

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Industrial communication networks – Fieldbus specifications –
Part 6-20: Application layer protocol specification – Type 20 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 6-20: Spécification du protocole de la couche application – Eléments
de type 20**



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2014 IEC, Geneva, Switzerland

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

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

IEC 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.

IEC Catalogue - webstore.iec.ch/catalogue

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub

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 It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and 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 14 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

More than 55 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

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

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

A propos de l'IEC

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

A propos des publications IEC

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

Catalogue IEC - webstore.iec.ch/catalogue

Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

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

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

IEC Just Published - webstore.iec.ch/justpublished

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

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne 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 14 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary

Plus de 55 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

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 6-20: Application layer protocol specification – Type 20 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 6-20: Spécification du protocole de la couche application – Eléments
de type 20**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX

XC

ICS 25.040.40; 35.100.70; 35.110

ISBN 978-2-8322-1766-5

**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	7
INTRODUCTION	9
1 Scope	10
2 Normative references	10
3 Terms, definitions, symbols, abbreviations and conventions	11
3.1 Terms and definitions from other ISO/IEC standards	11
3.2 IEC 61158-1 terms	12
3.3 Type 20 fieldbus application-layer specific definitions	14
3.4 Abbreviations and symbols	17
3.5 Conventions	18
3.6 Conventions used in state machines	18
4 Abstract syntax	20
5 Transfer syntax	20
5.1 Common APDU fields	20
5.2 Common APDU structure	22
5.3 Device application service-specific APDU structures	24
5.4 Data coding rules	51
6 Common procedures	56
6.1 Delayed response	56
6.2 Publish mode procedure	56
7 FAL protocol state machines	56
7.1 General	56
7.2 AREP mapping to data link layer	57
7.3 Client ARPM	58
7.4 Server ARPM	60
7.5 Functions used by FAL state machines	61
Annex A (normative) Application process status	63
A.1 General	63
A.2 Device malfunction	63
A.3 Configuration changed	63
A.4 Cold start	64
A.5 More status available	64
A.6 Loop current fixed	64
A.7 Loop current saturated	64
A.8 Non-primary variable out of limits	65
A.9 Primary variable out of limits	65
Annex B (normative) Device and dynamic variable	66
B.1 Device variable	66
B.2 Dynamic variable	66
B.3 Primary variable	67
B.4 Device variable classification	68
B.5 Device families	68
B.6 Device variable status	68
Annex C (normative) Common tables	70
C.1 Overview	

C.1.1	General	70
C.1.2	Enumeration	70
C.1.3	Bit Field	70
C.2	Table definitions	70
C.2.1	Publish mode control codes	70
C.2.2	Write device variable codes	70
C.2.3	Device variable family codes	70
C.2.4	Device variable classification codes	70
C.2.5	Analog channel saturated codes	70
C.2.6	Analog channel fixed codes	70
C.2.7	Standardized status 0 codes	71
C.2.8	Standardized status 1 codes	71
C.2.9	Standardized status 2 codes	71
C.2.10	Standardized status 3 codes	71
C.2.11	Publish trigger mode codes	71
C.2.12	Transfer function codes	71
C.2.13	Alarm Selection Codes	71
C.2.14	Write Protect Codes	72
C.2.15	Physical layer signalling codes	72
C.2.16	Flag Assignment codes	72
C.2.17	Loop current mode codes	73
C.2.18	Trim point codes	73
C.2.19	Analog channel flag codes	73
C.2.20	Device variable codes	73
C.2.21	Device profile codes	74
Annex D (normative)	Command requirements	75
D.1	General	75
D.2	Stateless request and response	75
D.3	Read command	75
D.4	Write command	75
D.5	Action command	75
D.6	Indexed command	76
D.7	Multi-transaction command	76
Bibliography		77
Figure 1	– Request APDU	22
Figure 2	– Normal response APDU	22
Figure 3	– Command error response from slave to master	23
Figure 4	– Aggregated command APDU	24
Figure 5	– Coding without identification	51
Figure 6	– Coding of Integer type data	51
Figure 7	– Coding of Integer16 type data	52
Figure 8	– Coding of Unsigned type data	52
Figure 9	– Coding of Unsigned16 type data	52
Figure 10	– Coding of single precision Floating Point type data	52
Figure 11	– Coding of double precision Floating Point type data	53
Figure 12	– Coding of Date type data	53

Figure 13 – Client state machine	58
Figure 14 – Server state machine	60
Figure A.1 – Loop current saturation and alarm levels	65
Figure B.1 – Device and Dynamic variables	66
Figure B.2 – Primary variable domains	67
Figure B.3 – Device variable status	69
Table 1 – Conventions used for state machines	19
Table 2 – Response code values	20
Table 3 – Application process status values	21
Table 4 – Extended status values	21
Table 5 – Identify request APDU	24
Table 6 – Identify response value field	25
Table 7 – Identify command specific response codes	25
Table 8 – Read primary variable response value field	26
Table 9 – Read primary variable command specific response codes	26
Table 10 – Read loop current and percent of range value field	26
Table 11 – Read loop current and percent of range command specific response codes	27
Table 12 – Read dynamic variables and loop current value field	27
Table 13 – Read dynamic variables and loop current command specific response codes	27
Table 14 – Write loop configuration value field	28
Table 15 – Write loop configuration command specific response codes	28
Table 16 – Read loop configuration value field	29
Table 17 – Read loop configuration command specific response codes	29
Table 18 – Read dynamic variable families classifications value field	29
Table 19 – Read dynamic variable families classifications command specific response codes	29
Table 20 – Read device variables with status request value field	30
Table 21 – Read device variables with status value field	30
Table 22 – Read device variables with status command specific response codes	31
Table 23 – Read message response value field	32
Table 24 – Read message command specific response codes	32
Table 25 – Read tag, descriptor, date response value field	33
Table 26 – Read tag, descriptor, date command specific response codes	33
Table 27 – Read primary variable transducer information response value field	33
Table 28 – Read primary variable transducer information command specific response codes	34
Table 29 – Read device information response value field	34
Table 30 – Read device information command specific response codes	35
Table 31 – Read final assembly number response value field	35
Table 32 – Read final assembly number command specific response codes	35
Table 33 – Write message value field	35
Table 34 – Write message command specific response codes	35

Table 35 – Write tag, descriptor, date value field	36
Table 36 – Write tag, descriptor, date command specific response codes	36
Table 37 – Write final assembly number value field	37
Table 38 – Write final assembly number command specific response codes	37
Table 39 – Read long tag response value field	37
Table 40 – Read long tag command-specific response codes	37
Table 41 – Write long tag value field	38
Table 42 – Write long tag command specific Response codes	38
Table 43 – Write primary variable range value field	39
Table 44 – Write primary variable range command specific response codes	39
Table 45 – Enter-exit fixed current mode request value field	40
Table 46 – Enter-exit fixed current mode response value field	40
Table 47 – Enter-exit fixed current mode command specific response codes	40
Table 48 – Write primary variable unit value field	41
Table 49 – Write primary variable unit command specific response codes	41
Table 50 – Trim loop current zero request value field	41
Table 51 – Trim loop current zero command specific response codes	41
Table 52 – Trim loop current gain request value field	42
Table 53 – Trim loop current gain command specific response codes	42
Table 54 – Read dynamic variable assignment response value field	43
Table 55 – Read dynamic variable assignment command specific response codes	43
Table 56 – Write dynamic variable assignment value field	44
Table 57 – Write dynamic variable assignment command specific response codes	44
Table 58 – Write number of response preambles value field	45
Table 59 – Write number of response preambles command specific response codes	45
Table 60 – Read device variable trim points request value field	45
Table 61 – Read device variable trim points response value field	46
Table 62 – Read device variable trim points command specific response codes	46
Table 63 – Read device variable trim guidelines request value field	46
Table 64 – Read device variable trim guidelines response value field	46
Table 65 – Read device variable trim points command specific response codes	47
Table 66 – Write device variable trim point value field	47
Table 67 – Write device variable trim point command specific response codes	48
Table 68 – Reset device variable trim value field	48
Table 69 – Reset device variable trim command specific response codes	49
Table 70 – Aggregated command specific response codes	50
Table 71 – Coding for Date type	53
Table 72 – Coding for one octet Enumerated Type	54
Table 73 – One octet bit field	54
Table 74 – Packed ASCII character set	55
Table 75 – Acceptable subset of ISO Latin-1 characters	56
Table 76 – Client machine state transitions	59
Table 77 – Server machine state transitions	61

Table 78 – Function FormReqApdu	61
Table 79 – Function Command	61
Table 80 – Function CommErr.....	61
Table 81 – Function RespCode	62
Table 82 – Function Commcode.....	62
Table 83 – Function ApStatus	62
Table 84 – Function Value	62
Table A.1 – Commands that cause configuration change	63
Table C.1 – Transfer function codes	71
Table C.2 – Alarm Selection codes	72
Table C.3 – Write Protect codes	72
Table C.4 – Physical layer signalling codes	72
Table C.5 – Flag Assignment codes	73
Table C.6 – Loop current mode codes.....	73
Table C.7 – Trim point codes	73
Table C.8 – Analog channel flag codes	73
Table C.9 – Device variable codes	74

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**INDUSTRIAL COMMUNICATION NETWORKS –
FIELDBUS SPECIFICATIONS –****Part 6-20: Application layer protocol specification –
Type 20 elements**

FOREWORD

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

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

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

International Standard IEC 61158-6-20 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation.

This third edition cancels and replaces the second edition published in 2010. This edition constitutes a technical revision.

The main changes with respect to the previous edition are listed below:

- a) added protocol for new services that are added to IEC 61158-5-20;
- b) added normative annexes;
- c) updated then references, terms, definitions, symbols, abbreviations;
- d) corrected the editorial errors and the text.

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

FDIS	Report on voting
65C/764/FDIS	65C/774/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 publication has been drafted in accordance with ISO/IEC Directives, Part 2.

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

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

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

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

INTRODUCTION

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

The application 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.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 6-20: Application layer protocol specification – Type 20 elements

1 Scope

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 20 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 Type 20 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 define

- a) the wire-representation of the service primitives defined in IEC 61158-5-20, and
- b) the externally visible behavior associated with their transfer.

This standard specifies the protocol of the Type 20 IEC fieldbus application layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498-1) and the OSI Application Layer Structure (ISO/IEC 9545).

2 Normative references

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

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

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

IEC 61158-5-20:2014, *Industrial communication networks – Fieldbus specifications – Part 5-20: Application layer service definition – Type 20 elements*

IEC 62591:2010, *Industrial communication networks – Wireless communication network and communication profiles – WirelessHART™*

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

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

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

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

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

IEEE 802.15.4: *IEEE Standard for Local and metropolitan area networks – Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*

3 Terms, definitions, symbols, abbreviations and conventions

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

3.1 Terms and definitions from other ISO/IEC standards

3.1.1 Terms and definitions from ISO/IEC 7498-1

- a) abstract syntax
- b) application entity
- c) application process
- d) application protocol data unit
- e) application service element
- f) application entity invocation
- g) application process invocation
- h) transfer syntax

3.1.2 Terms and definitions from ISO/IEC 9545

- a) application-entity-invocation
- b) application-process-invocation
- c) application-service-element

3.1.3 Terms and definitions from ISO/IEC 8824-1

- a) object identifier
- b) type
- c) value
- d) simple type

- e) structured type
- f) component type
- g) tag
- h) true
- i) false
- j) integer type
- k) octet string type
- m) null type

3.1.4 Terms and definitions from ISO/IEC 8825-1

- 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

3.2 IEC 61158-1 terms

3.2.1

application

function or data structure for which data is consumed or produced

3.2.2

application object

object class that manages and provides the run time exchange of messages across the network and within the network device

Note 1 to entry: Multiple types of application object classes may be defined.

3.2.3

application process

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

3.2.4

application process object

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

Note 1 to entry: Application process object definitions are composed of a set of values for the attributes of their class (see the definition for Application Process Object Class Definition). Application process object definitions may be accessed remotely using the services of the FAL Object Management ASE. FAL Object Management services can be used to load or update object definitions, to read object definitions, and to dynamically create and delete application objects and their corresponding definitions.

3.2.5

application process object class

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

3.2.6

application relationship

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

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

3.2.7**application relationship endpoint**

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

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

3.2.8**attribute**

description of an externally visible characteristic or feature of an object

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

3.2.9**behaviour**

indication of how the object responds to particular events

Note 1 to entry: Its description includes the relationship between attribute values and services.

3.2.10**class**

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

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

3.2.11**class attributes**

attribute that is shared by all objects within the same class

3.2.12**class code**

unique identifier assigned to each object class

3.2.13**class specific service**

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

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

3.2.14**client**

- (a) an object which uses the services of another (server) object to perform a task
- (b) an initiator of a message to which a server reacts, such as the role of an AR endpoint in which it issues confirmed service request APDUs to a single AR endpoint acting as a server

3.2.15**comm error**

detectable error in receiving a PhPDU or DLPDU, also 'Communication error code' octet of APDU

3.2.16**conveyance path**

unidirectional flow of APDUs across an application relationship

3.2.17

cyclic

term used to describe events which repeat in a regular and repetitive manner

3.2.18

endpoint

one of the communicating entities involved in a connection

3.2.19

error

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

3.2.20

error code

identification of a specific type of error within an error class

3.2.21

management information

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

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

3.2.22

server

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

3.2.23

service

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

Note 1 to entry: A set of common services is defined and provisions for the definition of object-specific services are provided. Object-specific services are those which are defined by a particular object class to perform a required function which is not performed by a common service.

3.3 Type 20 fieldbus application-layer specific definitions

3.3.1

analog channel

continuously varying electrical signal connecting a field device to the remainder of the data acquisition or control system

Note 1 to entry: Some field devices support multiple analog channels (input or output) Each analog channel transmits a single dynamic variable to or from the field device.

3.3.2

broadcast

process of sending a PDU to all devices that are connected to the network and are able to receive the transmission

3.3.3

broadcast address

address used by a master to send a command to all devices

3.3.4**burst mode**

initiation of communication activity by a slave device at cyclic interval without request from a master

3.3.5**busy**

state of a device which is active on different tasks and cannot execute a command at the time

3.3.6**comm error**

detectable error in receiving a PhPDU or DLPDU, also 'Communication error code' octet of APDU

3.3.7**device**

any entity containing an implementation of Type 20 fieldbus

3.3.8**device ID**

serial number for a device that is unique among all instances of one type of device

Note 1 to entry: The manufacturer is required to assigned unique value for every device that has the identical values for Manufacturer ID and Device Type.

3.3.9**device type**

manufacturer's type of a device, e.g. its product name

Note 1 to entry: The value of this attribute is unique among all manufacturers and all type of devices. Its value specifies the set of commands and data objects supported by the device.

3.3.10**device variable**

uniquely defined data item within a Field Device that is always associated with the process-related information

Note 1 to entry: A device variable's value varies in response to changes and variations in the process to which the device is connected.

3.3.11**dynamic variable**

a device variable that is assigned as the dynamic variable and possibly associated with an analog channel

Note 1 to entry: A device may contain up to four variables – primary, secondary, tertiary, and quaternary variables. These are collectively called the dynamic variables.

3.3.12**expanded device type**

the manufacturer's type of the device

Note 1 to entry: The value of this attribute is unique among all manufacturers and all type of devices. Its value specifies the set of commands and data objects supported by the device.

3.3.13**field device**

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

Note 1 to entry: It directly connects to the sensor or actuator or performs process control function and it is directly connected to the physical layer specified in this standard. It may generate or receive an analog signal in addition to a digital signal.

3.3.14

long tag

32-character restricted ISO Latin-1 string used to identify a field device

3.3.15

loop current

value measured by a milli-ammeter in series with the field device

Note 1 to entry: The loop current is a near DC analog 4-20 mA signal used to communicate a single value between the control system and the field device. Voltage mode devices use "Volts DC" as their engineering units where "loop current" values are used.

3.3.16

manufacturer ID

string identifying the manufacturer that produced the device

Note 1 to entry: A manufacturer is required to use the value assigned to it and is not permitted to use the value assigned to another manufacturer.

3.3.17

master

device that initiates communication activity by sending request PDUs to another device and expecting a response frame from that device

3.3.18

network

a single pair of cable, connectors, associated signaling elements by which a given set of signaling devices are interconnected and non-signaling elements that are attached to the same pair of cable

Note 1 to entry: An installation using multiple-pair wire and a common network power supply is considered as multiple networks.

3.3.19

polling address

identifier assigned to a device such that it is unique within the network to which the device is connected

Note 1 to entry: The polling address is used to construct a one-octet address.

3.3.20

primary master

master device that can always initiate the communication

3.3.21

secondary master

master device that can initiate the communication only through an arbitration process and when primary master has relinquished the initiation of the communication

3.3.22

slave

device that initiates communication activity only after it receives a request PDU from a master device, and that is required to send a response to that request

3.3.23

tag

8-character ASCII string used to identify a field device

3.3.24**transaction**

exchange of related, consecutive frames between two peer medium access control entities, required for a successful transmission

Note 1 to entry: A transaction consists of either (a) a single PhPDU transmission from a source device, or (b) one PhPDU from the source device followed by a second, link-level acknowledgement PhPDU from the destination device.

3.3.25**unique ID**

identifier assigned to a device which is unique among all instances of the devices compliant to this standard

3.4 Abbreviations and symbols

AE	Application entity
AL	Application layer
AP	Application process
APDU	Application protocol data unit
APO	Application object
AR	Application relationship
AREP	Application relationship endpoint
ARPM	Application Relationship Protocol Machine
ASCII	American Standard Code for Information Interchange
ASE	Application Service Element
Cnf	Confirmation
DL-	(as a prefix) data-link-
DLC	Data-link connection
DLE	DL entity (the local active instance of the data-link layer)
DLL	Data-link layer
DLM	Data-link-management
DLME	Data-link-management entity
DLPDU	Data link protocol data unit
DLSAP	Data-link service access point
DLSDU	DL-service-data-unit
DR	Delayed response
DRM	Delayed response mechanism
FAL	Fieldbus application layer
HCF	HART™ Communication Foundation
ID	Identifier
IEC	International Electrotechnical Commission
Ind	Indication
LRV	Low range value
LSO	Least significant octet
MSO	Most significant octet

OSI	Open Systems Interconnect
PDU	Protocol data unit
PhL-	Physical layer (as a prefix)
PhPDU	PhL-protocol-data-unit
PV	Primary variable
QV	Quaternary variable
Req	Request
Rsp	Response
SAP	Service access point
SDU	Service data unit
SN	Sign bit
SOP	Standard Operating Procedure
SV	Secondary variable
TV	Tertiary variable
URV	Upper range value
VFD	Virtual field device

3.5 Conventions

3.5.1 General concept

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of three parts: its class definitions, its services, and its protocol specification. The first two are contained in IEC 61158-5-20. The protocol specification for each of the ASEs is defined in this standard.

The class definitions define the attributes of the classes supported by each ASE. The service specification defines the services that are provided by the ASE.

3.5.2 Conventions for class definitions

Class definitions are described using templates. Each template consists of a list of attributes for the class. The general form of the template is defined in IEC 61158-5-20, 3.5.2.

3.5.3 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 Conventions used in state machines

3.6.1 State machine diagram

This standard uses a diagram to represent a state machine. The following convention is used.

- The labeled rectangles represent states that a component can be in.
- State transitions are directed lines. They show which state a component leaves and which state it transitions to.
- A transition is labeled with the events that caused it and the corresponding actions (this may be empty). The events (above the line) causing the transition is separated from the resulting actions (below the line) by a horizontal line.
- A tilde (" ~ ") means the one's complement or negation.

3.6.2 State machine table

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

NOTE The name for the transition is not included in all state machine specifications.

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

4 Abstract syntax

The abstract syntax of APDUs is combined with their transfer syntax and is specified in Clause 5.

5 Transfer syntax

5.1 Common APDU fields

5.1.1 Command number

It is command number that is used to represent the Numeric identifier of Read, Write or Information report service. It is Unsigned8 field and it identifies the type of service and the object in the Slave that has to be accessed or action that has to be performed.

5.1.2 Extended command number

It is command number that is used to extend the Numeric Identifier of Read, Write or Information report service to Unsigned16 value. If the extended command number is used then command number field shall be set to '31'. It is Unsigned16 field and it identifies the type of service and the object in the Slave that has to be accessed or action that has to be performed.

5.1.3 Octet count

This field represents the total number of octets in all of the fields that follow this field. The data type of this field is Unsigned8 and its value can be '0' to '255'. If the value of this field is zero, then this is the last field in the APDU.

5.1.4 Response code

Response code field indicates either the successful completion of the requested command, or completion of the requested command with a warning or an error in performing the requested command, as shown in Table 2. Some of the values depend upon the value of Command field. Each service-specific APDU structure shows the response code values assigned to that service. A device shall use values specified in the service-specific APDU. The device can respond with an unused value as long it indicates a condition that is not defined in this standard. It is one octet Enumeration data type field.

Table 2 – Response code values

Value	Class	Definition
0	Success	Command (read or Write) was executed properly
1 – 127	Warning	Command (Write) was executed with the deviation as described in response (for example a value was set to the nearest allowed value)
1 – 127	Error	Command execution was not properly completed and the Response code indicates the reason (for example the device is in Write Protect mode)
1 – 127	Reserved	A code may be reserved for future use by this standard protocol. A device is not permitted to return reserved value

5.1.5 Value

The Value field is the user data which is transferred between application layer and its user. The ALE assembles it from the parameters of a service primitive or parses it into parameters of a service primitive. Its structure depends upon the type of service-specific APDU.

5.1.6 Application process status

This field is provided by the FAL user. This field is one octet Bit Field data type. Its value should be set by the FAL user as shown in Table 3, so that all user applications can interoperate with field devices compliant to this standard.

NOTE Application process status is also known as Device status, because it indicates the current operating status of the field device as a whole and is not associated with the completion of any command.

Table 3 – Application process status values

Bit mask	Name	Description
0x80	Device malfunction	The device detected a serious error or failure that compromises device operation.
0x40	Configuration changed	An operation was performed that changed the device's configuration.
0x20	Cold start	A power failure or device reset has occurred.
0x10	More status available	More status information is available via another Read service.
0x08	Loop current fixed	The Loop current or signalling voltage (for a device using voltage output) is being held at a fixed value and is not responding to process variations.
0x04	Loop current saturated	The Loop current or signalling voltage (for a device using voltage output) has reached its upper (or lower) endpoint limit and cannot increase (or decrease) any further.
0x02	Non-primary variable out of limits	A Device variable that is not mapped to the Primary variable is beyond its operating limits.
0x01	Primary variable out of limits	The Primary variable is beyond its operating limits.

5.1.7 Extended status

This field indicates additional status of the Application process. The possible values of this field are shown in Table 4. Its data type is one octet Bit Field. This field is used as part of the FAL user data or as part of the header in some of the APDUs.

Table 4 – Extended status values

Code	Value	Description
0x01	Maintenance required	The device requires maintenance, although it has not malfunctioned.
0x02	Device variable alert	One or more of the 'Device variables' are in an Alarm or Warning state.
0x04	Critical power failure	The available power is becoming critically low; used by devices that operate from stored power.

NOTE Extended status is also known as Extended device status, because it indicates the current operating status of the field device as a whole and is not associated with the completion of any command.

5.1.8 Formats of APDU structures

The sending application layer prepares an APDU to transfer it to the receiving application layer. It uses the parameters from the service primitives to do so. There are several formats of the APDU:

- request APDU from Master to Slave device using one octet command number,
- request APDU from Master to Slave device using extended command number,

- normal response from Slave to Master device using one octet command number,
- normal response from Slave to Master device using extended command number,
- command error response from Slave to Master device using one octet command number, and
- command error response from Slave to Master device using extended command number.

The common format and coding rules for all APDU structures are specified in 5.2. The formats for service-specific APDU structures are specified in 5.3. The command number field is included in the request and response both. Therefore, the requester does not have to store the request APDU to associate it with the response APDU.

5.2 Common APDU structure

5.2.1 Request APDU

The request APDU shall have a structure as shown in Figure 1. The Figure 1 (a) shows the APDU for Command number values that are one octet long and its value is not thirty-one (31). The Figure 1 (b) shows the APDU for Command values that are two octets long, also called extended command number.

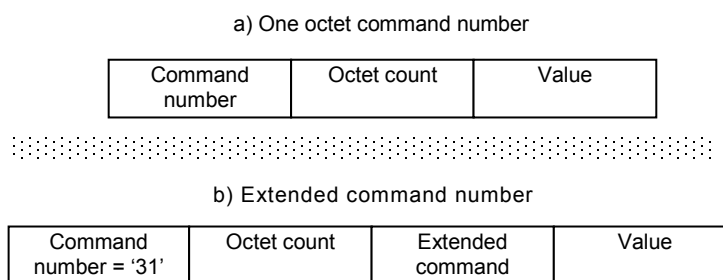


Figure 1 – Request APDU

For a Read service, Value field shall be either Null or a Variable Sub Index. For a Write service, Value field shall be Variable Sub Index, if required followed by the value that has to be written to the Slave objects identified by the Command. The ALE assembles it from the parameters of a service primitive. Its structure depends upon the service-specific APDU.

5.2.2 Normal response APDU

Figure 2 shows the structure of the application layer fields that are used for a response from Slave to Master without any error in executing the requested command. The Figure 2 (a) shows the APDU for Command number values that are one octet long and its value is not thirty-one (31). The Figure 2 (b) shows the APDU for Command values that are two octets long, also called extended command number.

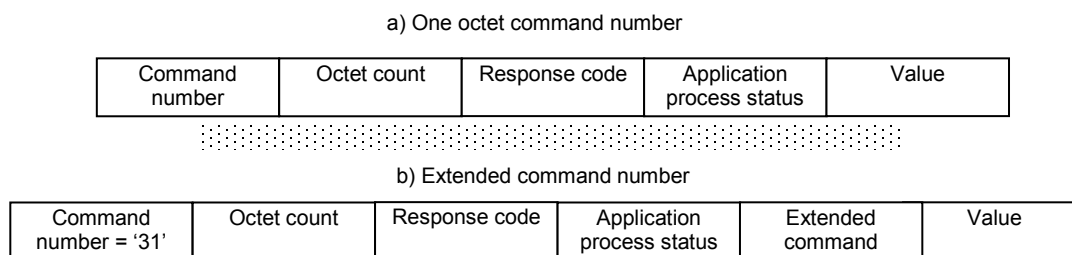


Figure 2 – Normal response APDU

The Slave sends this APDU in response to a request from Master. The value of Command field in this APDU is identical to the Command field and Extended command field (if present) in the corresponding request APDU. The minimum value of Octet count field is two '2', because Response code and Application process status fields are always included.

If the response is for a Read request, then Value field is used for the value of the objects that are read from Slave device. If the response is for a Write request, then this field is used for the value of the objects that were written in the Slave device. In the response to the Write request, it is same as the value received in the request, unless the value received was not permitted and the server wrote a different value. In that case, it is the value written in the Slave device.

5.2.3 Command error response APDU

Figure 3 shows the structure of the application layer fields that are used for a response from Slave to Master, when there is an error in executing the command. The Figure 3 (a) shows the APDU for Command number values that are one octet long and its value is not thirty-one (31). The Figure 3 (b) shows the APDU for Command values that are two octets long, also called extended command number. The Value field does not exist. The value of Octet count field shall be '2' for Command number values that are one octet long and it shall be '4' for extended command number.

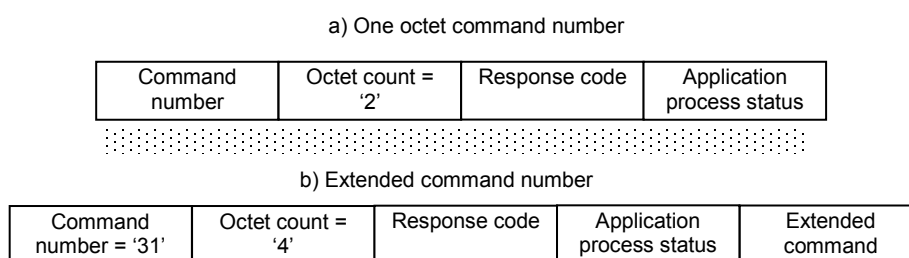


Figure 3 – Command error response from slave to master

The responding device shall send this APDU in response to a request from Master. The value of Command field in this APDU shall be identical to the Command number field in the corresponding request APDU. The Application process status shall be the current status.

5.2.4 Information report APDU

A slave device can publish data without requiring a request from Master for every publishing. It shall use the Information report service for this purpose using the response APDU specified in 5.2.2 or command aggregation APDU specified in 5.2.5. The publishing is setup to use one of the command numbers (see IEC 62591:2010, 8.2.3.32).

5.2.5 Aggregated command APDU

This standard allows multiple commands and associated data to be transported in a single request and response APDU. The command aggregation is especially useful when reading device configurations.

The request APDU format shall be as shown in Figure 1 (a) with Command number equal to '78'. The octet count shall be the length of the value field as shown in Figure 4 (a). The Request count is equal to the number of individual requests in the value field. The format of individual request is shown in Figure 4 (b). The Command number field in each command request shall be two octets long; the Octet count shall be the length of the individual request value field.

The response APDU format shall be as shown in Figure 2 (a) with Command number equal to '78'. The octet count shall be the length of the value field as shown in Figure 4 (c). The

Response count is equal to the number of individual responses in the value field. The format of individual response is shown in Figure 4 (d). The Command number field in each command response shall be two octets long; the Octet count shall be the length of the individual response code and value fields combined.

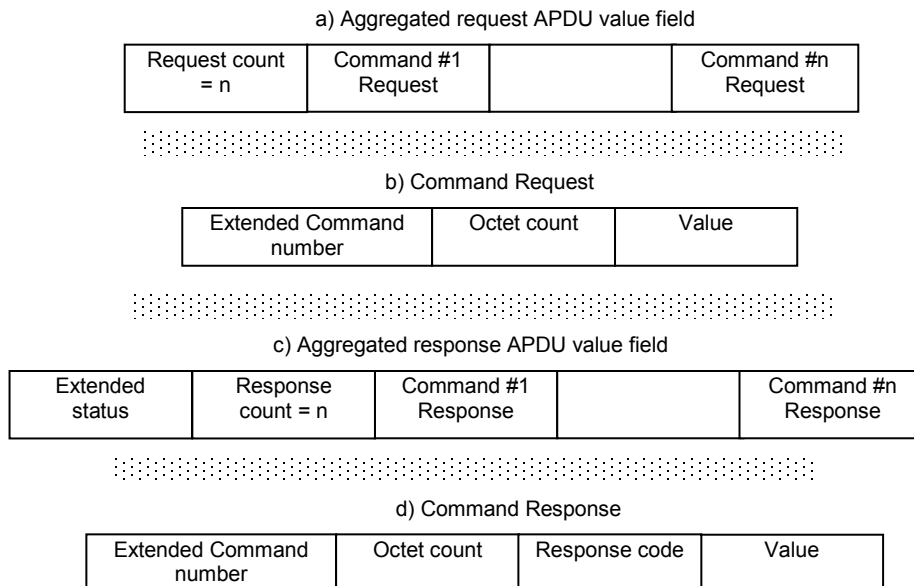


Figure 4 – Aggregated command APDU

If the device has been configured to publish more than one messages (see IEC 62591:2010, 8.2.3.32), it can combine these into one message using command aggregation, except that “Publish data Message 0” shall never be aggregated.

Whenever possible, the field device shall aggregate multiple pending “Publish data Message” into one transaction using Information report request. If the multiple “Publish data Message” does not fit into a single transaction, then these can be sent in separately.

5.3 Device application service-specific APDU structures

5.3.1 Identify FAL PDU

5.3.1.1 Request primitive

The value of Command field can be either 0 or 11 or 21. The APDU format is shown in Table 5.

Table 5 – Identify request APDU

Command	Octet count	Data
0	0	Null
11	6	Tag (Packed ASCII)
21	32	Long Tag (Latin-1)

If the value of Command field is 0, then Value field is null. If the value of Command field is 11, then Value field contains six octet value of Tag parameter. If the value of Command field is 21, then Value field contains 32 octet value of Long Tag parameter.

5.3.1.2 Response primitive

The Value field of Identify response is shown in Table 6. The Response code values are shown in Table 7.

Table 6 – Identify response value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	None	is always 254
1	Enumeration	Expanded Device Type	is a 2-octet enumeration and uniquely identifies the type of the device
3	Unsigned8	Preamble Count	indicates the minimum number of Preambles to be sent with the request message from the Master to the responding device
4	Unsigned8	Command Rev	indicates the major revision level of the Protocol supported by the device
5	Unsigned8	Device Rev	describes the revision level of the device. The value of this parameter shall be defined by the manufacturer.
6	Unsigned8	Soft Rev	indicates the revision level of the firmware in the device. The manufacturer shall increment the value of this parameter for every new release of the device's firmware
7	Bit Field	Hard Rev Phy Type	a) the five most significant bits of this octet represent the revision level of the device hardware. The manufacturer shall increment the value of this attribute for every major change of the device's hardware. It is not necessary to track individual hardware component changes. b) the least three significant bits of this octet represent the type of PhL signalling used by the device
8	Bit Field	Device Flag	indicates other information about the device such as multi-sensor, non-volatile memory control, protocol bridge, etc.
9	Unsigned24	Device ID	indicates a serial number for the device. The manufacturer shall assign a unique value for every device that has the identical values for Manufacturer ID and Expanded Device Type
12	Unsigned8	Preamble Count	indicates the minimum number of preambles to be sent with the response message from the slave to the master
13	Unsigned8	Variable Count	specifies the maximum number of objects (variables) that can be accessed from the device. The value of this parameter indicates the last variable code that a Client application can expect to be found in the device
14	Unsigned16	Configuration Change Count	keeps track of number of device configuration changes. The device shall increment the value of this parameter every time it receives a request to change the configuration using application layer services, or a user of the device changes the device configuration using local mean such as local operator's interface
16	Bit Field	Device ExtStatus	indicates the extended operational status of the device
17	Enumeration	Manufacturer ID	is a 2 octet enumeration and indicates the manufacturer that produced the device
19	Enumeration	Distributor code	is a 2 octet enumeration and indicates the private label manufacturer that distributed the device
21	Enumeration	Device profile	specifies the class to which the device belongs as shown in IEC 62591:2010, Table F.32.

Table 7 – Identify command specific response codes

Value	Class	Description
0	Success	No error

5.3.1.3 Procedure at responding device

If the request contains a Tag or Long Tag, then the device shall compare it with the Tag or Long Tag stored in the device. The device shall respond only if the value in request matches the value stored in the device.

5.3.2 Read primary variable FAL PDU

5.3.2.1 Request primitive

The value of Command is one (1); the value of Octet Count field is zero (0) and Data field is null.

5.3.2.2 Response primitive

The Value field of response is shown in Table 8. The Response Code values are shown in Table 9.

Table 8 – Read primary variable response value field

Octet offset	Data type	Parameter name	Description
0	Engineering unit	Primary Variable Unit	This parameter represents the engineering unit of primary variable.
1 – 4	Float32	Primary Variable	It is the current value of primary variable.

Table 9 – Read primary variable command specific response codes

Value	Class	Description
0	Success	No error
6	Error	Device-Specific Command Error
8	Warning	Update Failure
16	Error	Access Restricted

5.3.3 Read loop current and percent of range FAL PDU

5.3.3.1 Request primitive

The value of Command is two (2); the value of Octet Count field is zero (0) and Data field is null.

5.3.3.2 Response primitive

The Value field of response is shown in Table 10. The Response Code values are shown in Table 11.

Table 10 – Read loop current and percent of range value field

Octet offset	Data type	Parameter name	Description
0 – 3	Float32	Loop Current	It is the current value of the output of the device – either loop current in milli-amperes or voltage output in volts.
4 – 7	Float32	Primary Variable	It is the current value of primary variable in percent of range. It tracks the primary input to the device. Its value can be outside the normal range of 0 % to 100 %.

Table 11 – Read loop current and percent of range command specific response codes

Value	Class	Description
0	Success	No error
6	Error	Device-Specific Command Error
8	Warning	Update Failure
16	Error	Access Restricted

5.3.4 Read dynamic variables and loop current FAL PDU

5.3.4.1 Request primitive

The value of Command is three (3); the value of Octet Count field is zero (0) and Data field is null.

5.3.4.2 Response primitive

The Value field of response is shown in Table 12. The Response Code values are shown in Table 13. If a device does not support all four variables, then the response contains only the number of variables that the device supports. In that case, the remaining fields are absent from the response.

Table 12 – Read dynamic variables and loop current value field

Octet offset	Data type	Parameter name	Description
0 – 3	Float32	Loop Current	It is the current value of the output of the device – either loop current in milli-amperes or voltage output in volts.
4	Engineering unit	Primary Variable Unit	This parameter represents the engineering unit of primary variable.
5 – 8	Float32	Primary Variable	It is the current value of primary variable.
9	Engineering unit	Secondary Variable Unit	This parameter represents the engineering unit of secondary variable.
10 – 13	Float32	Secondary Variable	It is the current value of secondary variable.
14	Engineering unit	Tertiary Variable Unit	This parameter represents the engineering unit of tertiary variable.
15 – 18	Float32	Tertiary Variable	It is the current value of tertiary variable.
19	Engineering unit	Quaternary Variable Unit	This parameter represents the engineering unit of quaternary variable.
20 – 23	Float32	Quaternary Variable	It is the current value of quaternary variable.

Table 13 – Read dynamic variables and loop current command specific response codes

Value	Class	Description
0	Success	No error
6	Error	Device-Specific Command Error
8	Warning	Update Failure
16	Error	Access Restricted

5.3.5 Write loop configuration FAL PDU

5.3.5.1 Request primitive

The value of Command is six (6); the value of Octet Count field is two (2) and Data field is as shown in Table 14.

Table 14 – Write loop configuration value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Polling Address	This parameter represents the data-link address of the device.
1	Enumeration	Loop Current Mode	This parameter encodes the loop current mode as shown in Table C.6.

5.3.5.2 Response primitive

The Value field of response is shown in Table 14. The Response Code values are shown in Table 15.

Table 15 – Write loop configuration command specific response codes

Value	Class	Description
0	Success	No error
2	Error	Invalid Poll Address
5	Error	Too few data octets received
6	Error	Device-Specific Command Error
7	Error	In write protect mode
12	Error	Invalid mode selection
16	Error	Access Restricted
32	Error	Busy

5.3.5.3 Procedure at responding device

A devices shall disable loop current signalling when requested by the Master. When current signalling is disabled, the loop current shall be set to the minimum value required for field device operation. The Application process status bit 4 as shown in Table 3, Loop Current Fixed, shall be set to '1'.

5.3.6 Read loop configuration FAL PDU

5.3.6.1 Request primitive

The value of Command is seven (7); the value of Octet Count field is zero (0) and Data field is null.

5.3.6.2 Response primitive

The Value field of response is shown in Table 16. The Response Code values are shown in Table 17.

Table 16 – Read loop configuration value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Polling Address	This parameter represents the data-link address of the device.
1	Enumeration	Loop Current Mode	This parameter encodes the loop current mode as shown in Table C.6.

Table 17 – Read loop configuration command specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted

5.3.7 Read dynamic variable families classifications FAL PDU

5.3.7.1 Request primitive

The value of Command is eight (8); the value of Octet Count field is zero (0) and Data field is null.

5.3.7.2 Response primitive

The Value field of response is shown in Table 18. The Response Code values are shown in Table 19.

Table 18 – Read dynamic variable families classifications value field

Octet offset	Data type	Parameter name	Description
0	Enumeration	Primary Variable Classification	This parameter indicates the function performed by the device variable as specified in IEC 62591:2010, Table F.5.
1	Enumeration	Secondary Variable Classification	This parameter indicates the function performed by the device variable as specified in IEC 62591:2010, Table F.5.
2	Enumeration	Tertiary Variable Classification	This parameter indicates the function performed by the device variable as specified in IEC 62591:2010, Table F.5.
3	Enumeration	Quaternary Variable Classification	This parameter indicates the function performed by the device variable as specified in IEC 62591:2010, Table F.5.

Table 19 – Read dynamic variable families classifications command specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted

5.3.8 Read device variables with status FAL PDU

5.3.8.1 Request primitive

The value of Command is nine (9); the value of Octet count field is one to eight (1 – 8) and it is equal to number of requested variables. The request Value field is shown in Table 20.

Table 20 – Read device variables with status request value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Slot 0	This parameter is the code assigned to a device variable
1	Unsigned8	Slot 1	This parameter is the code assigned to a device variable
2	Unsigned8	Slot 2	This parameter is the code assigned to a device variable
3	Unsigned8	Slot 3	This parameter is the code assigned to a device variable
4	Unsigned8	Slot 4	This parameter is the code assigned to a device variable
5	Unsigned8	Slot 5	This parameter is the code assigned to a device variable
6	Unsigned8	Slot 6	This parameter is the code assigned to a device variable
7	Unsigned8	Slot 7	This parameter is the code assigned to a device variable

5.3.8.2 Response primitive

The Value field of response is shown in Table 21. The Response code values are shown in Table 22.

Table 21 – Read device variables with status value field

Octet offset	Data type	Parameter name	Description
0	Bit Field	Device ExtdStatus	This parameter indicates the extended operational status of the device.
1	Unsigned8	Variable Code	This is the code received in request at Value field for slot 0.
2	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 0 as specified in IEC 62591:2010, Table F.5.
3	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 0.
4	Float32	Variable Value	It is the current value of variable for slot 0.
8	Bit Field	Variable Status	This represents the status of variable for slot 0 as specified in Clause B.6.
9	Unsigned8	Variable Code	This is the code received in request at Value field for slot 1.
10	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 1 as specified in IEC 62591:2010, Table F.5.
11	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 1.
12	Float32	Variable Value	It is the current value of variable for slot 1.
16	Bit Field	Variable Status	This represents the status of variable for slot 1 as specified in Clause B.6.
17	Unsigned8	Variable Code	This is the code received in request at Value field for slot 2.
18	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 2 as specified in IEC 62591:2010, Table F.5.
19	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 2.
20	Float32	Variable Value	It is the current value of variable for slot 2.
24	Bit Field	Variable Status	This represents the status of variable for slot 2 as specified in Clause B.6.
25	Unsigned8	Variable Code	This is the code received in request at Value field for slot 3.
26	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 3 as specified in IEC 62591:2010, Table F.5.
27	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 3.

Octet offset	Data type	Parameter name	Description
28	Float32	Variable Value	It is the current value of variable for slot 3.
32	Bit Field	Variable Status	This represents the status of variable for slot 3 as specified in Clause B.6.
33	Unsigned8	Variable Code	This is the code received in request at Value field for slot 4.
34	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 4 as specified in IEC 62591:2010, Table F.5.
35	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 4.
36	Float32	Variable Value	It is the current value of variable for slot 4.
40	Bit Field	Variable Status	This represents the status of variable for slot 4 as specified in Clause B.6.
41	Unsigned8	Variable Code	This is the code received in request at Value field for slot 5.
42	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 5 as specified in IEC 62591:2010, Table F.5.
43	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 5.
44	Float32	Variable Value	It is the current value of variable for slot 5.
48	Bit Field	Variable Status	This represents the status of variable for slot 5 as specified in Clause B.6.
49	Unsigned8	Variable Code	This is the code received in request at Value field for slot 6.
50	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 6 as specified in IEC 62591:2010, Table F.5.
51	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 6.
52	Float32	Variable Value	It is the current value of variable for slot 6.
56	Bit Field	Variable Status	This represents the status of variable for slot 6 as specified in Clause B.6.
57	Unsigned8	Variable Code	This is the code received in request at Value field for slot 7.
58	Enumeration	Variable classification	This indicates the function performed by the device variable for slot 7 as specified in IEC 62591:2010, Table F.5.
59	Engineering unit	Variable Unit	This represents the engineering unit of variable for slot 7.
60	Float32	Variable Value	It is the current value of variable for slot 7.
64	Bit Field	Variable Status	This represents the status of variable for slot 7 as specified in Clause B.6.
65	Time	Time stamp	Slot 0 data time stamp

Table 22 – Read device variables with status command specific response codes

Value	Class	Description
0	Success	No error
2	Error	Invalid selection
5	Error	Too few data octets received
6	Error	Device-Specific Command Error
8	Warning	Update Failure
14	Warning	Dynamic variables returned for device variables
16	Error	Access restricted
30	Warning	Command response truncated

5.3.8.3 Procedure at responding device

If the request does not contain any data octets (Value field) then the device shall respond with error code equal to 5; otherwise the respond code shall be a value other than 5.

The device shall respond with values of exactly as many variables as in the request.

If a variable requested is not supported in the device, then in the response APDU, the corresponding:

Variable classification field shall be set to '0';

Variable Unit field shall be set to '250';

Variable Value field shall be set to a floating point value of 'Not-a-number' (0x7F, 0xA0, 0x00, 0x00); and

Variable Status field shall be set to 0x30.

If the Variable classification is not supported for a requested variable, then Variable classification field in the response APDU for that variable shall be set to '0' (not yet classified) and the respond code shall be set to '14'.

The device shall support at least four slots.

5.3.9 Read message FAL PDU

5.3.9.1 Request primitive

The value of Command is twelve (12); the value of Octet Count field is zero (0) and Data field is null.

5.3.9.2 Response primitive

The Value field of response is shown in Table 23. The Response Code values are shown in Table 24.

Table 23 – Read message response value field

Octet offset	Data type	Parameter name	Description
0 – 23	Packed ASCII	Message	This is the value of Message stored in the device.

Table 24 – Read message command specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted
32	Error	Busy

5.3.10 Read tag, descriptor and date FAL PDU

5.3.10.1 Request primitive

The value of Command is thirteen (13); the value of Octet Count field is zero (0) and Data field is null.

5.3.10.2 Response primitive

The Value field of response is shown in Table 25. The Response Code values are shown in Table 26.

Table 25 – Read tag, descriptor, date response value field

Octet offset	Data type	Parameter name	Description
0 – 5	Packed ASCII	Tag	This is the value of Tag stored in the device.
6 – 17	Packed ASCII	Descriptor	This is the value of Descriptor stored in the device.
18 – 20	Date	Date	This is the value of Date stored in the device.

Table 26 – Read tag, descriptor, date command specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted
32	Error	Busy

5.3.11 Read primary variable transducer information FAL PDU

5.3.11.1 Request primitive

The value of Command is fourteen (14); the value of Octet Count field is zero (0) and Data field is null.

5.3.11.2 Response primitive

The Value field of response is shown in Table 27. The Response Code values are shown in Table 28.

Table 27 – Read primary variable transducer information response value field

Octet offset	Data type	Parameter name	Description
0 – 2	Unsigned24	Transducer Serial Number	This is the value of Transducer Serial Number.
3	Engineering unit	Transducer Unit	This is the code for Transducer Limits and Minimum Span Units
4 – 7	Float32	Upper Transducer Limit	It is the value of Upper Transducer Limit variable.
8 – 11	Float32	Lower Transducer Limit	It is the value of Lower Transducer Limit variable.
12 – 15	Float32	Minimum Span	It is the value of Minimum Span variable.

Table 28 – Read primary variable transducer information command specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted
32	Error	Busy

5.3.11.3 Procedure at responding device

If the Transducer Serial Number is not applicable to the device, then the device shall respond with following values:

- a) Transducer Serial Number equal to '0';
- b) Transducer Unit equal to '250';
- c) all other values equal to a floating point value of 'Not-a-number' (0x7F, 0xA0, 0x00, 0x00).

5.3.12 Read device information FAL PDU

5.3.12.1 Request primitive

The value of Command is fifteen (15); the value of Octet Count field is zero (0) and Data field is null.

5.3.12.2 Response primitive

The Value field of response is shown in Table 29. The Response Code values are shown in Table 30.

Table 29 – Read device information response value field

Octet offset	Data type	Parameter name	Description
0	Enumeration	PV Alarm Selection	This is the value of PV Alarm Selection as shown in Table C.2. It indicates the action taken by the device under error conditions. For transmitters, the code indicates the action taken by the Loop Current. For Actuators, the action taken by the positioner is indicated.
1	Enumeration	PV Transfer Function	This is the value of PV Transfer Function as shown in Table C.1. If the device does not support transfer functions, then this value is '0'.
2	Engineering unit	PV Range Unit	This is the value of PV upper and lower Range Unit.
3 – 6	Float32	PV Upper Range	PV Upper Range Value
7 – 10	Float32	PV Lower Range	PV Lower Range Value
11 – 14	Float32	PV Damping	PV Damping Value in seconds
15	Enumeration	Write Protect	This is the value of Write Protect as shown in Table C.3. If the device does not implement write protection, then this value is '251'.
16	Enumeration	Reserved	This field is reserved for future use. Its value shall be '250'.
17	Bit Field	PV Analog Channel	This is the value of PV Analog Channel flags as shown in Table C.8.

Table 30 – Read device information command specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted
32	Error	Busy

5.3.13 Read final assembly number FAL PDU**5.3.13.1 Request primitive**

The value of Command is sixteen (16); the value of Octet Count field is zero (0) and Data field is null.

5.3.13.2 Response primitive

The Value field of response is shown in Table 31. The Response Code values are shown in Table 32.

Table 31 – Read final assembly number response value field

Octet offset	Data type	Parameter name	Description
0 – 2	Unsigned24	Final Assembly Number	This is the value of Final Assembly Number

Table 32 – Read final assembly number command specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted
32	Error	Busy

5.3.14 Write message FAL PDU**5.3.14.1 Request primitive**

The value of Command is seventeen (17); the value of Octet Count field is twenty-four (24) and Data field is as shown in Table 33.

Table 33 – Write message value field

Octet offset	Data type	Parameter name	Description
0 – 23	Packed ASCII	Message	This is a string used by a Master for record keeping

5.3.14.2 Response primitive

The Value field of response is shown in Table 33. The Response Code values are shown in Table 34.

Table 34 – Write message command specific response codes

Value	Class	Description
0	Success	No error
2	Error	Invalid Poll Address
5	Error	Too few data octetsreceived
6	Error	Device-Specific Command Error
7	Error	In write protect mode
16	Error	Access Restricted
32	Error	Busy

5.3.15 Write tag, descriptor and date FAL PDU

5.3.15.1 Request primitive

The value of Command is eighteen (18); the value of Octet Count field is twenty-one (21) and Data field is as shown in Table 35.

Table 35 – Write tag, descriptor, date value field

Octet offset	Data type	Parameter name	Description
0 – 5	Packed ASCII	Tag	This is the value of Tag to be stored in the device.
6 – 17	Packed ASCII	Descriptor	This is the value of Descriptor to be stored in the device.
18 – 20	Date	Date	This is the value of Date to be stored in the device.

5.3.15.2 Response primitive

The Value field of response is shown in Table 35. The Response Code values are shown in Table 36.

Table 36 – Write tag, descriptor, date command specific response codes

Value	Class	Description
0	Success	No error
5	Error	Too few data octetsreceived
6	Error	Device-Specific Command Error
7	Error	In write protect mode
9	Error	Invalid Date code
16	Error	Access Restricted
32	Error	Busy

5.3.16 Write final assembly number FAL PDU

5.3.16.1 Request primitive

The value of Command is nineteen (19); the value of Octet Count field is three (3) and Data field is as shown in Table 37.

Table 37 – Write final assembly number value field

Octet offset	Data type	Parameter name	Description
0 – 2	Unsigned24	Final Assembly Number	This is the value of Final Assembly Number. It is normally changed when electronics or other device components are upgraded in the field.

5.3.16.2 Response primitive

The Value field of response is shown in Table 37. The Response Code values are shown in Table 38.

Table 38 – Write final assembly number command specific response codes

Value	Class	Description
0	Success	No error
5	Error	Too few data octets received
6	Error	Device-Specific Command Error
7	Error	In write protect mode
16	Error	Access Restricted
32	Error	Busy

5.3.17 Read long tag FAL PDU**5.3.17.1 Request primitive**

The value of Command is twenty (20); the value of Octet Count field is zero (0) and Data field is null.

5.3.17.2 Response primitive

The Value field of response is shown in Table 39. The Response Code values are shown in Table 40.

Table 39 – Read long tag response value field

Octet offset	Data type	Parameter name	Description
0 – 31	Latin-1	Long Tag	This is the value of Long Tag stored in the device.

Table 40 – Read long tag command-specific response codes

Value	Class	Description
0	Success	No error
16	Error	Access Restricted
32	Error	Busy

5.3.18 Write long tag FAL PDU**5.3.18.1 Request primitive**

The value of Command is twenty-two (22); the value of Octet Count field is thirty-two (32) and Data field is as shown in Table 41.

Table 41 – Write long tag value field

Octet offset	Data type	Parameter name	Description
0 – 31	Latin-1	Long Tag	This is the value of Long Tag to be stored in the device.

5.3.18.2 Response primitive

The Value field of response is shown in Table 41. The Response Code values are shown in Table 42.

Table 42 – Write long tag command specific Response codes

Value	Class	Description
0	Success	No error
5	Error	Too few data octets received
6	Error	Device-Specific Command Error
7	Error	In write protect mode
16	Error	Access Restricted
32	Error	Busy
33	Error	DR initiated
34	Error	DR running
35	Error	DR dead
36	Error	DR conflict

5.3.19 Reset configuration changed flag FAL PDU

This Command is specified in IEC 62591:2010, 8.2.3.19.

5.3.20 Perform self test FAL PDU

This Command is specified in IEC 62591:2010, 8.2.3.20.

5.3.21 Perform device reset FAL PDU

This Command is specified in IEC 62591:2010, 8.2.3.21.

5.3.22 Read additional device status FAL PDU

This Command is specified in IEC 62591:2010, 8.2.3.22.

5.3.23 Reset more status available FAL PDU

This Command is specified in IEC 62591:2010, 8.2.3.23.

5.3.24 Read device variable information FAL PDU

This Command is specified in IEC 62591:2010, 8.2.3.24.

5.3.25 Write device variable FAL PDU

This Command is specified in IEC 62591:2010, 8.2.3.25.

5.3.26 Write primary variable range FAL PDU

5.3.26.1 Request primitive

The value of Command is thirty-five (35); the value of Octet count field is nine (9) and Value field is as shown in Table 43.

Table 43 – Write primary variable range value field

Octet offset	Data type	Parameter name	Description
0	Engineering unit	Range unit	Measurement unit of the Range value
1	Float32	Upper range value	Range value to be written
5	Float32	Lower range value	Range value to be written

5.3.26.2 Response primitive

The Value field of response is shown in Table 43. The Response code values are shown in Table 44.

Table 44 – Write primary variable range command specific response codes

Value	Class	Description
0	Success	No error
2	Error	Invalid selection
5	Error	Too few data octets received
6	Error	Device-specific Command error
7	Error	In write protect mode
8	Warning	Set to the nearest possible value
9	Error	Lower range value too high
10	Error	Lower range value too low
11	Error	Upper range value too high
12	Error	Upper range value too low
13	Error	Upper and Lower range values out of limits
14	Warning	Span too small (device accuracy may be impaired).
16	Error	Access Restricted
18	Error	Invalid Units code
29	Error	Invalid span
32	Error	Busy

5.3.26.3 Procedure at responding device

The Upper range value of the Primary variable can be lower than its Lower range value, enabling the device to be operated with reverse action.

NOTE The device-specific document will indicate if this capability has not been implemented.

The range values shall be returned in the same units as received. The value returned in the response shall be the rounded or truncated value actually used by the device. The Primary variable range Unit code received with this command shall not affect the Primary variable Unit of the device. For a current output device, the Range values allow the Primary variable value to be converted to a percent for transmission via the Loop current. For a current input device,

the Range values allow the Loop current to be converted to a percent for use by the actuator (e.g., to use as the actuator setpoint).

5.3.27 Enter-exit fixed current mode FAL PDU

5.3.27.1 Request primitive

The value of Command is forty (40); the value of Octet count field is four (4) and Value field is as shown in Table 45.

Table 45 – Enter-exit fixed current mode request value field

Octet offset	Data type	Parameter name	Description
0	Float32	Fixed current level	Loop current value to be set

5.3.27.2 Response primitive

The Value field of response is shown in Table 46. The Response code values are shown in Table 47.

Table 46 – Enter-exit fixed current mode response value field

Octet offset	Data type	Parameter name	Description
0	Float32	Actual current level	Actual current value

Table 47 – Enter-exit fixed current mode command specific response codes

Value	Class	Description
0	Success	No error
3	Error	Fixed current level too high
4	Error	Fixed current level too low
5	Error	Too few data octets received
6	Error	Device-specific Command error
7	Error	In write protect mode
11	Error	Loop current not active (device in multidrop mode)
16	Error	Access Restricted
32	Error	Busy

5.3.27.3 Procedure at responding device

If the Fixed current level value in the request is zero, then the device shall exit the fixed Loop current mode. Else, if the value is valid then the device shall be put in the fixed Loop current mode using the value in the request. The value returned in the response shall be the rounded or truncated value actually used by the device.

Fixed Loop current mode shall be exited when power is removed and reapplied to the device.

5.3.28 Write primary variable unit FAL PDU

5.3.28.1 Request primitive

The value of Command is forty (44); the value of Octet count field is one (1) and Value field is as shown in Table 48.

Table 48 – Write primary variable unit value field

Octet offset	Data type	Parameter name	Description
0	Engineering unit	Primary variable unit code	Primary variable measurement unit code

5.3.28.2 Response primitive

The Value field of response is shown in Table 48. The Response code values are shown in Table 49.

Table 49 – Write primary variable unit command specific response codes

Value	Class	Description
0	Success	No error
2	Error	Invalid selection
5	Error	Too few data octets received
6	Error	Device-specific Command error
7	Error	In write protect mode
16	Error	Access Restricted
32	Error	Busy

5.3.28.3 Procedure at responding device

The device shall select the units in which the Primary variable and its range are measured. This command also selects the unit for transducer limits and minimum span.

5.3.29 Trim loop current zero FAL PDU**5.3.29.1 Request primitive**

The value of Command is forty (45); the value of Octet count field is four (4) and Value field is as shown in Table 50. In the request PDU it is the externally measured Loop current.

Table 50 – Trim loop current zero request value field

Octet offset	Data type	Parameter name	Description
0	Float32	Zero loop current	Loop current level in mA

5.3.29.2 Response primitive

The Value field of response is shown in Table 50. In the response PDU it is the actual Loop current used by the device. The Response code values are shown in Table 51.

Table 51 – Trim loop current zero command specific response codes

Value	Class	Description
0	Success	No error
3	Error	Zero loop current level too high
4	Error	Zero loop current level too low
5	Error	Too few data octets received
6	Error	Device-specific Command error

Value	Class	Description
7	Error	In write protect mode
9	Error	Incorrect Loop current mode or value
11	Error	Loop current not active (device in multidrop mode)
16	Error	Access Restricted
32	Error	Busy

5.3.29.3 Procedure at responding device

The Field device shall trim its zero calibration of the Loop current so that the Loop current value matches the value received from the Master. The value returned in the response shall be the rounded or truncated value actually used by the device.

NOTE This trim is typically performed by adjusting the Loop current to 4,00 mA and sending the measured value to the Field device.

Response code 9 (Incorrect Loop current mode or value) shall be returned if the device is not in the proper mode to allow the Loop current to be calibrated or if the current was not set to exactly the minimum value.

If the device is a voltage output device, then it shall interpret the Loop current value as output value in Volts DC.

5.3.30 Trim loop current gain FAL PDU

5.3.30.1 Request primitive

The value of Command is forty (46); the value of Octet count field is four (4) and Value field is as shown in Table 52. In the request PDU it is the externally measured Loop current.

Table 52 – Trim loop current gain request value field

Octet offset	Data type	Parameter name	Description
0	Float32	Gain loop current	Loop current level in mA

5.3.30.2 Response primitive

The Value field of response is shown in Table 52. In the response PDU it is the actual Loop current used by the device. The Response code values are shown in Table 53.

Table 53 – Trim loop current gain command specific response codes

Value	Class	Description
0	Success	No error
3	Error	Gain loop current level too high
4	Error	Gain loop current level too low
5	Error	Too few data octets received
6	Error	Device-specific Command error
7	Error	In write protect mode
9	Error	Incorrect Loop current mode or value
11	Error	Loop current not active (device in multidrop mode)
16	Error	Access Restricted

Value	Class	Description
32	Error	Busy

5.3.30.3 Procedure at responding device

The Field device shall trim its gain calibration of the Loop current so that the Loop current value matches the value received from the Master. The value returned in the response shall be the rounded or truncated value actually used by the device.

NOTE This trim is typically performed by adjusting the Loop current to 20,00 mA and sending the measured value to the Field device.

Response code 9 (Incorrect Loop current mode or value) shall be returned if the device is not in the proper mode to allow the Loop current to be calibrated or if the current was not set to exactly the maximum value.

If the device is a voltage output device, then it shall interpret the Loop current value as output value in Volts DC.

5.3.31 Read dynamic variable assignment FAL PDU

5.3.31.1 Request primitive

The value of Command is fifty (50); the value of Octet count field is zero (0) and Value field is null.

5.3.31.2 Response primitive

The value of Octet count field is four (4) and Value field is shown in Table 54. The Response code values are shown in Table 55.

Table 54 – Read dynamic variable assignment response value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Primary variable assignment	Device variable assigned to the Primary variable as specified in Table C.9.
1	Unsigned8	Secondary variable assignment	Device variable assigned to the Secondary variable as specified in Table C.9.
2	Unsigned8	Tertiary variable assignment	Device variable assigned to the Tertiary variable as specified in Table C.9.
3	Unsigned8	Quaternary variable assignment	Device variable assigned to the Quaternary variable as specified in Table C.9.

Table 55 – Read dynamic variable assignment command specific response codes

Value	Class	Description
0	Success	No error
6	Error	Device-specific Command error
16	Error	Access Restricted

5.3.31.3 Procedure at responding device

The device shall respond with the Device variable numbers that are assigned to the Primary, Secondary, Tertiary, and Quaternary Variables. The Field device shall return all response data octets. Unsupported Dynamic variable assignment shall be set to "250" (not used).

5.3.32 Write dynamic variable assignment FAL PDU

5.3.32.1 Request primitive

The value of Command is fifty-one (51); the value of Octet count field is four (4) and Value field is as shown in Table 56.

Table 56 – Write dynamic variable assignment value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Primary variable assignment	Device variable assigned to the Primary variable as specified in Table C.9.
1	Unsigned8	Secondary variable assignment	Device variable assigned to the Secondary variable as specified in Table C.9.
2	Unsigned8	Tertiary variable assignment	Device variable assigned to the Tertiary variable as specified in Table C.9.
3	Unsigned8	Quaternary variable assignment	Device variable assigned to the Quaternary variable as specified in Table C.9.

5.3.32.2 Response primitive

The Value field of response is shown in Table 56. The Response code values are shown in Table 57.

Table 57 – Write dynamic variable assignment command specific response codes

Value	Class	Description
0	Success	No error
2	Error	Invalid selection
5	Error	Too few data octets received
6	Error	Device-specific Command error
7	Error	In write protect mode
16	Error	Access Restricted
32	Error	Busy

5.3.32.3 Procedure at responding device

The device shall write the Device variable numbers that are assigned to the Primary, Secondary, Tertiary and Quaternary variables. If the request has one or more data octets, it shall not respond with "Too few data octets received". The variable assignment not included in the request may be set to any valid Device variable in the Field device. The Field device shall always return all four response data octets. Unsupported Dynamic variable assignment shall be set to "250" (not used).

5.3.33 Write number of response preambles FAL PDU

5.3.33.1 Request primitive

The value of Command is fifty-nine (59); the value of Octet count field is one (1) and Value field is as shown in Table 58.

Table 58 – Write number of response preambles value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Preamble count	Number of preambles characters to be sent in the PhPDU from the Slave to the Master

5.3.33.2 Response primitive

The Value field of response is shown in Table 58. The Response code values are shown in Table 59.

Table 59 – Write number of response preambles command specific response codes

Value	Class	Description
0	Success	No error
3	Error	Preamble count too high
4	Error	Preamble count too low
5	Error	Too few data octets received
6	Error	Device-specific Command error
7	Error	In write protect mode
8	Warning	Set to the nearest possible value
16	Error	Access Restricted
32	Error	Busy

5.3.33.3 Procedure at responding device

The device shall use the Preamble count parameter received in this request as the length of preamble in all PhPDU that is sent by the device. If the received value is less than five (5) or more than twenty (20), then the device shall use either reject the command and send appropriate response code.

5.3.34 Read device variable trim points FAL PDU**5.3.34.1 Request primitive**

The value of Command is eighty (80); the value of Octet count field is one (1) and Value field is as shown in Table 60.

Table 60 – Read device variable trim points request value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Device variable	Device variable code as specified in Table C.9

5.3.34.2 Response primitive

The value of Octet count field is ten (10) and Value field is shown in Table 61. The Response code values are shown in Table 62.

Table 61 – Read device variable trim points response value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Device variable	Device variable code as specified in Table C.9
1	Engineering unit	Trim point unit	Measurement unit code of trim points
2	Float32	Lower trim point	Lower or single trim point value
6	Float32	Upper trim point	Upper trim point value

Table 62 – Read device variable trim points command specific response codes

Value	Class	Description
0	Success	No error
5	Error	Too few data octets received
6	Error	Device-specific Command error
17	Error	Invalid Device variable code i.e. the requested Device variable does not exist.
19	Error	Device Variable code not allowed for this command
32	Error	Busy

5.3.34.3 Procedure at responding device

The device shall respond with the most recent value used trim point values. If the Device variable does not support a trim operation, then both the upper and lower trim points shall be 'Not-A-Number'. If the Device variable supports only a single trim point, then the upper trim point shall be 'Not-A-Number'.

5.3.35 Read device variable trim guidelines FAL PDU**5.3.35.1 Request primitive**

The value of Command is eighty-one (81); the value of Octet count field is one (1) and Value field is as shown in Table 63.

Table 63 – Read device variable trim guidelines request value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Device variable	Device variable code as specified in Table C.9

5.3.35.2 Response primitive

The value of Octet count field is twenty-three (23) and Value field is shown in Table 64. The Response code values are shown in Table 65.

Table 64 – Read device variable trim guidelines response value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Device variable	Device variable code as specified in Table C.9
1	Enumeration	Trim points supported	Trim points that can be set in the device as enumerated in Table C.7
2	Engineering unit	Trim point unit	Measurement unit code of trim points

Octet offset	Data type	Parameter name	Description
3	Float32	Minimum lower trim point	Minimum value that will be accepted by the device during a lower trim procedure
7	Float32	Maximum lower trim point	Maximum value that will be accepted by the device during a lower trim procedure
11	Float32	Minimum upper trim point	Minimum value that will be accepted by the device during a upper trim procedure
15	Float32	Maximum upper trim point	Maximum value that will be accepted by the device during a upper trim procedure
19	Float32	Minimum trim point difference	Minimum difference between upper and lower trim points that will be accepted by the device during a trim procedure

Table 65 – Read device variable trim points command specific response codes

Value	Class	Description
0	Success	No error
5	Error	Too few data octets received
6	Error	Device-specific Command error
17	Error	Invalid Device variable code i.e. the requested Device variable does not exist.
19	Error	Device Variable code not allowed for this command
32	Error	Busy

5.3.35.3 Procedure at responding device

The device shall respond with the information that a Host will need to guide a user through a correct selection of trim points. If the Device variable does not support a trim operation, then Trim points supported shall be zero (0), Trim point unit shall be 'Not used' and all trim point values shall be 'Not-A-Number'. If the Device variable supports only a single trim point, then the upper trim point and trim point difference shall be 'Not-A-Number'.

5.3.36 Write device variable trim point FAL PDU

5.3.36.1 Request primitive

The value of Command is eighty-two (82); the value of Octet count field is seven (7) and Value field is as shown in Table 66.

Table 66 – Write device variable trim point value field

Octet offset	Data type	Parameter name	Description
0	Unsigned8	Device variable	Device variable code as specified in Table C.9
1	Enumeration	Trim points type	Trim points that can be set in the device as enumerated in Table C.7
2	Engineering unit	Trim point unit	Measurement unit code of trim point
3	Float32	Trim point value	Trim point value to be written

5.3.36.2 Response primitive

The Value field of response is shown in Table 66. The Response code values are shown in Table 67.