

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Industrial communication networks –Fieldbus specifications –
Part 5-28: Application layer service definition – Type 28 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 5-28: Définition des services de la couche application –
Éléments de type 28**



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2023 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

IEC Secretariat
3, rue de Varembe
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
info@iec.ch
www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigendum or an amendment might have been published.

IEC publications search - webstore.iec.ch/advsearchform

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee, ...). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and once a month by email.

IEC Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: sales@iec.ch.

IEC Products & Services Portal - products.iec.ch

Discover our powerful search engine and read freely all the publications previews. With a subscription you will always have access to up to date content tailored to your needs.

Electropedia - www.electropedia.org

The world's leading online dictionary on electrotechnology, containing more than 22 300 terminological entries in English and French, with equivalent terms in 19 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

A propos de l'IEC

La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC

Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Recherche de publications IEC -

webstore.iec.ch/advsearchform

La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études, ...). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et une fois par mois par email.

Service Clients - webstore.iec.ch/csc

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: sales@iec.ch.

IEC Products & Services Portal - products.iec.ch

Découvrez notre puissant moteur de recherche et consultez gratuitement tous les aperçus des publications. Avec un abonnement, vous aurez toujours accès à un contenu à jour adapté à vos besoins.

Electropedia - www.electropedia.org

Le premier dictionnaire d'électrotechnologie en ligne au monde, avec plus de 22 300 articles terminologiques en anglais et en français, ainsi que les termes équivalents dans 19 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Industrial communication networks –Fieldbus specifications –
Part 5-28: Application layer service definition – Type 28 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 5-28: Définition des services de la couche application –
Éléments de type 28**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 25.040

ISBN 978-2-8322-6583-3

**Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**

CONTENTS

FOREWORD.....	5
INTRODUCTION.....	7
1 Scope.....	8
1.1 Overview.....	8
1.2 Specifications	9
1.3 Conformance	9
2 Normative references	9
3 Terms, definitions, symbols, abbreviated terms and conventions	10
3.1 Referenced terms and definitions.....	10
3.1.1 ISO/IEC 7498-1 terms.....	10
3.1.2 ISO/IEC 8822 terms.....	10
3.1.3 ISO/IEC 9545 terms.....	11
3.1.4 ISO/IEC 8824 terms.....	11
3.2 Additional terms and definitions for this IEC 61158-5-28	11
3.3 Abbreviations and symbols	12
3.4 Conventions.....	13
3.4.1 Overview	13
3.4.2 Conventions for class definitions	13
3.4.3 Conventions for service definitions	14
4 Concepts	15
5 Data type ASE	15
5.1 General.....	15
5.1.1 Overview	15
5.1.2 Overview of basic types	16
5.1.3 Overview of fixed-length types.....	16
5.1.4 Overview of constructed types.....	17
5.2 AL defined data types	17
5.2.1 Fixed length types	17
5.2.2 String type	19
5.2.3 Structure type.....	20
6 Type 28 communication model specification	25
6.1 General.....	25
6.2 Protocol stack for Type 28 fieldbus	26
6.3 Data mapping modelling.....	27
6.4 Overview of Type 28 communication model	29
6.4.1 General	29
6.4.2 P/S	29
6.4.3 C/S.....	30
6.5 AL related management information table	30
6.5.1 General	30
6.5.2 Time management information table	31
6.5.3 Network management information table	32
6.6 ASEs	33
6.6.1 Overview	33
6.6.2 RT data ASE.....	34
6.6.3 nRT data ASE.....	36

6.6.4	Time ASE	38
6.6.5	Resource ASE	41
6.6.6	Addressing ASE	43
6.6.7	Management ASE	48
6.6.8	Virtualization ASE	55
6.7	Application service	58
6.7.1	Overview	58
6.7.2	Read service	58
6.7.3	Write service	58
Annex A (informative)	Example of service data mapping DTU message	60
Annex B (informative)	Example of OPC UA data model and Type 28 data model mapping	61
Annex C (informative)	Example of RTA service processing	62
Annex D (informative)	Type 28 virtualization solution example	63
Bibliography		66
Figure 1	– Data type class hierarchy example	16
Figure 2	– BITMAP data type	20
Figure 3	– AL in fieldbus protocol stack architecture	26
Figure 4	– Type 28 AL protocol architecture	26
Figure 5	– Data mapping modelling object structure	27
Figure 6	– Diagram of mapping data buffers and service data variables	28
Figure 7	– Diagram of P/S model	29
Figure 8	– P/S model of PUSH mode	30
Figure 9	– P/S model of PULL mode	30
Figure 10	– Diagram of C/S communication model	30
Figure 11	– RT data transmission model	35
Figure 12	– Diagram of nRT data request response model	37
Figure 13	– Diagram of nRT data AP interaction based on C/S communication model	38
Figure 14	– Diagram of nRT data AP interaction based on P/S communication model	38
Figure 15	– Time synchronization application interaction process	40
Figure 16	– Time query process	41
Figure 17	– Resource AP interaction based on C/S communication model	43
Figure 18	– Resource ASE local service function diagram	43
Figure 19	– NETWORKID identification field structure	44
Figure 20	– NETWORKID interaction process diagram	46
Figure 21	– Diagram of MAC mapping table service function	46
Figure 22	– IP mapping table service function diagram	47
Figure 23	– Type 28 AL data message mapped to IP payload data payload	47
Figure 24	– Type 28 AL protocol header and IP protocol header mapping	48
Figure 25	– IP data message is mapped to valid data of Type 28 AL data message	48
Figure 26	– Static configuration initialization process	51
Figure 27	– Join the network interaction process	52
Figure 28	– Passively exit the network interaction process	53

Figure 29 – Actively exit network interaction process	54
Figure 30 – Diagram of diagnostic object mapping	54
Figure 31 – Diagram of the logging process	55
Figure 32 – A physical bus divided into two virtual bus domain diagrams	56
Figure 33 – Diagram of virtualized AP interaction based on C/S communication model	57
Figure 34 – Virtualization ASE local implementation diagram	58
Figure A.1 – Data transmission unit message content example	60
Figure B.1 – OPC UA data model and mapping of Type 28 data model	61
Figure C.1 – Example of RTA service processing	62
Figure D.1 – Diagram of Type 28 industrial bus service deployment	63
Figure D.2 – Diagram of networking topology based on the virtualized bus	63
Figure D.3 – Virtual network topology based on logical business function RT1	64
Table 1 – TIMEV type data coding	17
Table 2 – TIMEDATE type coding	18
Table 3 – TIMEOFDAY type coding	18
Table 4 – TIMEDIFFER type coding	19
Table 5 – VISIBLESTRING type data coding	19
Table 6 – Description of time information structure TIMEINFO_S	20
Table 7 – Description of clock option information structure CLOCK_OPTION_INFO_S	21
Table 8 – Description of network configuration parameter structure NETWORK_CFG_PARA_S	22
Table 9 – Description of MAC mapping table NETWORKID_MAC_MAP_TABLE_S	23
Table 10 – Description of IP mapping table structure IP_MAP_TABLE_S	24
Table 11 – Service data mapping table parameter list	29
Table 12 – Time management information table	31
Table 13 – Network management information table	32
Table 14 – Communication models supported by Type 28 AL ASEs	33
Table 15 – Application service interface return value list	33
Table 16 – RT data service interface parameters	34
Table 17 – nRT data service interface parameters	36
Table 18 – Timing service interface parameters	39
Table 19 – Resource service interface parameters	42
Table 20 – Addressing service interface parameters	45
Table 21 – Management service interface parameters	49
Table 22 – Virtual service interface parameters	56
Table 23 – Read service parameters	58
Table 24 – Write service parameters	59
Table 25 – Application service errcode	59
Table D.1 – Virtual bus environment node NETWORKID list	65

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**INDUSTRIAL COMMUNICATION NETWORKS –
FIELDBUS SPECIFICATIONS –****Part 5-28: Application layer service definition –
Type 28 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 the IEC 61784-1 series and the IEC 61784-2 series.

IEC 61158-5-28 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation. It is an International Standard.

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

Draft	Report on voting
65C/1206/FDIS	65C/1235/RVD

Full information on the voting for its approval can be found in the report on voting indicated in the above table.

The language used for the development of this International Standard is English.

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1 and ISO/IEC Directives, IEC Supplement, available at www.iec.ch/members_experts/refdocs. The main document types developed by IEC are described in greater detail at www.iec.ch/publications.

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 document will remain unchanged until the stability date indicated on the IEC website under webstore.iec.ch in the data related to the specific document. At this date, the document will be

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

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 can 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 5-28: Application layer service definition – Type 28 elements

1 Scope

1.1 Overview

The fieldbus Application Layer (FAL) provides user programs with a means to access the fieldbus communication environment. In this respect, the FAL can be considered 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 28 fieldbus. The term "time-critical" is used to represent the presence of a time-window, in which one or more specified actions are required to be completed with some defined level of certainty.

This document defines in an abstract way the externally visible service provided by the different Types of the fieldbus Application Layer in terms of

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

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

- the FAL user at the boundary between the user and the Application Layer of the Fieldbus Reference Model, and
- 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 IEC 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 to request and response 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 behavioural 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 behaviour. 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 the IEC 61158-6 series.

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

- the sizes and octet ordering of various multi-octet service parameters, and
- 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 fulfils the Type 28 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 the IEC 61784-1 series and the IEC 61784-2 series 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 61158-1:2023, *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-4-28:2023, *Industrial communication networks – Fieldbus specifications – Part 4-28: Data-link layer protocol specification – Type 28 elements*

IEC 61158-6-28:2023, *Industrial communication networks – Fieldbus specifications – Part 6-28: Application layer protocol specification – Type 28 elements*

ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

ISO/IEC 7498-3:1997, *Information technology – Open Systems Interconnection – Basic Reference Model: Naming and addressing*

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

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

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

ISO/IEC 10731:1994, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services*

ISO/IEC/IEEE 8802-3:2021, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 3: Standard for Ethernet*

IETF RFC 791, J. Postel, *Internet Protocol*, September 1981, available at <https://www.rfc-editor.org/info/rfc791> [viewed 2022-02-18]

IETF RFC 2460, S. Deering and R. Hinden, *“Internet Protocol, Version 6 (IPv6) Specification”*, December 1998, available at <https://www.rfc-editor.org/info/rfc2460> [viewed 2022-02-18]

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

3.1 Referenced terms and definitions

3.1.1 ISO/IEC 7498-1 terms

For the purposes of this document, the following terms given in ISO/IEC 7498-1 apply:

- application entity
- application process
- application protocol data unit
- application service element
- application entity invocation
- application process invocation
- application transaction
- real open system
- transfer syntax

3.1.2 ISO/IEC 8822 terms

For the purposes of this document, the following terms given in ISO/IEC 8822 apply:

- abstract syntax

- presentation context

3.1.3 ISO/IEC 9545 terms

For the purposes of this document, the following terms given in ISO/IEC 9545 apply:

- application-entity-invocation
- application-entity-type
- application-process-invocation
- application-process-type
- application-service-element

3.1.4 ISO/IEC 8824 terms

For the purposes of this document, the following terms given in ISO/IEC 8824 apply:

- object identifier
- type

3.2 Additional terms and definitions for this IEC 61158-5-28

The following terms and definitions apply to this document.

3.2.1

clock query

get clock information from the management node or terminal node device

3.2.2

clock synchronization

clock calibration of the terminal node device

3.2.3

control device

device that controls all field devices for logical operations, timing, calculations, etc.

3.2.4

cyclic

events which repeat in a regular and repetitive manner

3.2.5

cyclic time

time of cyclic processing of a device or module

3.2.6

field device

physical entity that is connected to the process or to factory equipment and has at least one signalling element that communicates with other signalling element(s) via a cable

3.2.7

management node

device for allocating and managing Type 28 network physical communication resources

3.2.8

non-RT Data

data insensitive to time deterministic requirements

3.2.9

RT Data

data sensitive to time deterministic requirements

3.2.10

terminal node

device in Type 28 network that communicates based on allocated physical communication resources

3.3 Abbreviations and symbols

AL	Application layer
ALCE	Application layer clock entity
ALDE	Application layer data entity
ALME	Application layer management entity
ALS	Application layer service
AE	Application entity
AP	Application process
APO	AP object
AR	Application relationship
ASE	Application service elements
C/S	Client/Server
DL-	Data-link layer (as a prefix)
DLCE	Data-link clock entity
DLDE	Data-link data entity
DLE	Data-link entity
DLL	Data-link layer
DLS	Data-link service
DLP	Data-link protocol
DLS-user	Data-link service user
DTC	Data transmission channel
DTS	Data transmission service
DTU	Data transmission unit
DLSDU	Data-link service data unit
DLPDU	Data-link protocol data unit
MAC	Medium access control
MN	Management node
NodeID	Node identifier
nRT	non-Real-Time
PhL	Physical layer
RT	Real-Time
RTA	Real-Time acyclic
RTC	Real-Time cyclic
TN	Terminal node

3.4 Conventions

3.4.1 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, including its class specification and its service specification.

The class specification defines the attributes of the class. The service specification defines the services that are provided by the ASE.

3.4.2 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
3.1	(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.
 - 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 a 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.

3.4.3 Conventions for service definitions

3.4.3.1 General

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

3.4.3.2 Service parameters

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

The service specifications of this document use 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

- a) parameter name,
- b) request primitive,
- c) indication primitive,
- d) response primitive, and

e) 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 can be provided or not depending on dynamic usage of the service user. When not provided, a default value for the parameter is assumed.

— (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.

3.4.3.3 Service procedures

The procedures are defined in terms of

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

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

4 Concepts

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

5 Data type ASE

5.1 General

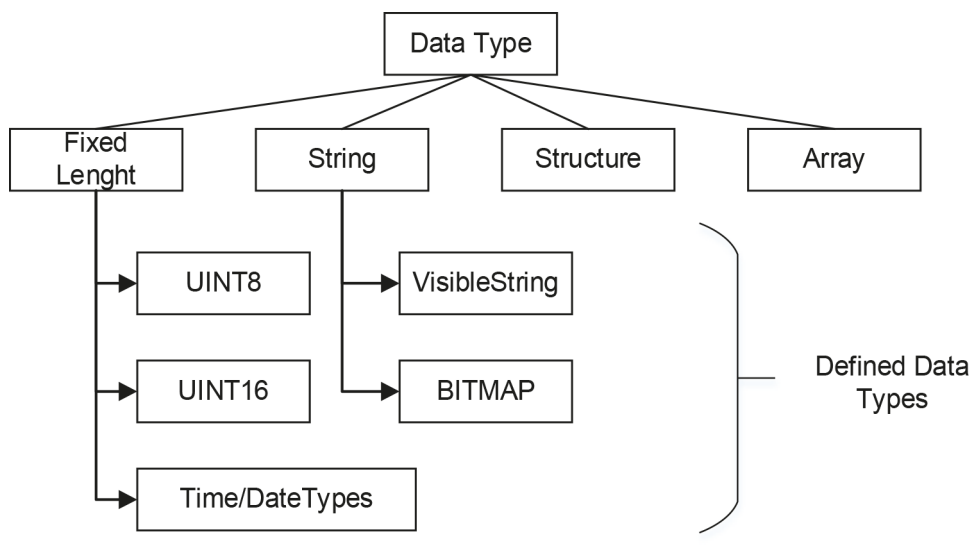
5.1.1 Overview

In the communication Type 28 FAL, Data type ASE complies with IEC 61158-1, Clause 10, and the other protocol-specific data types are not defined in this document.

Fieldbus data types specify the machine independent syntax for the data conveyed by each layer services. The fieldbus application supports the definition and transmission of basic and constructed data types.

Basic types are atomic data types that cannot be decomposed into more elemental types. Constructed data types are composed of basic types and other constructed types. Their complexity and the depth of nesting are not constrained by this document.

Data types are defined as instances of the data type class, as shown in Figure 1. Only a subset of the data types defined in Clause 5 are shown in Figure 1. Defining new types is accomplished by providing a new numeric ID and supplying values for the attributes defined for the data type class.



IEC

Figure 1 – Data type class hierarchy example

The data type definitions in Figure 1 are represented as a class/format/instance structure beginning with data type class entitled "Data Type". The formats for data types are defined by the data type class and represented in Figure 1.

The basic data classes are used to define fixed length and bit string data types. Standard types taken from ISO/IEC 8824 are referred to as simple data types. Other standard basic data types are defined specifically for fieldbus applications and are referred to as specific types.

The constructed types specified in this document are strings, arrays and structures. There are no standard types defined for arrays and structures.

5.1.2 Overview of basic types

Most basic types are defined from a set of ISO/IEC 8824 types (simple types). Some ISO/IEC 8824 types have been extended for fieldbus specific use (specific types).

Simple types are ISO/IEC 8824 universal types. They are defined in this document to provide them with fieldbus class identifiers.

Specific types are basic types defined specifically for use in the fieldbus environment. They are defined as simple class subtypes.

Basic types have a constant length. Two variations are defined, one for defining data types whose length is an integral number of octets, and one for defining data types whose length is a number of bits.

NOTE Boolean, Integer, OctetString, VisibleString, and UniversalTime are defined in this document for the purpose of assigning fieldbus class identifiers to them. This document does not change their definitions as specified in ISO/IEC 8824.

5.1.3 Overview of fixed-length types

The length of fixed-length types is an integral number of octets.

5.1.4 Overview of constructed types

5.1.4.1 Strings

A string is composed of an ordered set, variable in number, of homogeneously typed fixed-length elements.

5.1.4.2 Arrays

An array is composed of an ordered set of homogeneously typed elements. There are no restrictions on the data type of array elements placed, but each element shall be of the same type. Once defined, the number of elements in an array can be changed, or not.

5.1.4.3 Structures

A structure is made of an ordered set of heterogeneously-typed elements called fields. Like arrays, this document does not restrict the data type of fields. However, the fields within a structure do not have to be of the same type.

5.1.4.4 Nesting level

This document permits arrays and structures to contain other arrays and structures. It places no restriction on the number of nesting levels allowed.

When an array or structure contains constructed elements, access to a single element in its entirety is provided. Access to sub elements of the constructed element is also provided.

5.2 AL defined data types

5.2.1 Fixed length types

5.2.1.1 Date/Time types

5.2.1.1.1 Time type (TIMEV)

The time data type is an unsigned integer of 64 bits, which represents the time incremented with 1 ns. Table 1 shows the coding.

Table 1 – TIMEV type data coding

octet	bit							
	7	6	5	4	3	2	1	0
1	2^{63}	2^{62}	2^{61}	2^{60}	2^{59}	2^{58}	2^{57}	2^{56}
2	2^{55}	2^{54}	2^{53}	2^{52}	2^{51}	2^{50}	2^{49}	2^{48}
3	2^{47}	2^{46}	2^{45}	2^{44}	2^{43}	2^{42}	2^{41}	2^{40}
4	2^{39}	2^{38}	2^{37}	2^{36}	2^{35}	2^{34}	2^{33}	2^{32}
5	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}
6	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}
7	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8
8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

CLASS:

DATA TYPE

ATTRIBUTES:

- | | | |
|---|------------------------------|---------|
| 1 | Data Type Numeric Identifier | = 16 |
| 2 | Data Type Name | = TIMEV |

3 Format = FIXED LENGTH

4.1 Octet Length = 8

5.2.1.1.2 Date type coding (TIMEDATE)

The TIMEDATE data type length is seven octets, indicating date and time. Table 2 shows the meaning of each octet.

Table 2 – TIMEDATE type coding

octet	Bit								Description
	7	6	5	4	3	2	1	0	
1	2 ¹⁵	2 ¹⁴	2 ¹³	2 ¹²	2 ¹¹	2 ¹⁰	2 ⁹	2 ⁸	Indicates the millisecond value less than a minute
2	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	
3	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	Indicates the minute value less than an hour
4	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	Indicates the hour values less than a day
5	2 ²	2 ¹	2 ⁰	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	The first 3 bits means the day of the week, and the last 5 bits is the day of the current month
6	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	Indicates the month
7	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	Indicates the year. The value range from 0 to 50 corresponds to range from 2000 to 2050, and the value range from 51 to 99 corresponds to range from 1951 to 1999.

CLASS: DATA TYPE

ATTRIBUTES:

1	Data Type Numeric Identifier	= 17
2	Data Type Name	= TIMEDATE
3	Format	= FIXED LENGTH
4.1	Octet Length	= 7

5.2.1.1.3 Time of day type coding (TIMEOFDAY)

TIMEOFDAY data type length is six octets and indicates the time of day. Table 3 shows the meaning of each octet.

Table 3 – TIMEOFDAY type coding

octet	Bit								Description
	7	6	5	4	3	2	1	0	
1	2 ³¹	2 ³⁰	2 ²⁹	2 ²⁸	2 ²⁷	2 ²⁶	2 ²⁵	2 ²⁴	Indicates the nanosecond count starting at 0:00 midnight
2	2 ²³	2 ²²	2 ²¹	2 ²⁰	2 ¹⁹	2 ¹⁸	2 ¹⁷	2 ¹⁶	
3	2 ¹⁵	2 ¹⁴	2 ¹³	2 ¹²	2 ¹¹	2 ¹⁰	2 ⁹	2 ⁸	
4	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	
5	2 ¹⁵	2 ¹⁴	2 ¹³	2 ¹²	2 ¹¹	2 ¹⁰	2 ⁹	2 ⁸	Indicates the number of days that have been counted since 1 January 1984
6	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	

CLASS: DATA TYPE**ATTRIBUTES:**

1	Data Type Numeric Identifier	= 18
2	Data Type Name	= TIMEOFDAY
3	Format	= FIXED LENGTH
4.1	Octet Length	= 6

5.2.1.1.4 Time difference type coding (TIMEDIFFER)

TIMEDIFFER data type length is six octets and indicates time difference. Table 4 shows the meaning of each octet.

Table 4 – TIMEDIFFER type coding

octet	Bit								Description
	7	6	5	4	3	2	1	0	
1	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	Indicates the deviation of nanosecond values less than a day.
2	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	
3	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8	
4	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
5	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8	Indicates the number of days of deviation
6	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	

CLASS: DATA TYPE**ATTRIBUTES:**

1	Data Type Numeric Identifier	= 19
2	Data Type Name	= TIMEDIFFER
3	Format	= FIXED LENGTH
4.1	Octet Length	= 6

5.2.2 String type**5.2.2.1 Visible string type (VISIBLESTRING)**

VISIBLESTRING type is a set of octets in order. The length is not fixed, and it is 1 to N octet. Table 5 shows the string coding. For strings of N octets, the MSB of the highest valid octet of this type of data is transmitted at first.

Table 5 – VISIBLESTRING type data coding

octet	bit							
	7	6	5	4	3	2	1	0
1	2^{8N-1}	2^{8N-2}	2^{8N-3}	2^{8N-4}	2^{8N-5}	2^{8N-6}	2^{8N-7}	2^{8N-8}
2	2^{8N-9}	2^{8N-10}	2^{8N-11}	2^{8N-12}	2^{8N-13}	2^{8N-14}	2^{8N-15}	2^{8N-16}
...	...	...	...	...	...	...	...	...
N	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

CLASS: DATA TYPE**ATTRIBUTES:**

1	Data Type Numeric Identifier	= 32
2	Data Type Name	= VISIBLESTRING
3	Format	= STRING
4.1	Octet Length	= 1 to n

5.2.2.2 Bitmap type coding (BITMAP)

BITMAP type length ranges in value from 2 to 256 octets, including Width and Mask_Info information.

Figure 2 shows the bitmap type coding.

NOTE Width has 1octet, its value indicates the number of the octets of Mask_Info.

NOTE Mask_Info represented by bit. The octet length is Width value and the bit length is Width × 8.

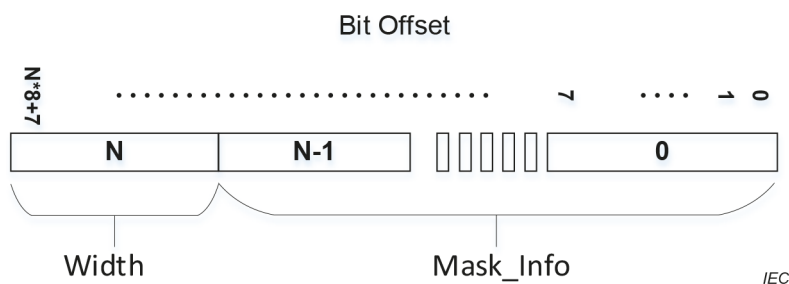


Figure 2 – BITMAP data type

CLASS:	DATA TYPE
ATTRIBUTES:	
1	Data Type Numeric Identifier = 33
2	Data Type Name = BITMAP
3	Format = STRING
4.1	Octet Length = 1 to n

5.2.3 Structure type

5.2.3.1 Time information structure

- Structure name: TIMEINFO_S
- Number of structural members: 5
- Description of structure member: see Table 6.

Table 6 – Description of time information structure TIMEINFO_S

Member name	Type of Data	Description
Time_fmt	UINT8	Identify the valid bits of three time formats based on bits, bit0, bit1, and bit2 respectively. Identify whether the time values are valid based on hour, minute and second, millisecond and microsecond and second. When the corresponding position is 1, it means that it is valid, otherwise it is invalid.
Time_ns	TIMEV	Time count based on nanosecond.
TimeDate	TIMEDATE	Convert to date value of year, month, day, minute, minute, second, and precision is millisecond.
TimeOfDay	TIMEOFDAY	The number of days based on nanosecond.
TimeDiffer	TIMEDIFFER	Time offset value based on nanosecond.

CLASS: DATA TYPE**ATTRIBUTES:**

2	Data Type Name	= TIMEINFO_S
5	Format	= STRUCTURE
5.1	Number of Fields	= 5
5.2	List of Fields	
5.2.1	Field Name	= Time_fmt
5.2.2	Field Data Type	UINT8
5.2.3	Field Name	= Time_ns
5.2.4	Field Data Type	TIMEV
5.2.5	Field Name	= TimeDate
5.2.6	Field Data Type	TIMEDATE
5.2.7	Field Name	= TimeOfDay
5.2.8	Field Data Type	TIMEOFDAY
5.2.9	Field Name	= TimeDiffer
5.2.10	Field Data Type	TIMEDIFFER

5.2.3.2 Clock option information structure

- Structure name: CLOCK_OPTION_INFO_S
- Number of structural members: 5
- Description of structure member, see Table 7.

Table 7 – Description of clock option information structure CLOCK_OPTION_INFO_S

Member name	Type of Data	Description
Tid_cnt	UINT8	Number of clocks on the device or module
Tid_val	UINT8	Clock ID value
Tid_attr	UINT8	The control-field attribute of the clock, 0 is the control clock (list), 1 is the field clock (list)
Tcd_lst	UINT8	TCD command for the corresponding clock, the value range of TCD is identical to IEC 61158-6-28, Clause 4
Tid_errcode	UINT8	A clock error exception encoding list

CLASS: DATA TYPE**ATTRIBUTES:**

2	Data Type Name	= CLOCK_OPTION_INFO_S
5	Format	= STRUCTURE
5.1	Number of Fields	= 5
5.2	List of Fields	
5.2.1	Field Name	= Tid_cnt
5.2.2	Field Data Type	UINT8
5.2.3	Field Name	= Tid_val
5.2.4	Field Data Type	UINT8
5.2.5	Field Name	= Tid_attr
5.2.6	Field Data Type	UINT8
5.2.7	Field Name	= Tcd_lst
5.2.8	Field Data Type	UINT8
5.2.9	Field Name	= Tid_errcode

5.2.10

Field Data Type

UINT8

5.2.3.3 Network configuration parameter structure

- Structure name: NETWORK_CFG_PARA_S
- Number of structural members: 15
- Description of structure member: see Table 8.

**Table 8 – Description of network configuration parameter structure
NETWORK_CFG_PARA_S**

Member name	Type of Data	Description
Network_id	UINT32	Identify the NETWORKID (see 6.6.6.1.2) of device or module
Ip_switch	UINT8	Whether the IPv4 (defined in IETF RFC 791) address or IPv6 (defined in IETF RFC 2460) address is supported by bit. Bit 0: whether IPv4 is supported or not. Bit 1: whether IPv6 is supported, other bits are not used yet; Corresponding bit = 1 indicates support.
Multicast_id0 to 15	UINT32	NETWORKID of the multicast type, multicast group to which the identified device or module belongs. Each device or module belongs to maximum of 16 multicast groups.
Mac_addr	ARRAY	Unique MAC (defined in ISO/IEC/IEEE 8802-3) address of the device or module
Nodeid	UINT8	Node ID address of the device
Dctime	UINT16	Data calculation cycle time
Bctime	UINT16	Bus service cycle time
Wctime	UINT16	Bus service data write cycle time
Rctime	UINT16	Bus service data read cycle time
Ctime	UINT16	Bus service competition cycle time
Totcnt	UINT16	Number of timeouts allowed by the device or module
Rt_data_addr	UINT32	Start address of RT data buffer
Rt_data_size	UINT32	Size of RT data buffer
nRt_data_addr	UINT32	Start address of nRT buffer
nRt_data_size	UINT32	Size of nRT data buffer

CLASS:

DATA TYPE

ATTRIBUTES:

2	Data Type Name	= NETWORK_CFG_PARA_S
5	Format	= STRUCTURE
5.1	Number of Fields	= 15
5.2	List of Fields	
5.2.1	Field Name	= Network_id
5.2.2	Field Data Type	UINT32
5.2.3	Field Name	= Ip_switch
5.2.4	Field Data Type	UINT8
5.2.5	Field Name	= Multicast_id0 to Multicast_id15
5.2.6	Field Data Type	UINT8
5.2.7	Field Name	= Mac_addr
5.2.8	Field Data Type	ARRAY
5.2.9	Field Name	= Nodeid

5.2.10	Field Data Type	UINT8
5.2.11	Field Name	= Dctime
5.2.12	Field Data Type	UINT16
5.2.13	Field Name	= Bctime
5.2.14	Field Data Type	UINT16
5.2.15	Field Name	= Wctime
5.2.16	Field Data Type	UINT16
5.2.17	Field Name	= Rctime
5.2.18	Field Data Type	UINT16
5.2.19	Field Name	= Ctime
5.2.20	Field Data Type	UINT16
5.2.21	Field Name	= Totcnt
5.2.22	Field Data Type	UINT16
5.2.23	Field Name	= Rt_data_addr
5.2.24	Field Data Type	UINT32
5.2.25	Field Name	= Rt_data_size
5.2.26	Field Data Type	UINT32
5.2.27	Field Name	= nRt_data_addr
5.2.28	Field Data Type	UINT32
5.2.29	Field Name	= nRt_data_size
5.2.30	Field Data Type	UINT32
6	Constraint	Format = ARRAY
6.1	Number of Array Elements	6
6.2	Array Element Data Type	UINT8

5.2.3.4 MAC mapping table structure

- Structure name: NETWORKID_MAC_MAP_TABLE_S
- Number of structural members: 7
- Description of structure member: see Table 9.

Table 9 – Description of MAC mapping table NETWORKID_MAC_MAP_TABLE_S

Member name	Type of Data	Description
Index	UINT16	Entry index
NodeID	UINT8	Node address
RoleID	UINT8	Control-field attribute of device or module, 0x00 is the control device, 0x01 is the field device, 0x02 is the unknown. Other values are reserved.
IP_Flag	UINT8	The device or module supports which type of IP. Bit 0 identifies IPv4 (defined in IETF RFC 791). Bit 1 identifies IPv6 (defined in IETF RFC 2460). The corresponding bit position 1 indicates supporting. Other bits are reserved.
StatusID	UINT8	The status ID of the device or module. The value 0x00 indicates the initialization status is OK, and the value 0x01 indicates the normal working status. Other values are reserved.
Network_ID	UINT32	Unique NETWORKID ID of the device or module
MacAddr	ARRAY	MAC (defined in ISO/IEC/IEEE 88023-3) address of the device or module

CLASS:		DATA TYPE
ATTRIBUTES:		
2	Data Type Name	= NETWORKID_MAC_MAP_TABLE_S
5	Format	= STRUCTURE
5.1	Number of Fields	= 7
5.2	List of Fields	
5.2.1	Field Name	= Index
5.2.2	Field Data Type	UINT16
5.2.3	Field Name	= NodeID
5.2.4	Field Data Type	UINT8
5.2.5	Field Name	= RoleID
5.2.6	Field Data Type	UINT8
5.2.7	Field Name	= IP_Flag
5.2.8	Field Data Type	UINT8
5.2.9	Field Name	= StatusID
5.2.10	Field Data Type	UINT8
5.2.11	Field Name	= Network_ID
5.2.12	Field Data Type	UINT32
5.2.13	Field Name	= MacAddr
5.2.14	Field Data Type	ARRAY
6	Constraint	Format = ARRAY
6.1	Number of Array Elements	= 6
6.2	Array Element Data Type	UINT8

5.2.3.5 Mapping table structure

- Structure name: IP_MAP_TABLE_S
- Number of structural members: 9
- Description of structure member: see Table 10.

Table 10 – Description of IP mapping table structure IP_MAP_TABLE_S

Member name	Type of Data	Description
Index	UINT16	Entry index
Stated	UINT8	Value that identifies the status of the entry. 0x00 is invalid; 0x01 is valid; 0x02 is pending. Other values are reserved.
Network_id	UINT32	Unique NETWORKID of the device or module
ipv4_addr	UINT32	IPv4 address (defined in IETF RFC 791).
ipv4_mask	UINT32	IPv4 mask (defined in IETF RFC 791).
ipv6_addr	UINT128	IPv6 address (defined in IETF RFC 2460)
ipv6_mask	UINT128	IPv6 mask (defined in IETF RFC 2460)
Dmac_addr	ARRAY	Unique physical MAC address of device or module
Vmac_addr	ARRAY	Virtual MAC address of device or module. One physical device has one or more sub-devices or sub-modules, and each sub-device or sub-module uniquely corresponds to one virtual MAC address.

CLASS:**DATA TYPE****ATTRIBUTES:**

2	Data Type Name	= IP_MAP_TABLE_S
5	Format	= STRUCTURE
5.1	Number of Fields	= 9
5.2	List of Fields	
5.2.1	Field Name	= Index
5.2.2	Field Data Type	UINT16
5.2.3	Field Name	= Stated
5.2.4	Field Data Type	UINT8
5.2.5	Field Name	= Network_id
5.2.6	Field Data Type	UINT32
5.2.7	Field Name	= ipv4_addr
5.2.8	Field Data Type	UINT32
5.2.9	Field Name	= ipv4_mask
5.2.10	Field Data Type	UINT32
5.2.11	Field Name	= ipv6_addr
5.2.12	Field Data Type	UINT128
5.2.13	Field Name	= ipv6_mask
5.2.14	Field Data Type	UINT128
5.2.15	Field Name	= Dmac_addr
5.2.16	Field Data Type	ARRAY
5.2.17	Field Name	= Vmac_addr
5.2.18	Field Data Type	ARRAY
6	Constraint	Format = ARRAY
6.1	Number of Array Elements	6
6.2	Array Element Data Type	UINT8
6.3	Number of Array Elements	6
6.4	Array Element Data Type	UINT8

6 Type 28 communication model specification

6.1 General

The Type 28 AL processes RT and nRT application services based on the Type 28 DLL and PhL, which provides three kinds of service functions: time service, data service and system management service. The position of AL in the Type 28 protocol stack architecture is shown in Figure 3.

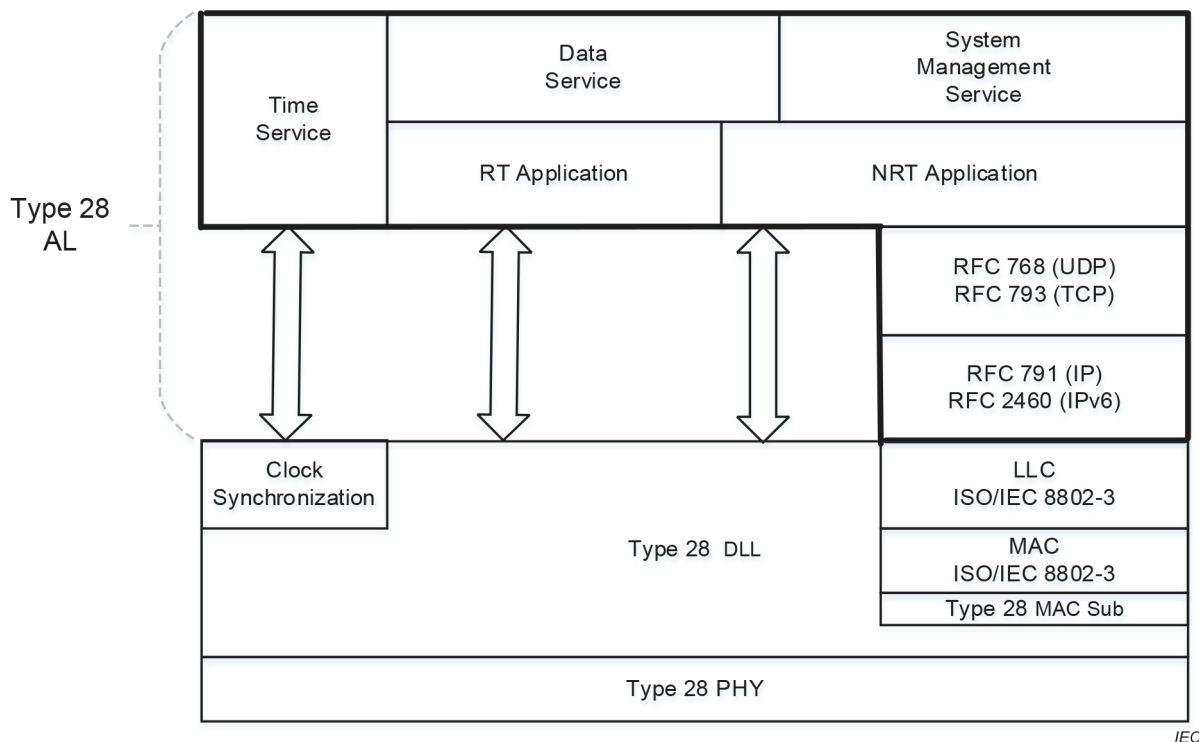


Figure 3 – AL in fieldbus protocol stack architecture

6.2 Protocol stack for Type 28 fieldbus

Figure 4 shows the Type 28 AL protocol architecture.

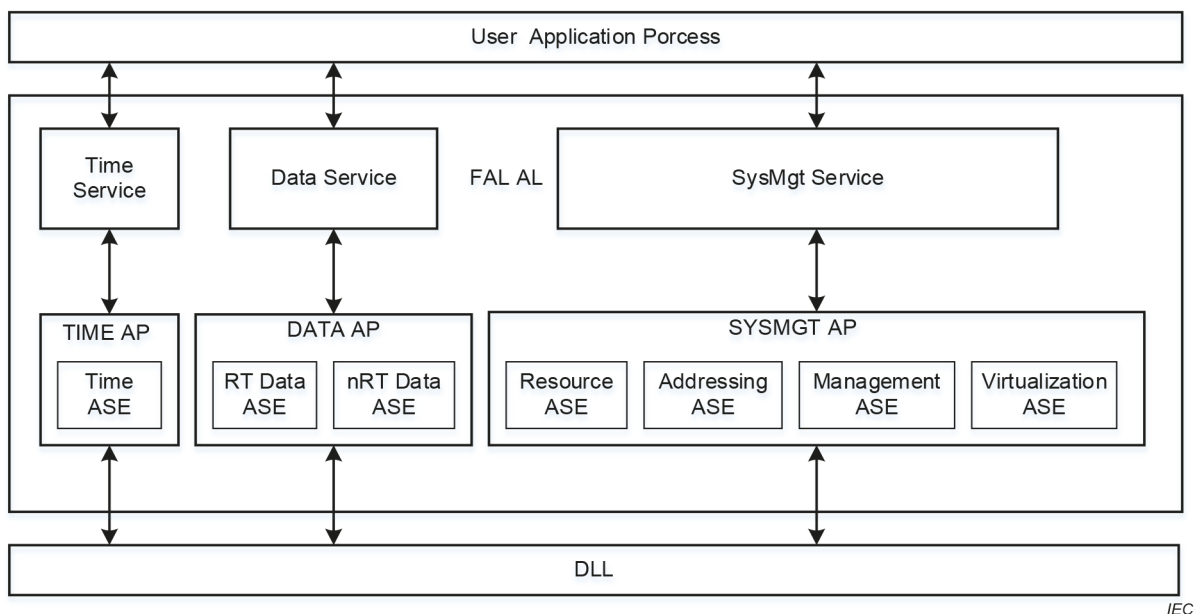


Figure 4 – Type 28 AL protocol architecture

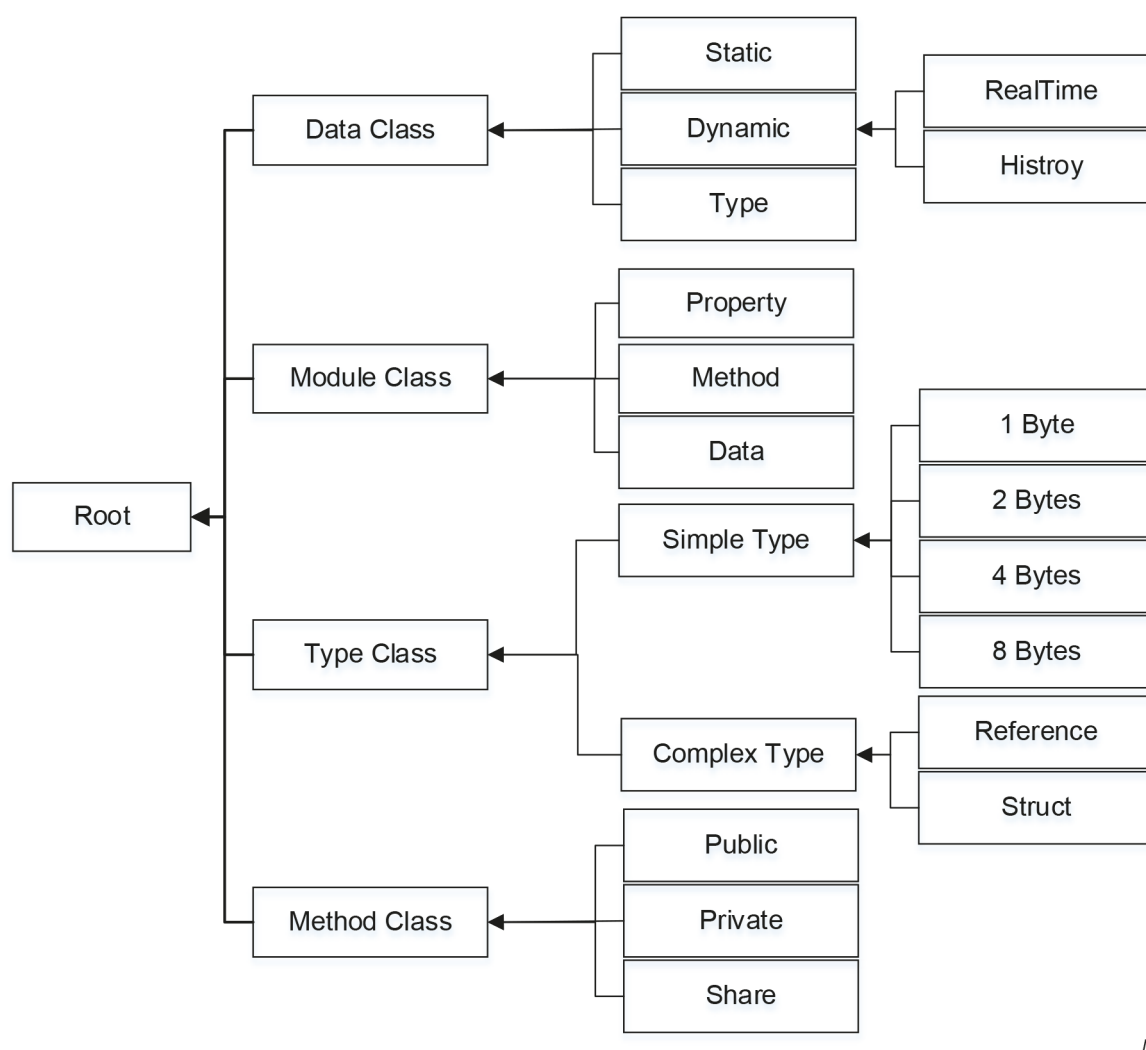
Type 28 AL provides users with time service, data service and system management service, which are implemented by the application service element (ASE) of the corresponding Application Process (AP). Time service implements time synchronization and time query based on time ASE. Data service processes the data with different real-time requirements based on RT data ASE and nRT data ASE. System management service provides the network system

with resource application service, addressing application services, managing application services and virtualization application services. The Type 28 AL protocol implements the specific AL service functions through the Type 28 DLL and service interfaces provided by the lower layer.

6.3 Data mapping modelling

In the Type 28 network, there is a service data buffer structure on the device or module with RT data buffer or the nRT data buffer based on the RT of the service data processing. The RT data buffer distinguishes the real-time cyclic (RTC) data buffer and real-time acyclic (RTA) data buffer according to the different cyclic processing methods of the data. There are several receiving and transmitting addresses for each type of data buffer. The AP related service data in the Type 28 network maps to the corresponding position in the data buffer using data modelling.

The APO definition involved in the AP of Type 28 AL protocol adopts the object-oriented method. All APO classes inherit from the unique root node (Root), and the definition of all APOs includes four object classes, Data Class, Module Class, Type Class, and Method Class as shown in Figure 5.



IEC

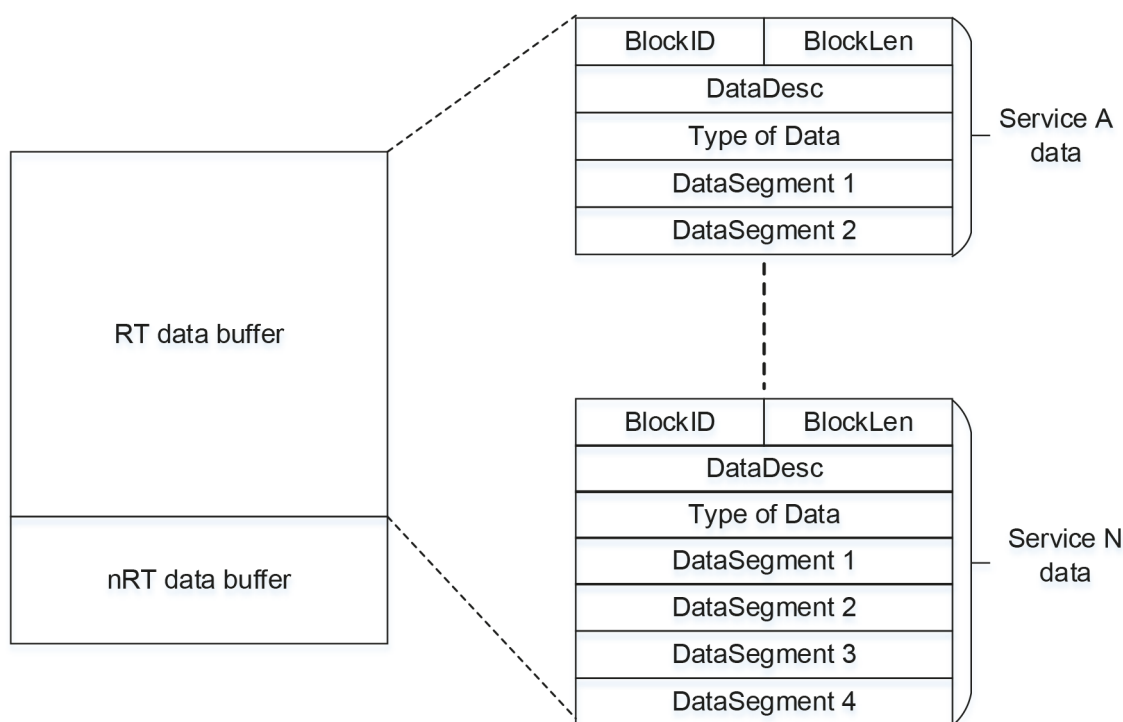
Figure 5 – Data mapping modelling object structure

In the Type 28 network environment, devices or modules describe specific AP interaction using data and method. The data content includes the attributes, features, types and data that need to be processed interactively of device or module. The data is described in simple data type.

complex data type. The methods can be private methods, public methods or shared methods. When device or module performs specific AP interaction, it needs to describe with the specific data and methods, and maps the buffer data to the actual data message to transmit. This modelling methods and data transmission methods are directly related to the data message format, which can improve the communication efficiency and performance of the bus (from the perspective of data message encapsulation and parsing). See Annex A for an example of service data mapping for DTU messages.

The data mapping modelling method provided by the Type 28 AL protocol is suitable to various industrial bus environments. If the object dictionary is used to define the AL data, based on the object class defined by the Type 28 AL protocol, different object data is instantiated separately, and then mapped into data messages transmitted by the communication to meet the AP interaction requirements. The mapping model method is well compatible with the data model of OPC UA. See Annex B for a data modelling description using a sensor as an example.

Modelling the APO data in the Type 28 network environment, the actual service data variables are represented by buffer blocks in the data buffer, and each block has deterministic information such as the block ID and the block length as shown in Figure 6.



IEC

Figure 6 – Diagram of mapping data buffers and service data variables

Description:

- BlockID: The starting address of data buffer block corresponding to the service data.
- BlockLen: Valid data length of the data buffer block corresponding to the service data, that is, the total length from the beginning of the data segment to the end of the data segment.
- DataDesc: Description of the association with the actual application service.
- Type of Data: Determines the size of the data segment.
- DataSegment: Corresponds to the actual service data variable.

In the Type 28 network, the interaction of AP should access the buffer block data by the globally unique DataID. There is a service data mapping table keyed using the unique DataID that exists

on the device or module. The content of every unique entry is associated with the actual data buffer block data. The parameters of the table are shown in Table 11.

Table 11 – Service data mapping table parameter list

Entry name	Type of Data	Description of entry
DataID	UINT16	ID that uniquely identifies the service data
BuffType	UINT8	Buffer type, 0x00: RT, 0x04: nRT, other values are reserved.
DataState	UINT8	Status of the data block, 0x00: valid, 0x04: in use, 0x0F: invalid, other values are reserved.
BlockID	UINT32	Data buffer block ID
NetworkID	UINT32	NETWORKID of device or module to which the service data belongs

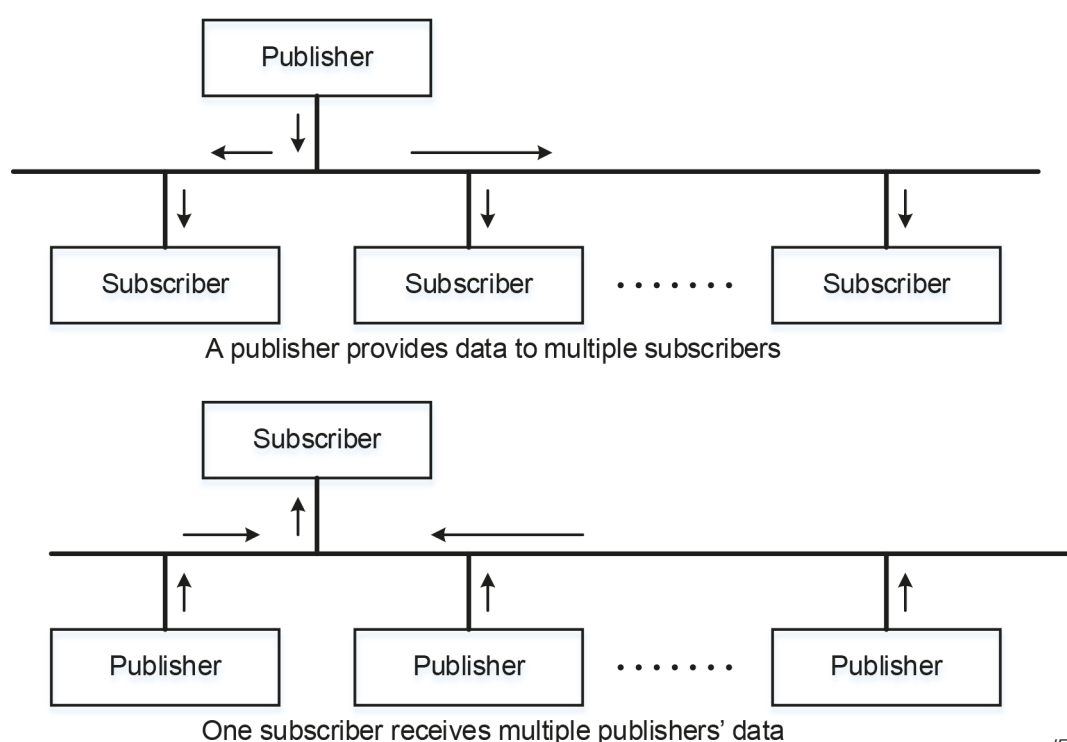
6.4 Overview of Type 28 communication model

6.4.1 General

The AP in the Type 28 network implements the determined service function objectives through interaction. The interactions between the APs can be based on the request/response message or on the data/events issued by each other APs. The communication models of these interactions methods are publisher/subscriber model (P/S) and client/server model (C/S).

6.4.2 P/S

In the process of AP interaction based on P/S model, one publisher provides data to one or more subscribers, and one subscriber receives data provided by one or more publishers. Diagram of P/S model is shown in Figure 7.



IEC

Figure 7 – Diagram of P/S model

On the network, the device or module provides service data as a publisher or receives service data as a subscriber based on AP interaction demand. The AP interaction of RT data and nRT

data between APs should adopt the P/S model. In the Type 28 network, the P/S-based communication model supports both push (PUSH) mode and pull (PULL) mode as shown in Figure 8 and Figure 9.

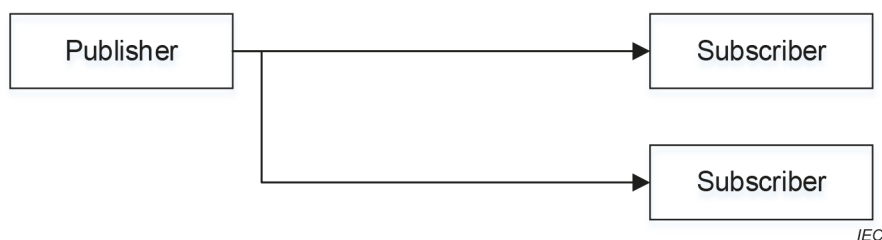


Figure 8 – P/S model of PUSH mode

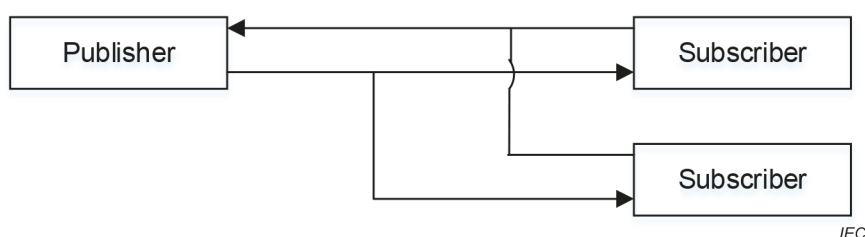


Figure 9 – P/S model of PULL mode

For PUSH mode, publisher actively sends data to the subscriber. This interactive communication mode does not need to confirm whether the subscriber receives the data correctly. For PULL mode, the subscriber actively requests the publisher to send data when data is needed, and application service based on this mode is usually required to be confirmed.

6.4.3 C/S

The AP interaction based on the C/S model uses bidirectional data stream to implement the determined service function by request and response. Based on this model, there are two modes: confirmed and non-confirmed. In the confirmed mode, the server needs to confirm the status and completion of the interaction process according to the response of the client to the response message. In non-confirmed mode, the server will confirm the status and the completion of the interaction process after making response processing against the client's request. The user's choice of confirmed and non-confirmed mode depends on the specific application interaction service. A C/S communication model diagram is shown in Figure 10.

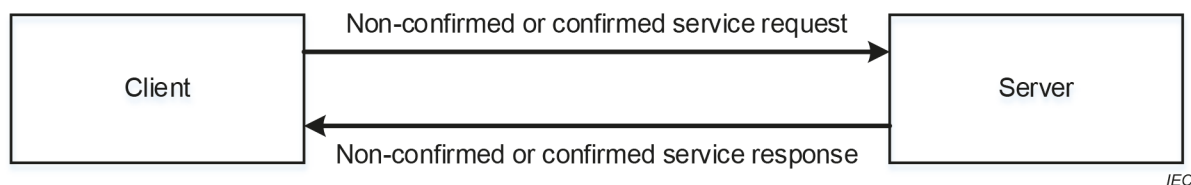


Figure 10 – Diagram of C/S communication model

6.5 AL related management information table

6.5.1 General

The AL management information is consistent with the corresponding items in the system management configuration file, and exists in the form of software variables in the device. AL management information includes time management information and network management information. The AL system management user or the AL user should access the corresponding information entry through the AL management service interface.

6.5.2 Time management information table

Table 12 – Time management information table

Parameter index	Parameter name	RW	Data Type	Description
0x2000	time_fmt	R/W	UINT8	Different time formats represented by bit Bit 0: indicates that time format is based on hour, minute and second. Bit 1: indicates that time format is based on milliseconds and microseconds. Bit 2: indicates that the time format is based on second pulse. The corresponding bit set to 0 is invalid, set to 1 is valid Bit 3 to bit 7: reserved
0x2001	time_ns	R/W	TIMEV	Time count based on nanosecond
0x2002	timedate	R/W	TIMEDATE	Convert to date value of year, month, day, minute, , and second, which is accurate to milliseconds
0x2003	timeofday	R/W	TIMEOFDAY	The number of days based on ns
0x2004	timediffer	R/W	TIMEOFDIFF	The time difference based on nanosecond
0x2005	tid_cnt	R/W	UINT8	Number of clocks on the device or module Value range: 1 to 8 Value 0 is invalid. Other values are reserved
0x2006	tid_val	R/W	UINT8	Clock ID Value range: 0 to 7 Other values are reserved
0x2007	tid_attr	R/W	UINT8	Control field attribute of clock Value range: 0x00 or 0x01 0x00: control clock 0x01: field clock Other values are reserved
0x2008	tcd_lst	R/W	ARRAY	ARRAY of type UINT8 TCD command for the corresponding clock Value range is specified in IEC 61158-6-28, Clause 4.
0x2009 to 0x208F	reserved			Protocol reserved
0x2090 to 0x20FF	reserved			Reserve for equipment manufacturer

6.5.3 Network management information table

Table 13 – Network management information table

Parameter index	Parameter name	RW	Data Type	Description
0x2100	network_id	R/W	UINT32	Indicate NETWORKID of node or module See 6.6.6.1.2.
0x2101	ip_switch	R/W	UINT8	Whether supporting the IPv4 or the IPv6 represented by bit Bit 0: Whether supporting IPv4 Bit 1: Whether supporting IPv6 The corresponding bit set to 0 indicates "not support" and set to 1 indicates "support". Bit 2 to bit 7: reserved
0x2102	multicast_id0 to multicast_id15	R/W	UINT32	NETWORKID with group type indicates the multicast group to which the device or module belongs to. Each device or module belongs to a maximum of 16 multicast groups.
0x2103	mac_addr	RW	UINT64	Unique MAC address of the device or module Value range: 0 to $(2^{48} - 1)$ Other values are reserved.
0x2104	nodeid	RW	UINT8	Node ID address of the device Value range: 0 to 255 See IEC 61158-4-28, Clause 4.
0x2105	dctime	RW	UINT16	Data calculation cycle time, unit is μ s Value range: 0 to 65 535
0x2106	bctime	RW	UINT16	Bus service cycle time, unit is μ s Value range: 0 to 65 535
0x2107	wctime	RW	UINT16	Bus service data writing cycle time, unit is μ s Value range: 0 to 65 535
0x2108	rctime	RW	UINT16	Bus service data reading cycle time, unit is μ s Value range: 0 to 65 535
0x2109	cctime	RW	UINT16	Bus service competition cycle time, unit is μ s Value range: 0 to 65 535
0x210A	totcnt	RW	UINT16	Number of timeouts allowed by device or module Value range: 0 to 65 535
0x210B	rt_data_addr	RO	UINT32	Real-time data buffer start address Value range: 0x00000000 to 0xFFFFFFFF
0x210C	rt_data_size	RW	UINT32	Real-time data buffer size Value range: 0x00000000 to 0xFFFFFFFF
0x210D	nrt_data_addr	RW	UINT32	Non-real-time data buffer start address Value range: 0x00000000 to 0xFFFFFFFF
0x210E	nrt_data_size	RW	UINT32	Non-real-time data buffer size Value range: 0x00000000 to 0xFFFFFFFF
0x210F to 0x28FF	reserved			Protocol reserved
0x2900 to 0x2FFF	reserved			Reserved for equipment manufacturer

6.6 ASEs

6.6.1 Overview

Type 28 AL provides users with time service, data service and system management service based on the DLL service interface. The AL user addresses and accesses devices or modules in the Type 28 network through NETWORKID. The time AP provides time synchronization and time query service functions through time ASE. The data AP provides reading and writing operations of RT data and nRT data through RT Data ASE and nRT Data ASE. The system management AP provides application service functions of system such as resource management, address management, network system management, virtualization management through resource ASE, addressing ASE, management ASE and virtualization ASE. These application entities implement specific data communication transmission services through C/S or P/S communication models. Table 14 shows the description of the communication model used by Type 28 AL ASEs.

Table 14 – Communication models supported by Type 28 AL ASEs

ASE	C/S	P/S
RT Data cyclic	Not support	Support
RT Data acyclic	Support	Support
nRT Data	Support	Support
Time synchronization	Not support	Support
Time query	Support	Not support
Resource	Support	Support
Addressing	Support	Not support
Management	Support	Support
Virtualization	Support	Not support

The return value of the AL service is shown in Table 15. The AL user determines whether the service is successful and the reason for the failure according to the return value.

Table 15 – Application service interface return value list

Wrong name	Wrong coding	Wrong description
SUCCESS	0x00000000	Service processing succeeded
FAILED	0x00000001	Service processing failed
ERR_NOTSUPPORT	0x00000002	Device or module does not support
ERR_NOTINITIALIZED	0x00000003	Device or module has not been initialized
ERR_NOTEXIST	0x00000004	Device or module does not exist, service is unavailable
ERR_NOTREADY	0x00000005	Device initialization finishes but no valid status is set
ERR_INVALIDPARA	0x00000006	Invalid parameter information
ERR_UNKNOWN	0x00000007	Unknown error
ERR_TIMEOUT	0x00000008	Service processing timeout

6.6.2 RT data ASE

6.6.2.1 General

RT data ASE defines service interfaces and interface parameters that support RT application services. The RT application service processes the RT data in cyclic or acyclic manner. The RTC data is cyclically read or written according to the specified cycle time. The processing of RTA data does not have a fixed cycle time, but it should be completed within the specified time when once occurred.

The RTC data is transmitted based on the P/S communication model, and the RTA data is transmitted based on the P/S communication model or the C/S communication model. Before processing the RT application service, the user needs to pre-configure the channel resource based on the resource ASE, and the channel resource will be occupied by the user all the time. If the channel resource for the RT data transmission is changed, the system should be restarted to meet the latest requirement of RT data service.

6.6.2.2 Interface

ALS-RT-DATA-Proc (

Dst_NetworkID,
Src_NetworkID,
ProcessID,
DataID,
OptionRTLst,
ALS_Result

)

Table 16 defines RT Data service interface parameters.

Table 16 – RT data service interface parameters

Parameter name	Data type	Input/output
Dst_NetworkID	UINT32	IN
Src_NetworkID	UINT32	IN
ProcessID	UINT16	IN
DataID	UINT32	IN OUT
OptionRTLst	STRUCT	IN OUT
ALS_Result	UINT32	OUT

- Dst_NetworkID: NETWORKID of the destination device or module.
- Src_NetworkID: NETWORKID of the source device or module.
- ProcessID: RT data processing mode ID, which identifies the RT data processing method based on bits and if the corresponding bit position is 1, execute the corresponding processing, defined as following:
 - Update (bit 0): Updates data after comparing the content of the data based on the bit value;
 - Reset (bit 1): Restores the data in data area content to the default value;
 - Clear (bit 2): Clears the data of the specified data area and set the zero operation;
 - Read (bit 3): Reads the data of the specified data area, and does not clear the corresponding data content;
 - Read clear (bit 4): Reads the data of the specified data area, and clear the data of the corresponding data area;
 - Write (bit 5): Writes data directly to the specified data area, and overwrites the original data;

- Write clear (bit 6): Clears the operation for the specified data area, and then write new data;
 - Query (bit 7): Queries data status;
 - Request (bit 8): Requests data;
 - Response (bit 9): Responds to the request, response data;
 - Confirmation (bit 10): Confirms the data transmission status;
 - Send (bit 11): Sends data to the specified destination device or module;
 - Receive (bit 12): Receives data sent by a specified source device or module;
- DataID: Service data encoding ID, unique ID of the specific service data. The ID can be associated with the data area corresponding to the data buffer. For RTC data and RTA data, the corresponding DataID is different.
 - OptionRTLst: A list of RT data processing information options, including the status of the data information after processing, as described below:
 - StateInfo: UINT16 type value, status information of the data area;
 - DataSize: UINT16 type value, the valid data size required for reading and writing;
 - RetCode: UINT32 type value, RT data information processing result encoding, the default value is 0;
 - ALS_Result: The value that the RT data service returns, as specified in Table 15.

6.6.2.3 Interaction process

The processing of RT data is deterministic and timely. The determinacy means that the data should be processed at a certain time, while the timeliness means that the time required for data to be transmitted from the source device or module to the destination device or module is expected. RTC data needs to be cyclically processed in a fixed cycle, while RTA data does not need to be cyclically processed in the fixed cycle.

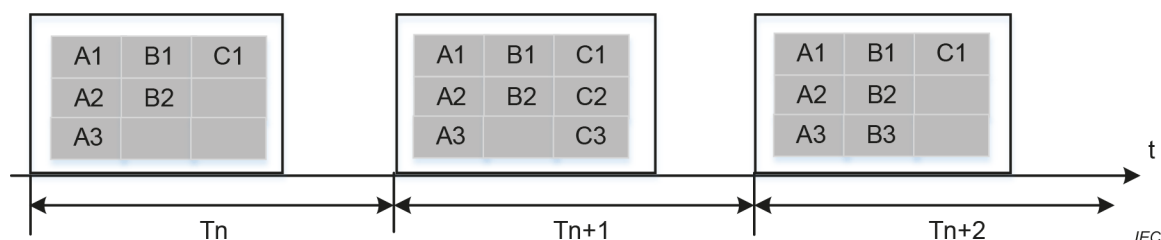


Figure 11 – RT data transmission model

As shown in Figure 11, the service data blocks A1, A2, A3, B1, B2 and C1 are always cyclically processed in each fixed period, like T_n , T_{n+1} , T_{n+2} , those are RTC data interactions. Service data blocks B3, C2 and C3 are not processed in every cycle. When transmission is required, RT processing is completed within a certain time period, that is RTA data interaction.

Since RTA data are usually control, configuration, alarms or exceptions information, they are not cyclically processed. See Annex C for an example of RTA service processing procedure. When the RTA data application service adopts the C/S communication model, the client application sends a request to the server. The server application acquires the request for the corresponding processing and sends the response data to the client application according to the requested content within a specified time period. The client application obtains the response data content sent by the server and confirms the completion of the interactive service.

6.6.3 nRT data ASE

6.6.3.1 General

In the Type 28 network, for nRT application service data, time-critical is not necessary and the length of message is not fixed. Before the nRT data transmission is performed, the resource ASE is used to apply for the channel resource for transmission to the MN. The maximum times of this channel resources applied to the user should be specified.

nRT data ASE defines interfaces and related parameters for nRT data processing, also provides users with operations such as reading and writing nRT data. The user uses the resource ASE to apply for channel resources for nRT data processing, and the user needs to specify the channel resource information in nRT data processing.

When the nRT data AP interacts based on C/S communication model, the client application sends a request for reading or writing to the server application, and the server application performs response processing. When the nRT data AP interacts based on P/S communication model, the data is transmitted through the idle default channel resources allocated by the system. The system schedules the channel resources according to the pre-assigned nRT application service priority, and preferentially allocates the idle channel resources to nRT application service users with high priority.

6.6.3.2 Interface

```
ALS-NRT-DATA-Proc (
    Dst_NetworkID,
    Src_NetworkID,
    ProcessID,
    DataID,
    OptionNRTLst,
    ALS_Result
)
```

Table 17 defines nRT data service interface parameters.

Table 17 – nRT data service interface parameters

Parameter name	Data type	Input/output
Dst_NetworkID	UINT32	IN
Src_NetworkID	UINT32	IN
ProcessID	UINT16	IN
DataID	UINT32	IN
OptionNRTLst	STRUCT	IN OUT
ALS_Result	UINT32	OUT

- Dst_NetworkID: NETWORKID of the destination device or module.
- Src_NetworkID: NETWORKID of the source device or module.
- ProcessID: nRT data processing mode ID. The data processing mode is represented by bit, wherein the bit 15 position 1 indicates that the data is transmitted based on the TCP/IP protocol stack, and the IP information is obtained by the NETWORKID. The bit 15 position 0 indicates that the processing is based on the Type 28 DLL layer, and see the description of the ProcessID described in Subclause 6.6.2.2 for the description of remaining bit.
- DataID: Service data ID, which is associated with the corresponding address of the data buffer. For the nRT data corresponding to the service interface, you can choose not to associate with the buffer. In this case, the ID is invalid.
- OptionNRTLst: A list of option parameters, including the following information:

- Opt_Cmd: UINT8 type value, optional command word, which identifies whether the service needs confirmation; 0x00: the service does not need to be confirmed; 0x01: the service needs to be confirmed;
 - DataSize: UINT16 type value, the effective data size that needs to be read and written;
 - DataAddr: UINT32 type data storage area address, this value is valid when the DataID is invalid;
 - Data_Flag: UINT8 type value, which identifies the release operation after reading and writing of nRT data block; 0x00: no processing; 0x01: clear release; 0x02: no clear release;
- ALS_Result: The nRT data service processes the return value. See Table 15 for the specific return values.

6.6.3.3 Interaction process

In the Type 28 network, when the nRT data AP interaction uses the C/S communication model, the nRT data service interface provides confirmed mode and non-confirmed mode to meet the requirement of different services. In the confirmed mode, the client sends a confirmed message to the server after correctly receiving and processing the response message sent by the server. In the non-confirmed mode, the client should not send the confirmed message to the server after correctly receiving and processing the response message sent by the server. In the process of implementing the nRT data application service by using the C/S communication model, after the client application initiates the request, it waits for the asynchronous response of the server application, and the asynchronous interval time period is usually one or more bus cycles as shown in Figure 12. The time period of the interval is related to the configured bus cycle and the specific size of the nRT data that needs to be transmitted.

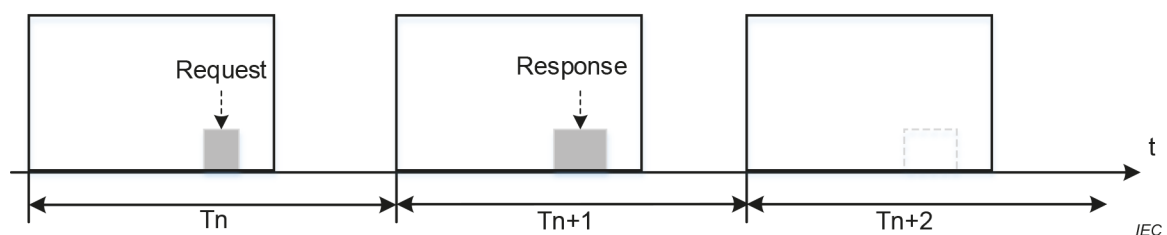


Figure 12 – Diagram of nRT data request response model

When the nRT data AP interaction adopts the C/S communication model, the client application initiates one nRT data service request. The server application responds to the client application request, and completes the data processing locally or sends the response data to the remote client application. According to the different specific content of the client application request, the data transmission service based on the request and response is continuously completed between the server application and the client application.

Figure 13 shows the nRT data AP interaction based on C/S communication model.

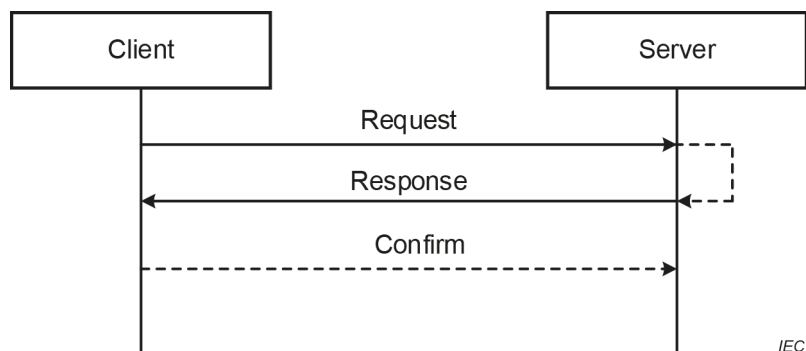


Figure 13 – Diagram of nRT data AP interaction based on C/S communication model

When nRT data AP interaction adopts P/S model, it will schedule channel resources according to the priority of nRT data application service and send nRT data through the idle default channel. According to the features of nRT data, nRT data application service users will use different default channels and occupancy periods. A diagram of nRT data AP interaction based on P/S communication model is shown in Figure 14, where both the nRT service 1 and the nRT service 2 can use one default channel for data transmission, and the nRT service 3 has higher priority and needs to use default channel twice to complete the corresponding data transmission.

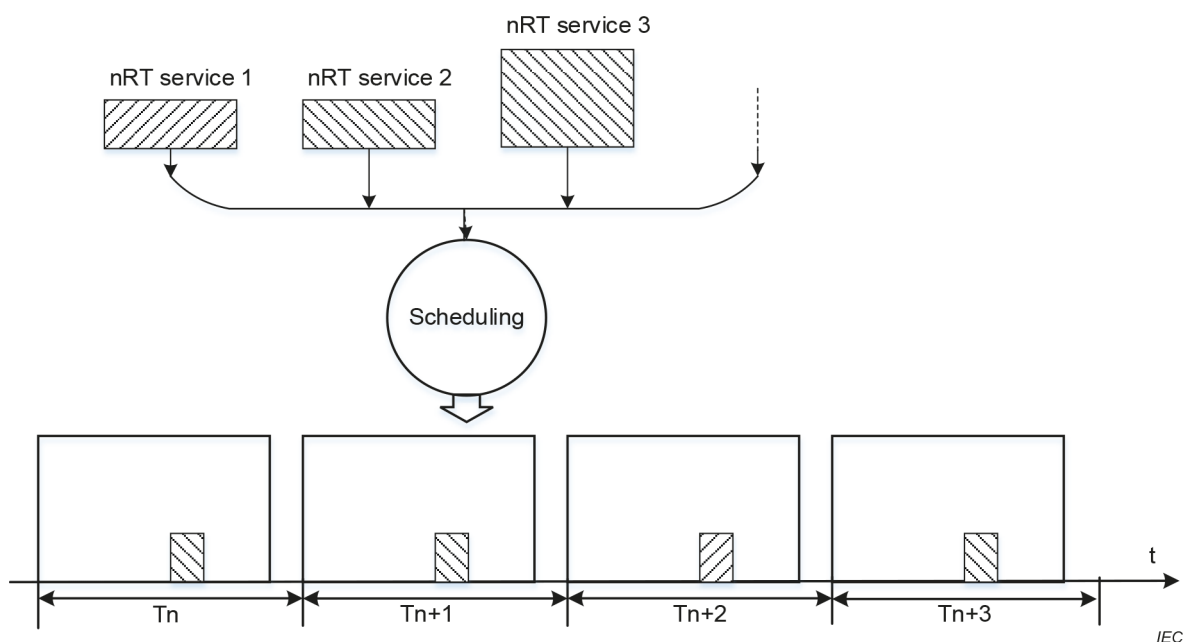


Figure 14 – Diagram of nRT data AP interaction based on P/S communication model

6.6.4 Time ASE

6.6.4.1 General

The time ASE defines the interface and parameters of the time service. The devices and modules in the Type 28 network implement the time synchronization and time query service functions based on the time ASE.

On the network, the AL user triggers the time synchronization service at the MN. The MN will send message including the transmission delay time information between TN and all TNs, and then the TN will parse the delay time information from the message sent by the MN, and set the time of the station to complete synchronization.

The time query service provides a query for the synchronizing state of TN's time. The object of the query service on MN is any node in the network but the node itself on TN.

6.6.4.2 Interface

ALS-TIMING-Proc (

Dst_NetworkID,
Src_NetworkID,
ProcessID,
OptionClkInfo,
TimeInfo,
ALS_Result

)

Table 18 defines timing service interface parameters.

Table 18 – Timing service interface parameters

Parameter name	Data type	Input/output
Dst_NetworkID	UINT32	IN
Src_NetworkID	UINT32	IN
ProcessID	UINT16	IN
OptionClkInfo	STRUCT	IN OUT
TimeInfo	STRUCT	IN OUT
ALS_Result	UINT32	OUT

- Dst_NetworkID: NETWORKID of the destination device or module.
- Src_NetworkID: NETWORKID of the source device or module.
- ProcessID: Time Service Processing ID:
 - 0x0000: Time synchronization request;
 - 0x0001: Time synchronization response;
 - 0x0002: Time synchronization exception;
 - 0x0004: Time query request;
 - 0x0008: Time query response;
 - 0x0010: Time query error.
- OptionClkInfo: Time option information. When implementing time synchronization service, this parameter is used as an input parameter, and specifies clock device information list that needs to be set. When this parameter is used as an output parameter, it shall specify the set output information. If the setting is successful, it will output the current state, current value and current attribute of the output time; if the setting fails, the corresponding error state code is output. See Table 7 for the description of the structure CLOCK_OPTION_INFO_S. When the time query service function is implemented, this parameter is an output parameter.
- TimeInfo: Time information that needs to be set for the time. See Table 6 for the description of the structure TIMEINFO_S.
- ALS_Result: Time Service processing return value. See Table 15 for details of the return value.

6.6.4.3 Interaction process

6.6.4.3.1 Time synchronization interaction process

The MN sends its own time information and the measured transmission delay between the corresponding nodes to each TN, and the TN synchronizes time and updates the local time information based on the time information of MN. The time synchronization interaction process adopts the P/S communication model, and the time application of the MN acts as a publisher and periodically sends time information to the time application of the TN, but the release period is set by the MN. If a synchronization failure or error occurs in the client time application, a time synchronization reset operation can be requested from the MN time application.

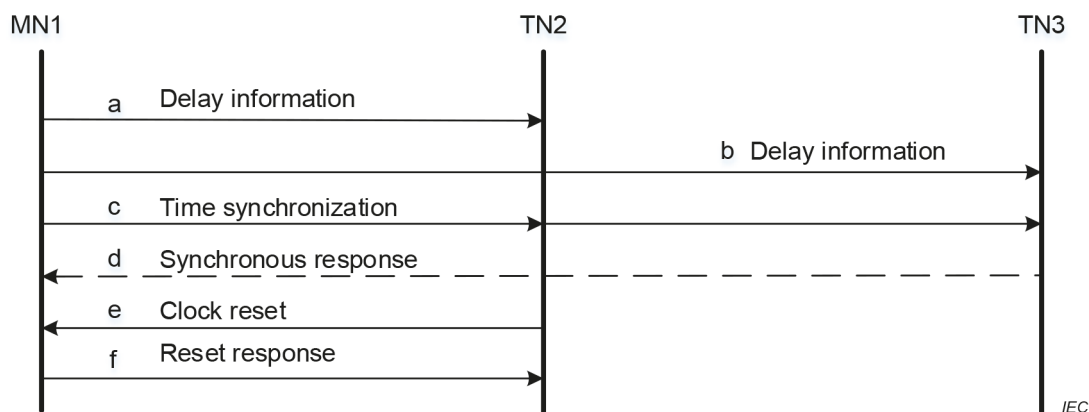


Figure 15 – Time synchronization application interaction process

See Figure 15. Time synchronization is initiated by the MN:

- The MN sends the transmission delay value of corresponding MN to the TN.
- The MN initiates time synchronization, sends the time value to the TN through the time service notification message, and requests the TN to respond to the synchronization result.
- After receiving the time information in the time service notification message sent by the MN, the TN implements the local time synchronization through the local DLL interface, and determines whether to respond to the synchronization result according to the requirements of the MN.
- If an error occurs on the TN during the time synchronization, the response error message shall be transmitted to the MN.
- If an error occurs on the TN during the time synchronization, the TN can actively send a time reset request to the MN, and ask the MN to update the time information and the line transmission delay information. At the same time the MN should update the time information and line transmission delay information to the designated TN.
- The MN can select periodic time synchronization or single time synchronization operation according to system requirements and configuration.

6.6.4.3.2 Time query interaction process

The time query service provides the function of querying the time synchronization state and synchronization quality of the TN. The time query service uses the C/S communication model to interact. The time application of the MN acts as a client and requests time synchronization state and synchronization quality information from time application at the TN as a server. The MN sends a time service notification message to trigger the time query service. For the procedure, see Figure 16.

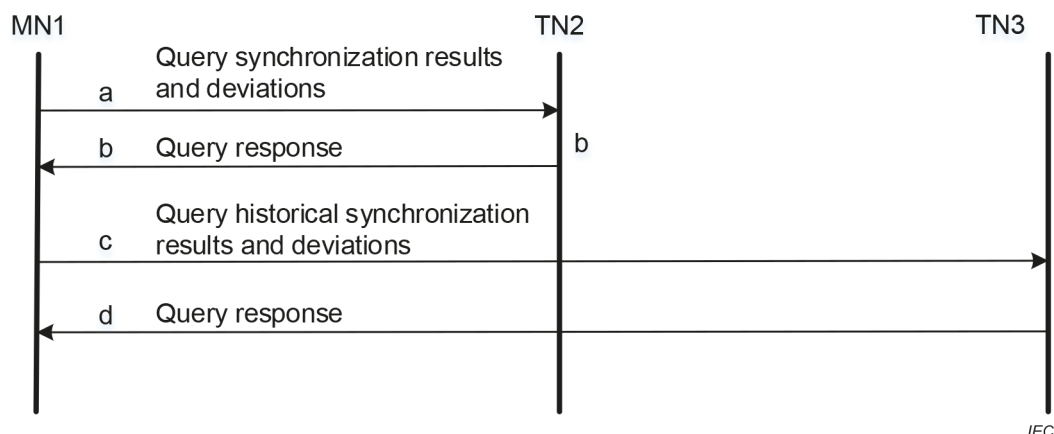


Figure 16 – Time query process

See Figure 16. The time query is initiated by the MN:

- a) The MN queries the time of the last synchronization of the corresponding TN and the deviation value during synchronization.
- b) After receiving the query, the TN replies to the local record information of the MN, and the MN determines the quality and deviation range of the synchronization of the TN based on this.
- c) The MN can query other terminal information of corresponding TN, such as the average value, root mean square value, time span, etc. of deviation values of the synchronization.
- d) The TN replies to the corresponding content of the MN based on the content of the query.

6.6.5 Resource ASE

6.6.5.1 General

The resource ASE of the Type 28 AL protocol defines the service interface definition and parameter settings of the channel resources required for AP interaction on the network. The resource ASE performs channel resource allocation and usage based on the service interface provided by the Type 28 DLL, and thus implements the operations of applying, establishing, updating, and releasing channel resources.

Channel resources include fixed bandwidth channel resources and variable bandwidth channel resources. The fixed bandwidth channel resources are created during Type 28 configuration or network initialization, and cannot be updated and released afterwards; variable bandwidth channel resources are created and applied by users when needed. The size of variable bandwidth channel resources and resources usage time is determined by the user at the time of application, and users can dynamically update and release the resource channel during use. The MN is responsible for the allocation and management of all channel resources of the Type 28 network.

The resource ASE implements the AP interaction by using the confirmed C/S communication model. When the resource AP interacts, the server application is always at the MN, while the client application can be at the MN or the TN. The client application sends a resource application request, and server application responds to the resource application request, sends the response data, allocates the resource channel or rejects the resource application request, while the client application acknowledges receipt of the data and sends a confirmation message to the server application, and then completes the local resource AP interaction.