NOTE 3 OUT_{min} and OUT_{max} refer to the minimum and maximum values of the output data type of the conversion function, as defined in Clause 5.

NOTE 4 In conversions of bit string types, if the number of bits in the input variable IN is less than the number of the bits in the output variable OUT, the bits of the input is copied to the corresponding least significant bits of the result and the remainder of the result is zero-filled. If the number of bits of IN is greater than the number of bits of OUT, the least significant bits of IN is copied to the corresponding bits of the result. For instance:

SWORD_TO_WORD(16#FF) = 16#00FF
and
WORD_TO_SWORD (16#0FF0) = 16#F0

NOTE 5 Bit numbering in this expression is as specified in IEC 61158-6-2.

NOTE 6 The result of conversion of a bit string variable to type STRING consists of a string containing the base 16 literal representation of the variable value, as defined in IEC 61131-3:2003, 2.2.1, in characters taken from the ISO/IEC 646 character set.

NOTE 7 The result of conversion of a date and/or time of day variable to type STRING consists of a string containing the literal representation of the variable value, as defined in IEC 61131-3:2003, 2.2.1, in characters taken from the ISO/IEC 646 character set.

NOTE 8 Conversion of REAL and LREAL to integer types is accomplished by rounding as specified in ISO/IEC/IEEE 60559.

NOTE 9 Rounding errors can occur if the number of significant bits in the input variable is larger than the number of significant bits in the output floating-point representation. Also, range errors of the type noted for ANY_NUM_TO_ANY_INT can occur in LREAL_TO_REAL.

NOTE 10 Conversion of a variable of a numeric type to DATE has the same result as conversion of the variable to UINT, with the result being interpreted as the number of days since 1972-01-01.

NOTE 11 Conversion of a variable of a numeric type to type STRING consists of a string containing the literal representation of the variable value, as defined in IEC 61131-3:2003, 2.2.1, in characters taken from the ISO/IEC 646 character set.

NOTE 12 Conversion of a variable of a numeric type to TIME has the same result as conversion of the variable to DINT, with the result being interpreted as a duration in milliseconds.

NOTE 13 Conversion of a variable of a numeric type to TOD has the same result as conversion of the variable to UDINT, with the result being interpreted as a time since midnight in milliseconds.

NOTE 14 Conversion of a variable of type DATE to a numerical type is the same as the conversion of a variable of type UINT to the corresponding numerical type, with the result being the numerical equivalent of the days since 1972-01-01.

NOTE 15 It is an error if the STRING data to be converted is not in the format for external representation of the output data type as specified in IEC 61131-3:2003, 2.2, or if the result of the conversion is outside the range $\{OUT_{min}, OUT_{max}\}$.

NOTE 16 Conversion of a variable of type TIME to a numerical type is the same as the conversion of a variable of type DINT to the corresponding numerical type, with the result being the numerical equivalent of the corresponding time interval expressed in milliseconds.

NOTE 17 Conversion of a variable of type TOD (TIME_OF_DAY) to a numerical type is the same as the conversion of a variable of type UDINT to the corresponding numerical type, with the result being the numerical equivalent of the time since midnight expressed in milliseconds.

6.1.5.3.2 Implementation dependant parameters

The values of implementation dependent parameters from IEC 61131-3:2003, Table D.1, are as shown in Table 5.

NOTE Values of other implementation dependent parameters are defined in other standards or in the specifications of individual functional units as appropriate.

Table 5 – Values of implementation-dependent parameters

Subclause of IEC 61131-3:2003	Parameter	Value
2.2.3.1	Range of values of duration	Same as LINT in microseconds
2.3.1	Range of values for variables of type TIME	Same as DINT in milliseconds
	Precision of representation of seconds in types TIME_OF_DAY and DATE_AND_TIME	1 ms
2.3.3.1	Maximum number of enumerated values	256
2.3.3.2	Default maximum length of STRING variables	256
	Maximum allowed length of STRING variables	65 536
2.4.1.2	Maximum number of subscripts	8
	Maximum range of subscript values	0 – 255
	Maximum number of levels of structures	8
2.5.1.5	Maximum inputs of extensible functions	8
2.5.1.5.1	Effects of type conversions on accuracy	As defined in Table 4
2.5.1.5.2	Accuracy of functions of one variable	As defined in ISO/IEC/IEEE 60559
2.5.2.3.3	PVmin, PVmax of counters	0, 65 535

6.1.5.3.3 Language extensions

The extensions to IEC 61131-3 defined in this part are listed in Table 6. When these extensions are used in a particular device, the subclause references in this table shall be replaced by references to the descriptions of the corresponding extensions in the functional unit's documentation.

Table 6 – Extensions to IEC 61131-3:2003

Subclause	Description
2.1.1	ITIME data type FTIME data type LTIME data type STRING2 data type STRINGN data type SHORT_STRING data type STRINGI data type EPATH data type
2.1.4	Structured bit string types
2.2	Operations on STRING2 variables
2.2.4.1	Numbered bit string access
2.2.4.2	Structured bit string access