IEC 61158-6-18

Edition 1.0 2007-12

INTERNATIONAL STANDARD

Industrial communication networks – Fieldbus specifications –
Part 6-18: Application layer protocol specification – Type 18 elements

IEC 61158-6-18:2007(E)

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland
Email: inmail@iec.ch
Web: 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.

- Catalogue of IEC publications: www.iec.ch/searchpub
The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub
Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.

- Electropedia: www.electropedia.org
The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv
If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00

IEC 61158-6-18

Edition 1.0 2007-12

INTERNATIONAL STANDARD

Industrial communication networks – Fieldbus specifications – Part 6-18: Application layer protocol specification – Type 18 elements

INTERNATIONAL ELECTROTECHNICAL COMMISSION

PRICE CODE XB

ICS 35.100.70; 25.040.40

ISBN 2-8318-9496-4

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 6-18: Application layer service definition – Type 18 elements

FOREWORD

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

NOTE Use of some of the associated protocol types is restricted by their intellectual-property-right holders. In all cases, the commitment to limited release of intellectual-property-rights made by the holders of those rights permits a particular data-link layer protocol type to be used with physical layer and application layer protocols in Type combinations as specified explicitly in the IEC 61784 series. Use of the various protocol types in other combinations may require permission from their respective intellectual-property-right holders.

IEC draws attention to the fact that it is claimed that compliance with this standard may involve the use of patents as follows, where the [xx] notation indicates the holder of the patent right:

Type 18:

3343036/Japan	[MEC]	"Network System for a Programmable Controller"
5896509/USA	[MEC]	"Network System for a Programmable Controller"
246906/Korea	[MEC]	"Network System for a Programmable Controller"
Pending/Germany	[MEC]	"Network System for a Programmable Controller"

IEC takes no position concerning the evidence, validity and scope of these patent rights.

The holders of these patent rights have assured IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holders of these patent rights are registered with IEC. Information may be obtained from:

[MEC] Mitsubishi Electric Corporation
Corporate Licensing Deivsion
7-3, Marunouchi 2-chome, Chiyoda-ku,
Tokyo 100-8310, Japan

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

International Standard IEC 61158-6-18 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation.

This first edition and its companion parts of the IEC 61158-6 subseries cancel and replace IEC 61158-6:2003. This edition of this part constitutes a technical addition.

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

a) deletion of the former Type 6 fieldbus for lack of market relevance;

b) addition of new types of fieldbuses;

c) partition of part 6 of the third edition into multiple parts numbered -6-2, -6-3, ...

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

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

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

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

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

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

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

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

INTRODUCTION

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

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

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

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

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 6-18: Application layer protocol specification – Type 18 elements

1 Scope

1.1 General

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

This standard provides common elements for basic time-critical and non-time-critical messaging communications between application programs in an automation environment and material specific to Type 18 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 specifies interactions between remote applications and defines the externally visible behavior provided by the Type 18 fieldbus application layer in terms of

a) the formal abstract syntax defining the application layer protocol data units conveyed between communicating application entities;
b) the transfer syntax defining encoding rules that are applied to the application layer protocol data units;
c) the application context state machine defining the application service behavior visible between communicating application entities;
d) the application relationship state machines defining the communication behavior visible between communicating application entities.

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

1) define the wire-representation of the service primitives defined in IEC 61158- 5-18, and
2) define the externally visible behavior associated with their transfer.

This standard specifies the protocol of the Type 18 fieldbus application layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498) and the OSI application layer structure (ISO/IEC 9545).

1.2 Specifications

The principal objective of this standard is to specify the syntax and behavior of the application layer protocol that conveys the application layer services defined in IEC 61158-5-18.

A secondary objective is to provide migration paths from previously-existing industrial communications protocols. It is this latter objective which gives rise to the diversity of protocols standardized in the IEC 61158-6 series.

1.3 Conformance

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

Conformance is achieved through implementation of this application layer protocol specification.

2 Normative references

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

IEC 61158-5-18, *Industrial communication networks – Fieldbus specifications – Part 5-18: Application layer service definition – Type 18 elements*

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

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

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

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

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

3 Terms and definitions

3.1 Terms and definitions from other ISO/IEC standards

3.1.1 ISO/IEC 7498-1 terms

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

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

3.1.2 ISO/IEC 8822 terms

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

a) abstract syntax
b) presentation context

3.1.3 ISO/IEC 9545 terms

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

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

3.1.4 ISO/IEC 8824 terms

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

a) object identifier
b) type

3.2 Other terms and definitions

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

3.2.1
active connection control object
instance of a certain FAL class that abstracts the interconnection facility (as Consumer and Provider) of an automation device

3.2.2
alarm
activation of an event that shows a critical state

3.2.3
alarm ack
acknowledgment of an event that shows a critical state

3.2.4
alarm data object
object(s) which represent critical states referenced by device/slot/subslot/alarm type

3.2.5
allocate
take a resource from a common area and assign that resource for the exclusive use of a specific entity

3.2.6
application
function or data structure for which data is consumed or produced

3.2.7
application layer interoperability
capability of application entities to perform coordinated and cooperative operations using the services of the FAL

3.2.8
application objects
multiple object classes that manage and provide a run time exchange of PDUs across the network and within the network device

3.2.9
application process
part of a distributed application on a network, which is located on one device and unambiguously addressed

3.2.10
application process identifier
distinguishes multiple application processes used in a device

NOTE Application process identifier is assigned by PROFIBUS International (PI).

3.2.11
application process object
component of an application process that is identifiable and accessible through an FAL application relationship

NOTE 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.

3.2.12
application process object class
a class of application process objects defined in terms of the set of their network-accessible attributes and services

3.2.13
application relationship
cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation. This relationship is activated either by the exchange of application-protocol-data-units or as a result of preconfiguration activities

3.2.14
application relationship application service element
application-service-element that provides the exclusive means for establishing and terminating all application relationships

3.2.15
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 Each application process involved in the application relationship maintains its own application relationship endpoint.

3.2.16
attribute
description of an externally visible characteristic or feature of an object

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

3.2.17
backup
status of the IO AR, which indicates that it, is in the standby state

3.2.18
behavior
indication of how an object responds to particular events

3.2.19
channel
representation of a single physical or logical link of an input or output application object of a server to the process in order to support addressing of diagnosis information

NOTE The channel typically represents a single connector or clamp as a real interface of a module or sub-module. This reference is used to identify points of failure within diagnosis PDUs.

3.2.20
channel related diagnosis
information concerning a specific element of an input or output application object, provided for maintenance purposes

EXAMPLE open loop.

3.2.21
class
a set of objects, all of which represent the same kind of system component

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

3.2.22
class attributes
attribute that is shared by all objects within the same class

3.2.23
class code
unique identifier assigned to each object class

3.2.24
class specific service
service defined by a particular object class to perform a required function which is not performed by a common service

NOTE A class specific object is unique to the object class which defines it.

3.2.25
clear
status of the IO controller, which indicates that the control algorithm is currently not running

3.2.26
client

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

b) initiator of a PDU to which a server reacts

3.2.27
common profile
a collection of device independent information and functionality providing consistency between all devices

3.2.28
communication data object
object(s) which are parameter of communication relationships and referenced by device/slot/ subslot/ index

3.2.29
configuration check
comparison of the expected IO-Data object structuring of the client with the real IO-Data object structuring to the server in the start-up phase

3.2.30
configuration data base
interconnection information maintained by the ACCO ASE

3.2.31
configuration fault
an unacceptable difference between the expected IO-Data object structuring and the real IO-Data object structuring, as detected by the server

3.2.32
configuration identifier
representation of a portion of IO Data of a single input- and/or output-module of a server

3.2.33
connection
the logical link between sink and source of attributes and services at different custom interfaces of Custom RT-Auto objects

3.2.34
connection channel
description of a connection between a sink and a source of data items

3.2.35
consume
act of receiving data from a provider

3.2.36
consumer
node or sink that is receiving data from a producer

3.2.37
consumerID
unambiguous identifier within the scope of the ACCO assigned by the consumer to recognize the internal data of a configured interconnection sink.

3.2.38
context management
network-accessible information (communication objects) that supports managing the operation of the fieldbus system, including the application layer

NOTE Managing includes functions such as controlling, monitoring, and diagnosing.

3.2.39
conveyance path
unidirectional flow of APDUs across an application relationship

3.2.40
cyclic
repetitive in a regular manner

3.2.41
data consistency
means for coherent transmission and access of the input- or output-data object between and within client and server

3.2.42
data marshaling
the encoding of parameters of the FAL service primitives with respect to their interface definition

NOTE This is part of the abstract ORPC model.

3.2.43
device
physical hardware connected to the link

NOTE A device may contain more than one node.

3.2.44
device ID
a vendor assigned device type identification

3.2.45
device profile
a collection of device dependent information and functionality providing consistency between similar devices of the same device type

3.2.46
diagnosis data object
object(s) which contains diagnosis information referenced by device/slot/subslot/index

3.2.47
diagnosis information
all data available at the server for maintenance purposes

3.2.48
dynamic reconfiguration
change of IO data objects without interruption of an established application relationship and continuous updating of non-changed IO data objects

3.2.49
endpoint
one of the communicating entities involved in a connection

3.2.50
engineering
abstract term that characterizes the client application or device responsible for configuring an automation system via interconnecting data items

3.2.51
error
discrepancy between a computed, observed or measured value or condition and the specified or theoretically correct value or condition

3.2.52
error class
general grouping for related error definitions and corresponding error codes

3.2.53
error code
identification of a specific type of error within an error class

3.2.54
event
an instance of a change of conditions

3.2.55
extended channel related diagnosis
information concerning a specific element of a specific application object, provided for maintenance purposes

EXAMPLE Link Fail.

3.2.56
frame
denigrated term for DLPDU, unit of data transfer on a link

3.2.57
identification data object
object(s) that contain information about device, module and sub-module manufacturer and type referenced by device/slot/subslot/index

3.2.58
implicit AR endpoint
AR endpoint that is defined locally within a device without use of the create service

3.2.59
index
address of a record data object within an application process

3.2.60
instance
the actual physical occurrence of an object within a class that identifies one of many objects within the same object class

3.2.61
instance attributes
attribute that is unique to an object instance and not shared by the object class

3.2.62
instantiated
object that has been created in a device

3.2.63
interface
collection of FAL class attributes and services that represents a specific view on the FAL class

3.2.64
interface definition language
syntax and semantic of describing service parameters in a formal way

NOTE This description is the input for the ORPC model, especially for the ORPC wire protocol.

3.2.65
interface pointer
key attribute that unambiguously addresses an object interface instance

3.2.66
invocation
act of using a service or other resource of an application process

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

3.2.67
IO controller
controlling device, which acts as client for several IO devices (field devices)

NOTE This is usually a programmable controller or a distributed control system.

3.2.68
IO data object
object designated to be transferred cyclically for the purpose of processing and referenced by device/slot/subslot

3.2.69
IO device
field device which acts as server for IO operation

3.2.70
IO parameter server
server for application parameter of IO devices (client)

NOTE This is usually a device to backup parameter data and to log online changes of device parameter.

3.2.71
IO subsystem
subsystem composed of one IO controller and all its associated IO devices

3.2.72
IO supervisor
engineering device which manages commissioning and diagnosis of an IO system

3.2.73
IO system
system composed of all its IO subsystems

NOTE As an example a PLC with more than one IO controller (network interface) controls one IO system composed of an IO subsystems for each IO controller.

3.2.74
Isochronous mode
IO system operating tightly synchronized with a jitter of less than 1 µs

3.2.75
logical device
a certain FAL class that abstracts a software component or a firmware component as an autonomous self-contained facility of an automation device

3.2.76
member
piece of an attribute that is structured as an element of an array

3.2.77
method
<object> a synonym for an operational service which is provided by the server ASE and invoked by a client

3.2.78
module
hardware or logical component of a physical device

3.2.79
network
a set of nodes connected by some type of communication medium, including any intervening repeaters, bridges, routers and lower-layer gateways

3.2.80
object
abstract representation of a particular component within a device, usually a collection of related data (in the form of variables) and methods (procedures) for operating on that data that have clearly defined interface and behavior

3.2.81
object remote procedure call
model for object oriented or component based remote method invocation

3.2.82
object specific service
service unique to the object class which defines it

3.2.83
operate
status of the IO controller that indicates that the control algorithm is currently running

3.2.84
peer
role of an AR endpoint in which it is capable of acting as both client and server

3.2.85
physical device
a certain FAL class that abstracts the hardware facilities of an automation device

3.2.86
point-to-point connection
connection that exists between exactly two application objects

3.2.87
primary
status of the IO AR that indicates that it is in the operating state

NOTE Besides a primary IO AR a backup IO AR may exist. In example used for redundancy and dynamic reconfiguration of IO data.

3.2.88
property
a synonym for ASE attributes which are readable or writeable via operational ASE services

NOTE These services are generally named “get_<Attribute Name>” or “set_<Attribute Name>” and correspond with the IDL keywords “propget” and “propput”.

3.2.89
provider
a) source of a data connection

b) node or source sending data to one or many consumer

3.2.90
providerID
an unambiguous identifier within the scope of the ACCO assigned by the provider to recognize the internal data of a configured connection source

3.2.91
PTCP domain
a certain number of PTCP subdomains in one IP subnet

3.2.92
PTCP subdomain
a certain amount of DTEs with synchronized clocks

3.2.93
quality code
additional status information of a data item

3.2.94
quality code aware
attribute of the RT-Auto class that indicates that an RT-Auto object uses a status code for its data items

3.2.95
quality code unaware
opposite of quality code aware

3.2.96
record data object
object(s) which are already pre-processed and transferred acyclically for the purpose of information or further processing and referenced by device/slot/subslot/index

3.2.97
resource
processing or information capability

3.2.98
RT-Auto
an FAL class that abstracts the automation function as a process-related component of an automation device

3.2.99
run
status of the IO controller which indicates that the control algorithm is currently operating

3.2.100
runtime object model
objects that exist in a device together with their interfaces and methods that are accessible

3.2.101
server

a) role of an AREP in which it returns a confirmed service response APDU to the client that initiated the request

b) object which provides services to another (client) object

3.2.102
service
operation or function than an object and/or object class performs upon request from another object and/or object class

3.2.103
slot
address of a structural unit within an IO device

NOTE Within a modular device, a slot typically addresses a physical module. Within compact devices, a slot typically addresses a logical function or virtual module.

3.2.104
stop
status of the IO controller which indicates that the control algorithm is currently not running

3.2.105
submodule
hardware or logical component of a module

3.2.106
subslot
address of a structural unit within a slot

NOTE A subslot may address a physical interface for submodules within a module. Generally, a subslot is a second level to structure data within a device.

3.2.107
vendor ID
central administrative number used as manufacturer identification

NOTE The vendor ID is assigned by PROFIBUS International (PI).

3.3 .Abbreviations and symbols

ACCO	Active connection control object
IDL	Interface definition language
IP	Internet protocol
DNS	Domain name service
LDev	Logical device
ORPC	Object remote procedure call
PDev	Physical device
QoS	Quality of service
QC	Quality code
RT	Runtime
RT-Auto	Runtime automation object
TCP	Transmission control protocol

3.4 Additional abbreviations and symbols for decentralized periphery

AE	Application entity
AL	Application layer
ALME	Application layer management entity
ALP	Application layer protocol
ALPMI	Alarm protocol machine initiator
ALPMR	Alarm protocol machine responder
AP	Application process
APDU	Application protocol data unit
API	Application process identifier
APO	Application object

AR	Application relationship
AREP	Application relationship endpoint
ARP	Address resolution protocol
ASCII	American Standard Code for Information Interchange
ASE	Application service element
BMC	Best master clock
CM	Context management
Cnf	Confirmation
CR	Communication relationship
CREP	Communication relationship endpoint
DCE	OSF distributed computing environment
DCP	Discovery and basic configuration protocol
DCPMCR	DCP multicast receiver
DCPMCS	DCP multicast sender
DCPUCR	DCP unicast receiver
DCPUCS	DCP unicast sender
DHCP	Dynamic host configuration protocol
DIM	Device interface module
DL-	(as a prefix) data-link-
DLC	Data-link connection
DLL	Data-link layer
DLPDU	Data-link protocol data unit
DLSDU	DL-service data unit
DNS	Domain name service
DTE	Date terminal equipment
FAL	Fieldbus application layer
FIFO	First-in first-out
GSDML	Generic station description markup language
I&M	Identification and maintenance profile
IANA	Internet Assigned Numbers Authority
ICMP	Internet control message protocol
ID	Identifier
IEC	International Electrotechnical Commission
IFW	RT_CLASS_3 forwarding protocol machine
Ind	Indication
IOCS	Input output object consumer status
IOPS	Input output object provider status
IP	Internet protocol
IR	Isochronous relay
IRT	Isochronous real time protocol
ISO	International Organization for Standardization
IsoM	Isochronous mode
LED	Light emitting diode
LLDP	Link layer discovery protocol
LME	Layer management entity
lsb	Least significant bit
LT	Length/type
MAC	Medium access control
msb	Most significant bit
NCA	Network computing architecture
OSI	Open Systems Interconnect
PDU	Protocol data unit
PI	PROFIBUS International see www.profibus.com
PL	Physical layer
PTCP	Precision transparent clock protocol
QoS	Quality of service
Req	Request
RPC	Remote procedure call
Rsp	Response
RT	Real time protocol
RTA	Real time protocol acyclic
RTC	Real time protocol cyclic

RTE	Real time Ethernet
SDU	Service Data Unit
TLV	Type length value (coding rule)
UDP	User datagram protocol
UUID	Universal unique identifier
VLAN	Virtual local area network

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 61158-5-18. 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 attributes are accessible from instances of the class using the Management ASE services specified in IEC 61158-5-18 standard. The service specification defines the services that are provided by the ASE.

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

4 Abstract syntax

4.1 M1 device manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 1.

Table 1 – M1 device manager attribute format

Attribute	Format	Size (bits)
Management information	Structure of 6 elements:	9 octets
Transmission speed	Unsigned8	8
Number of occupied stations	Unsigned8	8
Station number	Unsigned8	8
Vendor code	Unsigned16	16
Model code	3 octets, bit mapped	24
Software version	Unsigned8	8
Connected Slave management information	Array of 64 members:	640 octets
Slave information 1 : 64	Structure of 5 elements:	10 octets
Station number	Unsigned8	8
Vendor code	Unsigned16	16
Model code	3 octets, bit mapped	24
Software version	Unsigned8	8
Reserved field	3 octets	24

4.2 M2 device manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 2.

Table 2 – M2 device manager attribute format

Attribute	Format	Size (bits)
Slave station information	Array of 64 members:	128 octets
Slave station information 1 : 64	Word	16
Slave station status information	Array of 64 members:	32 octets
Slave station status information 1 : 64	4 Bits	4

4.3 S1 device manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 3.

Table 3 – S1 device manager attribute format

Attribute	Format	Size (bits)
Management information	Structure of 5 elements:	10 octets
Station number	Unsigned8	8
Vendor code	Unsigned16	16
Model code	3 octets, bit mapped	24
Software version	Unsigned8	8
Reserved field	3 octets	24

4.4 S2 device manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 4.

Table 4 – S2 device manager attribute format

Attribute	Format	Size (bits)
Slave station information	Word	16
Slave station status information	4 Bits	4

4.5 M1 connection manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 5.

Table 5 – M1 connection manager attribute format

Attribute	Format	Size (bits)
Parameter information	Structure of 14 elements:	452 octets
Number of connected modules	Unsigned16	16
Number of intelligent devices	Unsigned16	16
Station information	Array of 64 members:	128 octets
Station information 1 : 64	Word	16
Number of automatic return modules	Unsigned16	16
Number of retries	Unsigned16	16
Delay time setting	Unsigned16	16
Standby Master station specification	Unsigned16	16
Operation during Master error state	Unsigned16	16
Data-link during Master error state	Unsigned16	16
Scan mode specification	Unsigned16	16
Reserved station specification	Array of 8 members:	8 octets
Reserved station number	Unsigned8	8
Error invalid station specification	Array of 8 members:	8 octets
Invalid station number	Unsigned8	8
Reserved 1	4 Words	8 octets
Reserved 2	78 Words	156 octets
Network status information	Structure of 4 elements:	260 octets
Master status information	Word	16
Slave status information	Array of 64 members:	128 octets
Slave status information 1 : 64	Word	16
Master transmitted status field	Word	16
Slave transmitted status field	Array of 64 members:	128 octets
Slave transmitted status field 1 : 64	Word	16
Network information	Structure of 3 elements:	6 octets
Current link scan time	Unsigned16	16
Minimum link scan time	Unsigned16	16
Maximum link scan time	Unsigned16	16

4.6 M2 connection manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 6.

Table 6 – M2 connection manager attribute format

Attribute	Format	Size (bits)
Parameter information	Structure of 4 elements:	15
Transmission speed	2 Bits	2
Last station number	Unsigned8	8
Point mode setting	2 Bits	2
Master station i/o point mode setting	3 Bits	3
Network status information	Structure of 2 elements:	48 octets
Reception status information	Structure of 64 elements:	32 octets
Reception status 1 : 64	4 Bits	4
Slave status information	Structure of 64 elements:	16 octets
Slave status 1 : 64	2 Bits	2

4.7 S1 connection manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 7.

Table 7 – S1 connection manager attribute format

Attribute	Format	Size (bits)
Process data support level	2 Bits	2
Network status information	Structure of 1 element:	16
Master transmitted status field	Word	16

4.8 S2 connection manager PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 8.

Table 8 – S2 connection manager attribute format

Attribute	Format	Size (bits)
Parameter information	Structure of 2 elements:	11
Slave station number	Unsigned8	8
Number of occupied slots	3 Bits	3

4.9 M1 cyclic transmission PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 9.

Table 9 – M1 cyclic transmission attribute format

Attribute	Format	Size (bits)
Master status	Word	16
Data out	Structure of 2 elements:	x + y [1]
RY data	Bit-oriented data structure	x [1]
RWw data	Word-oriented data structure	y [1]
Data in	Structure of 2 elements	z + 16 [1]
Number of Modules	Unsigned16	16
Slave input data	Array of up to 64 members:	z [1]
Station number	Unsigned16	16
Slave status	Word	16
RX data	Bit-oriented data structure	x [1]
RWr data	Word-oriented data structure	y [1]
[1] The values of x, y and z are dependent upon the values of the corresponding configuration settings in the Master status.		

4.10 M2 cyclic transmission PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 10.

Table 10 – M2 cyclic transmission attribute format

Attribute	Format	Size (bits)
RY data	Bit-oriented data structure	64 n [1]
RX data	Bit-oriented data structure	64 n [1]
[1] n = the number of points specified by the point-mode-setting value of the M2 connection manager.		

4.11 S1 cyclic transmission PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 11.

Table 11 – S1 cyclic transmission attribute format

Attribute	Format	Size (bits)
Slave status	Word	16
Data out	Structure of 2 elements:	x + y [1]
RY data	Bit-oriented data structure	x [1]
RWw data	Word-oriented data structure	y [1]
Master status	Word	16
Data in	Structure of 2 elements:	x + y [1]
RX data	Bit-oriented data structure	x [1]
RWr data	Word-oriented data structure	y [1]
[1] The values of x and y are dependent upon the values of the corresponding configuration settings in the Master status.		

4.12 S2 cyclic transmission PDU abstract syntax

The abstract syntax for attributes belonging to this class are described in Table 12.

Table 12 – S2 cyclic transmission attribute format

Attribute	Format	Size (bits)
RY data	Bit-oriented data structure	n [1]
RX data	Bit-oriented data structure	n [1]
[1] n = the number of points specified by the point-mode-setting value of the corresponding M2 connection manager times the number of occupied Slots as specified by the S2 connection manager.		

5 Transfer syntax

5.1 M1 device manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 13.

Table 13 – M1 device manager attribute encoding

<table>
<tr><th>Attribute</th><th colspan="4">Encoding</th></tr>
<tr><td>Management information</td><td colspan="4">Specifies the configuration of the Master device</td></tr>
<tr><td>Transmission speed</td><td colspan="4">Enumerated list of baud rate values in kbit/s

0 = 156
1 = 625
2 = 2 500
3 = 5 000
4 = 10 000
5 : 255 = reserved</td></tr>
<tr><td>Number of occupied stations</td><td colspan="4">1 : 4 = allowable range</td></tr>
<tr><td>Station number</td><td colspan="4">0 = Master
1 : 127 = reserved
128 = Standby Master
129 : 255 = reserved</td></tr>
<tr><td>Vendor code</td><td colspan="4">The assignment and management of vendor codes is best handled by applicable trade organizations and is therefore beyond the scope of this specification</td></tr>
<tr><td>Model code</td><td colspan="4"><table>
<tr><th>Octet</th><th>Bit</th><th>Description</th><th>Value</th></tr>
<tr><td rowspan="4">0</td><td>1 – 0</td><td>Total number of used bit-oriented data bits (both RX and RY combined)</td><td>0 = full use
1 = 8 used
2 = 32 used
3 = 16 used</td></tr>
<tr><td>3 – 2</td><td>Distribution of used bit-oriented data bits</td><td>0 = RX and RY in equal sizes
1 = RX only
2 = RY only
3 = other RX / RY mix</td></tr>
<tr><td>5 – 4</td><td>Number of occupied station slots</td><td>0 = 1 slot
1 = 2 slots
2 = 3 slots
3 = 4 slots</td></tr>
<tr><td>7 – 6</td><td>reserved</td><td>(set = 0)</td></tr>
<tr><td rowspan="4">1</td><td>0</td><td>FAL-user switch setting</td><td>0 = normal
1 = abnormal</td></tr>
<tr><td>1</td><td>FAL-user output upon fault</td><td>0 = clear
1 = hold</td></tr>
<tr><td>5 – 2</td><td>reserved</td><td>(set = 0)</td></tr>
<tr><td>7 – 6</td><td>DLE support level</td><td>0 = level A
1 = level B
2 = level C
3 = reserved</td></tr>
<tr><td rowspan="2">2</td><td>6 – 0</td><td>reserved</td><td>(set = 0)</td></tr>
<tr><td>7</td><td>Acyclic data format supports messaging type commands.</td><td>0 = not supported
1 = supported</td></tr>
</table></td></tr>
<tr><td>Software version</td><td colspan="4">0 = undefined
1 : 26 = version A : Z
27 : 52 = version AA : AZ
53 : 255 = version BA : IU</td></tr>
</table>

<table>
<tr><th>Attribute</th><th colspan="4">Encoding</th></tr>
<tr><td>Connected Slave management information</td><td colspan="4">Specifies the configuration of the connected Slaves</td></tr>
<tr><td>Slave information 1 : 64</td><td colspan="4">Array of 64 elements, each encoded as:</td></tr>
<tr><td>Station number</td><td colspan="4">1 : 64 = allowable range</td></tr>
<tr><td>Vendor code</td><td colspan="4">The assignment and management of vendor codes is best handled by applicable trade organizations and is therefore beyond the scope of this specification</td></tr>
<tr><td rowspan="11">Model code</td><td>Octet</td><td>Bit</td><td>Description</td><td>Value</td></tr>
<tr><td rowspan="4">0</td><td>1 – 0</td><td>Total number of used bit-oriented data bits (both RX and RY combined)</td><td>0 = full use
1 = 8 used
2 = 32 used
3 = 16 used</td></tr>
<tr><td>3 – 2</td><td>Distribution of used bit-oriented data bits</td><td>0 = RX and RY in equal sizes
1 = RX only
2 = RY only
3 = other RX / RY mix</td></tr>
<tr><td>5 – 4</td><td>Number of occupied station slots</td><td>0 = 1 slot
1 = 2 slots
2 = 3 slots
3 = 4 slots</td></tr>
<tr><td>7 – 6</td><td>reserved</td><td>(set = 0)</td></tr>
<tr><td rowspan="4">1</td><td>0</td><td>FAL-user switch setting</td><td>0 = normal
1 = abnormal</td></tr>
<tr><td>1</td><td>FAL-user output upon fault</td><td>0 = clear
1 = hold</td></tr>
<tr><td>5 – 2</td><td>reserved</td><td>(set = 0)</td></tr>
<tr><td>7 – 6</td><td>DLE support level</td><td>0 = level A
1 = level B
2 = level C
3 = reserved</td></tr>
<tr><td rowspan="2">2</td><td>6 – 0</td><td>reserved</td><td>(set = 0)</td></tr>
<tr><td>7</td><td>Acyclic data format supports messaging type commands.</td><td>0 = not supported
1 = supported</td></tr>
<tr><td>Software version</td><td colspan="4">0 = undefined
1 : 26 = version A : Z
27 : 52 = version AA : AZ
53 : 255 = version BA : IU</td></tr>
<tr><td>Reserved field</td><td colspan="4">Reserved for future specification definition</td></tr>
</table>

5.2 M2 device manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 14.

Table 14 – M2 device manager attribute encoding

Attribute	Encoding	
Slave station information	Specifies the configuration of the connected Slaves	
Slave station information 1 : 64	Array of 64 Words, each encoded as:	
	Bit	**Definition**
	2 – 0	0 = 1 i/o point 1 = 2 i/o points 2 = 4 i/o points 3 = 8 i/o points 4 = 16 i/o points 5 = 12 i/o points 6 = reserved 7 = reserved
	3	output i/o type present (0 = false; 1 = true)
	4	input i/o type present (0 = false; 1 = true)
	5	device type: (0 = remote i/o station; 1 = remote device station)
	6	configured as a head station (for number of occupied DLE station slots > 1) (0 = false; 1 = true)
	7	input time constant (0 = normal; 1 = high speed)
	8	output state for abnormal operating states (0 = clear; 1 = hold)
	15 – 9	reserved
Slave station status information	The status fields transmitted from each connected Slave	
Slave station status information 1 : 64	Array of 64, 4-Bit fields, each encoded as:	
	Bit	**Definition**
	0	Slave status (0 = Normal; 1 = Error)
	1	Configuration data transmitted since connected (0 = false; 1 = true)
	2	Parity (provides even parity for status field)
	3	Reserved (set = 0)

5.3 S1 device manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 15.

Table 15 – S1 device manager attribute encoding

Attribute	Encoding			
Management information	Specifies the configuration of the Slave device			
Station number	1 : 64 = allowable range			
Vendor code	The assignment and management of vendor codes is FAL user specific and beyond the scope of this specification			
Model code	**Octet**	**Bit**	**Description**	**Value**
	0	1 – 0	Total number of used bit-oriented data bits (both RX and RY combined)	0 = full use 1 = 8 used 2 = 32 used 3 = 16 used
		3 – 2	Distribution of used bit-oriented data bits	0 = RX and RY in equal sizes 1 = RX only 2 = RY only 3 = other RX / RY mix
		5 – 4	Number of occupied station slots	0 = 1 slot 1 = 2 slots 2 = 3 slots 3 = 4 slots
		7 – 6	reserved	(set = 0)
	1	0	FAL-user switch setting	0 = normal 1 = abnormal
		1	FAL-user output upon fault	0 = clear 1 = hold
		5 – 2	reserved	(set = 0)
		7 – 6	DLE support level	0 = level A 1 = level B 2 = level C 3 = reserved
	2	6 – 0	reserved	(set = 0)
		7	Acyclic data format supports messaging type commands.	0 = not supported 1 = supported
Software version	0 = undefined 1 : 26 = version A : Z 27 : 52 = version AA : AZ 53 : 255 = version BA : IU			
Reserved field	Reserved for future specification definition			

5.4 S2 device manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 16.

Table 16 – S2 device manager attribute encoding

Attribute	Encoding	
Slave station information	**Bit**	**Definition**
	2 – 0	0 = 1 i/o point 1 = 2 i/o points 2 = 4 i/o points 3 = 8 i/o points 4 = 16 i/o points 5 = 12 i/o points 6 = reserved 7 = reserved
	3	output i/o type present (0 = false; 1 = true)
	4	input i/o type present (0 = false; 1 = true)
	5	device type: (0 = remote i/o station; 1 = remote device station)
	6	configured as a head station (for number of occupied DLE station slots > 1) (0 = false; 1 = true)
	7	input time constant (0 = normal; 1 = high speed)
	8	output state for abnormal operating states (0 = clear; 1 = hold)
	15 – 9	reserved
Slave station status information	**Bit**	**Definition**
	0	Slave status (0 = Normal; 1 = Error)
	1	Configuration data transmitted since connected (0 = false; 1 = true)
	2	Parity (provides even parity for status field)
	3	Reserved (set = 0)

5.5 M1 connection manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 17.

Table 17 – M1 connection manager attribute encoding

Attribute	Encoding		
Parameter information	Specifies the connection configuration		
Number of connected modules	1 : 64 = allowable range		
Number of intelligent devices	Specifies the number of Slave devices that support process data support level C 0 : 26 = allowable range		
Station information	Indicates the connection configurations of the connected Slave devices.		
Station information 1 : 64	**Bit**	**Description**	**Value**
	7 – 0	Station number	1 : 64 = allowable range
	11 – 8	Number of occupied station slots	1 : 4 = allowable range
	15 – 12	Station type in terms of process data support level	0 = Level A 1 = Level B 2 = Level C 3 : 15 = reserved
Number of automatic return modules	Specifies the number Slave devices that are able to resume connections. 1 : 10 = allowable range		
Number of retries	Specifies the number of times the M1 connection manager object will retry communications scans with consecutive DL errors. 1 : 7 = allowable range		
Delay time setting	Specifies the data-link Scan interval in µs. 0 : 5 000 = allowable range		
Standby Master station specification	0 : 64 = allowable range 0 = no standby Master		
Operation during Master error state	Specifies M1 connection manager object control over the DL during an FAL error state. 0 = Stop Scan 1 = Continue Scan 2 : 65535 = reserved		
Data-link during Master error state	Specifies the state for process data during an FAL error state. 0 = Hold data states 1 = Clear data states 2 : 65535 = reserved		
Scan mode specification	Specifies the scanning behavior. 0 = free-running (synchronous) 1 = triggered (asynchronous) 2 : 65535 = reserved		
Reserved station specification	Specifies the reserved Slave station numbers. Although reserved stations are counted as connected stations, a data-link error will not occur for reserved stations not connected.		
Reserved station number	1 : 64 = allowable range		
Error invalid station specification	Specifies the error invalid Slave station numbers. Slave station errors (at the Master) will not occur for error invalid Slave stations.		
Invalid station number	1 : 64 = allowable range		
Network status information	Master reception status		

Attribute	Encoding		
Master status information	**Bit**	**Description**	**Values**
	0	Master frame interval timeout occurred	0 = False 1 = True
	1	Refresh frame reception timeout occurred	0 = False 1 = True
	2	Consecutive transmission timeout occurred	0 = False 1 = True
	3	Consecutive reception timeout occurred	0 = False 1 = True
	4	Transmission route monitor 1 error occurred	0 = False 1 = True
	5	Transmission route monitor 2 error occurred	0 = False 1 = True
	6	Standby Master activation requested	0 = False 1 = True
	15 – 7	reserved	
Slave status information	Scanning results		
Slave status info 1 : 64	**Bit**	**Description**	**Values**
	0	Polling status error occurred	0 = False 1 = True
	1	CRC error occurred	0 = False 1 = True
	2	An abort frame was reported by the DL	0 = False 1 = True
	3	A polling response timeout occurred	0 = False 1 = True
	4	A received data buffer overflow occurred	0 = False 1 = True
	5	An invalid reception address filed was received	0 = False 1 = True
	6	At least one retry has occurred	0 = False 1 = True
	15 – 7	reserved	
Master transmitted status field	**Bit**	**Description**	**Values**
	0	FAL-user state	0 = Stop 1 = Run
	1	FAL-user status	0 = Normal 1 = Fault
	2	Cyclic refresh status	0 = Stop 1 = Run
	3	Acyclic status	0 = Normal 1 = Error
	4	Acyclic enabled	0 = Disable 1 = Enable
	6 – 5	reserved	
	7	Master type	0 = Active 1 = Standby

Attribute	Encoding		
	11 – 8	Size of bit-oriented data fields: RX, RY in Slots. A Slot is 4 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved
	15 – 12	Size of word-oriented data fields: RWr, RWw in Slots. A Slot is 8 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved
Slave transmitted status field	Transmitted status fields received from each Slave		
Slave status field 1 : 64	**Bit**	**Description**	**Values**
	0	FAL-user fuse status	0 = Normal 1 = Abnormal
	1	FAL-user status	0 = Normal 1 = Fault
	2	Cyclic refresh status	0 = Complete 1 = Not received
	3	Slave parameter receive status	0 = Complete 1 = Not received
	4	Slave's FAL-user switch status	0 = No change 1 = Changed
	5	Cyclic transmission enabled	0 = Enable 1 = Disable
	6	reserved	
	7	FAL-user watchdog timer error detected	0 = Normal 1 = Error
	8	Acyclic status	0 = Normal 1 = Error
	9	Acyclic enabled	0 = Disabled 1 = Enabled
	10	Acyclic type: Master/Slave or Peer/Peer	0 = M/S 1 = P/P
	11	reserved	
	12	Transmission status	0 = Normal 1 = Fault
	13	reserved	1
	15 – 14	Cyclic segmenting configuration	0 = 1x or not supported 1 = 2x 3 = 4x 4 = 8x
Network information	Real time performance parameters		
Current link scan time	ms		
Minimum link scan time	ms		
Maximum link scan time	ms		

5.6 M2 connection manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 18.

Table 18 – M2 connection manager attribute encoding

<table>
<tr><th>Attribute</th><th colspan="3">Encoding</th></tr>
<tr><td>Parameter information</td><td colspan="3"></td></tr>
<tr><td>Transmission speed</td><td colspan="3">0 = 156 (baud rate in kbit/s)
1 = 625
2 = 2 500
3 = reserved</td></tr>
<tr><td>Last station number</td><td colspan="3">1 : 64 = allowable range</td></tr>
<tr><td>Point mode setting</td><td colspan="3">0 = undefined
1 = 4 (points)
2 = 8
3 = 16</td></tr>
<tr><td>Master station i/o point mode setting</td><td colspan="3">0 = 16 (total points controlled by Master)
1 = 32
2 = 48
3 = 64
4 = 128
5 = 256
6 = 512
7 = 1 24</td></tr>
<tr><td>Network status information</td><td colspan="3">Connection status</td></tr>
<tr><td>Reception status information</td><td colspan="3">Reception and monitoring timer status</td></tr>
<tr><td rowspan="5">Reception status 1 : 64</td><td>Bit</td><td>Description</td><td>Values</td></tr>
<tr><td>0</td><td>CRC error occurred</td><td>0 = Normal
1 = Error</td></tr>
<tr><td>1</td><td>Parity error occurred</td><td>0 = Normal
1 = Error</td></tr>
<tr><td>2</td><td>No response error occurred</td><td>0 = Normal
1 = Error</td></tr>
<tr><td>3</td><td>reserved</td><td></td></tr>
<tr><td>Slave status information</td><td colspan="3">Slaves connection status</td></tr>
<tr><td rowspan="3">Slave status 1 : 64</td><td>Bit</td><td>Description</td><td>Values</td></tr>
<tr><td>0</td><td>Slave station connection status</td><td>0 = Not connected
1 = Connected</td></tr>
<tr><td>1</td><td>31 or more consecutively received errors including: CRC, parity or timeout</td><td>0 = Normal
1 = Fault</td></tr>
</table>

5.7 S1 connection manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 19.

Table 19 – S1 connection manager attribute encoding

Attribute	Encoding		
Process data support level	0 = Level A 1 = Level B 2 = Level C 3 : 15 = reserved		
Network status information	Status field transmitted by the connected Master.		
Master transmitted status field	**Bit**	**Description**	**Values**
	0	FAL-user state	0 = Stop 1 = Run
	1	FAL-user status	0 = Normal 1 = Fault
	2	Cyclic refresh status	0 = Stop 1 = Run
	3	Acyclic status	0 = Normal 1 = Error
	4	Acyclic enabled	0 = Disable 1 = Enable
	6 – 5	reserved	
	7	Master type	0 = Active 1 = Standby
	11 – 8	Size of bit-oriented data fields: RX, RY in Slots. A Slot is 4 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved
	15 – 12	Size of word-oriented data fields: RWr, RWw in Slots. A Slot is 8 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved

5.8 S2 connection manager PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 20.

Table 20 – S2 connection manager attribute encoding

Attribute	Encoding
Parameter information	Specifies the connection configuration of the Slave
Slave station number	1 : 64 = range of allowable values
Number of occupied slots	3 bits encoded as an unsigned integer 1 : 4 = range of allowable values

5.9 M1 cyclic transmission PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 21.

Table 21 – M1 cyclic transmission attribute encoding

Attribute	Encoding		
Master status	**Bit**	**Description**	**Values**
	0	FAL-user state	0 = Stop 1 = Run
	1	FAL-user status	0 = Normal 1 = Fault
	2	Cyclic refresh status	0 = Stop 1 = Run
	3	Acyclic status	0 = Normal 1 = Error
	4	Acyclic enabled	0 = Disable 1 = Enable
	6 – 5	reserved	
	7	Master type	0 = Active 1 = Standby
	11 – 8	Size of bit-oriented data fields: RX, RY in Slots. A Slot is 4 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved
	15 – 12	Size of word-oriented data fields: RWr, RWw in Slots. A Slot is 8 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved
Data out	Process data registers set by the Master for Slave device output.		
RY data	A position mapped field of bit-oriented output data for all connected Slave devices ordered by Slot with 32 bits per Slot.		
RWw data	A position mapped field of word-oriented output data for all connected Slave devices ordered by Slot with 4 words per Slot.		
Data in	Process data registers read by the Master representing Slave device inputs.		
Number of Modules	1 : 64 = range of allowable values		
Slave input data			
Station number	1 : 64 = range of allowable values		

Attribute	Encoding		
Slave status	**Bit**	**Description**	**Values**
	0	FAL-user fuse status	0 = Normal 1 = Abnormal
	1	FAL-user status	0 = Normal 1 = Fault
	2	Cyclic refresh status	0 = Complete 1 = Not received
	3	Slave parameter receive status	0 = Complete 1 = Not received
	4	Slave's FAL-user switch status	0 = No change 1 = Changed
	5	Cyclic transmission enabled	0 = Enable 1 = Disable
	6	reserved	
	7	FAL-user watchdog timer error detected	0 = Normal 1 = Error
	8	Acyclic status	0 = Normal 1 = Error
	9	Acyclic enabled	0 = Disabled 1 = Enabled
	10	Acyclic type: Master/Slave or Peer/Peer	0 = M/S 1 = P/P
	11	reserved	
	12	Transmission status	0 = Normal 1 = Fault
	13	reserved	1
	15 – 14	Cyclic segmenting configuration	0 = 1x or not supported 1 = 2x 3 = 4x 4 = 8x
RX data	A field containing the bit-oriented input data from Slave device n ordered by Slot with 32 bits per Slot. The number of Slots occupied by the Slave device determines the total length of this field.		
RWr data	A field containing the word-oriented input data from Slave device n ordered by Slot with 4 words per Slot. The number of Slots occupied by the Slave device determines the total length of this field.		

5.10 M2 cyclic transmission PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 22.

Table 22 – M2 cyclic transmission attribute encoding

Attribute	Encoding
RY data	A field containing the bit-oriented output data for the Slave devices ordered by Slot with n bits per Slot. Where n = 4, 8 or 16 based upon the point-mode-setting value of the M2 connection manager.
RX data	A field containing the bit-oriented input data from the Slave devices ordered by Slot with n bits per Slot. Where n = 4, 8 or 16 based upon the point-mode-setting value of the M2 connection manager.

5.11 S1 cyclic transmission PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 23.

Table 23 – S1 cyclic transmission attribute encoding

Attribute	Encoding		
Slave status	**Bit**	**Description**	**Values**
	0	FAL-user fuse status	0 = Normal 1 = Abnormal
	1	FAL-user status	0 = Normal 1 = Fault
	2	Cyclic refresh status	0 = Complete 1 = Not received
	3	Slave parameter receive status	0 = Complete 1 = Not received
	4	Slave's FAL-user switch status	0 = No change 1 = Changed
	5	Cyclic transmission enabled	0 = Enable 1 = Disable
	6	reserved	
	7	FAL-user watchdog timer error detected	0 = Normal 1 = Error
	8	Acyclic status	0 = Normal 1 = Error
	9	Acyclic enabled	0 = Disabled 1 = Enabled
	10	Acyclic type: Master/Slave or Peer/Peer	0 = M/S 1 = P/P
	11	reserved	
	12	Transmission status	0 = Normal 1 = Fault
	13	reserved	1
	15 – 14	Cyclic segmenting configuration	0 = 1x or not supported 1 = 2x 3 = 4x 4 = 8x
Data out	The process data received from the Master		
RY data	A field containing the bit-oriented input data ordered by Slot with 32 bits per Slot. The number of Slots occupied by the Slave device determines the total length of this field.		
RWw data	A field containing the word-oriented input data ordered by Slot with 4 words per Slot. The number of Slots occupied by the Slave device determines the total length of this field.		
Master status	**Bit**	**Description**	**Values**
	0	FAL-user state	0 = Stop 1 = Run
	1	FAL-user status	0 = Normal 1 = Fault
	2	Cyclic refresh status	0 = Stop 1 = Run
	3	Acyclic status	0 = Normal 1 = Error
	4	Acyclic enabled	0 = Disable 1 = Enable
	6 – 5	reserved	

Attribute	Encoding		
	7	Master type	0 = Active 1 = Standby
	11 – 8	Size of bit-oriented data fields: RX, RY in Slots. A Slot is 4 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved
	15 – 12	Size of word-oriented data fields: RWr, RWw in Slots. A Slot is 8 octets in length	0 = 0 1 = 8 2 = 16 3 = 24 4 = 32 5 = 40 6 = 48 7 = 56 8 = 64 9 : 15 = reserved
Data in	The process data transmitted to the Master		
RX data	A field containing the bit-oriented input data ordered by Slot with 32 bits per Slot. The number of Slots occupied by the Slave device determines the total length of this field.		
RWr data	A field containing the word-oriented input data ordered by Slot with 4 words per Slot. The number of Slots occupied by the Slave device determines the total length of this field.		

5.12 S2 cyclic transmission PDU encoding

The specific PDU encoding for attributes belonging to this class are described in Table 24.

Table 24 – S2 cyclic transmission attribute encoding

Attribute	Encoding
RY data	A field containing the bit-oriented output data from the Master device ordered by Slot with n bits per Slot. Where n = 4, 8 or 16 based upon the point-mode-setting value of the associated M2 connection manager. The total length of this field is determined by n times the number Slots occupied by the Slave device as specified by the S2 connection manager object.
RX data	A field containing the bit-oriented input data for the Master device ordered by Slot with n bits per Slot. Where n = 4, 8 or 16 based upon the point-mode-setting value of the associated M2 connection manager. The total length of this field is determined by n times the number Slots occupied by the Slave device as specified by the S2 connection manager object.

5.13 Acyclic transmission PDU encoding

5.13.1 Acyclic message encoding

The specific PDU encoding for acyclic messages is specified in Table 25.

Table 25 – Acyclic transmission – message data encoding

Field	Size (octets)	Value
Length	2	Length of message (in octets) not including the first 4 octets. (i.e., set to a value four less than the total length of the message)
reserved	1	for future use (set = 0)
Type and sequence	1	bits 3 – 0 = type (set = 0) M2 type: bits 7 – 4 = sequence number in the range 1-7. FAL user increments this field by 1 upon each successive request, rolling back to 1 after 7. S2 type: bits 6 – 4 = reserved field – used by DL-protocol for segmenting and reassembly Bit 7 = sequence flag, alternating 0 and 1. FAL user toggles this bit on each successive request
Segment number	1	reserved field – used by DL-protocol for segmenting and reassembly
Data type	1	b7 = priority (0 = low; 1 = high) b6 = response required (0 = true; 1 = false) b5 – b0 = reserved
Destination address	1	Station number of the intended destination AREP
Source address	1	Station number of the originating device.
Destination app type	1	FAL user target application type (set = 33)
Source app type	1	FAL user source application type (set = 33)
Destination app module	1	FAL-user target application handler module (0 = network, 1-255 = FAL user specific)
Source app module	1	DLS-user source application handler module (0 = network, 1-255 = FAL user specific)
Destination network id	1	for future extension (set = 0)
Destination address	1	for future extension (set = 0)
Destination id	2	bits 9 – 0 = fixed (set = 1023) bits 15 – 10 = Station number of the intended destination AREP
Source network id	1	for future extension (set = 0)
Source address	1	for future extension (set = 0)
Source id	2	bits 9 – 0 = fixed (set = 1 23) bits 15 – 10 = Station number of the originating device
Command header	8	Acyclic command header as specified in Table 26
Command parameters	0 – 960	Parameter field for the command as specified in 5.13.2

Table 26 – Command header format

Field	Size (octets)	Value
Length	2	length of command plus command data not including this field. (i.e., set to a value six more than the length of command data)
Command type	1	command code as specified in Table 27
reserved	1	(set = 0)
SAP	2	FAL user specific Service Access Point identifier
Response code	2	bits 7 – 0 = error fields bit 8 = priority (0 = Warning; 1 = Major error) bits 11 – 9 = error location (set by DLS-user) bits 15 – 12 = FAL user specific field

Table 27 – Command codes

Code	Command
0	undefined
1	Parameter block 1
2	Parameter block 2
3	System information acquisition
4	Memory access information acquisition
5 – 7	reserved
8	Run
9	Stop
10 – 14	reserved
15	Line test
16	Memory read
17	reserved
18	Memory write
19 – 31	reserved
32	Special command reserved for compliance with the semiconductor manufacturing industry the specifics of which are beyond the scope of this specification
33 – 95	reserved
96 – 127	open area for FAL user definition

5.13.2 Acyclic command parameter encoding

5.13.2.1 Parameter block 1

The encoding of the request parameter field for this command is specified in Figure 1.

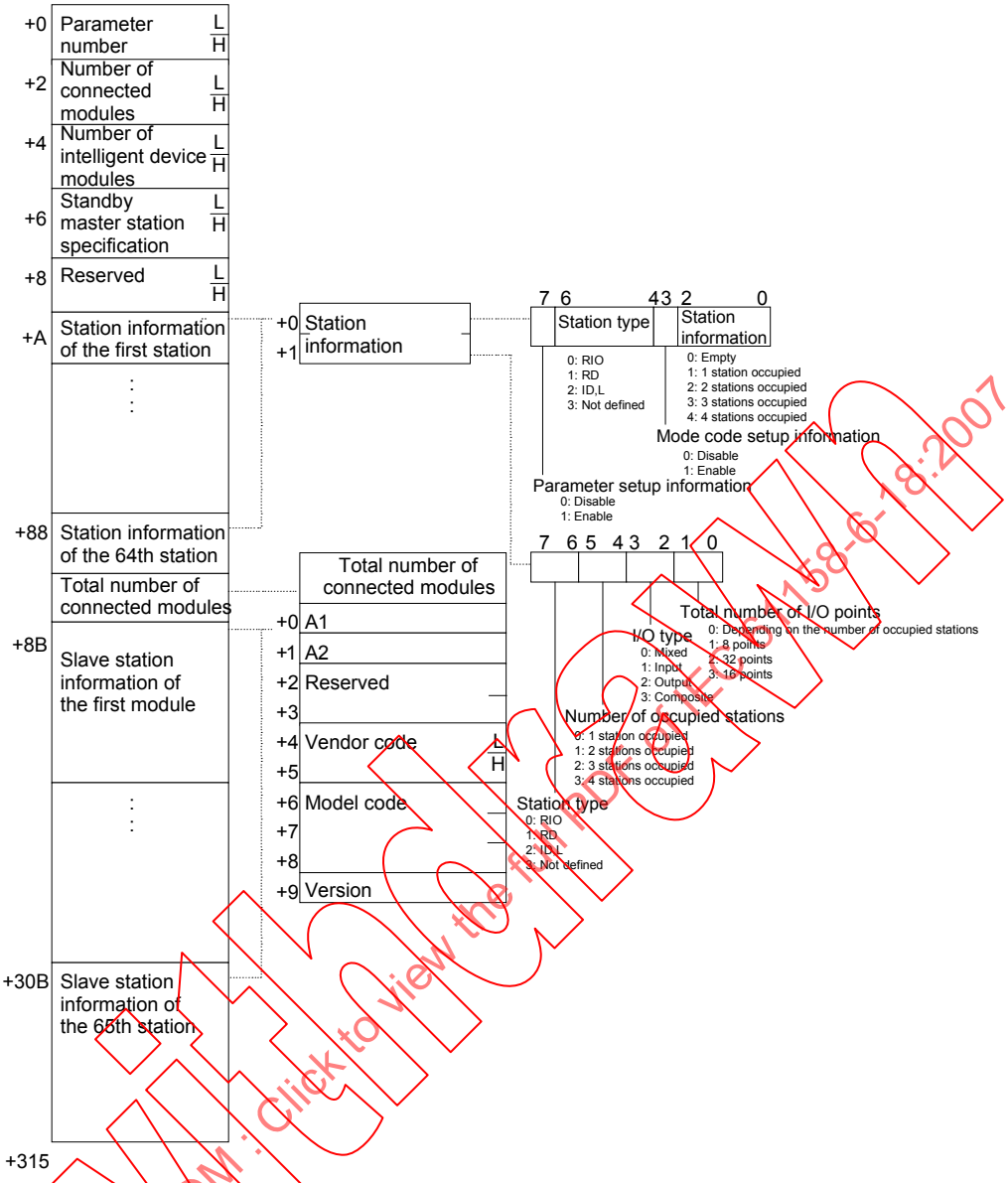

Figure 1 – Parameter block 1 command parameter field

5.13.2.2 Parameter block 2

The encoding of the request parameter field for this command is specified in Figure 2.

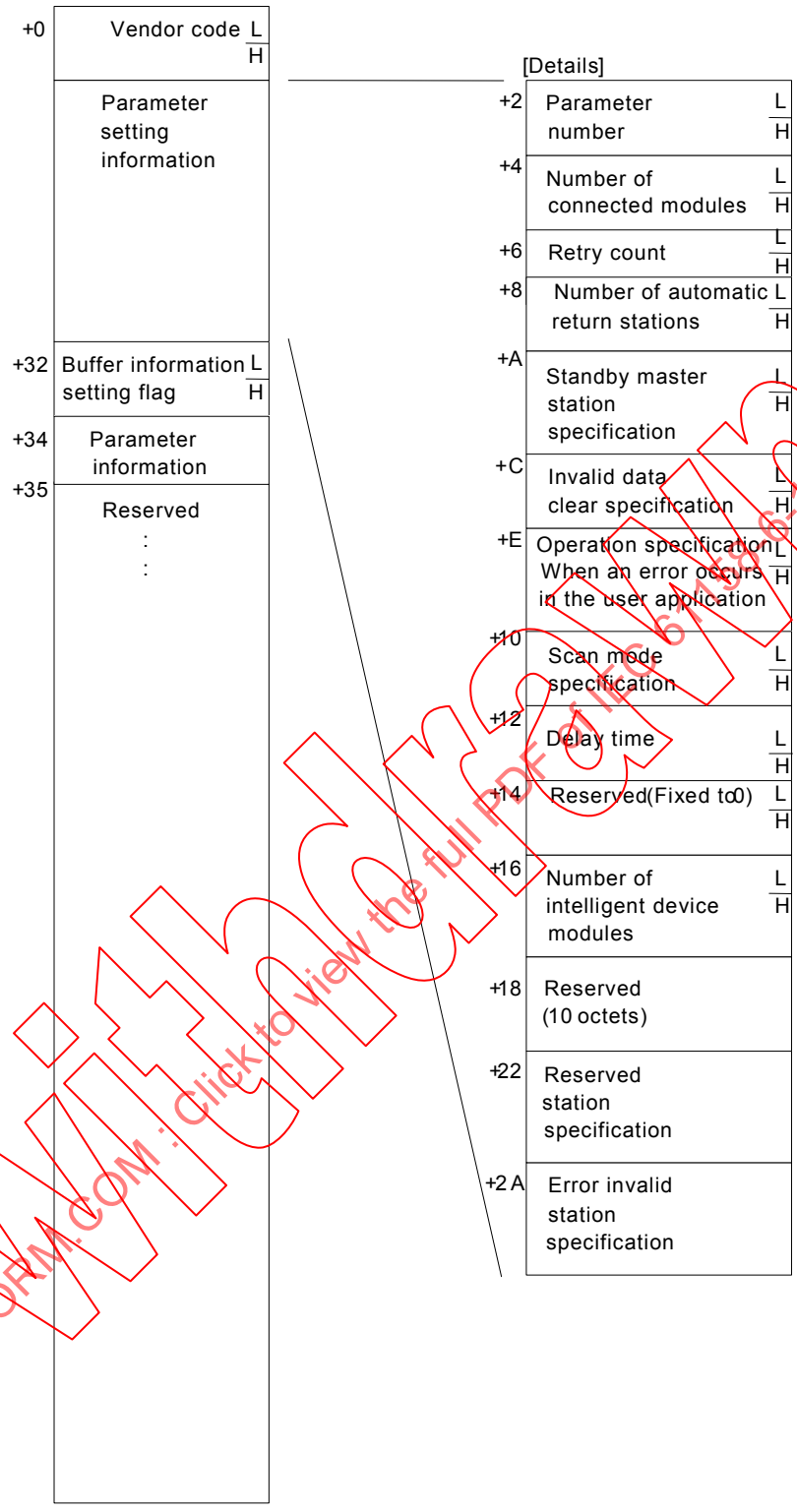

Figure 2 – Parameter block 2 command parameter field

5.13.2.3 System information acquisition

The encoding of the response parameter field for this command is specified in Table 28.

Table 28 – System information command parameter field

Octet	Field	Encoding
0 – 1	Vendor code	Field specified in M2 device manager
2 – 5	Model code	Field specified in M2 device manager
6 – 7	Software version	Field specified in M2 device manager
8 – 23	Supported commands	128 bit, bit-mapped field where the bit location corresponds to a command number (0: 127). The values contained in the bit indicates if the command is supported. 0 = not supported 1 = supported
24	Segmenting limit	Specifies the maximum number of segments supported for acyclic messaging 1 : 7 = range of allowable values
25 – 26	Data buffer limit	Specifies the maximum length for acyclic messaging 0 : 960 = range of allowable values

5.13.2.4 Memory access information acquisition

The encoding of the response parameter field for this command is specified in Table 29.

Table 29 – System information command parameter field

Octet	Field	Encoding
0 – 31	List of available access codes	256 bit, bit-mapped field where the bit location corresponds to an access code (0: 255). The values contained in the bit indicates if the access code is supported. 0 = supported 1 = not supported
32 – 35	Device name	Coded value to represent the device type. Specific value are FAL user specific and beyond the scope of this specification.
36 – 37	Number of access points	Specifies the number memory points accessible over the network.

5.13.2.5 Run

The encoding of the request parameter field for this command is specified in Table 30.

Table 30 – System information command parameter field

Octet	Field	Encoding
0 – 1	Run mode	0 = undefined 1 = forced run mode 2 = undefined 3 = normal run mode 4 : 65535 = reserved
2	Clear mode	0 = no not clear device 1 = clear all but locked range 2 = clear complete device 3 : 255 = reserved
3	Signal flow mode	reserved (set = 0)

5.13.2.6 Stop

The encoding of the request parameter field for this command is specified in Table 31.

Table 31 – System information command parameter field

Octet	Field	Encoding
0 – 1	Run mode	0 = undefined 1 = forced run mode 2 = undefined 3 = normal run mode 4 : 65535 = reserved

5.13.2.7 Line test

The encoding of the response parameter field for this command is specified in Table 32.

Table 32 – Line test command parameter field

Octet	bit	description	value
0 – 1	15 – 0	Vendor code	The assignment and management of vendor codes is FAL user specific and beyond the scope of this specification.
2	1 – 0	Total number of used bit-oriented data bits (both RX and RY combined)	0 = full use 1 = 8 used 2 = 32 used 3 = 16 used
	3 – 2	Distribution of used bit-oriented data bits	0 = RX and RY in equal sizes 1 = RX only 2 = RY only 3 = other RX / RY mix
	5 – 4	Number of occupied station slots	0 = 1 slot 1 = 2 slots 2 = 3 slots 3 = 4 slots
	7 – 6	reserved	(set = 0)
3	0	FAL user switch setting	0 = normal 1 = abnormal
	1	FAL user output upon fault	0 = clear 1 = hold
	5 – 2	reserved	(set = 0)
	7 – 6	Process data support level	0 = level A 1 = level B 2 = level C 3 = reserved
4	6 – 0	reserved	(set = 0)
	7	Messaging system support. Acyclic data format supports messaging type commands.	0 = not supported 1 = supported
5	5 – 0	software version	1 – 63
	7 – 6	Cyclic data segmenting support	0 = does not support cyclic data segmenting 1 = supports cyclic data segmenting 2 = reserved 3 = reserved
6 – 9	-	Master's test data	The 4 octets of test data receive by the Slave device form the Master during the connection process.

5.13.2.8 Memory read

The encoding of the request parameter field for this command is specified in Table 33.

Table 33 – Memory read command parameter field

Octet	Field	Encoding		
0 – 1	Quantity	Specifies the number of attribute specifications that follow 1 : 160 = range of allowable values		
2	Attribute type	**Bit**	**Description**	**Values**
		0	Memory location with respect to device	0 = internal 1 = external
		2 – 1	Access type	0 = bit 1 = octet 2 = word 3 = double word
		7 – 3	reserved	0
3	Access code	**Bit**	**Description**	**Values**
		0	Bit input data	0 = False 1 = True
		1	Bit output data	0 = False 1 = True
		2	Word data	0 = False 1 = True
		3	Timer	0 = False 1 = True
		4	Counter	0 = False 1 = True
		5	Link	0 = False 1 = True
		6	Status	0 = False 1 = True
		7	Retain	0 = False 1 = True
4 – 5	Address	Specifies the memory address		
6 – 7	Number of points to read	for Bit type attribute 0 : 7680 = range of allowable values for octet type attribute 0 : 960 = range of allowable values for Word type attribute 0 : 480 = range of allowable values		
…	…	…		
	Blocks of 6 octets Attribute Specifications as specified in Quantity			
	…			

The encoding of the response parameter field for this command is the data field specified by the request.

5.13.2.9 Memory write

The encoding of the request parameter field for this command is specified in Table 34.

Table 34 – Memory write command parameter field

Octet	Field	Encoding		
0 – 1	Quantity	Specifies the number of attribute specifications that follow 1 : 160 = range of allowable values		
2	Attribute type	**Bit**	**Description**	**Values**
		0	Memory location with respect to device	0 = internal 1 = external
		2 – 1	Access type	0 = bit 1 = octet 2 = word 3 = double word
		7 – 3	reserved	0
3	Access code and memory type	**Bit**	**Description**	**Values**
		0	Bit input data	0 = False 1 = True
		1	Bit output data	0 = False 1 = True
		2	Word data	0 = False 1 = True
		3	Timer	0 = False 1 = True
		4	Counter	0 = False 1 = True
		5	Link	0 = False 1 = True
		6	Status	0 = False 1 = True
		7	Retain	0 = False 1 = True
4 – 5	Address	Specifies the memory address		
6 – 7	Number of points to read	for Bit type attribute 0 : 7680 = range of allowable values for octet type attribute 0 : 960 = range of allowable values for Word type attribute 0 : 480 = range of allowable values		
...	... Blocks of 6 octets Attribute Specifications as specified in Quantity ...			
...	... Data to be written, encoded as specified in the request ...			

6 Structure of FAL protocol state machines

Interface to FAL services and protocol machines are specified in this subclause.

NOTE The state machines specified in this subclause and ARPMs defined in the following sections only define the valid events for each. It is a local matter to handle these invalid events.

The behavior of the FAL is described by three integrated protocol machines. The three protocol machines are: FAL Service Protocol Machine (FSPM), the Application Relationship Protocol Machine (ARPM), and the Data-link layer Mapping Protocol Machine (DMPM). The relationship among these protocol machines as well as primitives exchanged among them are depicted in Figure 3.

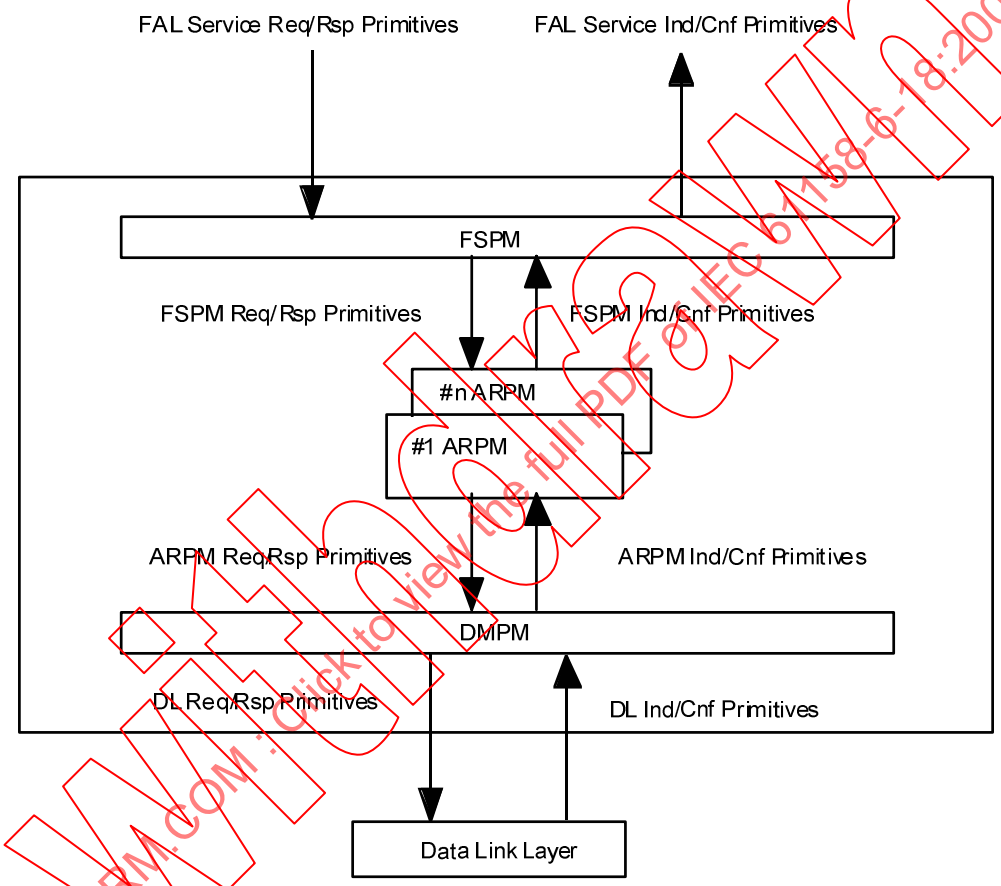

Figure 3 – Relationships among protocol machines and adjacent layers

The FSPM describes the service interface between the FAL-user and a particular AREP. The FSPM is common to all the AREP classes and does not have any state changes. The FSPM is responsible for the following activities:

a) to accept service primitives from the FAL service user and convert them into FAL internal primitives;

b) to select an appropriate ARPM state machine based on the AREP Identifier parameter supplied by the FAL-user and send FAL internal primitives to the selected ARPM;

c) to accept FAL internal primitives from the ARPM and convert them into service primitives for the FAL-user.

The ARPM describes the establishment and release of an AR and exchange of FAL-PDUs with a remote ARPM(s). The ARPM is responsible for the following activities:

a) to accept FAL internal primitives from the FSPM and create and send other FAL internal primitives to either the FSPM or the DMPM, based on the AREP and primitive types;

b) to accept FAL internal primitives from the DMPM and send them to the FSPM as a form of FAL internal primitives;

c) if the primitives are for the Establish or Release service, it shall try to establish or release the specified AR.

The DMPM describes the mapping between the FAL and the DLL. It is common to all the AREP types and does not have any state changes. The DMPM is responsible for the following activities:

a) to accept FAL internal primitives from the ARPM, prepare DLL service primitives, and send them to the DLL;

b) to receive DLL indication or confirmation primitives from the DLL and send them to the ARPM in a form of FAL internal primitives.

7 AP-context state machine

The type 18 FAL does not implement an AP-context state machine.

8 FAL service protocol machine (FSPM)

8.1 Overview

The FSPM provides the interface to the FAL user in the form of service handlers which convert service parameters into APDUs and process service requests from the FAL user or convert APDUs into service parameters and deliver service indications to the FAL user.

8.2 FAL service primitives

The FSPM operates in a single state with events defined by the receipt of service primitives. The descriptions of these receipt events are specified in Table 35.

Table 35 – FSPM events

Primitive	Description
Get	Retrieves the value of the attribute identified.
Set	Sets the value of the attribute identified as specified.
Error	Upon receipt of an error indication from the ARPM, an analogous error notification is delivered to the FAL user.
Connect	Upon receipt of a connect service request from the FAL user, a connect service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Disconnect	Upon receipt of a disconnect service request from the FAL user, a disconnect service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Start scan	Upon receipt of a start scan service request from the FAL user, a start scan service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Stop scan	Upon receipt of a connect service request from the FAL user, a stop scan service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Activate standby	Upon receipt of a connect service request from the FAL user, a stop scan service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Verify slave configuration	Upon receipt of a verify slave configuration service request from the FAL user, a verify slave configuration service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Trigger transmission	Upon receipt of a trigger transmissions service request from the FAL user, a trigger transmission service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Data received	Upon receipt of a data received indication from the ARPM, a data received indication is delivered to the FAL user.
Send message	Upon receipt of a send message service request from the FAL user, a send message service request is delivered to the ARPM and the confirmation is returned to the FAL user.
Message received	Upon receipt of a message received indication from the ARPM, a message received indication is delivered to the FAL user.

9 AR protocol machine (ARPM)

9.1 Overview

The ARPM manages the functions and behaviors of the ARs by

a) receiving, decoding and processing service primitives from the FSPM,
b) preparing, encoding and delivering service primitives to the DMPM,
c) receiving, decoding and processing service primitives from the DMPM,
d) preparing, encoding and delivering service primitives to the FSPM,
e) monitoring critical functions of the ARs including timeout times and other fault conditions,
f) delivering event notifications to the APCSM.

There are three types of AR: M1 Master, M2 Master and Slave.

9.2 M1 master ARPM

9.2.1 Overview

The M1 Master ARPM manages the behavioral states, transitions and interactions of an M1 Master AR. As shown in Figure 4, there are two states.