

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Industrial communication networks – Fieldbus specifications –
Part 6-11: Application layer protocol specification – Type 11 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 6-11: Spécification du protocole de la couche d'application – Éléments
de Type 11**

IECNORM.COM : Click to view the full PDF of IEC 61158-6-11:2007



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2007 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 la CEI ou du Comité national de la CEI du pays du demandeur.

Si vous avez des questions sur le copyright de la CEI ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de la CEI de votre pays de résidence.

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

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
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 corrigenda or an amendment might have been published.

Useful links:

IEC publications search - www.iec.ch/searchpub

The advanced search enables you 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It also gives information on projects, replaced and withdrawn publications.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 on-line and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary (IEV) on-line.

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: csc@iec.ch.

A propos de la CEI

La Commission Electrotechnique Internationale (CEI) 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 CEI

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

Liens utiles:

Recherche de publications CEI - www.iec.ch/searchpub

La recherche avancée vous permet de trouver des publications CEI 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.

Just Published CEI - webstore.iec.ch/justpublished

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

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (VEI) en ligne.

Service Clients - webstore.iec.ch/csc

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

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Industrial communication networks – Fieldbus specifications –
Part 6-11: Application layer protocol specification – Type 11 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 6-11: Spécification du protocole de la couche d'application – Éléments
de Type 11**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX



ICS 25.040.40; 35.100.70

ISBN 978-2-8322-1023-9

**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.....	4
INTRODUCTION.....	6
1 Scope.....	7
1.1 General.....	7
1.2 Specifications.....	8
1.3 Conformance.....	8
2 Normative references	8
3 Terms, definitions, symbols, abbreviations and conventions	9
3.1 Introduction	9
3.2 Terms and definitions from other ISO/IEC standards	9
3.3 Terms and definitions from IEC/TR 61158-1	10
3.4 Other terms and definitions	10
3.5 Abbreviations and symbols.....	11
3.6 Conventions	12
4 FAL syntax description	13
4.1 Concept	13
4.2 General.....	14
4.3 FAL-AR PDU abstract syntax	14
4.4 Abstract syntax of PDU body.....	15
4.5 Data type	15
5 Transfer syntax	15
5.1 Overview and FAL header	15
5.2 Encoding rule	16
5.3 Encoding of structured types.....	18
6 FAL protocol state machines structures	18
6.1 Overview	18
7 FAL service protocol machine (FSPM).....	19
7.1 General.....	19
7.2 Primitives definitions	19
7.3 FSPM state tables	20
8 Application relationship protocol machine (ARPM).....	21
8.1 General.....	21
8.2 Primitive definitions	21
8.3 DLL mapping of BNU-PEC AREP class	22
8.4 BNU-PEC ARPM states machine.....	23
9 DLL mapping protocol machine (DMPM).....	25
9.1 Overview.....	25
9.2 Primitive definitions	26
9.3 DLL mapping protocol machine (DMPM).....	27
9.4 Data-link layer service selection	30
Bibliography.....	31
Figure 1 – RTE-TCnet communication profile.....	14
Figure 2 – APDU overview	16
Figure 3 – Relationship between FSPM, ARPM, DMPM and external physical CM	19

Figure 4 – State transition diagram of FSPM.....	20
Figure 5 – State transition diagram of the BNU-PEC	23
Figure 6 – State transition diagram of DMPM.....	27
Table 1 – Conventions used for state machines	12
Table 2 – FAL header	16
Table 3 – Primitives issued by FAL user to FSPM	20
Table 4 – Primitives issued by FSPM to FAL user	20
Table 5 – FSPM state table – sender transactions	20
Table 6 – FSPM state table – receiver transactions	21
Table 7 – Function SelectArep	21
Table 8 – Primitives issued by FSPM to ARPM	22
Table 9 – Primitives issued by ARPM to FSPM	22
Table 10 – Parameters used with primitives exchanged between FSPM and ARPM	22
Table 11 – BNU-PEC state descriptions	23
Table 12 – BNU-PEC ARPM state table – sender transactions	24
Table 13 – BNU-PEC ARPM state table – receiver transactions	24
Table 14 – Function GetArepId ()	25
Table 15 – Function BuildFAL-PDU	25
Table 16 – Function FAL_Pdu_Type	25
Table 17 – Primitives issued by ARPM to DMPM	26
Table 18 – Primitives issued by DMPM to ARPM	26
Table 19 – Parameters used with primitives exchanged between ARPM and DMPM	26
Table 20 – Primitives exchanged between data-link layer and DMPM	27
Table 21 – DMPM state descriptions	27
Table 22 – DMPM state table – sender transactions	28
Table 23 – DMPM state table – receiver transactions	29
Table 24 – Function PickArep	29
Table 25 – Function FindAREP	29

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**INDUSTRIAL COMMUNICATION NETWORKS –
FIELD BUS SPECIFICATIONS –****Part 6-11: Application layer protocol specification – Type 11 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 provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
- 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.

NOTE Use of some of the associated protocol types is restricted by their 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 particular data-link layer protocol type to be used with physical layer and application layer protocols in Type combinations as specified explicitly in the IEC 61784 series. Use of the various protocol types in other combinations may require permission from their respective intellectual-property-right holders.

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this standard may involve the use of patents concerning as follows:

TOSHIBA has the patent applications listed below:

- US Publication Number 6711131 and its counterpart patents in other countries
- US Publication Number 5414813 and its counterpart patents in other countries
- US Publication Number 4930121 and its counterpart patents in other countries

IEC takes no position concerning the evidence, validity and scope of this patent right.

The holder of this patent right has assured the IEC that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with IEC. Information may be obtained from:

Toshiba Corporation
1-1, Shibaura 1-Chome
Minato-ku Tokyo 105-8001, Japan

Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights other than those identified above. IEC shall not be held responsible for identifying any or all such patent rights.

International standard IEC 61158-6-11 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation.

This first edition and its companion parts of the IEC 61158-6 subseries cancel and replace IEC 61158-6:2003. This edition of this part constitutes a technical addition. This part and its Type 11 companion parts also cancel and replace IEC/PAS 62406, published in 2005.

This bilingual version (2013-09) corresponds to the monolingual English version, published in 2007-12.

The text of this standard is based on the following documents:

FDIS	Report on voting
65C/476/FDIS	65C/487/RVD

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

This edition of IEC 61158-6 includes the following significant changes from the previous edition:

- a) deletion of the former Type 6 fieldbus for lack of market relevance;
- b) addition of new types of fieldbuses;
- c) partition of part 6 of the third edition into multiple parts numbered -6-2, -6-3, ...

The French version of this standard has not been voted upon.

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

The committee has decided that the contents of this publication will remain unchanged until the maintenance result 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.

NOTE The revision of this standard will be synchronized with the other parts of the IEC 61158 series.

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

INTRODUCTION

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

The application protocol provides the application service by making use of the services available from the data-link or other immediately lower layer. The primary aim of this standard is to provide a set of rules for communication expressed in terms of the procedures to be carried out by peer application entities (AEs) at the time of communication. These rules for communication are intended to provide a sound basis for development in order to serve a variety of purposes:

- as a guide for implementors and designers;
- for use in the testing and procurement of equipment;
- as part of an agreement for the admittance of systems into the open systems environment;
- as a refinement to the understanding of time-critical communications within OSI.

This standard is concerned, in particular, with the communication and interworking of sensors, effectors and other automation devices. By using this standard together with other standards positioned within the OSI or fieldbus reference models, otherwise incompatible systems may work together in any combination.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-11:2007

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 6-11: Application layer protocol specification – Type 11 elements

1 Scope

1.1 General

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

This standard provides common elements for basic time-critical and non-time-critical messaging communications between application programs in an automation environment and material specific to Type 11 fieldbus. The term “time-critical” is used to represent the presence of a time-window, within which one or more specified actions are required to be completed with some defined level of certainty. Failure to complete specified actions within the time window risks failure of the applications requesting the actions, with attendant risk to equipment, plant and possibly human life.

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

- a) the abstract syntax defining the application layer protocol data units conveyed between communicating application entities,
- b) the transfer syntax defining the application layer protocol data units conveyed between communicating application entities,
- c) the application context state machine defining the application service behavior visible between communicating application entities; and
- d) the application relationship state machines defining the communication behavior visible between communicating application entities; and.

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

- 1) define the wire-representation of the service primitives defined in IEC 61158-5-11, and
- 2) define the externally visible behavior associated with their transfer.

This standard specifies the protocol of the IEC fieldbus Application Layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498) and the OSI Application Layer Structure (ISO/IEC 9545).

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

Although these services specify, from the perspective of applications, how request and responses are issued and delivered, they do not include a specification of what the requesting and responding applications are to do with them. That is, the behavioral aspects of the applications are not specified; only a definition of what requests and responses they can send/receive is specified. This permits greater flexibility to the FAL users in standardizing

such object behavior. In addition to these services, some supporting services are also defined in this standard to provide access to the FAL to control certain aspects of its operation.

1.2 Specifications

The principal objective of this standard is to specify the syntax and behavior of the application layer protocol that conveys the application layer services defined in IEC 61158-5-11.

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 protocols standardized in parts of the IEC 61158-6 series.

1.3 Conformance

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

There is no conformance of equipment to the application layer service definition standard. Instead, conformance is achieved through implementation of this application layer protocol specification.

2 Normative references

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

IEC 60559, *Binary floating-point arithmetic for microprocessor systems*

IEC 61158-3-11, *Industrial communication networks – Fieldbus specifications – Part 3-11: Data-link layer service definition – Type 11 elements*

IEC 61158-5-11, *Industrial communication networks – Fieldbus specifications – Part 5-11: Application layer service definition – Type 11 elements*

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

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

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

ISO/IEC 8824, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*

ISO/IEC 8825, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*

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

3 Terms, definitions, symbols, abbreviations and conventions

3.1 Introduction

For the purposes of this documents, the followings apply.

3.2 Terms and definitions from other ISO/IEC standards

3.2.1 Terms and definitions from ISO/IEC 7498-1

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

3.2.2 Terms and definitions from ISO/IEC 8822

- a) abstract syntax
- b) presentation context

3.2.3 Terms and definitions from ISO/IEC 9545

- 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

3.2.4 Terms and definitions from ISO/IEC 8824

- a) object identifier
- b) type
- c) value
- d) simple type
- e) structured type
- f) component type
- g) tag
- h) Boolean type
- i) true
- j) false
- k) integer type
- l) bitstring type
- m) octetstring type
- n) null type
- o) sequence type
- p) sequence of type
- q) choice type
- r) tagged type
- s) any type
- t) module
- u) production

3.2.5 Terms and definitions from ISO/IEC 8825

- a) encoding (of a data value)
- b) data value
- c) identifier octets (the singular form is used in this standard)
- d) length octet(s) (both singular and plural forms are used in this standard)
- e) contents octets

3.3 Terms and definitions from IEC/TR 61158-1

- a) application relationship
- b) conveyance path
- c) client
- d) dedicated AR
- e) dynamic AR
- f) error class
- g) error code
- h) name
- i) numeric identifier
- j) peer
- k) pre-defined AR endpoint
- l) pre-established AR endpoint
- m) publisher
- n) subscriber
- o) server

3.4 Other terms and definitions

NOTE The following definitions may apply to all the types except if the same item is used with different definitions in a specific type. In this case the latter has precedence.

The following terms and definitions are used in this series of standards.

3.4.1

called

service user or a service provider that receives an indication primitive or a request APDU

3.4.2

calling

service user or a service provider that initiates a request primitive or a request APDU

3.4.3

common memory

virtual common memory over the Type 11 fieldbus, which is shared with the nodes participating in the Type 11 fieldbus and is primarily used for the real-time communications by the TCC data service

3.4.4

interoperability

capability of User Layer entities to perform coordinated and cooperative operations using the services of the FAL

3.4.5

management information

network accessible information that supports the management of the Fieldbus environment

3.4.6

receiving

service user that receives a confirmed primitive or an unconfirmed primitive, or a service provider that receives a confirmed APDU or an unconfirmed APDU

3.4.7**resource**

resource is a processing or information capability of a subsystem

3.4.8**sending**

service user that sends a confirmed primitive or an unconfirmed primitive, or a service provider that sends a confirmed APDU or an unconfirmed APDU

3.5 Abbreviations and symbols

AE	Application Entity
AE-I	Application Entity Invocation
AL	Application Layer
AP	Application Process
Ap_	Prefix for Data types defined for AP ASE
Ar_	Prefix for Data types defined for AR ASE
APDU	Application Protocol Data Unit
AR	Application Relationship
AREP	Application Relationship End Point
ASE	Application Service Element
ASN.1	Abstract Syntax Notation One
BCD	Binary Coded Decimal
BER	Basic Encoding Rule
BNU-PEC	Buffered Network-Scheduled Uni-directional Pre-Established Connection
CM	Common Memory
cnf	confirmation primitive
DI_	Prefix for Data types defined for data-link layer types
DL	Data-link
DLC	Data-link Connection
DLCEP	Data-link Connection End Point
DLPDU	Data-link Protocol Data Unit
DLSAP	Data-link Service Access Point
DLSDU	Data-link Service Data Unit
Dt_	Prefix for Data types defined for Data type ASE
Err	Error (used to indicate an APDU type)
Er_	Prefix for Error types defined
Ev_	Prefix for Data types defined for Event ASE
FAL	Fieldbus Application Layer
Fi_	Prefix for Data types defined for Function Invocation ASE
FIFO	First In First Out
Gn_	Prefix for Data types defined for general use
ID	Identifier
IEC	International Electrotechnical Commission
in	input primitive
ind	indication primitive
ISO	International Organization for Standardization
LAS	Link Active Scheduler
Lr_	Prefix for Data types defined for Load Region ASE
lsb	least significant bit
Mn_	Prefix for Data types defined for Management ASE
msb	most significant bit
out	output primitive
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
QoS	Quality Of Service
Req	Request (used to indicate an APDU type)
req	request primitive
Rsp	Response (used to indicate an APDU type)
rsp	response primitive

SAP	Service Access Point
SDU	Service Data Unit
TCC	Time-critical cyclic
ToS	Type Of Service
Vr_	Prefix for Data types defined for Variable ASE

3.6 Conventions

3.6.1 Conventions for class definitions

The data-link layer mapping definitions are described using templates. Each template consists of a list of attributes for the class. The general form of the template is defined in IEC 61158-5.

3.6.2 Abstract syntax conventions

When the "optionalParametersMap" parameter is used, a bit number which corresponds to each OPTIONAL or DEFAULT production is given as a comment.

3.6.3 Conventions used in state machines

The state machines are described in Table 1.

Table 1 – Conventions used for state machines

#	Current state	Event / condition => action	Next state
Name of this transition	The current state to which this state transition applies	Events or conditions that trigger this state transaction. => The actions that are taken when the above events or conditions are met. The actions are always indented below events or conditions	The next state after the actions in this transition is taken

The conventions used in the state machines are as follows:

:= Value of an item on the left is replaced by value of an item on the right. If an item on the right is a parameter, it comes from the primitive shown as an input event.

xxx A parameter name.

Example:

Identifier := reason

means value of a 'reason' parameter is assigned to a parameter called 'Identifier.'

"xxx" Indicates fixed value.

Example:

Identifier := "abc"

means value "abc" is assigned to a parameter named 'Identifier.'

= A logical condition to indicate an item on the left is equal to an item on the right.

< A logical condition to indicate an item on the left is less than the item on the right.

> A logical condition to indicate an item on the left is greater than the item on the right.

<> A logical condition to indicate an item on the left is not equal to an item on the right.

&& Logical "AND"

|| Logical "OR"

This construct allows the execution of a sequence of actions in a loop within one transition. The loop is executed for all values from start_value to end_value.

Example:
for (Identifier := start_value to end_value)
 actions
endfor

This construct allows the execution of alternative actions depending on some condition (which might be the value of some identifier or the outcome of a previous action) within one transition.

Example:
If (condition)
 actions
else
 actions
endif

Readers are strongly recommended to refer to the subclauses for the AREP attribute definitions, the local functions, and the FAL-PDU definitions to understand protocol machines. It is assumed that readers have sufficient knowledge of these definitions, and they are used without further explanations.

4 FAL syntax description

4.1 Concept

This standard specifies the Application layer protocol of the Type 11 essential for the ISO/IEC 8802-3-based Time-critical control network (TCnet), which is one of the communication networks for the Real-Time Ethernet(RTE) defined in IEC 61784-2 and is referred to as RTE-TCnet hereafter.

This standard meets the industrial automation market objective of providing predictable time deterministic and reliable time-critical data transfer and means, which allow co-existence with non-time-critical data transfer over the ISO/IEC 8802-3 series communications medium, for support of cooperation and synchronization between automation processes on field devices in a real-time application system. 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.

This standard specifies the part of the protocol set of the RTE-TCnet communication profile and/or of one or more communication profiles related to a common family of the RTE-TCnet. The RTE-TCnet communication profile, shown in Figure 1 as one of the profile sets, is based on the 7 layer OSI Basic Reference model. For regular ISO/IEC 8802-3 based applications the upper layers mapped over the data-link layer is in the ordinary way; on the other hand, for time-critical applications with Common Memory running in parallel, the specific application layer for RTE-TCnet is specified. The data-link layer for RTE-TCnet has the extension, but is compliant to the ISO/IEC 8802-3 MAC protocol in order to provide both services for time-critical communications and common memory applications respectively.

	Regular ISO/IEC 8802-3-based applications	Time-critical applications with common memory	
Application layer	TELNET, FTP, HTTP OPC XML-DA etc	Common memory	
Transport layer	RFC 768(UDP) RFC 793 (TCP)		null
Network layer	RFC 791 (IP)		
Data Link layer	ISO/IEC 8802-3 Specific scheduling extension		
Physical layer	ISO/IEC 8802-3 (Redundant)		

Figure 1 – RTE-TCnet communication profile

This standard specifies the data-link protocol as the essential parts of the RTE-TCnet profile, which are the extension part of the ISO/IEC 8802-3 based data-link layer and the Application layer exploiting the services of the data-link layer immediately below, in terms of the “three-layer” Fieldbus Reference Model which is based in part on the OSI Basic Reference Model. Other part of the RTE-TCnet profile is not in the scope of this document.

4.1.1 Field of applications

In industrial control systems, several kinds of field devices such as drives, sensors and actuators, programmable controllers, distributed control systems and human-machine interface devices are required to be connected with control networks. The process control data and the state data is transferred among these field devices in the system and the communications between these field devices requires simplicity in application programming and to be executed with adequate response time. In most industrial automation systems such as food, water, sewage, paper and steel, including a rolling mill, the control network is required to provide time-critical response capability for their application, as required in ISO/TR 13283 for time-critical communications architectures.

Plant production may be compromised due to errors, which could be introduced to the control system if the network does not provide a time-critical response. Therefore the following characteristics are required for a time-critical control network.

- A deterministic response time between the control device nodes.
- Ability to share process data seamlessly across the control system.

The RTE-TCnet is applicable to such industrial automation environment, in which time-critical communications is primarily required. 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.

4.2 General

FAL Syntax description of the RTE-TCnet defines unconfirmed send service and consists of the three parts as follows, FalArHeader, InvokeID and Unconfirmed Service Request.

4.3 FAL-AR PDU abstract syntax

4.3.1 Top level definition

FalArPDU ::= UnconfirmedSend-CommandPDU

4.3.2 Unconfirmed send service

```
UnconfirmedSend-CommandPDU ::= SEQUENCE {
    FalArHeader,
    InvokeID OPTIONAL,
    UnconfirmedServiceRequest
}
```

4.4 Abstract syntax of PDU body

4.4.1 FalArHeader

```
FalArHeader ::= Unsigned8 {
    -- bit 8      FAL Protocol Specifier      (Always 1)
    -- bit 7-4    Protocol Identifier          (Identifiers abstract syntax revision, and encoding rules)
    -- bit 3      Protocol Specific bit       (Reserved for each protocol to use)
    -- bit 2-1    PDU Identifier              (Identifies a PDU type within a Protocol Identifier)
}
```

4.4.2 InvokeID

```
InvokeID ::= Unsigned8
```

4.4.3 Unconfirmed PDUs

```
UnconfirmedServiceRequest ::= SEQUENCE {
    CMArep, -- Block number
    CMDData -- content of CM segment
}
```

4.4.4 CMArep

```
CMArep ::= Unsigned16 -- DLCEP address
```

4.4.5 CMDData

```
CMDData ::= SEQUENCE {
    when Unsigned16, -- CM word length
    data SEQUENCE { ANY { -- CM content
        BitString8, BitString16, BitString32,
        Integer16, Integer32,
        Unsigned16, Unsigned32,
        Floating32,
        OctetString2, OctetString4,
        VisibleString2, VisibleString4,
        BinaryTime0, BinaryTime1, BinaryTime2, BinaryTime3, BinaryTime4, BinaryTime5,
        BinaryTime6, BinaryTime7
    }
}
```

4.5 Data type

BitString8, BitString16, BitString32, Integer16, Integer32, Unsigned16, Unsigned32, Floating32, OctetString2, OctetString4, VisibleString2, VisibleString4, BinaryTime0, BinaryTime1, BinaryTime2, BinaryTime3, BinaryTime4, BinaryTime5, BinaryTime6.

5 Transfer syntax

5.1 Overview and FAL header

All the FAL PDUs shall have the common PDU-header called FalArHeader. The FalArHeader identifies abstract syntax, transfer syntax, and each of the PDUs. Table 2 defines how this header shall be used.

Table 2 – FAL header

Bit position of the FalArHeader			Abstract syntax	Encoding rule	PDU type	Revision
8	7	6 5 4 3				
1	1	1 1 1 1	10	ASN.1	RTE-TCnet	UnconfirmedSendPDU
Revision1						
NOTE All other definitions are reserved.						

5.2 Encoding rule

5.2.1 Overview

The Encoding Rule of the RTE-TCnet is a preferable encoding rule that is compatible with existing standards. The FAL-PDUs encoded with the TER(Traditional Encoding Rule) shall have a uniform format. The FAL-PDUs shall consist of two major parts, the “APDU Header” part and the “APDU Body” part as shown in Figure 2.

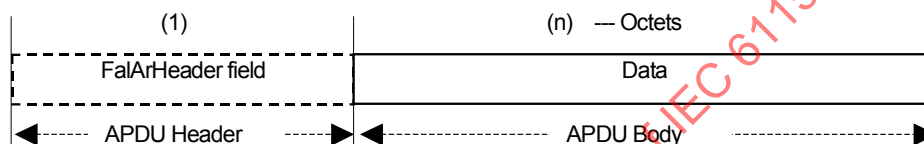


Figure 2 – APDU overview

5.2.2 APDU header encoding

The APDU Header part is always present in all APDUs which conform to this specification. It consists of one field: the FalArHeader field. Refer to 4.4.1 for the encoding rule of the FalArHeader field.

5.2.3 APDU body encoding

The Encoding Rule of FAL is based on the terms and definitions of the ISO/IEC 8825, and consists of the three encoding components given below. For time-critical and using fixed length data, Identifier octet and Length octets like TER (Traditional Encoding Rule) do not exist.

- Identifier octet
- Length octets
- Contents octets

NOTE Identification Octet and Content Length Octets do not exist in RTE-TCnet.

5.2.4 Encoding of simple variable

5.2.4.1 Encoding of a Boolean value

- The encoding of a Boolean value shall be primitive;
- the Identifier octet and Length octet(s) shall not be present;
- the Contents octets shall consist of a single octet;
- if the Boolean value is FALSE, the Contents octets shall be 0 (zero);
- if the Boolean value is TRUE, the Contents octets shall be 0xFF.

5.2.4.2 Encoding of a fixed length Integer value

- The encoding of a fixed-length Integer value of Integer8, Integer16 and Integer32 types shall be primitive;

- b) the Contents octets shall consist of exactly one, two or four octets, respectively;
- c) the Identifier octet and the Length octet(s) shall not be present;
- d) the Contents octets shall be a two's complement binary number equal to the integer value, and consist of bits 8 to 1 of the first octet, followed by bits 8 to 1 of the second octet, followed by bits 8 to 1 of each octet in turn up to and including the last octet of the Contents octets;
 - 1) the value of a two's complement binary number is derived by numbering the bits in the Content octets, starting with bit 1 of the first octet and ending the numbering with bit 8 of the last octet;
 - 2) each bit is assigned a numerical value of 2^{N-1} , where N is its position in the above numbering sequence;
 - 3) the value of the two's complement binary number is obtained by adding the numerical values assigned to each bit for those bits which are set to one, excluding bit 8 of the last octet, and then reducing the value by the numerical value assigned to bit 8 of the last octet if that bit is set to one.

5.2.5 Encoding of a fixed length Unsigned value

- a) The encoding of a fixed-length Integer value of Unsigned8, Unsigned16 and Unsigned32 types shall be primitive, and the Contents octets shall consist of exactly one, two or four octets, respectively;
- b) the Identifier octet and Length octet(s) shall not be present;
- c) the Contents octets shall be a binary number equal to the Unsigned value, and consist of bits 8 to 1 of the first octet, followed by bits 8 to 1 of the second octet, followed by bits 8 to 1 of each octet in turn up to and including the last octet of the Contents octets;
 - 1) the value of binary number is derived by numbering the bits in the Content octets, starting with bit 1 of the first octet as bit zero and ending the numbering with bit 8 of the last octet;
 - 2) each bit is assigned a numerical value of 2^{N-1} , where N is its position in the above numbering sequence;
 - 3) the value of the binary number is obtained by adding the numerical values assigned to each bit for those bits which are set to one.

5.2.6 Encoding of a Floating Point value

- a) The encoding of a Floating32 value shall be primitive, and the Contents octets shall consist of exactly four octets;
- b) the Identifier octet and Length octet(s) shall not be present;

NOTE The sign is encoded in bit 8 of the first octet. It is followed by the exponent starting from bit 7 of the first octet, and then the mantissa starting from bit 7 of the second octet for Floating32 and from bit 4 of the second octet for Floating64.

- c) the Contents octets shall contain floating point values defined in conformance with IEC 60559.

5.2.7 Encoding of a fixed length BitString value

- a) The encoding of a fixed-length Bitstring value of BitString8, BitString16, and BitString32 types shall be primitive, and the Contents octets shall consist of exactly one, two or four octets, respectively;
- b) the Identifier Octet and Length Octet(s) shall not be present;
- c) BitString value, commencing with the first bit and proceeding to the trailing bit, shall be placed in bits 8 to 1 of the first octet, followed by bits 8 to 1 of the second octet, followed by bits 8 to 1 of each octet up to and including the last octet of the Contents Octets.

5.2.8 Encoding of a fixed length Octet String value

- a) The encoding of a fixed-length OctetString value of OctetString2, and OctetString4 types shall be primitive, and the Contents octets shall consist of exactly two, or four octets respectively;
- b) the Identifier octet and Length octet(s) shall not be present;
- c) the Contents octets shall be equal in value to the octets in the data value, in the order they appear in the data value, and with the most significant bit of an octet of the data value aligned with the most significant bit of an octet of the Contents octets.

5.2.9 Encoding of a fixed length Visible String value

- a) The encoding of a fixed-length VisibleString value of VisibleString2, and VisibleString4 types shall be primitive, and the Contents octets shall consist of exactly two, or four octets respectively;
- b) the Identifier octet and Length octet(s) shall not be present;
- c) the Contents octets shall be equal in value to the octets in the data value, in the order they appear in the data value, and with the most significant bit of an octet of the data value aligned with the most significant bit of an octet of the Contents octets.

5.2.10 Encoding of BinaryTime value

- a) The encoding of a BinaryTime0, BinaryTime1, BinaryTime2, BinaryTime3, BinaryTime4, BinaryTime5, BinaryTime6, and BinaryTime7 value shall be primitive;
- b) the Identifier octet and Length octet(s) shall not be present;
- c) the Contents octets shall be a binary number equal to the binary time value, and consisting of bits 8 to 1 of the first octet, followed by bits 8 to 1 of the second octet, followed by bits 8 to 1 of each octet in turn up to and including the last octet of the Contents octets;
 - 1) the Contents octets of a BinaryTime0, BinaryTime1, BinaryTime2, and BinaryTime3 value shall consist of two octets;
 - 2) the Contents octets of a BinaryTime4, BinaryTime5, BinaryTime6, and BinaryTime7 value shall consist of four octets.

NOTE The value of the granularity of each BinaryTime type is defined in IEC 61158-5 subseries.

5.3 Encoding of structured types

5.3.1 General

When the structured type is also encoded, the identifier or length of the structure are not provided in RTE-Tcnet.

5.3.2 Encoding of a SEQUENCE value

The SEQUENCE type is comparable to a record. It represents a collection of user data of the same or of different Data types.

A SEQUENCE type value may contain a simple variable, or a further structured variable as its components. If a SEQUENCE type contains another structured type value, it shall be counted as a single component even if it contains several components.

6 FAL protocol state machines structures

6.1 Overview

As shown in Figure 3, the protocol machine of FAL consists of three, the FAL service protocol machine (FSPM), the application relationship protocol machine (ARPM) and the data-link layer mapping protocol machine (DMPM).

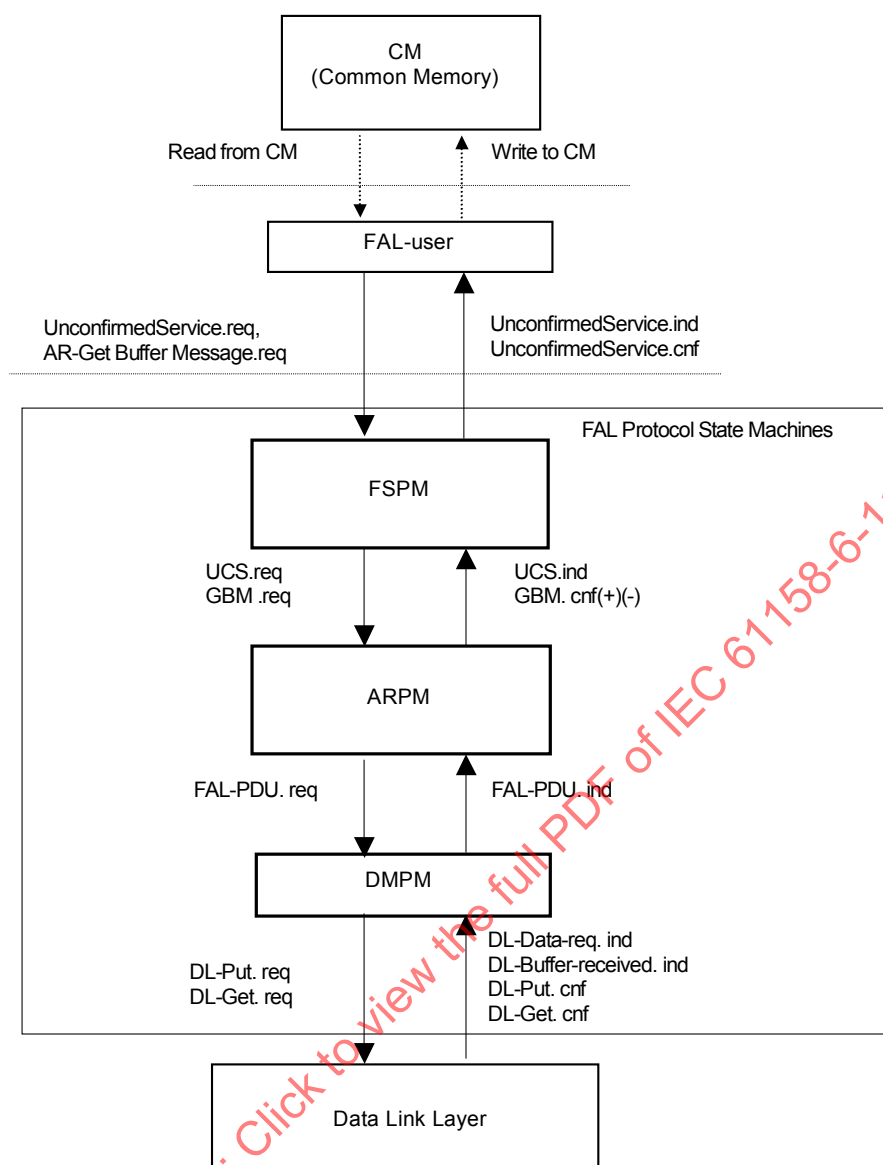


Figure 3 – Relationship between FSPM, ARPM, DMPM and external physical CM

7 FAL service protocol machine (FSPM)

7.1 General

The FAL Service Protocol Machine (FSPM) is common to all the AREP types. Only applicable primitives are different among different AREP types. It has one state called "ACTIVE".

NOTE Although now present, the type of AREP is only one.

7.2 Primitives definitions

7.2.1 Primitive exchanged between FAL user and FSPM

The primitive exchanged between the FAL user and the FSPM are described in Table 3 and Table 4.

Table 3 – Primitives issued by FAL user to FSPM

Primitive names	Source	Associated parameters	Function
UCS.req	FAL user	Arep-id, Data	This is an FAL internal primitive used to convey an Unconfirmed Send (UCS) request primitive from the FAL user to the FSPM
GBM.req	FAL user	Arep-id	This is an FAL internal primitive used to convey a Get-Buffered-Message (GBM) request primitive from the FAL user to the FSPM

Table 4 – Primitives issued by FSPM to FAL user

Primitive names	Source	Associated parameters	Function
UCS.ind	FSPM	Arep-id, Data	This is an FAL internal primitive used to convey an Unconfirmed Send (UCS) indication primitive from the FSPM to the FAL user
GBM.cnf(+)	FSPM	Arep-id, Data	This is an FAL internal primitive used to convey a Get-Buffered-Message (GBM) positive confirmation from the FSPM to the FAL user
GBM.cnf(-)	FSPM	Arep-id	This is an FAL internal primitive used to convey a Get-Buffered-Message (GBM) negative confirmation from the FSPM to the FAL user
FSTS.ind	FSPM	Arep-id, Reported-status	This is an FAL internal primitive used to convey a FAL-Status (FSTS) indication primitive from the FSPM to the FAL user

7.2.2 Parameters of FAL user /FSPM

All the parameters used in the primitives exchanged between the FAL user and the FSPM are identical to those defined in the "Operational Service" subclause.

7.3 FSPM state tables

7.3.1 General

The FSPM state machines are described in Figure 4, Table 5 and Table 6.

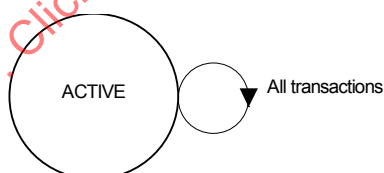


Figure 4 – State transition diagram of FSPM

Table 5 – FSPM state table – sender transactions

#	Current state	Event or condition => action	Next state
S1	ACTIVE	UCS.req && SelectArep (Arep-id) = "True" => UCS.req { user-data := Data }	ACTIVE
S2	ACTIVE	GBM.req && SelectArep (Arep-id) = "True" => GBM.req { }	ACTIVE

NOTE 1 A primitive parameter in the FSPM sender state machine is sent to an appropriate ARPM that is selected by the FSPM using the SelectArep function.
NOTE 2 If the SelectArep function return the value of False, it is a local matter to report such instance and the FSPM does not generate any primitive for the ARPM.

Table 6 – FSPM state table – receiver transactions

#	Current state	Event or condition => action	Next state
R1	ACTIVE	UCS.ind => UCS.ind { Arep-id := arep-id, Data := user-data }	ACTIVE
R2	ACTIVE	GBM.cnf(+) => GBM.cnf (+) { Arep-id := arep, Data := user-data }	ACTIVE
R3	ACTIVE	GBM.cnf(-) => GBM.cnf (-) { Arep-id := arep-id }	ACTIVE
R4	ACTIVE	FAL-STS.ind => FSTS.ind { Arep-id := arep-id, Reported-status := reported-status }	ACTIVE

7.3.2 Functions

The function used in this state machine is as shown in Table 7.

Table 7 – Function SelectArep

Name	SelectArep	Used in	FSPM
input		Output	
Arep-id		True False	
Function	Looks for the AREP entry that is specified by the Arep-id parameter. True means the AREP exists.		

8 Application relationship protocol machine (ARPM)

8.1 General

The RTE-TCnet define a Application Relation (AR) and their associated ARPM, which is the Buffered network-scheduled unidirectional - pre-established connection (BNU-PEC).

8.2 Primitive definitions

8.2.1 Primitives exchanged between ARPM and FSPM

Table 8 and Table 9 list the primitives exchanged between the FSPM and the ARPM.

Table 8 – Primitives issued by FSPM to ARPM

Primitive names	Source	Associated parameters	Function
UCS.req	FSPM	user-data	This is an FAL internal primitive used to convey Unconfirmed Send (UCS) request primitive from the FSPM to the ARPM
GBM.req	FSPM	(none)	This is an FAL internal primitive used to convey a Get-Buffered-Message (GBM) request primitive from the FSPM to the ARPM

Table 9 – Primitives issued by ARPM to FSPM

Primitive names	Source	Associated parameters	Function
UCS.ind	ARPM	arep-id, user-data	This is an FAL internal primitive used to convey an Unconfirmed Send (UCS) indication primitive from the ARPM to the FSPM
GBM.cnf(+)	ARPM	arep-id	This is an FAL internal primitive used to convey a Get-Buffered-Message (GBM) positive confirmation from the ARPM to the FSPM
GBM.cnf(-)	ARPM	arep-id	This is an FAL internal primitive used to convey a Get-Buffered-Message (GBM) negative confirmation from the ARPM to the FSPM
FSTS.ind	ARPM	arep-id, reported-status	This is an FAL internal primitive used to convey a FAL-Status (FSTS) indication primitive from the ARPM to the FSPM

8.2.2 Parameters of FSPM/ARPM primitives

The parameters used with the primitives exchanged between the FSPM and the ARPM are described in Table 10.

Table 10 – Parameters used with primitives exchanged between FSPM and ARPM

Parameter name	Description
arep-id	This parameter is used to unambiguously identify an instance of the AREP that has issued a primitive. A means for such identification is not specified by this specification
user-data	This parameter conveys a FAL-User data
identifier	This parameter conveys value that is used for the Identifier parameter
reason	This parameter conveys value that is used for the Reason-Code parameter
status	This parameter conveys value that is used for the Status parameter
reported-status	This parameter conveys a data-link layer event status

8.3 DLL mapping of BNU-PEC AREP class

This subclause describes the mapping of the BNU-PEC AREP class to the RTE data-link layer. It does not redefine the DLCEP attributes that are or will be defined in the data-link layer specification; rather, it defines how they are used by this AR class.

The DLL Mapping attributes, their permitted values and the DLL services used with the BNU-PEC AREP class are defined in this subclause.

CLASS: BNU-PEC

PARENT CLASS: BufferedNetworkScheduledUnidirectionalPre-EstablishedConnectionAREP

ATTRIBUTES:

1 (m) KeyAttribute: LocalDlcepIdentifier

2 (m) Attribute: Role (Publisher, Subscriber)

DLL SERVICES:

1 (m) OpsService: DL-Put
 2 (m) OpsService: DL-Get
 3 (m) OpsService: DL-Buffer-received
 4 (m) OpsService: DL-Data-req

8.3.1 Attributes

8.3.1.1 LocalDlcepldentifier

This attribute specifies the local DLCEP-identifier of a DL-Put or DL-Get primitive and thus it identifies the DLCEP.

8.3.1.2 Role

This attribute specifies the role of this AREP. The value of "Publisher" indicates that this AREP is used as a publisher. The value of "Subscriber" indicates that this AREP is used as a subscriber.

8.3.2 DLL services

Refer to 9.4, for the DLL service descriptions.

8.4 BNU-PEC ARPM states machine

8.4.1 BNU-PEC ARPM state

The defined states together with their descriptions of the BNU-PEC ARPM are listed in Table 11 and Figure 5.

Table 11 – BNU-PEC state descriptions

State name	Description
OPEN	The BNU-PEC in the OPEN state is defined and capable of sending or receiving FAL-PDUs



Figure 5 – State transition diagram of the BNU-PEC

8.4.2 BNU-PEC ARPM state table

Table 12 and Table 13 define the BNU-PEC state machines.

Table 12 – BNU-PEC ARPM state table – sender transactions

#	Current state	Event or condition => action	Next state
S1	OPEN	UCS.req && Role = "Publisher" => FAL-PDU.req { dmpm-service-name := "DMPM_Put_req", arep-id := GetArepld (), dlsdu := BuildFAL-PDU (fal-pdu-name := "UCS.PDU", fal-data := user-data) } }	OPEN
S2	OPEN	GBM.req && Role = "Subscriber" => FAL-PDU.req { dmpm-service-name := "DMPM_Get_req" , arep-id := GetArepld () } }	OPEN

Table 13 – BNU-PEC ARPM state table – receiver transactions

#	Current state	Event or condition => action	Next state
R1	OPEN	FAL-PDU.ind && Role = "Subscriber" && dmpm-service-name = "DMPM_Buffer_Received_ind" && FAL_Pdu_Type (fal-pdu) = "UCS-PDU" => UCS.ind { arep-id := GetArepld (), user-data := fal-pdu } }	OPEN
R2	OPEN	FAL-PDU.ind && Role = "Subscriber" && dmpm-service-name = "DMPM_Get_cnf " && reason = "success " => GBM.cnf (+) { arep-id := GetArepld (), user-data := fal-pdu } }	OPEN
R3	OPEN	FAL-PDU.ind && Role = "Subscriber" && dmpm-service-name = "DMPM_Get_cnf " && reason <> "success " => GBM.cnf(-) { arep-id := GetArepld () } }	OPEN
R4	OPEN	FAL-PDU.ind && Role = "Publisher" && dmpm-service-name = "DMPM_DATA_REQ_ind" => FSTS.ind { arep-id := GetArepld (), reported-status := "DATA-REQ" } }	OPEN
R5	OPEN	ErrorToARPM => (no action token, see note) NOTE It is a local matter to report this error status to network management entities.	OPEN

8.4.3 Functions used by BNU-PEC ARPM

Table 14, Table 15 and Table 16 define the function used by this service machine.

Table 14 – Function GetArepld ()

Name	GetArepld()	Used in	ARPM
input		Output	
(none)		AREP Identifier	
Function			
Returns a value that can unambiguously identify the current AREP.			

Table 15 – Function BuildFAL-PDU

Name	BuildFAL-PDU	Used in	ARPM	
input		Output		
fal-pdu-name, fal-data		dl pdu		
Function				
Builds an FAL-PDU out of the parameters given as input variables.				

Table 16 – Function FAL_Pdu_Type

Name	FAL_Pdu_Type	Used in	ARPM	
input		Output		
fal-pdu		One of the FAL-PDU types defined in the subclause 9.4.		
Function				
Decodes the FAL-PDU that is conveyed in the fal-pdu parameter and retrieves one of the FAL-PDU types.				

9 DLL mapping protocol machine (DMPM)

9.1 Overview

The DLL Mapping Protocol Machine is common to all the AREP types. Only applicable primitives are different among different AREP types.

NOTE Although now present, the type of AREP is only one.

a) Remarks about DLCEP-identifier

The RTE-TCnet data-link layer specification defines local DLCEP-identifiers to distinguish the pre-defined connections pre-established by a local means. This DLCEP-identifier is key property of FAL ARPMs, and included a parameter as DMPM primitive.

b) Remarks about DLSDU-length

The RTE-TCnet data-link layer specification defines the parameter of DLSDU-length to distinguish the end of each DLSDU.

In the specification of the RTE-TCnet data-link layer, the DLSDU-length is defined to partition the end of the DLSDU. In actual implementation, usage of this parameter depends on implementation. In this DMPM, DLSDU-length is not included.

c) Remarks about configuration and initialization of the data-link layer

The RTE-TCnet data-link layer specification defines the configuration service to set the resource of layer or the class of connection.

In the specification of the RTE-TCnet data-link layer, the configuration service is defined to set the class of resource and connection.

9.2 Primitive definitions

9.2.1 Primitives exchanged between DMPM and ARPM

Table 17 and Table 18 list the service primitives between the ARPM and the DMPM.

Table 17 – Primitives issued by ARPM to DMPM

Primitive names	Source	Associated parameters	Function
FAL-PDU.req	ARPM	dmpm-service-name, arep-id, local-dlcep-identifier, reason, response_request, dlsdu	This primitive is used to request the DMPM to transfer an FAL-PDU. It passes the FAL-PDU to the DMPM as a DLSDU. It also carries some of the data-link layer parameters that are referenced there

Table 18 – Primitives issued by DMPM to ARPM

Primitive names	Source	Associated parameters	Function
FAL-PDU.ind	DMPM	dmpm-service-name, reason, response_request, fal-pdu	This primitive is used to pass an FAL-PDU received as a data-link layer service data unit to a designated ARPM. It also carries some of the data-link layer parameters that are referenced in the ARPM
ErrorToARPM	DMPM	reason	This primitive is used to convey selected communication errors reported by the data-link layer to a designated ARPM

9.2.2 Parameters of ARPM/DMPM primitives

The parameters used with the primitives exchanged between the ARPM and the DMPM are described in Table 19.

Table 19 – Parameters used with primitives exchanged between ARPM and DMPM

Parameter name	Description
arep-id	This parameter carries a local identifier to specify the associated AR instance
dmpm-service-name	This parameter conveys a DMPM pseudo-service name or s data-link layer service name. Possible value are represented as DMPM_XXXX_yyy
dls_user_data	This parameter conveys the value of the dl-dls-user-data parameter
dlsdu	This parameter conveys the value of the dl-dls-user-data parameter
fal-pdu	This parameter conveys the value of the dl-dls-user-data parameter
local-dlcep-identifier	This parameter conveys the value of the Requesting pre-established AREP parameter. and the value of the Responding pre-established AREP parameter
reason	This parameter conveys the value of the dl_reason parameter
status	This parameter conveys the value of the dl-status parameter

9.2.3 Primitives exchanges between data-link layer and DMPM

Table 20 summarizes the primitives exchanged between the DLL and the DMPM.

Table 20 – Primitives exchanged between data-link layer and DMPM

Primitive names	Source	Associated parameters
DL-Buffer-received.ind	Data-link layer	dmpm-service-name, reason, response_request, fal-pdu
DL-Data-req.ind	Data-link layer	reason
DL-Get.req (out)	Data-link layer	dl_dlcep_identifier, dl_dlsdu, dl-status
DL-Put.req (out)	Data-link layer	dl_dlcep_identifier, dl-status
DL-Get.req (in)	DMPM	dl_dlcep_identifier,
DL-Put.req (in)	DMPM	dl_dlcep_identifier, dl-dls-user-data

9.2.4 Parameters of DMPM/data-link layer primitives

The parameters used with the primitives exchanged between the DMPM and the data-link layer are identical to those defined in IEC 61158-3-11. They are prefixed by "dl_" to indicate that they are used by the FAL.

9.3 DLL mapping protocol machine (DMPM)

9.3.1 DMPM States

The defined state of the DMPM together with its description are listed in Table 21 and Figure 6.

Table 21 – DMPM state descriptions

State name	Description
ACTIVE	The DMPM in the ACTIVE state is ready to transmit or receive primitives to or from the data-link layer and the ARPM

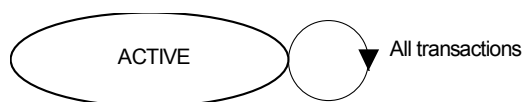


Figure 6 – State transition diagram of DMPM

9.3.2 DMPM state table

The DMPM state machines are defined in Table 22 and Table 23.

NOTE Although each primitive contains all the available parameters, only those applicable to particular ARPM are relevant.

Table 22 – DMPM state table – sender transactions

#	Current state	Event or condition => action	Next state
S1	ACTIVE	FAL-PDU.req && dmpm-service-name = "DMPM_Put_req" => PickArep (arep-id), DL-Put.req (in) { dl-local-dlcep-identifier := local-dlcep-identifier, dl-dls-user-data := dlsdu } DL-Put.req (out) -- immediate response && dl-status = "success" => FAL-PDU.ind { dmpm-service-name := "DMPM_Put_cnf" , reason := dl-status } DL-Put.req (out) -- immediate response && dl-status <> "success" => FAL-PDU.ind { dmpm-service-name := "DMPM_Put_cnf" , reason := dl-status } ErrorToARPM { reason := dl-status }	ACTIVE
S2	ACTIVE	FAL-PDU.req && dmpm-service-name = "DMPM_Get_req" => PickArep (arep-id), DL-Get.req (in) { dl-local-dlcep-identifier := local-dlcep-identifier } DL-Get.req (out) -- immediate response => FAL-PDU.ind { dmpm-service-name := "DMPM_Get_cnf" , reason := dl-status }	ACTIVE

Table 23 – DMPM state table – receiver transactions

#	Current state	Event or condition => action	Next state
R1	ACTIVE	DL-Buffer-received.ind && FindAREP () = "False" => (no action token)	ACTIVE
R2	ACTIVE	DL-Buffer-received.ind && FindAREP () = "True" => DL-Get.req (in) { dl-local-dlcep-identifier := local-dlcep-identifier } DL-Get.req (out) -- immediate response && dl-status = "success" => FAL-PDU.ind { dmpm-service-name := "DMPM_Buffer_Received_ind", fal-pdu := dl-dls-user-data, reason := dl-status } DL-Get.req (out) -- immediate response && dl-status <> "success" => FAL-PDU.ind { dmpm-service-name := "DMPM_Get_cnf", reason := dl-status } ErrorToAPM { reason := dl-status }	ACTIVE
R3	ACTIVE	DL-Data-req.ind && FindAREP () = "False" => (no action token)	ACTIVE
R4	ACTIVE	DL-Data-req.ind && FindAREP () = "True" => FAL-PDU.ind { dmpm-service-name := "DMPM_DATA_REQ" }	ACTIVE

9.3.3 Function used by DMPM

Table 24 and Table 25 define the function used by the DMPM.

Table 24 – Function PickArep

Name	PickArep	Used in	DMPM
input		Output	
arep-id		(all the attributes of the specified AREP)	
Function	Selects the attributes for the AREP specified by the arep-id parameter. After this function is executed, the attributes of the selected AREP are available to the state machine.		

Table 25 – Function FindAREP

Name	FindAREP	Used in	DMPM
input		Output	
local-dlcep-identifier		True False	
Function	Identifies the AREP that shall be bound with an active DMPM. True means the AREP exists. If it does, this function also returns a means to send a DMPM primitive to that AREP.		

9.4 Data-link layer service selection

9.4.1 General

This subclause briefly describes the data-link layer service utilized by the FAL. These data-link layer services are fully defined in IEC 61158-3-11.

NOTE The FAL assumes that a resource, such as buffers, is set up prior to any operations of FAL protocol machines by a local means. Therefore, this service is not listed in this subclause.

9.4.2 DL-Put

This service is used to copy an FAL-PDU to a buffer. It refers to the Put service.

9.4.3 DL-Get

This service is used to read an FAL-PDU from a buffer. It refers to the Get service.

9.4.4 DL-Buffer-received

The DL-Buffer-received service is used to inform the FAL of new update on the specified receive buffer.

9.4.5 DL-Data-req

The DL-Data-req service is used to inform the FAL that the specified buffer content became update timing.

IECNORM.COM : Click to view the full PDF of IEC 61158-6-11:2007

Bibliography

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

IEC/TR 61158-1 (Ed.2.0), *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-4-11, *Industrial communication networks – Fieldbus specifications – Part 4-11: Data-link layer protocol specification – Type 11 elements*

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

ISO/IEC 8649, *Information technology – Open Systems Interconnection – Service definition for the Association Control Service Element*

ISO/IEC 8650 (all parts), *Information technology – Open Systems Interconnection – Connection-oriented protocol for the Association Control Service Element: Protocol specification*

ISO/IEC 8802-3, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*

ISO/IEC 9506-2, *Industrial automation systems – Manufacturing Message Specification – Part 2: Protocol specification*

ISO/IEC 10646 (all parts), *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*

SOMMAIRE

AVANT-PROPOS	34
INTRODUCTION.....	36
1 Domaine d'application	37
1.1 Généralités.....	37
1.2 Spécifications	38
1.3 Conformité	38
2 Références normatives.....	38
3 Termes, définitions, symboles, abréviations et conventions	39
3.1 Introduction	39
3.2 Termes et définitions provenant d'autres normes ISO/CEI	39
3.3 Termes et définitions provenant de la norme CEI/TR 61158-1	40
3.4 Autres termes et définitions	40
3.5 Abréviations et symboles.....	41
3.6 Conventions	42
4 Description de la syntaxe FAL	44
4.1 Concept	44
4.2 Généralités.....	45
4.3 Syntaxe abstraite FAL-AR PDU	46
4.4 Syntaxe abstraite de corps PDU	46
4.5 Data type	46
5 Syntaxe de transfert	47
5.1 Présentation générale et en-tête de FAL	47
5.2 Règle de codage	47
5.3 Codage des types structurés	49
6 Structures des diagrammes d'états de protocole FAL	50
6.1 Présentation générale	50
7 Machine de protocole de service FAL (FSPM)	51
7.1 Généralités.....	51
7.2 Définitions des primitives.....	51
7.3 Tableaux d'états de la machine FSPM.....	52
8 Machine de protocole de relations entre applications (ARPM)	53
8.1 Généralités.....	53
8.2 Définitions des primitives.....	53
8.3 Mapping DLL de la classe BNU-PEC AREP	54
8.4 Diagramme d'états BNU-PEC ARPM	55
9 Machine de protocole de mapping DLL (DMPM)	57
9.1 Présentation générale	57
9.2 Définitions des primitives.....	58
9.3 Machine de protocole de mapping DLL (DMPM)	59
9.4 Choix d'un service de couche Liaison de données	62
Bibliographie.....	63
Figure 1 – Profil de communication RTE-TCnet.....	45
Figure 2 – Présentation générale de l'APDU	47
Figure 3 – Relations entre FSPM, ARPM, DMPM et CM physique externe	51

Figure 4 – Diagramme de passage d'états de la machine FSPM	52
Figure 5 – Diagramme de passage d'états de BNU-PEC	55
Figure 6 – Diagramme de passage d'états de la machine DMPM	59
Tableau 1 – Conventions utilisées dans les diagrammes d'états	43
Tableau 2 – En-tête de FAL	47
Tableau 3 – Primitives échangées entre l'utilisateur FAL et la FSPM	51
Tableau 4 – Primitives échangées entre la FSPM et l'utilisateur FAL	51
Tableau 5 – Tableau d'états de la machine FSPM – Transactions de l'émetteur.....	52
Tableau 6 – Tableau d'états de la machine FSPM – Transactions du récepteur	52
Tableau 7 – Fonction SelectArep	53
Tableau 8 – Primitives échangées entre les machines FSPM et ARPM	53
Tableau 9 – Primitives échangées entre les machines ARPM et FSPM	53
Tableau 10 – Paramètres utilisés avec les primitives échangées entre les machines FSPM et ARPM.....	54
Tableau 11 – Description des états BNU-PEC.....	55
Tableau 12 – Tableau d'états BNU-PEC ARPM – Transactions de l'émetteur.....	55
Tableau 13 – Tableau d'états BNU-PEC ARPM – Transactions du récepteur	56
Tableau 14 – Fonction GetArepId ()	56
Tableau 15 – Fonction BuildFAL-PDU.....	57
Tableau 16 – Fonction FAL_Pdu_Type	57
Tableau 17 – Primitives échangées entre les machines ARPM et DMPM	58
Tableau 18 – Primitives échangées entre les machines DMPM et ARPM	58
Tableau 19 – Paramètres utilisés avec les primitives échangées entre les machines ARPM et DMPM.....	58
Tableau 20 – Primitives échangées entre la couche Liaison de données et la machine DMPM	59
Tableau 21 – Description des états DMPM.....	59
Tableau 22 – Tableau d'états DMPM – Transactions de l'émetteur.....	60
Tableau 23 – Tableau d'états DMPM – Transactions du récepteur	61
Tableau 24 – Fonction PickArep	61
Tableau 25 – Fonction FindAREP	61

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 6-11: Spécification du protocole de la couche d'application – Éléments de Type 11

AVANT-PROPOS

- 1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) La CEI elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de la CEI. La CEI n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et de ne pas avoir signalé leur existence.

NOTE Certains des types de protocoles associés font l'objet d'une utilisation restreinte par les détenteurs de leurs droits de propriété intellectuelle. Dans tous les cas, l'engagement de renonciation partielle aux droits de propriété intellectuelle, pris par les détenteurs de ces droits, autorise l'utilisation d'un type de protocole de couche Liaison de données particulier avec des protocoles de couche physique et de couche Application dans les combinaisons de Types, comme spécifié explicitement dans la série CEI 61784. L'utilisation des divers types de protocoles dans d'autres combinaisons peut nécessiter l'autorisation de leurs détenteurs de droits de propriété intellectuelle respectifs.

La commission électrotechnique internationale (CEI) attire l'attention sur le fait qu'il est déclaré que la conformité avec les dispositions de la présente norme peut impliquer l'utilisation d'un brevet intéressant ce qui suit:

TOSHIBA possède les applications de brevets énumérées ci-dessous:

- Numéro de publication US 6711131 et ses brevets correspondants dans d'autres pays
- Numéro de publication US 5414813 et ses brevets correspondants dans d'autres pays
- Numéro de publication US 4930121 et ses brevets correspondants dans d'autres pays

La CEI ne prend pas position quant à la preuve, à la validité et à la portée de ces droits de propriété.

Le détenteur de ces droits de propriété a donné l'assurance à la CEI qu'il consent à négocier des licences avec des demandeurs du monde entier, à des termes et conditions raisonnables et non discriminatoires. A ce propos, la déclaration du détenteur des droits de propriété est enregistrée à la CEI. Des informations peuvent être demandées à:

Toshiba Corporation
1-1, Shibaura 1-Chome
Minato-ku Tokyo 105-8001, Japon

L'attention est d'autre part attirée sur le fait que certains des éléments de la présente Norme internationale peuvent faire l'objet de droits de propriété autres que ceux qui ont été mentionnés ci-dessus. La CEI ne saurait être tenue pour responsable de l'identification de ces droits de propriété en tout ou partie.

La Norme internationale CEI 61158-6-11 a été établie par le sous-comité 65C: Réseaux industriels, du comité d'études 65 de la CEI: Mesure, commande et automation dans les processus industriels.

Cette première édition et les parties associées de la sous-série CEI 61158-6 annulent et remplacent la CEI 61158-6:2003. L'édition de la présente partie constitue une addition technique. La présente partie et les parties de Type 11 associées annulent et remplacent également la CEI/PAS 62406 publiée en 2005.

La présente version bilingue (2013-09) correspond à la version anglaise monolingue publiée en 2007-12.

Le texte anglais de cette norme est issu des documents 65C/476/FDIS et 65C/487/RVD.

Le rapport de vote 65C/487/RVD donne toute information sur le vote ayant abouti à l'approbation de cette norme.

La version française de cette norme n'a pas été soumise au vote.

La présente édition de la CEI 61158-6 comprend les modifications significatives suivantes par rapport à l'édition précédente:

- a) suppression de l'ancien bus de terrain de Type 6 faute d'adaptation au marché;
- b) ajout de nouveaux types de bus de terrain;
- c) division de la partie 6 de la troisième édition en plusieurs parties numérotées -6-2, -6-3, ...

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de la CEI sous <http://webstore.iec.ch> dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite;
- supprimée;
- remplacée par une édition révisée, ou
- amendée.

NOTE La révision de la présente norme fera l'objet d'une synchronisation avec les autres parties de la série CEI 61158.

La liste de toutes les parties de la série CEI 61158, publiée sous le titre général *Réseaux de communications industriels – Spécifications des bus de terrain*, est disponible sur le site web de la CEI.

INTRODUCTION

La présente partie de la CEI 61158 s'inscrit dans une série créée pour faciliter l'interconnexion des composants de systèmes d'automatisation. Elle renvoie aux autres normes de l'ensemble défini par le modèle de référence de bus de terrain "à trois couches" décrit dans la CEI/TR 61158-1.

Le protocole d'application fournit le service d'application au moyen des services disponibles au niveau de la couche de liaison de données ou de la couche immédiatement inférieure. Le principal objectif de la présente norme est de définir un ensemble de règles de communication, exprimées en termes de procédures devant être suivies par les entités d'application (*Application Entity*, AE) homologues au moment de la communication. Ces règles de communication visent à fournir une base saine pour le développement, dans divers buts:

- en tant que guide pour les développeurs et les concepteurs;
- dans une optique d'utilisation lors des essais et de l'achat de matériel;
- dans le cadre d'un accord pour l'admission de systèmes dans l'environnement de systèmes ouverts;
- en tant que précision apportée à la compréhension des communications en temps critique dans le modèle OSI.

La présente norme traite, en particulier, de la communication et de l'interfonctionnement des capteurs, effecteurs et autres appareils d'automatisation. L'utilisation conjointe de la présente norme avec d'autres normes entrant dans les modèles de référence OSI ou de bus de terrain permet à des systèmes qui ne pourraient pas sans cela, de fonctionner ensemble dans toute combinaison.

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 6-11: Spécification du protocole de la couche d'application – Éléments de Type 11

1 Domaine d'application

1.1 Généralités

La couche d'application de bus de terrain (*Fieldbus Application Layer*, FAL) procure aux programmes de l'utilisateur un moyen d'accès à l'environnement de communication des bus de terrain. A cet égard, la FAL peut être considérée comme une "fenêtre entre programmes d'application correspondants."

La présente norme fournit des éléments communs pour les communications en temps critique ou non entre des programmes d'application dans un environnement et avec un matériel d'automatisation propres aux bus de terrain de Type 11. Le terme "en temps critique" signale l'existence d'une fenêtre temporelle dans laquelle est exigée la réalisation d'une ou de plusieurs actions spécifiées, avec un niveau de certitude défini. La non-réalisation des actions spécifiées dans le cadre de la fenêtre temporelle peut mener à l'échec des applications sollicitant les actions, avec un risque annexe pour les équipements, l'installation et éventuellement la vie humaine.

La présente norme définit de manière abstraite le comportement, visible par un observateur externe, assuré par la couche d'application de bus de terrain, en termes

- a) de syntaxe abstraite définissant les unités de données de protocole de la couche d'application transmises entre les entités d'application en communication,
- b) de syntaxe de transfert définissant les unités de données de protocole de la couche d'application, transmises entre les entités d'application en communication,
- c) de diagrammes d'états de contexte d'application définissant le comportement de service d'application observable entre les entités d'application en communication; et
- d) de diagrammes d'états de relations entre applications définissant le comportement de communication visible entre les entités d'application en communication.

La présente norme vise à spécifier le protocole mis en place pour définir

- 1) la représentation filaire des primitives de service définies dans la CEI 61158-5-11, et
- 2) le comportement, visible par un observateur externe, associé à leur transfert.

La présente norme spécifie le protocole de la couche d'application de bus de terrain CEI, en conformité avec le modèle de référence de base OSI (ISO/CEI 7498) et avec la structure de la couche d'application OSI (ISO/CEI 9545).

Les services et protocoles de couche FAL sont fournis par des entités AE de couche FAL contenues dans les processus d'application. Une entité AE de couche FAL se compose d'un ensemble d'éléments de service d'application (*Application Service Elements*, ASE) orientés objet et d'une entité de gestion de couches (*Layer Management Entity*, LME) qui gère l'entité AE. Les éléments ASE délivrent des services de communication agissant sur un ensemble de classes d'objets de processus d'application (*Application Process Object*, APO) associées. L'un des éléments ASE de couche FAL est un élément ASE de gestion qui fournit un ensemble commun de services destinés à la gestion des instances des classes de couche FAL.

Bien que ces services spécifient, du point de vue des applications, les modalités d'émission et de remise des demandes et des réponses, ils ne comprennent pas de spécification du traitement que sont tenues de faire les applications demandeuse et répondeuse. En d'autres termes, les aspects comportementaux des applications ne sont pas définis, seule une définition des demandes et des réponses que ces applications peuvent envoyer/recevoir est établie. Cela laisse une plus grande marge de manœuvre aux utilisateurs de la couche FAL dans la normalisation du comportement de ces objets. Outre ces services, la présente norme définit également certains services de soutien donnant accès à la couche FAL dans un but de commande de certains aspects de son fonctionnement.

1.2 Spécifications

La présente norme a pour principal objet de spécifier la syntaxe et le comportement du protocole de la couche d'application qui transfère les services correspondants définis dans la CEI 61158-5-11.

Un objectif secondaire consiste à fournir des voies d'évolution à partir des protocoles de communication industriels antérieurs. Ce dernier objectif explique la diversité des protocoles normalisés dans les parties de la série CEI 61158-6.

1.3 Conformité

La présente norme ne définit pas de mises en œuvre, ni de produits particuliers, pas plus qu'elle ne limite les mises en œuvre des entités de la couche d'application dans les systèmes d'automatisation industriels.

Il n'existe pas de conformité de l'équipement à la norme de définition de service de la couche d'application. En revanche, la mise en œuvre de cette spécification du protocole de la couche d'application permet d'obtenir cette conformité.

2 Références normatives

Les documents de référence suivants sont indispensables pour l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC 60559, *Binary floating-point arithmetic for microprocessor systems* (disponible en anglais seulement)

IEC 61158-3-11, *Industrial communication networks – Fieldbus specifications – Part 3-11: Data-link layer service definition – Type 11 elements* (disponible en anglais seulement)

IEC 61158-5-11, *Industrial communication networks – Fieldbus specifications – Part 5-11: Application layer service definition – Type 11 elements* (disponible en anglais seulement)

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

ISO/CEI 7498-1, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base – Partie 1: Le modèle de base*

ISO/CEI 8822, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Définition du service de présentation*

ISO/IEC 8824, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)* (disponible en anglais seulement)

ISO/IEC 8825, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)* (disponible en anglais seulement)

ISO/CEI 9545, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Structure de la couche d'application*

3 Termes, définitions, symboles, abréviations et conventions

3.1 Introduction

Pour les besoins du présent document, ce qui suit s'appliquent.

3.2 Termes et définitions provenant d'autres normes ISO/CEI

3.2.1 Termes et définitions provenant de l'ISO/CEI 7498-1

- a) entité d'application
- b) processus d'application
- c) unité de données de protocole d'application
- d) élément de service d'application
- e) invocation d'entité d'application
- f) invocation de processus d'application
- g) transaction d'application
- h) système ouvert réel
- i) syntaxe de transfert

3.2.2 Termes et définitions provenant de l'ISO/CEI 8822

- a) syntaxe abstraite
- b) contexte de présentation

3.2.3 Termes et définitions provenant de l'ISO/CEI 9545

- a) association d'applications
- b) contexte d'application
- c) nom de contexte d'application
- d) invocation d'entité d'application
- e) type d'entité d'application
- f) invocation de processus d'application
- g) type de processus d'application
- h) élément de service d'application
- i) élément de service de contrôle d'application

3.2.4 Termes et définitions provenant de l'ISO/CEI 8824

- a) identificateur d'objet
- b) type
- c) valeur
- d) type simple
- e) type structuré
- f) type composant
- g) balise
- h) type booléen
- i) true (vrai)
- j) false (faux)
- k) type entier
- l) type chaîne binaire
- m) type chaîne d'octets
- n) type null (nul)
- o) type de séquence
- p) séquence de type
- q) type de choix
- r) type balisé
- s) tout type
- t) module
- u) production

3.2.5 Termes et définitions provenant de l'ISO/CEI 8825

- a) codage (d'une valeur de donnée)
- b) valeur de donnée
- c) octets d'identification (la présente norme utilise le singulier)
- d) octet(s) de longueur (la présente norme utilise à la fois le singulier et le pluriel)
- e) octets de contenu

3.3 Termes et définitions provenant de la CEI/TR 61158-1

- a) relation entre applications
- b) chemin de transport
- c) client
- d) AR dédiée
- e) AR dynamique
- f) classe d'erreur
- g) code d'erreur
- h) nom
- i) identificateur numérique
- j) homologue
- k) point de fin de AR prédéfinie
- l) point de fin de AR préétablie
- m) éditeur
- n) abonné
- o) serveur

3.4 Autres termes et définitions

NOTE Les définitions suivantes peuvent s'appliquer à tous les types, sauf si le même élément est utilisé avec des définitions différentes d'un type spécifique. Dans ce cas, le dernier type prévaut.

Les termes et définitions suivants sont utilisés dans la présente série de normes.

3.4.1 appelé

utilisateur de service ou fournisseur de service qui reçoit une primitive d'indication ou un APDU de demande

3.4.2**appel**

utilisateur de service ou fournisseur de service qui initie une primitive de demande ou un APDU de demande

3.4.3**mémoire commune**

mémoire commune virtuelle du bus de terrain de Type 11, qui est partagée avec les nœuds participant à ce bus de terrain, et qui est utilisée principalement pour les communications en temps réel par le service de données TCC

3.4.4**interopérabilité**

capacité des entités Couche utilisateur à effectuer des opérations coordonnées et coopératives en utilisant les services de la FAL

3.4.5**informations de gestion**

informations accessibles via le réseau qui facilitent la gestion de l'environnement des bus de terrain

3.4.6**réception**

utilisateur de service qui reçoit une primitive confirmée ou une primitive non confirmée, ou fournisseur de service qui reçoit un APDU confirmé ou un APDU non confirmé

3.4.7**ressource**

capacité de traitement ou d'information d'un sous-système

3.4.8**envoi**

utilisateur de service qui envoie une primitive confirmée ou une primitive non confirmée, ou fournisseur de service qui reçoit un APDU confirmé ou un APDU non confirmé

3.5 Abréviations et symboles

AE	Entité d'application (en anglais Application Entity)
AE-I	Invocation de l'entité d'application (en anglais Application Entity Invocation)
AL	Couche d'application (en anglais Application Layer)
AP	Processus d'application (en anglais Application Process)
Ap_	Préfixe pour les data types définis pour ASE AP
Ar_	Préfixe pour les data types définis pour ASE AR
APDU	Ensemble (unité) de données de protocole d'application (en anglais Application Protocol Data Unit)
AR	Relation entre applications (en anglais Application Relationship)
AREP	Point de fin de relation entre applications (en anglais Application Relationship End Point)
ASE	Élément de service d'application (en anglais Application Service Element)
ASN.1	Notation de syntaxe abstraite numéro un (en anglais Abstract Syntax Notation One)
BCD	Décimal codé binaire (en anglais Binary Coded Decimal)
BER	Règle de codage de base (en anglais Basic Encoding Rule)
BNU-PEC	Connexion préétablie unidirectionnelle planifiée par le réseau en mémoire tampon (en anglais Buffered Network-Scheduled Uni-directional Pre-Established Connection)
CM	Mémoire commune (en anglais Common Memory)
cnf	Primitive de confirmation (en anglais confirmation primitive)
DI_	Préfixe pour les data types définis pour les types de couche Liaison de données

DL	Liaison de données (en anglais Data-link)
DLC	Connexion de couche de liaison de données (en anglais Data-link Connection)
DLCEP	Point de fin de connexion de couche de liaison de données (en anglais Data-link Connection End Point)
DLPDU	Ensemble (unité) de données de protocole de liaison de données (en anglais Data-link Protocole Data Unit)
DLSAP	Point d'accès de service de couche de liaison de données (en anglais Data-link Service Access Point)
DLSDU	Ensemble (unité) de données de service de liaison de données (en anglais Data-link Service Data Unit)
Dt_	Préfixe pour les data types définis pour l'ASE Data type
Err	Erreur (utilisée pour indiquer un type d'APDU)
Er_	Préfixe pour les types d'erreur définis
Ev_	Préfixe pour les data types définis pour l'ASE Événement
FAL	Couche Application de bus de terrain (en anglais Fieldbus Application Layer)
Fi_	Préfixe pour les data types définis pour l'ASE Lancement de fonctions
FIFO	Premier entré, premier sorti (en anglais First In First Out)
Gn_	Préfixe pour les Data types définis pour une utilisation générale
ID	Identificateur
CEI	Commission Electrotechnique Internationale
in	primitive d'entrée (en anglais input primitive)
ind	primitive d'indication (en anglais indication primitive)
ISO	Organisation internationale de normalisation
LAS	Échéancier actif de liaison (en anglais Link Active Scheduler)
Lr_	Préfixe pour les data types définis pour l'ASE Région de charge
lsb	bit de poids faible (en anglais least significant bit)
Mn_	Préfixe pour les data types définis pour l'ASE Gestion
msb	bit de poids fort (en anglais most significant bit)
out	primitive de sortie (en anglais output primitive)
OSI	Interconnexion de systèmes ouverts (en anglais Open Systems Interconnection)
PDU	Ensemble (Unité) de données de protocole (en anglais Protocol Data Unit)
PICS	Déclaration de conformité de mise en œuvre de protocole (en anglais Protocol Implementation Conformance Statement)
QoS	Qualité de service (en anglais Quality Of Service)
Req	Demande (utilisée pour indiquer un type d'APDU) (en anglais Request)
req	Primitive de demande (en anglais request primitive)
Rsp	Réponse (utilisée pour indiquer un type d'APDU) (en anglais Response)
rsp	primitive de réponse (en anglais response primitive)
SAP	Point d'accès de service (en anglais Service Access Point)
SDU	Ensemble (Unité) de données de service (en anglais Service Data Unit)
TCC	Cyclique en temps critique (en anglais Time-critical cyclic)
ToS	Type de service (en anglais Type Of Service)
Vr_	Préfixe pour les data types définis pour l'ASE Variable

3.6 Conventions

3.6.1 Conventions pour les définitions de classe

Les définitions de mapping de la couche de liaison de données sont décrites au moyen de modèles. Chaque modèle se compose d'une liste d'attributs associés à la classe. La forme générale du modèle est définie dans la CEI 61158-5.

3.6.2 Conventions de syntaxe abstraite

Lorsque le paramètre "optionalParametersMap" est utilisé, un numéro de bit correspondant à chaque production OPTIONAL ou DEFAULT est donné en commentaire.

3.6.3 Conventions utilisées dans les diagrammes d'états

Les diagrammes d'états sont décrits dans le Tableau 1.

Tableau 1 – Conventions utilisées dans les diagrammes d'états

#	État actuel	Événement / condition => action	État suivant
Nom du passage	L'état actuel auquel ce passage s'applique	Événements ou conditions déclenchant ce passage d'état. => Actions effectuées lorsque les événements ou conditions ci-dessus sont réunis. Elles figurent toujours au-dessous des événements ou conditions	État suivant dans lequel passe le diagramme d'états une fois les actions effectuées

Les conventions utilisées dans les diagrammes d'états sont les suivantes:

`:=` La valeur d'un élément, à gauche, est remplacée par la valeur d'un élément, à droite. Si un élément à droite est un paramètre, il provient de la primitive indiquée comme événement d'entrée.

`xxx` Indique un nom de paramètre.

Exemple:

Identifieur `:= reason`

signifie que la valeur d'un paramètre "reason" est attribuée à un paramètre appelé 'Identifieur.'

`"xxx"` Indique une valeur fixe.

Exemple:

Identifieur `:= "abc"`

signifie que la valeur "abc" est attribuée à un paramètre appelé 'Identifieur.'

`=` Condition logique indiquant qu'un élément à gauche est égal à un élément à droite.

`<` Condition logique indiquant qu'un élément à gauche est inférieur à l'élément à droite.

`>` Condition logique indiquant qu'un élément à gauche est supérieur à l'élément à droite.

`<>` Condition logique indiquant qu'un élément à gauche est différent d'un élément à droite.

`&&` indique le "ET" logique

`||` indique le "OU" logique

Cette construction permet d'exécuter une séquence d'actions en boucle au sein d'un passage. La boucle est exécutée pour toutes les valeurs comprises entre `start_value` et `end_value`.

Exemple:

```
for (Identifieur := start_value to end_value)
  actions
endfor
```

Cette construction permet d'exécuter d'autres actions dépendant d'une condition (qui peut être la valeur d'un identificateur ou le résultat d'une action antérieure) au sein d'un passage.

Exemple:

```
If (condition)
  actions
else
  actions
endif
```

Il est fortement recommandé aux lecteurs de se référer aux paragraphes traitant des définitions d'attributs de point AREP, des fonctions locales et des définitions d'unités FAL-PDU pour comprendre les machines de protocole. On suppose que les lecteurs possèdent une connaissance suffisante de ces définitions et savent les utiliser sans explications complémentaires.

4 Description de la syntaxe FAL

4.1 Concept

La présente norme spécifie le protocole de la couche d'application de Type 11, essentiel pour le réseau de commande en temps critique (TCnet) basé sur l'ISO/CEI 8802-3, qui est l'un des réseaux de communication pour le réseau Ethernet en temps réel (RTE) défini dans la CEI 61784-2, et désigné ci-après par le terme RTE-TCnet.

La présente norme satisfait à l'objectif du secteur de l'automatisation industrielle qui est de fournir un système de transfert de données déterministes de durée prévisible et de données en temps critique qui soient fiables, ainsi que des moyens qui permettent la coexistence avec le transfert de données en temps non critique sur le support de communications défini dans la série ISO/CEI 8802-3, afin de soutenir la coopération et la synchronisation entre les processus d'automatisation des appareils de terrain dans un système d'application en temps réel. Le terme «en temps critique» signale l'existence d'une fenêtre temporelle dans laquelle est exigée la réalisation d'une ou de plusieurs actions spécifiées, avec un niveau de certitude défini.

La présente norme spécifie la partie de l'ensemble de protocoles du profil de communication RTE-TCnet et/ou de un ou plusieurs profils de communication associés à une famille commune du RTE-TCnet. Le profil de communication RTE-TCnet, présenté à la Figure 1 comme l'un des ensembles de profils, est basé sur le modèle de référence de base OSI à 7 couches. Pour les applications régulières basées sur l'ISO/CEI 8802-3, les couches supérieures mises en correspondance sur la couche de liaison de données sont de forme ordinaire; par ailleurs, pour les applications en temps critique dont la mémoire commune fonctionne en parallèle, la couche d'application spécifique pour le RTE-TCnet est spécifiée. La couche de liaison de données pour le RTE-TCnet a une extension, mais est toutefois conforme au protocole MAC de l'ISO/CEI 8802-3 afin de fournir à la fois des services pour les communications en temps critique, et des applications à mémoire commune respectivement.

	Regular ISO/IEC 8802-3-based applications	Time-critical applications with common memory
Application layer	TELNET, FTP, HTTP OPC XML-DA etc	Common memory
Transport layer	RFC 768(UDP) RFC 793 (TCP)	null
Network layer	RFC 791 (IP)	
Data Link layer	ISO/IEC 8802-3 Specific scheduling extension	
Physical layer	ISO/IEC 8802-3 (Redundant)	

Légende

Anglais	Français
Regular ISO/IEC 8802-3 based applications	Applications régulières basées sur l'ISO/CEI 8802-3
Time-critical applications with common memory	Application en temps critique avec mémoire commune
Application layer	Couche Application
Common memory	Mémoire commune
Transport layer	Couche Transport
Network layer	Couche Réseau

Anglais	Français
null	nul
Data link layer	Couche Liaison de données
Physical layer	Couche physique
ISO/IEC 8802-3, Specific scheduling extension	ISO/CEI 8802-3, Extension de planification spécifique
ISO/IEC 8802-3, (Redundant)	ISO/CEI 8802-3, (Redondant)

Figure 1 – Profil de communication RTE-TCnet

La présente norme spécifie le protocole de liaison de données comme constituant les parties essentielles du profil RTE-TCnet, qui représentent l'extension de la couche de liaison de données basée sur l'ISO/CEI 8802-3 et la couche d'application qui exploite les services de la couche de liaison de données immédiatement inférieure, en termes de modèle de référence de bus de terrain "à trois couches", qui est basé en partie sur le modèle de référence de base OSI. L'autre partie du profil RTE-TCnet ne relève pas du domaine d'application du présent document.

4.1.1 Champ d'applications

Dans les systèmes de commande industriels, plusieurs types d'appareils de terrain tels que les entraînements, les capteurs et les actionneurs, les automates programmables, les systèmes de commande répartis et les appareils à interface homme-machine sont à connecter à des réseaux de commande. Les données de commande de processus et les données d'état sont transmises entre ces appareils de terrain du système et les communications entre ces appareils exigent de la simplicité dans la programmation des applications et sont à exécuter avec un temps de réponse adéquat. Dans la plupart des systèmes d'automatisation industriels, tels que les systèmes utilisés dans le secteur de l'alimentation, du traitement de l'eau, du traitement des eaux usées, du papier et de l'acier, y compris un laminier, le réseau de commande est tenu de fournir une capacité de réponse en temps critique pour l'application de ces systèmes, comme le requiert l'ISO/TR 13283 pour les architectures de communications en temps critique.

La production d'usine ou industrielle peut être remise en cause du fait d'erreurs, susceptibles de se produire dans le système de commande si le réseau ne fournit pas de réponse en temps critique. Un réseau de commande en temps critique requiert par conséquent les caractéristiques suivantes.

- Un temps de réponse déterministe entre les nœuds d'appareils de commande.
- Une capacité de partage des données de processus de manière continue dans le système de commande.

Le RTE-TCnet est applicable à un environnement d'automatisation industrielle, qui requiert principalement des communications en temps critique. Le terme «en temps critique» signale l'existence d'une fenêtre temporelle dans laquelle est exigée la réalisation d'une ou de plusieurs actions spécifiées, avec un niveau de certitude défini. La non-réalisation des actions spécifiées dans le cadre de la fenêtre temporelle peut mener à l'échec des applications sollicitant les actions, avec un risque annexe pour les équipements, l'installation et éventuellement la vie humaine.

4.2 Généralités

La description de la syntaxe FAL du RTE-TCnet définit un service d'envoi non confirmé et comprend les trois parties suivantes, FalArHeader, InvokeID et Unconfirmed Service Request (Demande de service non confirmé).

4.3 Syntaxe abstraite FAL-AR PDU

4.3.1 Définition de base

FalArPDU ::= UnconfirmedSend-CommandPDU

4.3.2 Service d'envoi non confirmé

```
UnconfirmedSend-CommandPDU ::= SEQUENCE {
    FalArHeader,
    InvokeID OPTIONAL,
    UnconfirmedServiceRequest
}
```

4.4 Syntaxe abstraite de corps PDU

4.4.1 FalArHeader

```
FalArHeader ::= Unsigned8 {
    -- bit 8      FAL Protocol Specifier (Spécificateur de protocole FAL)      (Toujours égal à 1)
    -- bit 7-4    Protocol Identifier (Identificateur de protocole)      (révision de la syntaxe abstraite des
    --            identificateurs et règles de codage)
    -- bit 3      Protocol Specific bit (bit spécifique au protocole)      (Réservé pour chaque protocole à utiliser)
    -- bit 2-1    PDU Identifier (Identificateur PDU)      (Identifie un type de PDU au sein d'un identificateur de
    --            protocole)
}
```

4.4.2 InvokeID

InvokeID ::= Unsigned8

4.4.3 PDU non confirmés

```
UnconfirmedServiceRequest ::= SEQUENCE {
    CMArep, -- Block number (Nombre de blocs)
    CMData -- content of CM segment (contenu de segment CM)
}
```

4.4.4 CMArep

CMArep ::= Unsigned16 -- DLCEP address (adresse DLCEP)

4.4.5 CMData

```
CMData ::= SEQUENCE {
    wlen Unsigned16, -- CM word length (longueur de mot CM)
    data SEQUENCE { ANY { -- CM content (contenu CM)
        BitString8, BitString16, BitString32,
        Integer16, Integer32,
        Unsigned16, Unsigned32,
        Floating32,
        OctetString2, OctetString4,
        VisibleString2, VisibleString4,
        BinaryTime0, BinaryTime1, BinaryTime2, BinaryTime3, BinaryTime4, BinaryTime5,
        BinaryTime6, BinaryTime7
    }
}
```

4.5 Data type

BitString8, BitString16, BitString32, Integer16, Integer32, Unsigned16, Unsigned32, Floating32, OctetString2, OctetString4, VisibleString2, VisibleString4, BinaryTime0, BinaryTime1, BinaryTime2, BinaryTime3, BinaryTime4, BinaryTime5, BinaryTime6.

5 Syntaxe de transfert

5.1 Présentation générale et en-tête de FAL

Tous les PDU FAL doivent avoir l'en-tête PDU commun appelé FalArHeader. Le FalArHeader identifie la syntaxe abstraite, la syntaxe de transfert et la syntaxe de chaque PDU. Le Tableau 2 définit de quelle manière cet en-tête doit être utilisé.

Tableau 2 – En-tête de FAL

Position binaire de FalArHeader			Syntaxe abstraite	Règle de codage	Type de PDU	Révision
8	7 6 5 4 3	2 1				
1	1 1 1 1 1	10	ASN.1	RTE-TCnet	UnconfirmedSendPDU	Revision1
NOTE Toutes les autres définitions sont réservées.						

5.2 Règle de codage

5.2.1 Présentation générale

La règle de codage de RTE-TCnet est une règle de codage préférentielle compatible avec les normes existantes. Les PDU FAL à codage TER (Traditional Encoding Rule ou Règle de codage conventionnelle) doivent avoir un format uniforme. Les PDU FAL doivent comporter deux parties principales, la partie "En-tête APDU" et la partie "Corps APDU", comme l'illustre la Figure 2.

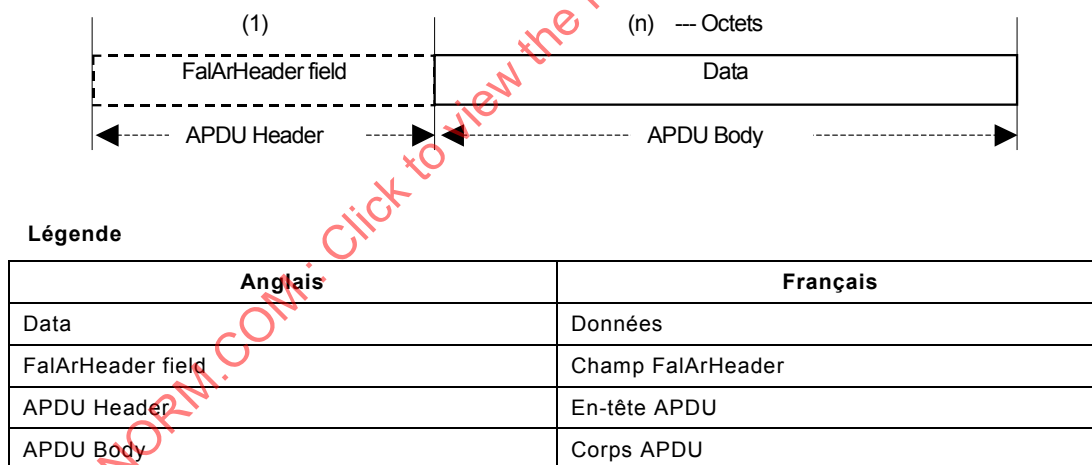


Figure 2 – Présentation générale de l'APDU

5.2.2 Codage de l'en-tête APDU

La partie En-tête APDU est toujours présente dans tous les APDU conformes à la présente spécification. Elle comporte un seul champ: le champ FalArHeader. Se reporter à 4.4.1 pour la règle de codage du champ FalArHeader.

5.2.3 Codage du corps APDU

La règle de codage de la FAL est basée sur les termes et définitions de l'ISO/CEI 8825, et comporte les trois composantes de codage indiquées ci-dessous. Pour les communications en temps critique et avec des données de longueur fixe, il n'existe pas d'octets d'identification ou d'octets de longueur de type TER (Règle de codage conventionnelle).

- Octet d'identification

- Octets de longueur
- Octets de contenu

NOTE L'octet d'identification et les octets de longueur et de contenu n'existent pas dans le RTE-TCnet.

5.2.4 Codage de variable simple

5.2.4.1 Codage d'une valeur booléenne

- a) Le codage d'une valeur booléenne doit être primitif;
- b) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;
- c) les octets de contenu doivent comporter un seul octet;
- d) si la valeur booléenne est FALSE, les octets de contenu doivent être 0 (zéro);
- e) si la valeur booléenne est TRUE, les octets de contenu doivent être 0xFF.

5.2.4.2 Codage d'une valeur Integer de longueur fixe

- a) Le codage d'une valeur Integer de longueur fixe des types Integer8, Integer16 et Integer32 doit être primitif;
- b) Les octets de contenu doivent comporter exactement un, deux ou quatre octets, respectivement;
- c) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;
- d) les octets de contenu doivent être un nombre binaire complément à deux égal à la valeur Integer, et comprendre les bits 8 à 1 du premier octet, suivis des bits 8 à 1 du deuxième octet, suivis des bits 8 à 1 de chaque octet restant, jusques et y compris le dernier octet des octets de contenu;
 - 1) la valeur d'un nombre binaire complément à deux est déduite par la numérotation des bits des octets de contenu, en commençant par le bit 1 du premier octet et en terminant la numérotation par le bit 8 du dernier octet;
 - 2) une valeur numérique de 2^N-1 est attribuée à chaque bit, où N désigne sa position dans la séquence de numérotation ci-dessus;
 - 3) la valeur du nombre binaire complément à deux est obtenue en ajoutant les valeurs numériques attribuées à chaque bit, pour les bits réglés sur un, en excluant le bit 8 du dernier octet, puis en réduisant la valeur par la valeur numérique attribuée au bit 8 du dernier octet si ce bit est réglé sur un.

5.2.5 Codage d'une valeur Unsigned de longueur fixe

- a) Le codage d'une valeur Integer de longueur fixe des types Unsigned8, Unsigned16 et Unsigned32 doit être primitif, et les octets de contenu doivent comporter exactement un, deux ou quatre octets, respectivement;
- b) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;
- c) les octets de contenu doivent être un nombre binaire égal à la valeur Unsigned, et comprendre les bits 8 à 1 du premier octet, suivis des bits 8 à 1 du deuxième octet, suivis des bits 8 à 1 de chaque octet restant, jusques et y compris le dernier octet des octets de contenu;
 - 1) la valeur d'un nombre binaire est déduite par la numérotation des bits des octets de contenu, en commençant par le bit 1 du premier octet utilisé comme bit nul, et en terminant la numérotation par le bit 8 du dernier octet;
 - 2) une valeur numérique de 2^N-1 est attribuée à chaque bit, où N désigne sa position dans la séquence de numérotation ci-dessus;
 - 3) la valeur du nombre binaire est obtenue en ajoutant les valeurs numériques attribuées à chaque bit pour les bits réglés sur un.

5.2.6 Codage d'une valeur Floating Point

- a) Le codage d'une valeur Floating32 doit être primitif, et les octets de contenu doivent comporter exactement quatre octets;

- b) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;

NOTE Le signe est codé dans le bit 8 du premier octet. Il est suivi de l'exposant qui commence par le bit 7 du premier octet, puis de la mantisse qui commence par le bit 7 du deuxième octet pour Floating32 et par le bit 4 du deuxième octet pour Floating64.

- c) les octets de contenu doivent comporter les valeurs en virgule flottante définies conformément à la CEI 60559.

5.2.7 Codage d'une valeur BitString de longueur fixe

- a) Le codage d'une valeur BitString de longueur fixe des types BitString8, BitString16 et BitString32 doit être primitif, et les octets de contenu doivent comporter exactement un, deux ou quatre octets, respectivement;
- b) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;
- c) la valeur BitString, qui commence par le premier bit et qui passe au bit de droite, doit être placée dans les bits 8 à 1 du premier octet, suivis des bits 8 à 1 du deuxième octet, suivi des bits 8 à 1 de chaque octet restant, jusques et y compris le dernier octet des octets de contenu.

5.2.8 Codage d'une valeur OctetString de longueur fixe

- a) Le codage d'une valeur OctetString de longueur fixe des types OctetString2 et OctetString4 doit être primitif, et les octets de contenu doivent comporter exactement deux ou quatre octets, respectivement;
- b) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;
- c) les octets de contenu doivent être équivalents en valeur aux octets de la valeur de donnée, dans leur ordre d'apparition dans cette dernière, le bit de poids fort d'un octet de la valeur de donnée étant aligné sur le bit de poids fort d'un octet des octets de contenu.

5.2.9 Codage d'une valeur VisibleString de longueur fixe

- a) Le codage d'une valeur VisibleString de longueur fixe des types VisibleString2 et VisibleString4 doit être primitif, et les octets de contenu doivent comporter exactement deux ou quatre octets, respectivement;
- b) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;
- c) les octets de contenu doivent être équivalents en valeur aux octets de la valeur de donnée, dans leur ordre d'apparition dans cette dernière, le bit de poids fort d'un octet de la valeur de donnée étant aligné sur le bit de poids fort d'un octet des octets de contenu.

5.2.10 Codage de la valeur BinaryTime

- a) Le codage d'une valeur BinaryTime0, BinaryTime1, BinaryTime2, BinaryTime3, BinaryTime4, BinaryTime5, BinaryTime6 et BinaryTime7 doit être primitif;
- b) l'octet d'identification et le(s) octet(s) de longueur ne doivent pas être présents;
- c) les octets de contenu doivent être un nombre binaire égal à la valeur BinaryTime, et comportant les bits 8 à 1 du premier octet, suivis des bits 8 à 1 du deuxième octet, suivis des bits 8 à 1 de chaque octet restant, jusques et y compris le dernier octet des octets de contenu;
 - 1) les octets de contenu d'une valeur BinaryTime0, BinaryTime1, BinaryTime2 et BinaryTime3 doivent comporter deux octets;
 - 2) les octets de contenu d'une valeur BinaryTime4, BinaryTime5, BinaryTime6 et BinaryTime7 doivent comporter quatre octets.

NOTE La valeur de granularité de chaque type BinaryTime est définie dans la sous-série de CEI 61158-5.

5.3 Codage des types structurés

5.3.1 Généralités

Lorsque le type structuré est également codé, le RTE-TCnet ne comporte pas l'identificateur ou la longueur de la structure.

5.3.2 Codage d'une valeur SEQUENCE

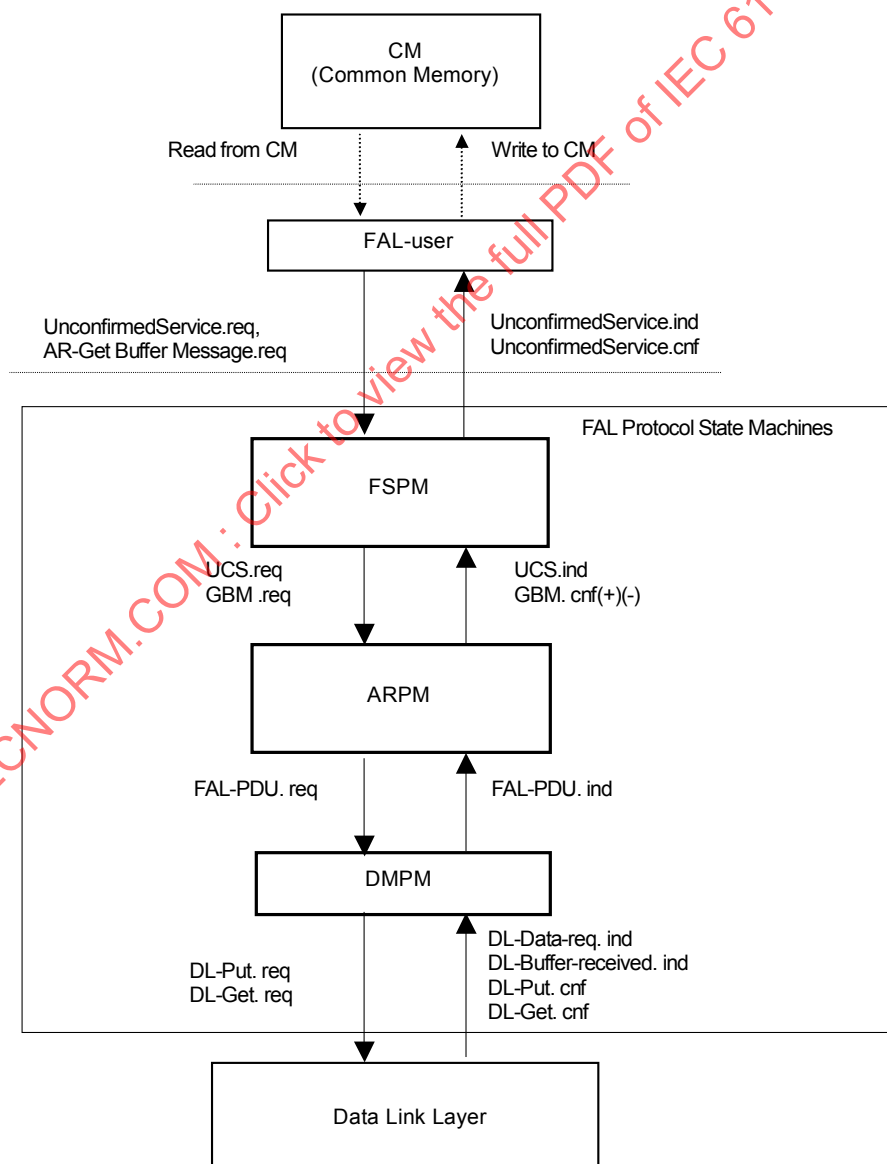
Le type SEQUENCE est comparable à un enregistrement. Il représente un ensemble de données utilisateur du même data type ou de data types différents.

Une valeur de type SEQUENCE peut contenir une variable simple, ou une autre variable structurée utilisée comme composantes. Si un type SEQUENCE contient une autre valeur de type structuré, il doit être considéré comme une composante simple, même s'il comporte plusieurs composantes.

6 Structures des diagrammes d'états de protocole FAL

6.1 Présentation générale

Comme l'illustre la Figure 3, la machine de protocole FAL comprend trois machines, à savoir la machine de protocole de service FAL (FSPM), la machine de protocole de relations entre applications (ARPM) et la machine de protocole de mapping de la couche de liaison de données (DMPM).



Légende

Anglais	Français
Common memory	Mémoire commune
Read from CM	Lecture à partir de la mémoire commune
Write to CM	Écriture dans la mémoire commune
FAL-user	Utilisateur FAL
FAL Protocol State Machines	Diagrammes d'états de protocole FAL
Data Link Layer	Couche de liaison de données

Figure 3 – Relations entre FSPM, ARPM, DMPM et CM physique externe**7 Machine de protocole de service FAL (FSPM)****7.1 Généralités**

La machine de protocole de service FAL (FSPM) est commune à tous les types d'AREP. Seules les primitives applicables sont différentes entre des types d'AREP également différents. Un état de cette machine est l'état "ACTIVE".

NOTE Bien que désormais présent, le type d'AREP est unique.

7.2 Définitions des primitives**7.2.1 Primitives échangées entre utilisateur FAL et FSPM**

Les primitives échangées entre l'utilisateur FAL et la FSPM sont décrites dans le Tableau 3 et le Tableau 4.

Tableau 3 – Primitives échangées entre l'utilisateur FAL et la FSPM

Noms des primitives	Source	Paramètres associés	Fonction
UCS.req	Utilisateur FAL	Arep-id, Data	Il s'agit d'une primitive interne FAL qui permet de transmettre une primitive de demande Unconfirmed Send (UCS) entre l'utilisateur FAL et la FSPM
GBM.req	Utilisateur FAL	Arep-id	Il s'agit d'une primitive interne FAL qui permet de transmettre une primitive de demande Get-Buffered-Message (GBM) entre l'utilisateur FAL et la FSPM

Tableau 4 – Primitives échangées entre la FSPM et l'utilisateur FAL

Noms des primitives	Source	Paramètres associés	Fonction
UCS.ind	FSPM	Arep-id, Data	Il s'agit d'une primitive interne FAL qui permet de transmettre une primitive d'indication Unconfirmed Send (UCS) entre la FSPM et l'utilisateur FAL
GBM.cnf(+)	FSPM	Arep-id, Data	Il s'agit d'une primitive interne FAL qui permet de transmettre une confirmation positive Get-Buffered-Message (GBM) entre la FSPM et l'utilisateur FAL
GBM.cnf(-)	FSPM	Arep-id	Il s'agit d'une primitive interne FAL qui permet de transmettre une confirmation négative Get-Buffered-Message (GBM) entre la FSPM et l'utilisateur FAL
FSTS.ind	FSPM	Arep-id, Reported-status	Il s'agit d'une primitive interne FAL qui permet de transmettre une primitive d'indication FAL-Status (FSTS) entre la FSPM et l'utilisateur FAL

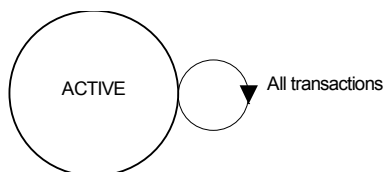
7.2.2 Paramètres utilisateur FAL /FSPM

Tous les paramètres utilisés dans les primitives échangées entre l'utilisateur FAL et la FSPM sont identiques à ceux définis dans le paragraphe "Service d'exploitation" ("Operational Service").

7.3 Tableaux d'états de la machine FSPM

7.3.1 Généralités

Les diagrammes d'états FSPM sont décrits dans la Figure 4, le Tableau 5 et le Tableau 6.



Légende

Anglais	Français
All transactions	Toutes les transactions

Figure 4 – Diagramme de passage d'états de la machine FSPM

Tableau 5 – Tableau d'états de la machine FSPM – Transactions de l'émetteur

#	État actuel	Événement ou condition => action	État suivant
S1	ACTIVE	UCS.req && SelectArep (Arep-id) = "True" => UCS.req { user-data := Data }	ACTIVE
S2	ACTIVE	GBM.req && SelectArep (Arep-id) = "True" => GBM.req { }	ACTIVE

NOTE 1 Un paramètre de primitive du diagramme d'états de l'émetteur FSPM est transmis à une machine ARPM appropriée, choisie par la machine FSPM à l'aide de la fonction SelectArep.
NOTE 2 Si la fonction SelectArep renvoie la valeur "False", le signalment de cette instance relève d'une initiative locale, et la machine FSPM ne génère aucune primitive pour la machine ARPM.

Tableau 6 – Tableau d'états de la machine FSPM – Transactions du récepteur

#	État actuel	Événement ou condition => action	État suivant
R1	ACTIVE	UCS.ind => UCS.ind { Arep-id := arep-id, Data := user-data }	ACTIVE
R2	ACTIVE	GBM.cnf(+) => GBM.cnf (+) { Arep-id := arep, Data := user-data }	ACTIVE
R3	ACTIVE	GBM.cnf(-) => GBM.cnf (-) { Arep-id := arep-id }	ACTIVE
R4	ACTIVE	FAL-STs.ind => FSTS.ind { Arep-id := arep-id,	ACTIVE