IEC 61158-5-3

Edition 3.0 2014-08

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Industrial communication networks – Fieldbus specifications – Part 5-3: Application layer service definition – Type 3 elements

Réseaux de communication industriels – Spécifications des bus de terrain – Partie 5-3: Définition des services de la couche application – Éléments de type 3

IEC 61158-5-3:2014-08(en-fr)

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

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

IEC 61158-5-3

Edition 3.0 2014-08

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Industrial communication networks – Fieldbus specifications – Part 5-3: Application layer service definition – Type 3 elements

Réseaux de communication industriels – Spécifications des bus de terrain – Partie 5-3: Définition des services de la couche application – Éléments de type 3

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

PRICE CODE XH
CODE PRIX

ICS 25.040.40; 35.100.70; 35.110

ISBN 978-2-8322-1732-0

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 5-3: Application layer service definition – Type 3 elements

FOREWORD

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

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

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

International Standard IEC 61158-5-3 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 change with respect to the previous edition is listed below:

- Correction of spelling and improved formating for a better reading.

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

FDIS	Report on voting
65C/763/FDIS	65C/773/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.

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 61158-1:2013.

The application service is provided by the application protocol making use of the services available from the data-link or other immediately lower layer. This standard defines the application service characteristics that fieldbus applications and/or system management exploit.

Throughout the set of fieldbus standards, the term “service” refers to the abstract capability provided by one layer of the OSI Basic Reference Model to the layer immediately above. Thus, the application layer service defined in this standard is a conceptual architectural service, independent of administrative and implementation divisions.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 5-3: Application layer service definition – Type 3 elements

1 Scope

1.1 General

This standard 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.

This sub-part contains material specific to Type 3 fieldbus.

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 3 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 service provided by the different Types of fieldbus Application Layer in terms of

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

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

a) the FAL user at the boundary between the user and the Application Layer of the Fieldbus Reference Model; and
b) Systems Management at the boundary between the Application Layer and Systems Management of the Fieldbus Reference Model.

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

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

object (APO) classes. One of the FAL ASEs is a management ASE that provides a common set of services for the management of the instances of FAL classes.

Although these services specify, from the perspective of applications, how request and responses are issued and delivered, they do not include a specification of what the requesting and responding applications are to do with them. That is, the behavioral aspects of the applications are not specified; only a definition of what requests and responses they can send/receive is specified. This permits greater flexibility to the FAL users in standardizing such object behavior. In addition to these services, some supporting services are also defined in this standard to provide access to the FAL to control certain aspects of its operation.

1.2 Specifications

The principal objective of this standard is to specify the characteristics of conceptual application layer services suitable for time-critical communications, and thus supplement the OSI Basic Reference Model in guiding the development of application layer protocols for time-critical communications.

A secondary objective is to provide migration paths from previously-existing industrial communications protocols. It is this latter objective which gives rise to the diversity of services standardized as the various Types of IEC 61158, and the corresponding protocols standardized in subparts of IEC 61158-6.

This specification may be used as the basis for formal Application Programming-Interfaces. Nevertheless, it is not a formal programming interface, and any such interface will need to address implementation issues not covered by this specification, including

a) the sizes and octet ordering of various multi-octet service parameters, and

b) the correlation of paired request and confirm, or indication and response, primitives.

1.3 Conformance

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

There is no conformance of equipment to this application layer service definition standard. Instead, conformance is achieved through implementation of conforming application layer protocols that fulfill any given Type of application layer services as defined in this part of IEC 61158.

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 61131-1, *Programmable controllers – Part 1: General information*

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

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

IEC 61158-4-3:2014, *Industrial communication networks – Fieldbus specifications – Part 4-3: Data-link layer protocol specification – Type 3 elements*

IEC 61158-5-10:2014, *Industrial communication networks – Fieldbus specifications – Part 5-10: Application layer service definition – Type 10 elements*

IEC 61158-6-3:2014, *Industrial communication networks – Fieldbus specifications – Part 6–3: Application layer protocol specification – Type 3 elements*

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

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

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

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

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

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

3 Terms, definitions, abbreviations, symbols and conventions

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

3.1 Referenced terms and definitions

3.1.1 ISO/IEC 7498-1 terms

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

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 8824-1 terms

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

a) object identifier
b) type

3.1.5 Fieldbus Data Link Layer terms

For the purposes of this document, the following terms as defined in IEC 61158-3-3 and IEC 61158-4-3 apply.

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

3.2 Fieldbus Application Layer Type 3 – specific terms and definitions

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

3.2.1
access protection
limitation of the usage of an application object to one client

3.2.2
address assignment table
mapping of the client's internal I/O-Data object storage to the decentralized input and output data objects

3.2.3
allocate
take a resource from a common area and assign that resource for the exclusive use of a specific entity

3.2.4
application
function or data structure for which data is consumed or produced

3.2.5
application layer interoperability
capability of application entities to perform coordinated and cooperative operations using the services of the FAL

3.2.6
application objects
multiple object classes that manage and provide a run time exchange of messages across the network and within the network device

3.2.7
application process identifier
distinguishes multiple application processes used in a device

3.2.8
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.

3.2.9
application process object class
a class of application process objects defined in terms of the set of their network-accessible attributes and services

3.2.10
application relationship
cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation.

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

3.2.11
application relationship application service element
application-service-element that provides the exclusive means for establishing and terminating all application relationships

3.2.12
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.

3.2.13
attribute
description of an externally visible characteristic or feature of an object

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

3.2.14
behavior
indication of how an object responds to particular events

3.2.15
channel
single physical or logical link of an input or output application object of a server to the process

3.2.16
channel related diagnosis
information concerning a specific element of an input or output application object, provided for maintenance purposes

EXAMPLE: validity of data

3.2.17
class
a set of objects, all of which represent the same kind of system component

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

3.2.18
class attributes
attribute that is shared by all objects within the same class

3.2.19
class code
unique identifier assigned to each object class

3.2.20
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.

3.2.21
client
<object view> object which uses the services of another (server) object to perform a task

3.2.22
client
<communication view> initiator of a message to which a server reacts

3.2.23
configuration check
comparison of the expected I/O-Data object structuring of the client with the real I/O-Data object structuring to the server in the start-up phase

3.2.24
configuration fault
an unacceptable difference between the expected I/O-Data object structuring and the real I/O-Data object structuring, as detected by the server

3.2.25
configuration identifier
representation of a portion of I/O Data of a single input- and/or output-module of a server

3.2.26
connection
logical binding between application objects that may be within the same or different devices

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

Note 2 to entry: The logical link between sink and source of attributes and services at different custom interfaces of RT-Auto ASEs is referred to as interconnection. There is a distinction between data and event interconnections. The logical link and the data flow between sink and source of automation data items is referred to as data interconnection. The logical link and the data flow between sink (method) and source (event) of operational services is referred to as event interconnection.

3.2.27
consume
act of receiving data from a producer

3.2.28
consumer
node or sink that is receiving data from a producer

3.2.29
consuming application
application that consumes data

3.2.30
cyclic
repetitive in a regular manner

3.2.31
data consistency
means for coherent transmission and access of the input- or output-data object between and within client and server

3.2.32
default DL-address
value 126 as an initial value for DL-address, which has to be changed (e.g. by assignment of an DL-address via the fieldbus) before operation with a DP-master (class 1)

3.2.33
device
physical hardware connected to the link

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

3.2.34
diagnosis information
all data available at the server for maintenance purposes

3.2.35
diagnosis information collection
system diagnosis information that is assembled at the client side

3.2.36
DP-master (class 1)
a controlling device which controls several DP-slaves (field devices)

Note 1 to entry: This is usually a programmable controller or a distributed control system.

3.2.37
DP-master (class 2)
controlling device which manages configuration data (parameter sets) and diagnosis data of a DP-master (Class 1), and that additionally can perform all communication capabilities of a DP-master (Class 1)

3.2.38
DP-slave
field device that can be assigned to one DP-master (Class 1) as a provider for cyclic I/O data exchange; in addition acyclic functions and alarms could be provided

3.2.39
endpoint
one of the communicating entities involved in a connection

3.2.40
engineering
abstract term that characterizes the client application or device responsible for configuring an automation system via interconnecting data items

3.2.41
error
discrepancy between a computed, observed or measured value or condition and the specified or theoretically correct value or condition

3.2.42
error class
general grouping for related error definitions and corresponding error codes

3.2.43
error code
identification of a specific type of error within an error class

3.2.44
event
an instance of a change of conditions

3.2.45
FAL subnet
subnetworks composed of one or more data link segments, identified by a subset of the network address

Note 1 to entry: FAL subnets are permitted to contain bridges but not routers.

3.2.46
freeze
function at the DP-slaves for simultaneous data transfer between the input data object and the process

3.2.47
group
<general> a general term for a collection of objects. Specific uses:

3.2.48
group
<addressing> when describing an address, an address that identifies more than one entity

3.2.49
group
<Type 3> set of DP-slaves which perform a Freeze or Sync function

3.2.50
interface
<general> shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics as appropriate

3.2.51
interface
<FAL> collection of FAL class attributes and services that represents a specific view on the FAL class

3.2.52
interface pointer
key attribute that unambiguously addresses an object interface instance

3.2.53
invocation
act of using a service or other resource of an application process

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

3.2.54
I/O data
object designated to be transferred cyclically for the purpose of processing

3.2.55
ident number
DP-master (Class 1) or DP-slave device type

3.2.56
identifier related diagnosis
information dedicated to modules for maintenance purpose

3.2.57
index
address of an object within an application process

3.2.58
instance
the actual physical occurrence of an object within a class that identifies one of many objects within the same object class

EXAMPLE California is an instance of the object class state.

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

3.2.59
instantiated
object that has been created in a device

3.2.60
manufacturer ID
identification of each product manufacturer by a unique number

3.2.61
master parameter set
the configuration and parameterization data of all DP-slaves that are assigned to the corresponding DP-master and the bus parameters

3.2.62
member
piece of an attribute that is structured as an element of an array

3.2.63
message router
object within a node that distributes messaging requests to appropriate application objects

3.2.64
method
a synonym for an operational service which is provided by the server ASE and invoked by a client

3.2.65
module
<general> hardware or logical component of a physical device

3.2.66
module
<Type 3> addressable unit inside the DP-slave

3.2.67
network
a set of nodes connected by some type of communication medium, including any intervening repeaters, bridges, routers and lower-layer gateways

3.2.68
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

3.2.69
physical device
<general> an automation or other network device

3.2.70
process data
object(s) which are already pre-processed and transferred acyclically for the purpose of information or further processing

3.2.71
produce
act of sending data to be received by a consumer

3.2.72
producer
node that is responsible for sending data

3.2.73
provider
source of a data connection

3.2.74
publisher
role of an AR endpoint that transmits APDUs onto the fieldbus for consumption by one or more subscribers

Note 1 to entry: A publisher may not be aware of the identity or the number of subscribers and it may publish its APDUs using a dedicated AR.

3.2.75
real configuration
input and output data structure of the DP-slave, including definition of data consistency

3.2.76
resource
a processing or information capability of a subsystem

3.2.77
server
< communication view> role of an AREP in which it returns a confirmed service response APDU to the client that initiated the request

3.2.78
server
< object view> object which provides services to another (client) object

3.2.79
service
operation or function than an object and/or object class performs upon request from another object and/or object class

3.2.80
slot
address of a module within a DP-slave

3.2.81
subscriber
role of an AREP in which it receives APDUs produced by a publisher

3.2.82
sync
function at the DP-slaves for simultaneous data transfer between the output data object and the process

3.2.83
target
end-node to which a connection is established

3.2.84
unconnected service
messaging service which does not rely on the set up of a connection between devices before allowing information exchanges

3.3 Abbreviations and symbols

AE	Application Entity
AL	Application Layer
ALME	Application Layer Management Entity
ALP	Application Layer Protocol
APO	Application Object
AP	Application Process
APDU	Application Protocol Data Unit
API	Application Process Identifier
AR	Application Relationship
AREP	Application Relationship End Point
ASE	Application Service Element
Cnf	Confirmation
CR	Communication Relationship
CREP	Communication Relationship End Point
DL-	(as a prefix) data-link-
DLL	Data-link layer
DLM	Data-link management
DLSAP	Data-link Service Access Point
DLSDU	DL-service-data-unit
DP	Decentralised Peripherals
FAL	Fieldbus Application Layer
ID	Identifier
IEC	International Electrotechnical Commission
Ind	Indication
ISO	International Organization for Standardization
LME	Layer Management Entity

OSI	Open Systems Interconnect
PDev	Physical Device
PDU	Protocol Data Unit
PL	Physical Layer
QoS	Quality of Service
Req	Request
Rsp	Response
SAP	Service Access Point

3.4 Conventions

3.4.1 Overview

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of two parts, its class specification, and its service specification.

The class specification defines the attributes of the class. The attributes are accessible from instances of the class using the Object Management ASE services specified in Clause 5 of this standard. The service specification defines the services that are provided by the ASE.

3.4.2 General conventions

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

3.4.3 Conventions for class definitions

Class definitions are described using templates. Each template consists of a list of attributes for the class. The general form of the template is shown below:

FAL ASE:			**ASE Name**
CLASS:			Class Name
CLASS ID:			#
PARENT CLASS:			Parent Class Name
ATTRIBUTES:			
1	(o)	Key Attribute:	numeric identifier
2	(o)	Key Attribute:	name
3	(m)	Attribute:	attribute name(values)
4	(m)	Attribute:	attribute name(values)
4.1	(s)	Attribute:	attribute name(values)
4.2	(s)	Attribute:	attribute name(values)
4.3	(s)	Attribute:	attribute name(values)
5.	(c)	Constraint:	constraint expression
5.1	(m)	Attribute:	attribute name(values)
5.2	(o)	Attribute:	attribute name(values)
6	(m)	Attribute:	attribute name(values)
6.1	(s)	Attribute:	attribute name(values)
6.2	(s)	Attribute:	attribute name(values)
SERVICES:			
1	(o)	OpsService:	service name
2.	(c)	Constraint:	constraint expression
2.1	(o)	OpsService:	service name
3	(m)	MgtService:	service name

(1) The "FAL ASE:" entry is the name of the FAL ASE that provides the services for the class being specified.

(2) The "CLASS:" entry is the name of the class being specified. All objects defined using this template will be an instance of this class. The class may be specified by this standard, or by a user of this standard.

(3) The "CLASS ID:" entry is a number that identifies the class being specified. This number is unique within the FAL ASE that will provide the services for this class. When qualified by the identity of its FAL ASE, it unambiguously identifies the class within the scope of the FAL. The value "NULL" indicates that the class cannot be instantiated. Class IDs between 1 and 255 are reserved by this standard to identify standardized classes. They have been assigned to maintain compatibility with existing national standards. CLASS IDs between 256 and 2 048 are allocated for identifying user defined classes.

(4) The "PARENT CLASS:" entry is the name of the parent class for the class being specified. All attributes defined for the parent class and inherited by it are inherited for the class being defined, and therefore do not have to be redefined in the template for this class.

NOTE The parent-class "TOP" indicates that the class being defined is an initial class definition. The parent class TOP is used as a starting point from which all other classes are defined. The use of TOP is reserved for classes defined by this standard.

(5) The "ATTRIBUTES" label indicate that the following entries are attributes defined for the class.

a) Each of the attribute entries contains a line number in column 1, a mandatory (m) / optional (o) / conditional (c) / selector (s) indicator in column 2, an attribute type label in column 3, a name or a conditional expression in column 4, and optionally a list of enumerated values in column 5. In the column following the list of values, the default value for the attribute may be specified.

b) Objects are normally identified by a numeric identifier or by an object name, or by both. In the class templates, these key attributes are defined under the key attribute.

c) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting is used to specify

i) fields of a structured attribute (4.1, 4.2, 4.3),

ii) attributes conditional on a constraint statement (5). Attributes may be mandatory (5.1) or optional (5.2) if the constraint is true. Not all optional attributes require constraint statements as does the attribute defined in (5.2).

iii) the selection fields of a choice type attribute (6.1 and 6.2).

(6) The "SERVICES" label indicates that the following entries are services defined for the class.

a) An (m) in column 2 indicates that the service is mandatory for the class, while an (o) indicates that it is optional. A (c) in this column indicates that the service is conditional. When all services defined for a class are defined as optional, at least one has to be selected when an instance of the class is defined.

b) The label "OpsService" designates an operational service (1).

c) The label "MgtService" designates a management service (2).

d) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting within the list of services is used to specify services conditional on a constraint statement.

3.4.4 Conventions for service definitions

3.4.4.1 General

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

3.4.4.2 Service parameters

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

The service specifications of this standard use a tabular format to describe the component parameters of the ASE service primitives. The parameters which apply to each group of service primitives are set out in tables. Each table consists of up to five columns for the

a) Parameter name,

b) request primitive,

c) indication primitive,

d) response primitive, and

e) confirm primitive.

One parameter (or component of it) is listed in each row of each table. Under the appropriate service primitive columns, a code is used to specify the type of usage of the parameter on the primitive specified in the column:

M parameter is mandatory for the primitive

U parameter is a User option, and may or may not be provided depending on dynamic usage of the service user. When not provided, a default value for the parameter is assumed.

C parameter is conditional upon other parameters or upon the environment of the service user.

— (blank) parameter is never present.

S parameter is a selected item.

Some entries are further qualified by items in brackets. These may be

a) a parameter-specific constraint:
"(=)" indicates that the parameter is semantically equivalent to the parameter in the service primitive to its immediate left in the table;

b) an indication that some note applies to the entry:
"(n)" indicates that the following note "n" contains additional information pertaining to the parameter and its use.

3.4.4.3 Service procedures

The procedures are defined in terms of

– the interactions between application entities through the exchange of fieldbus Application Protocol Data Units, and

– the interactions between an application layer service provider and an application layer service user in the same system through the invocation of application layer service primitives.

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

3.4.5 Overview

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of two parts, its class specification, and its service specification.

The class specification defines the attributes of the class. The attributes are accessible from instances of the class using the Object Management ASE services specified in clause 5 of this standard. The service specification defines the services that are provided by the ASE.

3.4.6 General conventions

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

3.4.7 Conventions for class definitions

Class definitions are described using templates. Each template consists of a list of attributes for the class. The general form of the template is shown below:

FAL ASE:			**ASE Name**
CLASS:			Class Name
CLASS ID:			#
PARENT CLASS:			Parent Class Name
ATTRIBUTES:			
1	(o)	Key Attribute:	numeric identifier
2	(o)	Key Attribute:	name
3	(m)	Attribute:	attribute name(values)
4	(m)	Attribute:	attribute name(values)
4.1	(s)	Attribute:	attribute name(values)
4.2	(s)	Attribute:	attribute name(values)
4.3	(s)	Attribute:	attribute name(values)
5.	(c)	Constraint:	constraint expression
5.1	(m)	Attribute:	attribute name(values)
5.2	(o)	Attribute:	attribute name(values)
6	(m)	Attribute:	attribute name(values)
6.1	(s)	Attribute:	attribute name(values)
6.2	(s)	Attribute:	attribute name(values)
SERVICES:			
1	(o)	OpsService:	service name
2.	(c)	Constraint:	constraint expression
2.1	(o)	OpsService:	service name
3	(m)	MgtService:	service name

(1) The "FAL ASE:" entry is the name of the FAL ASE that provides the services for the class being specified.

(2) The "CLASS:" entry is the name of the class being specified. All objects defined using this template will be an instance of this class. The class may be specified by this standard, or by a user of this standard.

(3) The "CLASS ID:" entry is a number that identifies the class being specified. This number is unique within the FAL ASE that will provide the services for this class. When qualified by the identity of its FAL ASE, it unambiguously identifies the class within the scope of the FAL. The value "NULL" indicates that the class cannot be instantiated. Class IDs

between 1 and 255 are reserved by this standard to identify standardized classes. They have been assigned to maintain compatibility with existing national standards. CLASS IDs between 256 and 2 048 are allocated for identifying user defined classes.

(4) The "PARENT CLASS:" entry is the name of the parent class for the class being specified. All attributes defined for the parent class and inherited by it are inherited for the class being defined, and therefore do not have to be redefined in the template for this class.

NOTE The parent-class "TOP" indicates that the class being defined is an initial class definition. The parent class TOP is used as a starting point from which all other classes are defined. The use of TOP is reserved for classes defined by this standard.

(5) The "ATTRIBUTES" label indicate that the following entries are attributes defined for the class.

a) Each of the attribute entries contains a line number in column 1, a mandatory (m) / optional (o) / conditional (c) / selector (s) indicator in column 2, an attribute type label in column 3, a name or a conditional expression in column 4, and optionally a list of enumerated values in column 5. In the column following the list of values, the default value for the attribute may be specified.

b) Objects are normally identified by a numeric identifier or by an object name, or by both. In the class templates, these key attributes are defined under the key attribute.

c) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting is used to specify

i) fields of a structured attribute (4.1, 4.2, 4.3),

ii) attributes conditional on a constraint statement (5). Attributes may be mandatory (see 5.1) or optional (see 5.2) if the constraint is true. Not all optional attributes require constraint statements as does the attribute defined in 5.2.

iii) the selection fields of a choice type attribute (see 6.1 and 6.2).

(6) The "SERVICES" label indicates that the following entries are services defined for the class.

a) An (m) in column 2 indicates that the service is mandatory for the class, while an (o) indicates that it is optional. A (c) in this column indicates that the service is conditional. When all services defined for a class are defined as optional, at least one has to be selected when an instance of the class is defined.

b) The label "OpsService" designates an operational service (1).

c) The label "MgtService" designates a management service (2).

d) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting within the list of services is used to specify services conditional on a constraint statement.

3.4.8 Conventions for service definitions

3.4.8.1 General

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

3.4.8.2 Service parameters

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

The service specifications of this standard use a tabular format to describe the component parameters of the ASE service primitives. The parameters which apply to each group of service primitives are set out in tables. Each table consists of up to five columns for the

1) Parameter name,
2) request primitive,
3) indication primitive,
4) response primitive, and
5) confirm primitive.

One parameter (or component of it) is listed in each row of each table. Under the appropriate service primitive columns, a code is used to specify the type of usage of the parameter on the primitive specified in the column:

M parameter is mandatory for the primitive

U parameter is a User option, and may or may not be provided depending on dynamic usage of the service user. When not provided, a default value for the parameter is assumed.

C parameter is conditional upon other parameters or upon the environment of the service user.

— (blank) parameter is never present.

S parameter is a selected item.

Some entries are further qualified by items in brackets. These may be

a) a parameter-specific constraint:
"(=)" indicates that the parameter is semantically equivalent to the parameter in the service primitive to its immediate left in the table.

b) an indication that some note applies to the entry:
"(n)" indicates that the following note "n" contains additional information pertaining to the parameter and its use.

3.4.8.3 Service procedures

The procedures are defined in terms of

– the interactions between application entities through the exchange of fieldbus Application Protocol Data Units, and

– the interactions between an application layer service provider and an application layer service user in the same system through the invocation of application layer service primitives.

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

4 Concepts

IEC 61158-1, Clause 9 describes the concepts of the application layer service descriptions and the templates used in this standard.

5 Data type ASE

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

IEC 61158-5-10, Clause 5 provides a collection of all data types required by FAL services of Type 3 and Type 10 as well as data types allowed to be used by application profiles.

Encoding rules for the FAL data types used by Type 3 are provided in IEC 61158-6-3.

6 Communication model specification

6.1 DP concepts

6.1.1 User requirements of the fieldbus DP system

The typical automation system consists of one or several programmable controllers connected by I/O-systems to the machine/process. The I/O-system is mostly implemented as a set of single point connections with a well-defined electrical interface such as 24 V digital signals or 4 mA to 20 mA analogue signals. The overall structure of the programmable controller is defined in IEC 61131-1 and widely used.

To switch from I/O-systems plugged in a programmable controller to decentralized I/O-systems a serial link is necessary not only for the exchange of process data, but also to transmit diagnosis, configuration data and parameters.

Furthermore many applications in factory automation require very short reaction times.

Therefore the user interface in the programmable controller for the exchange of process data should be designed in a way to allow direct access to these variables without specific function calls.

In addition a reliable data transfer is required for industrial applications.

The integration of simple field devices requires a simple protocol architecture to allow low cost implementations.

To support the integration of smart field devices acyclic communication services and alarms are needed.

In order to achieve an automation system consisting of devices from different manufacturers strict interoperability of all components is necessary.

In accordance with the IEC 61131-1 structure the decentralized structure should be invisible to an existing control program.

For efficient communication between DP-slaves, as required in motion control applications, the fieldbus provides a Publisher/Subscriber communication mechanism.

The fieldbus provides a mechanism for the synchronization of the application of the DP-master (Class 1) and the DP-slaves with the communication cycle for highly synchronized real time applications.

For time stamping and other actions with precise timing requirements a mechanism to synchronies clocks is offered.

Field devices requires a mechanism to upload (Pull) and/or to download (Push) unstructured LR Data and a mechanism to invoke predefined functions within the DP-slave.

Fieldbus DP provides mechanism to build redundant DP-slaves, which may operate in control systems with several master implementations.

6.1.2 Features of fieldbus DP

Combining the structure defined in IEC 61131-1 with fieldbus DP the following advantages are achieved.

– Changes in the plant cause only small changes in wiring.
– Critical signals can be transferred over short distances to the remote field device (sensor/actuator) and to/from the programmable controller over long distances via fieldbus DP with high accuracy.
– Parameterization and diagnosis can be done without additional wiring.
– Commissioning of subparts without additional installations.

Table 1 gives an overview of the requirements and features of a fieldbus DP system.

Table 1 – Requirements and features of fieldbus DP

Requirement	Feature of fieldbus DP
Short reaction time	Exchange of more than 1 000 Inputs and Outputs with 32 field devices in less than 10 ms
Automation system consists of one or several programmable controller	Mono-master or Multi-master Operation
Simple field devices	Simple protocol, low cost communication interface
Reliable data transfer	Hamming Distance 4 guaranteed; Error recovery mechanism; Use of proven transmission technology
Excellent diagnosis	Various diagnosis in Masters as well as in Slaves
User interface with direct access to the input and output data	Buffer interface for the input and output data
Smart field devices	Acyclic communication which provides a flexible and enhanced addressing scheme of data within field devices. The possibility to transfer an alarm from the field device to the programmable controller with an explicit acknowledgement
Interoperability	Precise and complete definitions including the definition of the system behavior
Efficient communication between DP-slaves for motion control applications	The Publisher/Subscriber communication mechanism allows direct communication between DP-slaves without interactions from the User of the DP-master (Class 1)
Synchronization of the application of the DP-master (Class 1) and the DP-slaves	The Isochronous Mode provides a mechanism for the synchronization of the application of the DP-master (Class 1) and the DP-slaves with the communication cycle. The cycle jitter is less than 1 µs for data rates above 1,5 Mbit/s
Mechanism to load unstructured data blocks	To load unstructured data blocks fieldbus DP provides the Load Region mechanism
Invocation of predefined functions in the DP-slave	With the Function Invocation model predefined functions within the DP-slave may be invoked
Synchronization of clocks	With the Time Synchronization an uniform Network Time (UTC) with an accuracy of less than 1 ms can be achieved

6.1.3 DP communication model

6.1.3.1 Associations

The fieldbus communication model supports the communication of field devices with a single controlling device (e.g. programmable controller or distributed control system) via associations (see Figure 1). Furthermore, the communication between several controlling devices and their assigned field devices is also supported (see Figure 2). In addition to these basic communication models, fieldbus supports also the cyclic communication between field devices (see Figure 3).

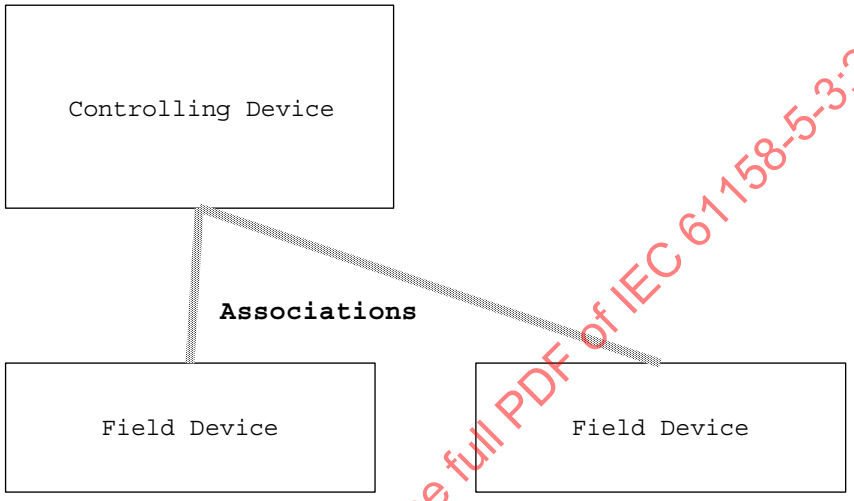


Figure 1 – Example of DP communication with a single controlling device

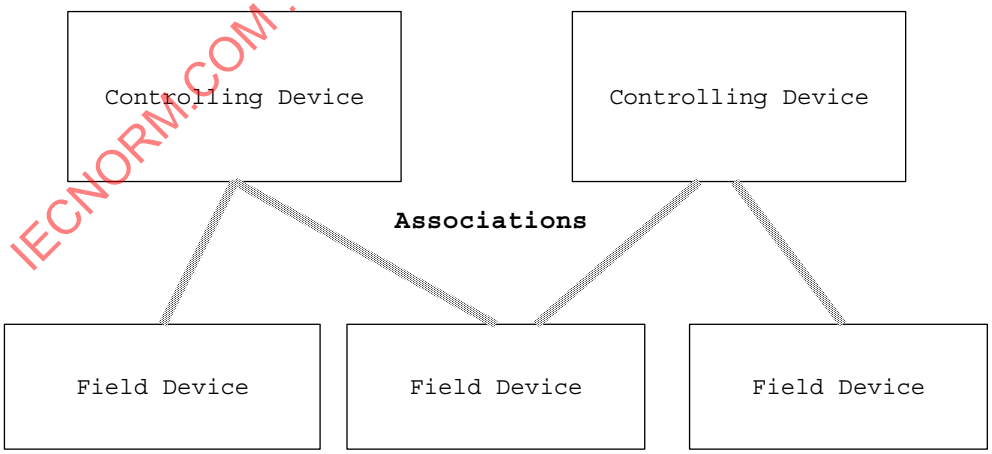


Figure 2 – Example of DP communication with several controlling devices

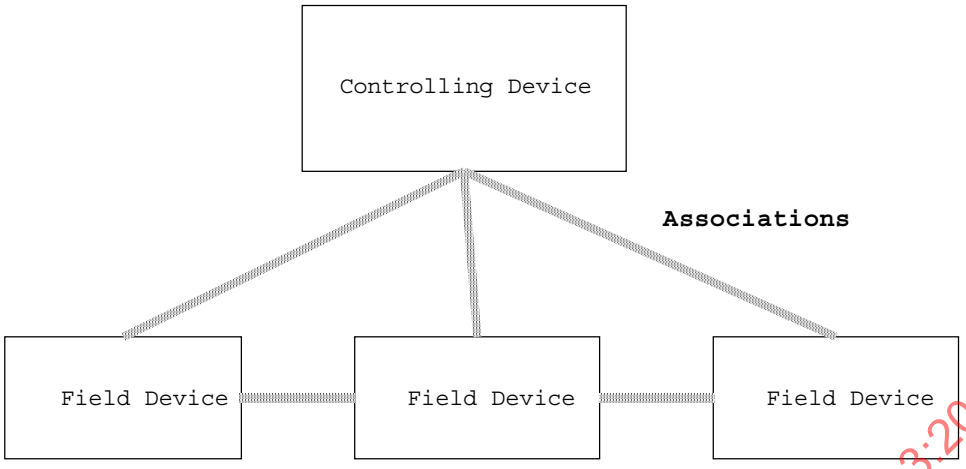


Figure 3 – Example of DP communication between field devices

6.1.3.2 Device types

6.1.3.2.1 DP-master (class 1)

The DP-master (Class 1) is a controlling device which is associated with one or more DP-slaves (field devices). The DP-master (Class 1) performs one or more of the following basic functionalities:

– cyclic exchange of I/O data with related DP-slaves,
– diagnosis with the DP-slaves,
– configuration of DP-slaves,
– treatment of configuration and diagnosis requests of a DP-master (Class 2).

The extended functionalities of a DP-master (Class 1) are:

– acyclic access to process data of the DP-slaves,
– treatment of alarms from DP-slaves,
– support of the Isochronous Mode for synchronization purposes with DP-slaves,
– support of the DXB mechanism for cyclic data exchange between DP-slaves,
– support of the up- and/or download of LR Data with DP-slaves,
– invocation of predefined functions within DP-slaves,
– support for clock synchronisation of the DP-slave's and other DP-master's clocks.

6.1.3.2.2 DP-master (class 2)

The DP-master (Class 2) is a controlling device which manages provision of configuration data (parameter sets) and collection of Diagnosis data for/from a DP-master (Class 1). Additionally the DP-master (Class 2) can perform all basic and some extended communication capabilities of a DP-master (Class 1) to a DP-slave.

6.1.3.2.3 DP-slave

6.1.3.2.3.1 General

The DP-slave is a field device and performs the following activities depending on the functionality.

The basic functionalities are:

– cyclic exchange of I/O data with the assigned DP-master,
– diagnosis with the assigned DP-master,
– treatment of configuration requests of a DP-master.

The extended functionalities of a DP-slave are:

– provision of acyclic access to process data for DP-masters,
– provision of alarms to the assigned DP-master,
– support of the Isochronous Mode for synchronization purposes with the DP-master (Class 1),
– cyclic data exchange between DP-slaves using the Publisher/Subscriber communication,
– support of the up- and/or download of LR Data,
– support of predefined functions which may be invoked by the DP-master (Class 1) or the DP-master (Class 2),
– provision to synchronies the local clock with a DP-master's clock,
– support of DP-slave Redundancy.

DP-slaves are composed of one or more modules. These modules usually reflect the hardware components of the field device.

6.1.3.2.3.2 Module

The DP Application Layer uses modules to reflect the structuring of components (e.g. hardware units, logical units) inside a DP-slave. A Module may have one or more channels which represent the real structure of the Input and/or Output Data. These channels may be a further subdivision of the Input and/or Output Data object.

Each module will be addressed by a slot number (1 to 254). Numbering is without gaps, ascending, beginning with 1. If a slot is not occupied with a module, an empty slot will be registered under the corresponding slot number in the configuration. The slot number "zero" (Slot = 0) refers to the DP-slave itself. Modules contain data which is index addressable. The concrete use of slot number and index to address data within the device is manufacturer specific.

Furthermore to each module a Configuration Identifier has to be assigned. Numbering is without gaps, ascending, beginning with 0. If a slot is not occupied with a module, a Cfg Identifier with the Input and Output Data length equal zero has to be assigned in the configuration (empty slot).

A module provides one or several of the following data elements:

– Diagnosis,
– I/O Data,
– Alarm,
– Process Data,
– LR Data,
– Context,
– Function Invocation Data.

In modular DP-slaves modules determine the actual configuration of the field device (see Figure 4). The number of modules is fixed according to the device specification (maximum number of modules is 244).

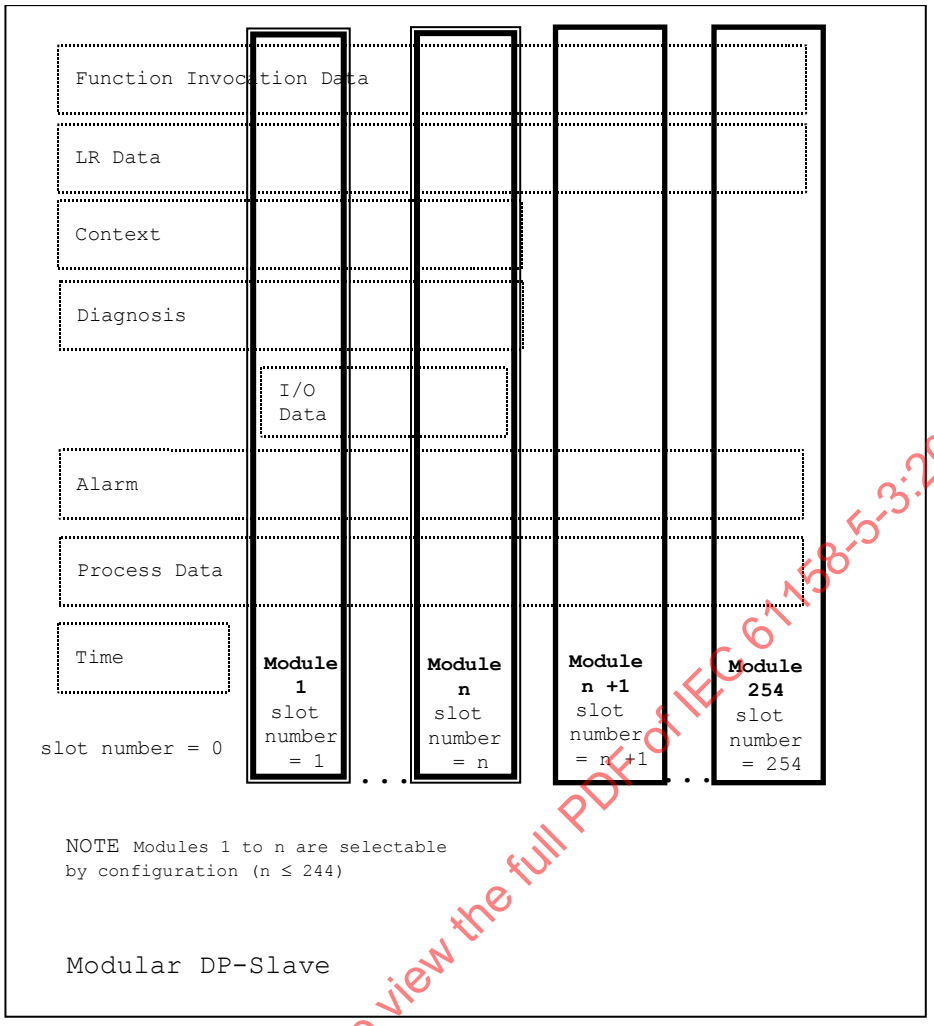


Figure 4 – DP-slave model (modular DP-slave)

In compact DP-slaves with respect to the device specification the number of modules is fixed and therefore also the configuration (for example only one module, see Figure 5).

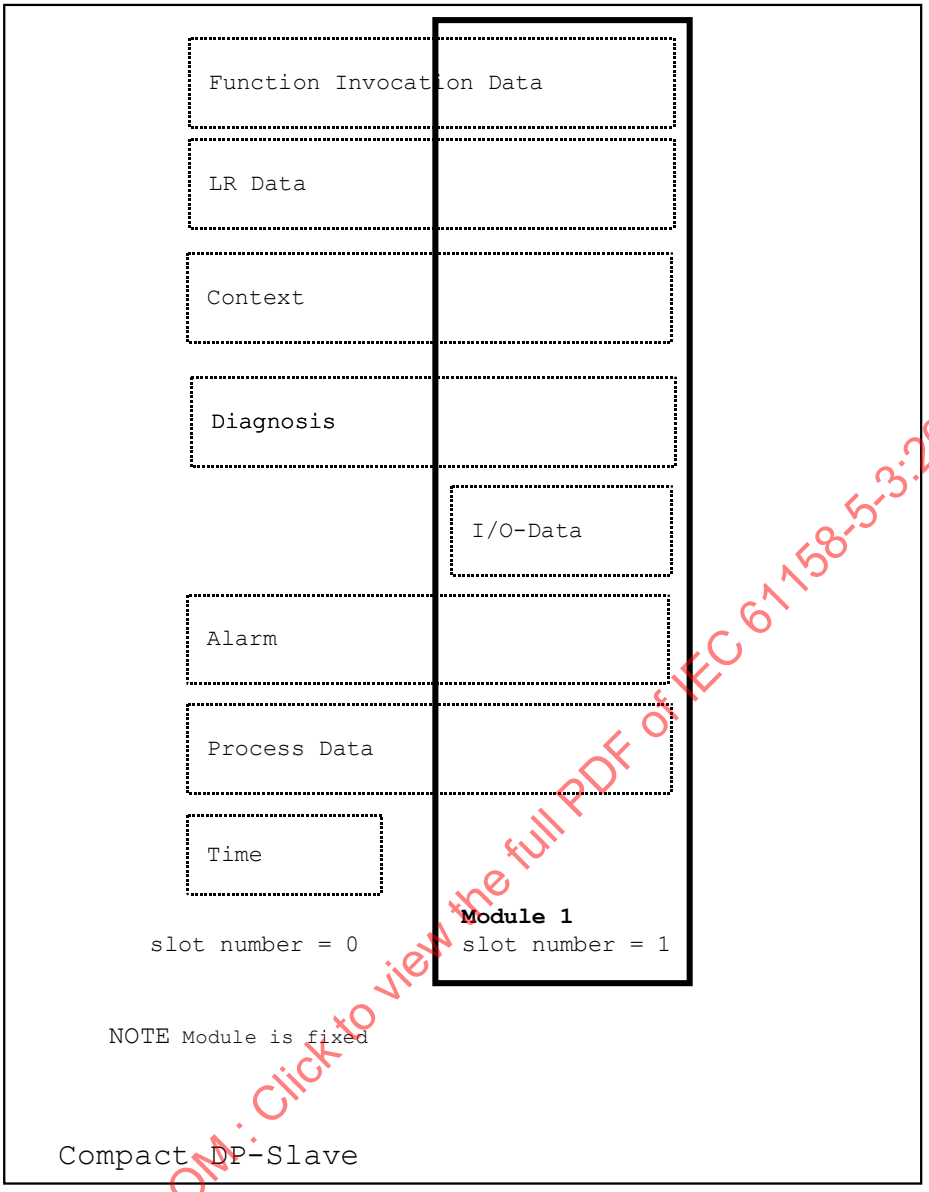


Figure 5 – DP-slave model (compact DP-slave)

6.1.3.3 Device attributes

Device attributes indicate that besides the basic functionality, extended functionality will be supported additionally.

These additional functionalities can imply certain rules for already defined data structures. A device attribute indicates that the device supports additional communication relationships or different communication behavior.

6.1.3.4 Application process

6.1.3.4.1 Overview

In the fieldbus DP environment, an application may be partitioned and distributed to a number of devices on the network. Each of these partitions is referred to as an application process (AP), (see Figure 6). A device may have several APs. In this case each individual AP is uniquely identified by an AP Identifier (API).

Every DP-slave shall have a Default AP with API = 0. The Default AP may be related to I/O data, to diagnosis and alarms, to configuration, to process data and to LR Data. All other DP-slave APs shall be related to process data only.

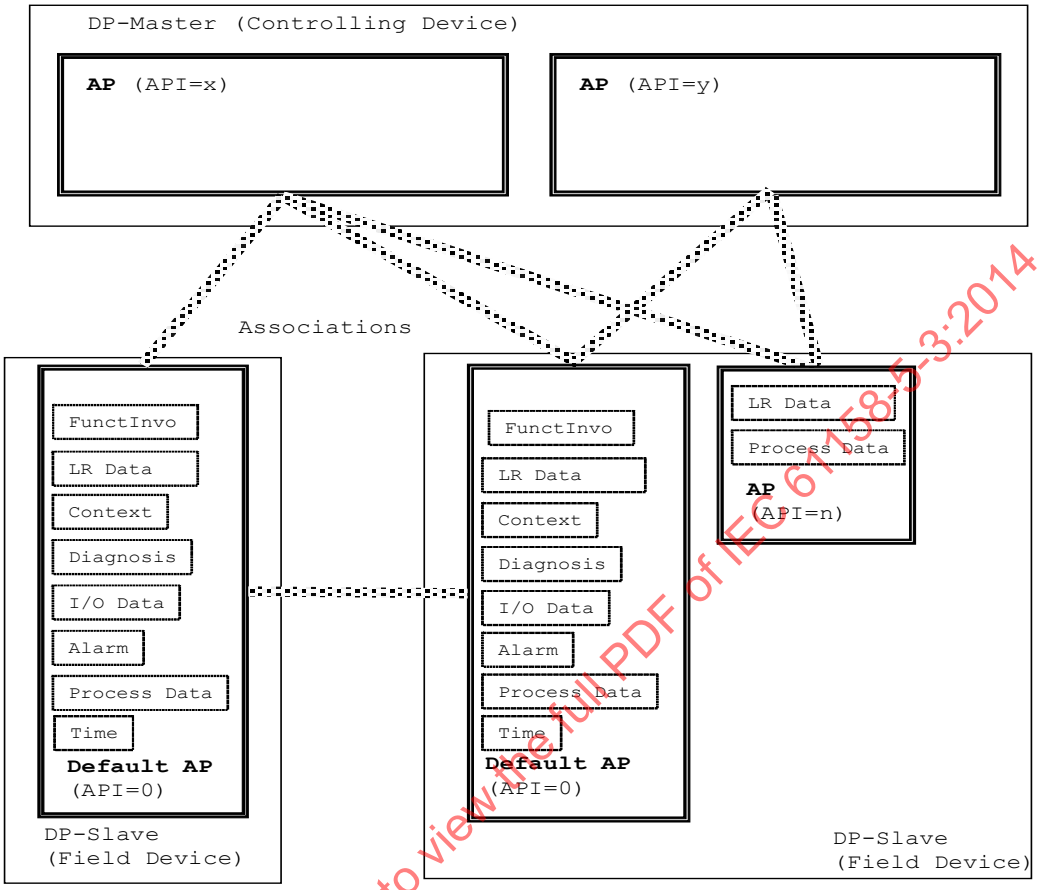


Figure 6 – Overview of application processes

An AP may be distributed to several modules. Each module within an application process is identified uniquely by a slot number (Slot = 1 to 254). The slot number "zero" (Slot = 0) refers to the AP itself within the field device.

The Context shall be related to the Default AP itself and to all its modules containing I/O Data.

The Diagnosis information may only be related to the Default AP itself and to all its modules containing I/O Data.

The I/O Data shall be related to modules of the Default AP and not to the AP itself. The number of modules with I/O Data is limited and depends on the configuration and I/O Data structure of the modules (see 6.1.3.2.3.2).

The Alarm data may be related to the Default AP itself and to all its modules.

The Process Data may be related to all APs and to all their modules.

The LR Data may be related to all APs and to all their modules.

The Time shall be related to the Default AP itself and not to its modules.

Figure 7 shows the relationship between the APs, the data elements and the modules of a DP-slave.

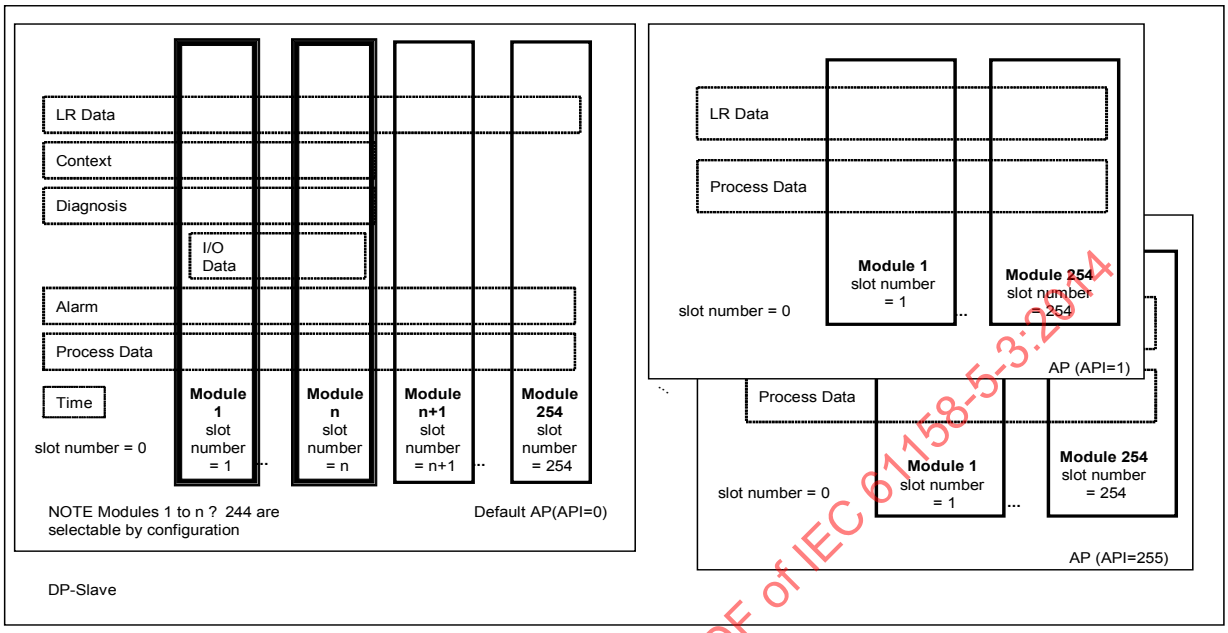


Figure 7 – DP-slave model (modular DP-slave)

The DP-master (Class 1) shall have only its Default AP. The DP-master (Class 1) shall communicate with the Default AP of a DP-slave only.

The DP-master (Class 2) may have several APs. Communicating with a DP-slave the DP-master (Class 2) may be able to address all APs of a DP-slave.

6.1.3.4.2 Application service element

An application service element (ASE), as defined in ISO/IEC 9545, is a set of application functions that provide a capability for the interworking of application processes for a specific purpose. ASEs provide a set of services for conveying requests and responses to and from application processes and their objects.

The DP Application Layer offers the following ASEs (see Figure 8):

I/O Data ASE
The I/O Data ASE provides a set of services to convey I/O data cyclically. These data always belong to those modules that have been configured in terms of the Context ASE. A multicast service provides the synchronization of the I/O data and reports the operational state of the assigned DP-master. Optionally the I/O Data ASE offers the possibility to share the Input Data of one DP-slave with other DP-slaves.

Process Data ASE
The Process Data ASE provides a set of services to convey acyclically data. The application of the DP-master requests each transmission individually. The Process Data ASE can be related to all APs of a DP-slave.

Diagnosis ASE
The Diagnosis ASE provides services for the DP-master to fetch Diagnosis information from a DP-slave which is related to the Default AP or to its modules that are part of the configuration.

Alarm ASE
The Alarm ASE provides a set of services to convey alarms or status messages issued by the DP-slave. The assigned DP-master acknowledges the Alarms. Status messages will not have to be acknowledged by the DP-master.

Management ASE
The Management ASE provides a set of services to manage a DP-master (Class 1) by a DP-master (Class 2). The DP-master (Class 1) receives the bus parameter set or the slave parameter set. Afterwards the received sets will be activated by the DP-master (Class 2). The DP-master (Class 1) will provide the system diagnosis information if requested.

Context ASE
The Context ASE provides a set of services to

- parameterize and configure the Default AP itself and its modules containing I/O data,
- transfer switchover commands for redundant structures and commands to reset MS1 or change operation mode to the DP-slave,
- set the device address of a DP-slave,
- to provide the real configuration of the DP-slave to be read by a DP-master (Class 2),

and to establish or release an association between an individual AP of a DP-master (Class 2) and an individual AP of the DP-slave for acyclic communication. This association exclusively belongs to the Process Data ASE.

Load Region ASE
The Load Region ASE provides a set of services to convey LR Data acyclically. The application of the DP-master requests each transmission individually. The Load Region ASE can be related to all APs of a DP-slave.

Function Invocation ASE
The Function Invocation ASE provides a set of services to invoke stateless and/or state-oriented Function Invocation objects. The application of the DP-master has to invoke each function individually. The Function Invocation ASE can be related to all APs of a DP-slave.

Time ASE
The Time ASE provides a set of services to synchronies the clocks of several or all devices on a fieldbus segment. The application of the DP-master (Class 1) requests each synchronization individually. The Time ASE can be related to the Default AP of a DP-slave.

Application Relationship ASE
The Application Relationship ASE (AR ASE) provides a description model for the separate AR types. This includes their transfer characteristics as well as their current communication states.

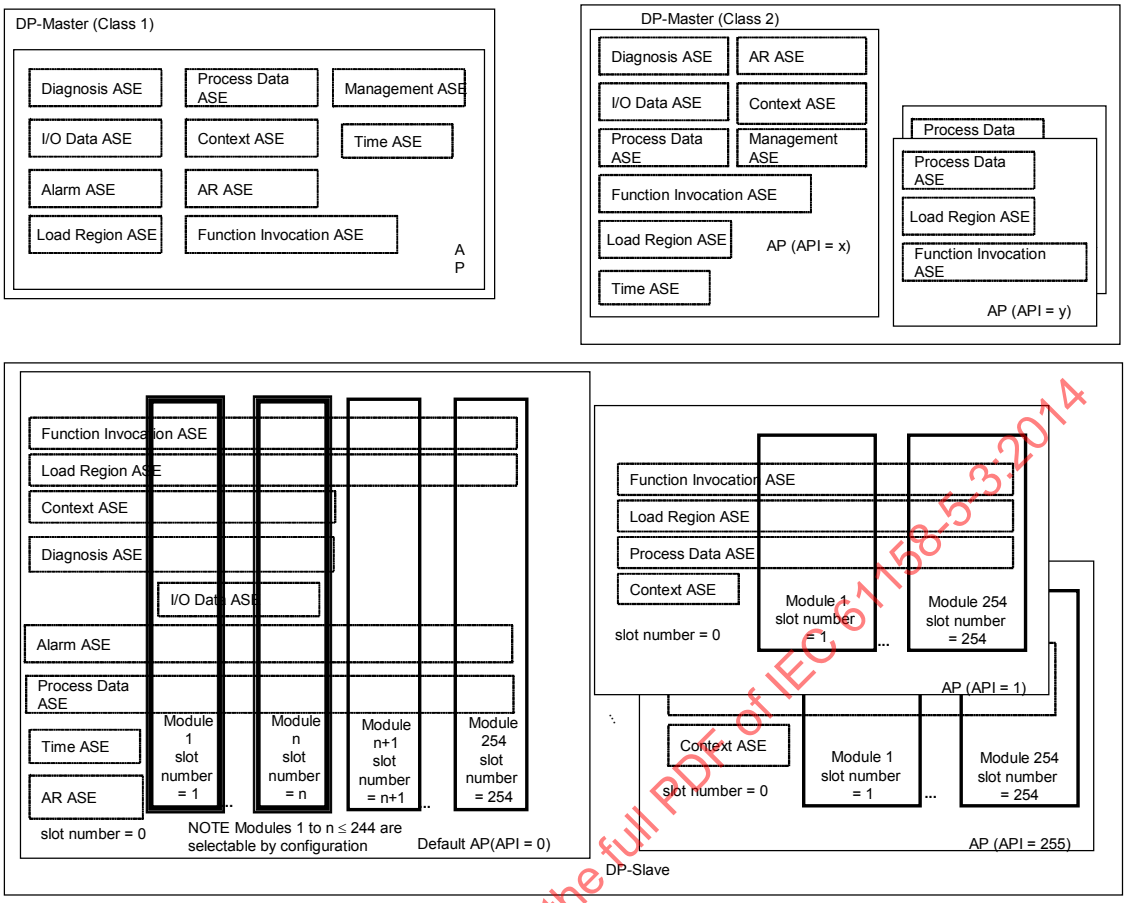


Figure 8 – Application Service Elements (ASEs)

6.1.3.4.3 Application objects

An application object (APO) is a network representation of a specific aspect of an AP. Each APO represents a set of information and processing capabilities of an AP that are accessible through services of the DP Application Layer. APOs are used to represent these capabilities to other APs in a fieldbus DP system.

In order to permit an AP to communicate with an AP of another device, APOs have to be available. Application objects (virtual objects) represent existing process objects (real objects), that an application process has made visible and accessible to the communication (see Figure 9).

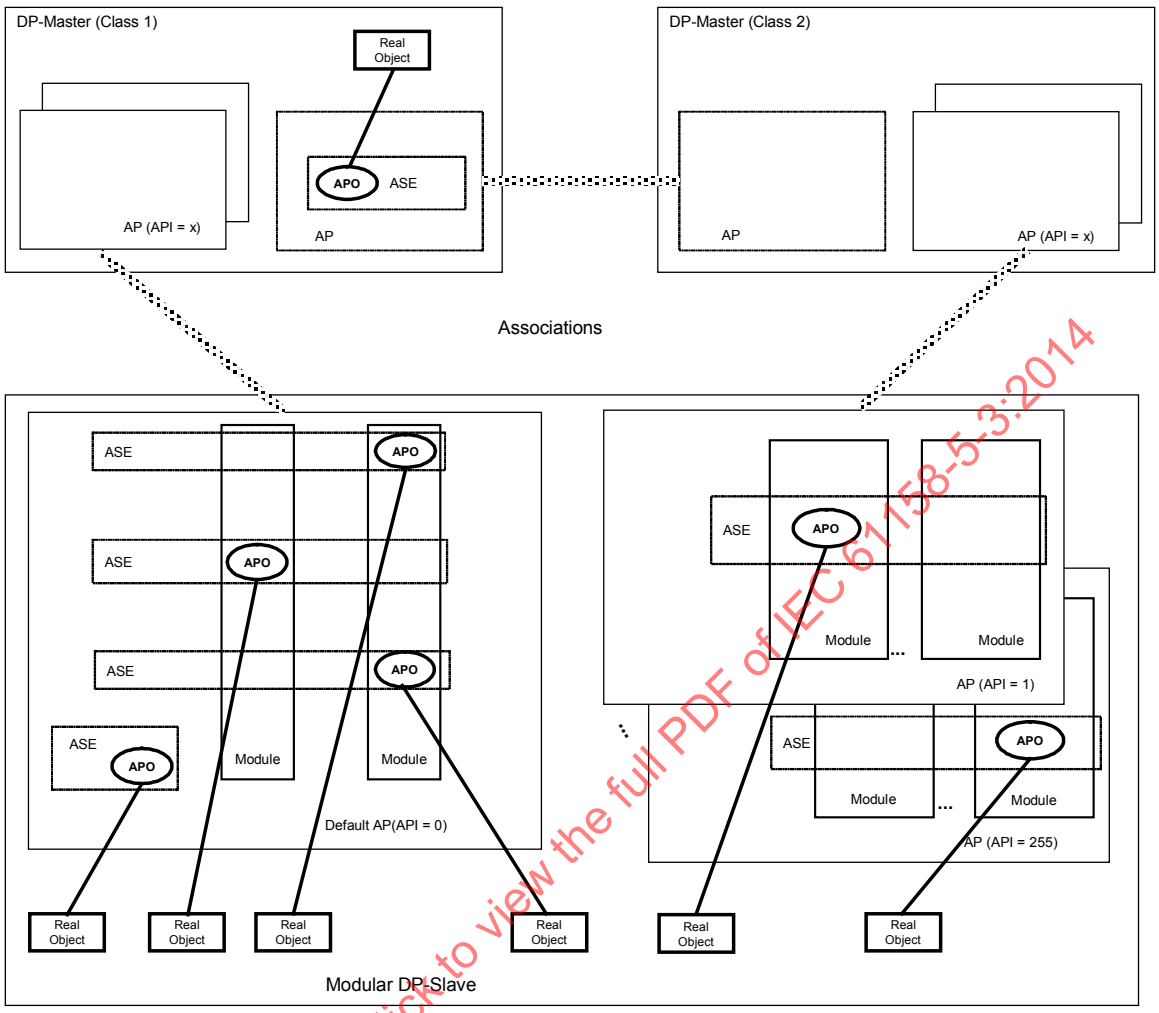


Figure 9 – Application Process with application Objects (APOs)

In Figure 10, a remote AP acting as Client may access the real object by sending requests through the APO that represents the real object. Local aspects of the AP convert between the network view (the APO) of the real object and the internal AP view of the real object.

Within the AP an APO is identified by slot and index. The address space which is defined by slot and index can be used by several APs.

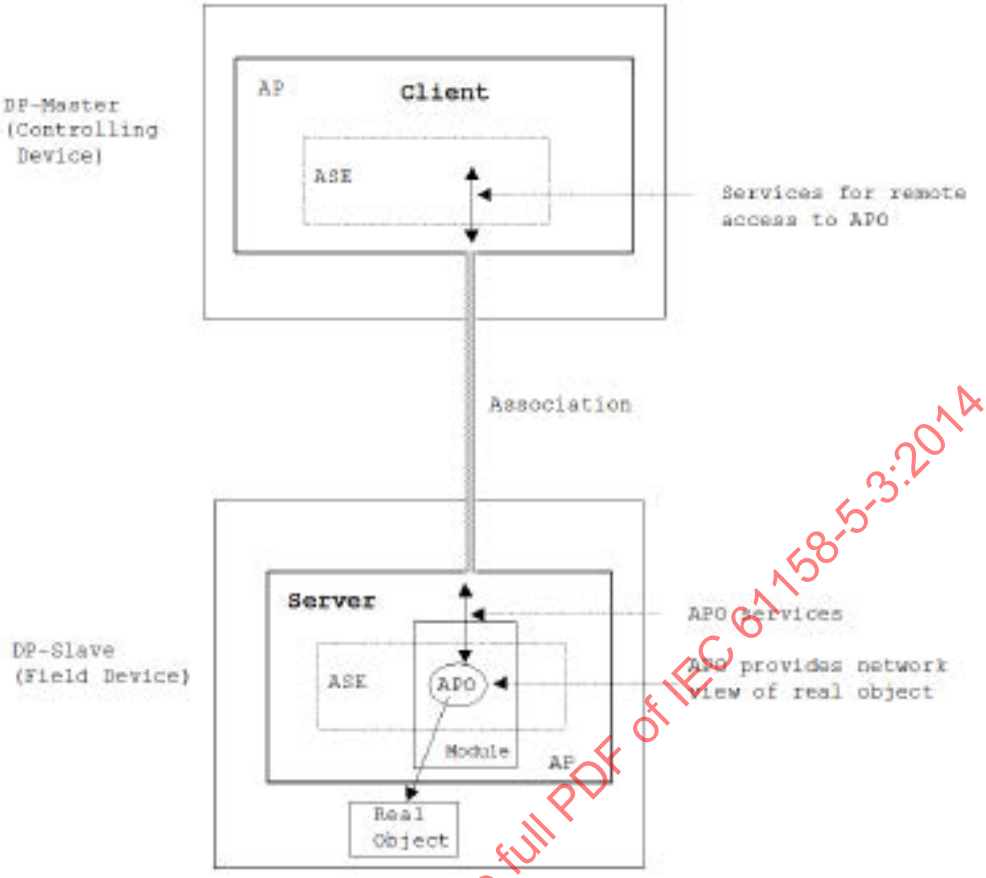


Figure 10 – Access to a remote APO

In Figure 11, a Client/Server and a Publisher/Subscriber association is shown. The Client may access the real object by sending requests through the APO that represents the real object. The Subscriber may subscribe to the whole or to a part of the remote APO. The Publisher/Subscriber association is always related to the Default AP and the I/O Data ASE. Local aspects of the AP convert between the network view (the APO) of the real object and the internal AP view of the real object.

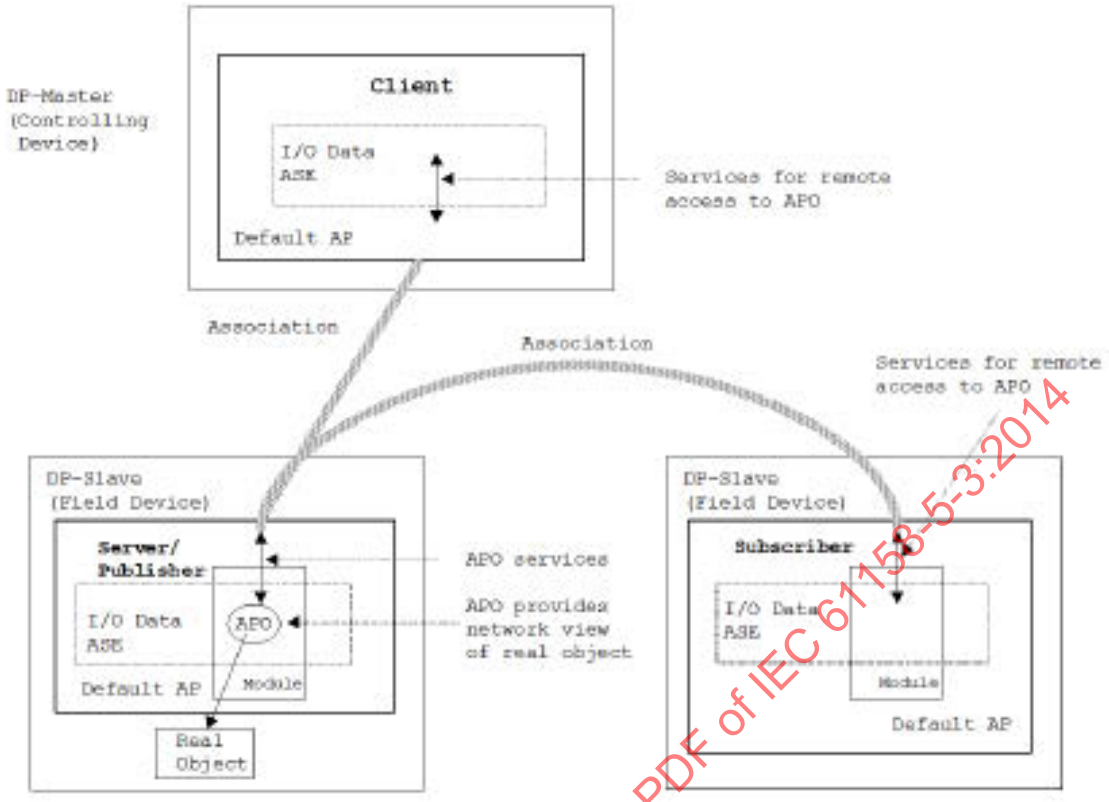


Figure 11 – Access to a remote APO for publisher/subscriber association

6.1.3.5 Application relationship

6.1.3.5.1 General

An Application Relationship (AR) is a co-operative relationship between two or more APs for the purpose of exchange of information and co-ordination of their joint operation (see Figure 12). This relationship is activated by the exchange of Application Protocol Data Units (APDU). Fieldbus DP uses different types of ARs which distinguish in their conveyance characteristic.

6.1.3.5.2 Application relationship endpoint

An AP has access to the communication using application relationship endpoints (AREP, see ISO/IEC 7498-3). One or more AREPs are fixed and uniquely assigned to an AP. These endpoints are addressed by the AP through an identifier (AREP ID). These identifiers are device specific and not defined by the communication itself. ARs are defined as a set of co-operating AREPs. Between two APs one or more ARs may exist, each one having unique AREPs. Figure 12 shows an example of one AR with two AREPs.

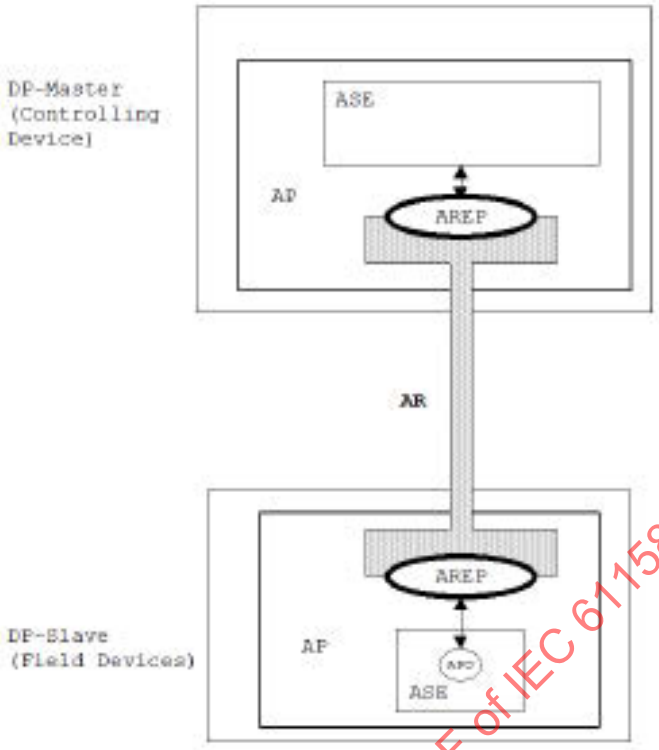


Figure 12 – Example of one AR with two AREPs

6.1.3.5.3 Overview of application relationships

The DP Application Layer offers the following AR types:

MS0: application relationship between the application process of one DP-master (Class 1) and all related DP-slaves and optional between the AP of one or several DP-master (Class 2) and all related DP-slaves and optional between the AP of one or several DP-slaves with all related DP-slaves for the following purposes:

– cyclic exchange of the I/O data with the DP-master (Class 1),
– cyclic exchange of the Input Data between DP-slaves (DXB),
– acyclic data transfer for parameterization, configuration and diagnosis (DP-master (Class 1))
– acyclic transfer of commands to a set of field devices (DP-master (Class 1)),
– cyclic transfer of synchronization messages to a set of field devices (DP-master (Class 1)),
– acyclic read of the I/O data (DP-master (Class 2)),
– acyclic read of configuration information (DP-master (Class 2)),
– acyclic read of diagnosis information (DP-master (Class 2)),
– acyclic write of remanent parameter (DP-master (Class 2)).

MS1: connection-oriented relationship between the application processes of one DP-master (Class 1) and of one related DP-slave for the following purposes:

– acyclic read and write of variables,
– acyclic transfer of alarms,
– up- and/or download of LR Data,

– invocation of stateless and/or state-oriented functions.

MS2: connection-oriented relationship between the application processes of one DP-master (Class 2) and of one related DP-slave for the purpose:

– acyclic read and write of variables,
– up- and/or download of LR Data,
– invocation of stateless and/or state-oriented functions.

MS3: connectionless relationship between the application processes of one DP-master and of a set of related DP-slaves for the purpose:

– synchronisation of time.

MM1: connectionless relationship between the application processes of one configuration device (Master Class 2) and of one related controlling device (Master Class 1) for the following purposes:

– upload and download for configuration information,
– upload of diagnosis information,
– activation of the previously transferred configuration.

MM2: connectionless relationship between the application processes of one configuration device (Master Class 2) and of a set of related controlling devices (Master Class 1) for the purpose:

– activation of the previously transferred configurations.

6.1.4 Dynamical behavior of fieldbus DP

Before commissioning a DP system, all stations shall have been assigned a unique address. In the case of DP-slaves, this address may be set up using the MS0 AR (see 6.2.5.3.9).

All DP-slaves that have not yet been assigned an individual address, shall start with the default address 126. Only one device with this address shall be allowed on the network. A DP-master address shall not be set to the default address. A DP-master (Class 2) may access a DP-slave via the default address and may assign a specific address to that device.

For security reasons, the DP-master (Class 1) shall not exchange I/O data with a DP-slave whose address is 126.

To exchange I/O data a DP-master (Class 1) shall have a valid Master Parameter Set which may be loaded and activated by the DP-master (Class 2) via the MM1 AR and MM2 AR. The Master Parameter Set consists of several sets of configuration data and parameters, each dedicated to a DP-slave. This set of data shall contain all necessary information (for the DP system) to describe a DP-slave. Additionally, the Master Parameter Set shall include the bus parameters as well as the address assignment table which assigns each individual remote I/O data a unique address in the I/O space of the DP-master's User.

If there exists a valid Master Parameter Set in a DP-master (Class 1), this device starts to check whether the DP-slaves dedicated to that DP-master are present or not (this will be done with a read of the diagnosis of a DP-slave). After an appropriate answer the DP-master (Class 1) will set the parameters and check the configuration of that DP-slave. A DP-slave will accept a check configuration request only from the Master who sets the parameters.

After submitting these two functions correctly, a DP-master (Class 1) checks the status of the DP-slave by reading the Diagnosis information. If the configuration check was successful and parameterization was correct, the DP-master (Class 1) will enter the user data exchange mode. A DP-slave accepts data exchange requests only from a DP-master (Class 1) which

has previously submitted parameters and configuration. If a diagnosis event occurred in the application process of a DP-slave, this event is signaled by sending a response with high priority. The DP-master (Class 1) will read the diagnosis and inform its local User.

The DP-master (Class 1) sends information about his own operation mode (CLEAR, OPERATE) to his dedicated DP-slaves for synchronization. This will be done at every change of the operation mode of the DP-master (Class 1) and additionally at regular intervals (Data Control Time/2).

Only in the data exchange mode the extended functionalities like the acyclic read and write of variables, the acyclic transfer of alarms, the up- and/or download of LR Data, the invocation of stateless and/or state-oriented functions, the DXB functionality and the Isochronous Mode are possible if supported by both the DP-master (Class 1) and the DP-slave. Errors on the acyclic communication (MS1 AR) also influence the cyclic data transfer (MS0 AR) and vice versa. This means that both communication relationships will be aborted. Each alarm has to be acknowledged by the assigned DP-master (Class 1).

Every DP-master (Class 2) can read the diagnosis information, the configuration, the input and output data (via the MS0 AR) of every DP-slave although the DP-slave is under control of a DP-master (Class 1). A DP-master (Class 2) can take a DP-slave under its control. In this case the DP-slave will stop the data exchange mode with the DP-master (Class 1). In this case the DP-master (Class 1) starts to read the diagnosis cyclically. If the DP-master (Class 2) has finished its communication with this DP-slave, the DP-slave will set the field "Master Add(ress)" to invalid. This event triggers the DP-master (Class 1) to gain control over this DP-slave again. As described above it will send parameters and configuration first.

If supported by both the DP-master (Class 2) and the DP-slave the extended functionality like the acyclic read and write of variables, the up- and/or download of LR Data and the invocation of stateless and/or state-oriented functions is possible (via the MS2 AR). The MS2 AR is independent of the MS0 AR and MS1 AR.

Fieldbus DP defines a redundancy model with the following features:

– One slave redundancy model for all redundancy structures (even for non-redundant systems)
– Scalability: Independent master, slave, and line redundancy
– Easy engineering: No additional user efforts, no complex tools necessary
– Complete monitoring of all components
– No influence on bus load and timing
– High reliability
– Short switch over time
– No loss of data during fault tolerance

A redundant DP-slave may consist of two communication interfaces with special redundancy extensions and an independent redundancy communication channel between these two communication interfaces. One of the slave interface modules acts as the primary and one as the backup slave.

The redundancy model of fieldbus DP is based on a switchover command, commands to stop, start and reset the MS1 AR, provided by the MS0 AR via the Check User Prm service to the application of the DP-slave and a Diagnosis object (Red Status) which provides information about the state of the redundant DP-slave if it is primary or backup and if a previous provided command (switchover, start, stop reset of the MS1 AR) has been processed.

During the processing of a switchover command or a stop, start or reset command the MS0 AR remains in the data exchange mode. Only the MS1 AR will be influenced, therefore a seamless takeover of a DP-slave may be accomplished.

The communication with a redundant DP-slave will be done in the following way:

MS0 AR:

The primary slave uses the MS0 AR (cyclic communication) to send and receive the I/O data. In addition the primary slave sends all diagnosis information about itself and the backup slave.

The backup slave may also send and receive values for the I/O data via the MS0 AR, but these values are not relevant and shall be ignored by the backup slave. The backup slave may transmit its own diagnosis information, but, it shall also be part of the primary diagnosis.

MS1 AR:

Only the primary slave will receive and transmit MS1 AR services. When the backup slave receives a MS1 AR service, it shall respond negatively.

MS2 AR:

Both the primary and the backup slave may receive acyclic MS2 AR services. Both slave interface modules shall process these services to allow an individual communication with the specific device. If values will be sent with MS2 AR services, these data may be sent to one slave interface module only and shall be exchanged with the other using the redundancy communication channel between the two communication interfaces. Only data, which will be held in both slave interface modules, shall be exchanged to keep them consistent.

The features for redundancy could be used for systems with backup masters as well and provide means for continuous cyclic operation.

6.1.5 Fieldbus DP status values

Table 2 shows the Status values of the service primitives.

Table 2 – Status values of the service primitives

Short Name	Meaning
ABT_FE	Invalid request APDU received
ABT_IA	Invalid additional address information
ABT_IV	Invalid service from the User
ABT_OC	Waiting for DL-DATA-REPLY.cnf
ABT_RE	Invalid response APDU received
ABT_SE	Sequence error; service not allowed in this state
ABT_STO	Send Timeout requested was too small
ABT_TO	Timeout of the connection
AD	Access denied
DH	Positive acknowledgement for sent data, reply data with high priority available
DI	Data Incomplete
DS	Local-DL/PHY Entity is not in logical token-ring or disconnected from line
EA	Area too large (Up-/Download)
FE	Format-Error in a Request-APDU
GE	Group Error, invalid group at service request (Global Control)
IP	Invalid Parameter
IV	Invalid parameters in request
LE	Data-block-length too large (Up-/Download)
LR	Resources local not available or not sufficient
LS	Service at local DLSAP or local DLSAP not activated.
NA	Negative ack, no reaction from remote station
NC	Master Parameter Set not compatible
NE	Area non-existent
NI	Service not implemented
NO	Service in this state not possible
NR	No Response data
RDH	Response data high and no resource for sent data
RDL	Response data low and no resource for sent data
RE	Format-Error in a Response-APDU
RR	Resources of the remote-DL Entity not sufficient or not available
RS	Service or remote-address at remote-DLSAP or remote-DLSAP not activated - remote-station is no DP-Station - remote-station is not yet ready for these Service - remote-station is associated with another Requester - optional service not available
SC	Sequence Conflict
SV	Sequence violation (second Time Master active)
SE	Sequence Error
TO	Function-Timeout expired
UE	Remote-DMPM / DL interface error

6.2 ASEs

6.2.1 Process data ASE

6.2.1.1 Overview

In the DP Application Layer environment, application processes contain data that remote applications are able to read and write. The Process Data ASE defines attributes of general purpose Process Data objects and provides a set of services used to read, and write their values. A Process Data ASE is specific for an AP and every AP may contain one Process Data ASE. Process Data objects within an AP are addressed by Slot Numbers and Indices. The Process Data object can be read and written partially or entirely.

The DP-masters (Class 1) and the DP-master (Class 2) are able to read or write the value of the Process Data object within the Default AP of a DP-slave. Process Data objects outside the Default AP of a DP-slave can only be accessed by the DP-master (Class 2).

The DP-master (Class 2) has to use the Context ASE (see 6.2.5.3.10) to establish an association to get access to the Process Data objects of a DP-slave. The Process Data object in combination with the Context ASE provides the possibility of access protection (see 6.2.1.3).

The DP-master (Class 1) can access the values of the Process Data object within the Default AP after the DP-slave has entered the data exchange mode. This association provides no access protection for the accessible Process Data object.

The access to Process Data objects is performed according to the Client/Server access model. The Client/Server model is characterized by a Client application sending a read or write request to a Server application that responds accordingly. The Server's activity is stimulated by Clients on the network.

The formal model of the Process Data ASE is presented by the Process Data class specification, containing a description of its attributes, services as well as invocations, followed by a detailed service specification.

6.2.1.2 Process data class specification

6.2.1.2.1 General

The Process Data ASE defines two Process Data object types:

– Simple objects,
– Combined objects.

Simple objects may have a one to one relation to real objects. Combined objects may have a one to two relation to real objects. Combined objects may be read and written with one service interaction. This type of objects may be used for communication which requires consistency between read and write or high protocol efficiency.

6.2.1.2.2 Simple process data class specification

6.2.1.2.2.1 Template

A Simple Process Data object is always related to one real object as shown in Figure 13.

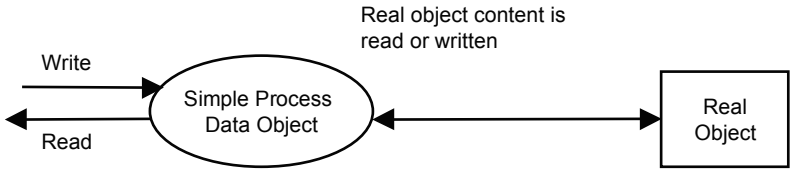


Figure 13 – Relation of a simple process data object to the real object

A Process Data object is described by the following template:

DP ASE:			**Process Data ASE**
CLASS:			**Simple Process Data**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attributes:	Identifier
2.	(m)	Attribute:	Process Data Description
3.	(m)	Attribute:	Partial Access
4.	(m)	Attribute:	Access Rights MS1
5.	(m)	Attribute:	Access Rights MS2
6.	(c)	Constraint:	Access Rights MS2 = Right to Read or Right to Read/Write
6.1.	(m)	Attribute:	List of SCL Read
6.1.1.	(m)	Attribute:	SCL
7.	(c)	Constraint:	Access Rights MS2 = Right to Write or Right to Read/Write
7.1.	(m)	Attribute:	List of SCL Write
7.1.1.	(m)	Attribute:	SCL
8.	(o)	Attribute:	Process Data Name
9.	(o)	Attribute:	Local Detail
SERVICES:			
1.	(o)	OpsService:	Read
2.	(o)	OpsService:	Write

6.2.1.2.2.2 Attributes

Identifier

This key attribute is a triplet composed of API, Slot Number, and Index to define to which Application Process Identifier, module and to which index within the specified module the Simple Process Data object belongs. This Identifier shall not be used by another Action object, Function Invocation object, Load Region object or Process Data object.

Attribute Type: Unsigned8, Unsigned8, Unsigned8

Allowed triplet values: [0 to 254, 0 to 254, 0 to 254]

Process Data Description

This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Partial Access

This Boolean attribute defines whether a partial access to the Simple Process Data object is supported or not. Partial access equal TRUE means that the content of the Simple Process Data object is read or written from the first octet up to the octet defined by the length given in the service. The length given in the service shall be less or equal the value of the attribute length in case of a write access. For a read access the entire object will be read if the length given in the service is greater than the length of the object. Partial access equal FALSE means that the content of the Simple Process Data object can only be read or written entirely. That implies that the length given in the service shall be equal to or (in case of read access) larger than the length of the object.

Attribute Type: Boolean

Access Rights MS1
This attribute defines the type of access rights defined for a MS1 AR for the Simple Process Data object.

The allowed values are shown in Table 3.

Table 3 – Access Rights MS1

Value	Meaning
0	Right to Write
1	Right to Read
2	Right to Read/Write
3	No Access

Access Rights MS2
This attribute defines the type of access rights defined for a MS2 AR for the Simple Process Data object.

The allowed values are shown in Table 4.

Table 4 – Access Rights MS2

Value	Meaning
0	Right to Write
1	Right to Read
2	Right to Read/Write
3	No Access

List of SCL Read
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one octet value for read access protection to this Simple Process Data object. The rules for accepting or rejecting a read access are described in 6.2.1.3.

Attribute Type: Unsigned8

List of SCL Write
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one octet value for write access protection to this Simple Process Data object. The rules for accepting or rejecting a write access are described in 6.2.1.3.

Attribute Type: Unsigned8

Process Data Name
This optional attribute defines the name of the Process Data object.

Attribute Type: Visible String(32)

Local Detail
This optional attribute contains additional local information about the Simple Process Data object.

Attribute Type: Octet String

6.2.1.2.2.3 Invocation of the simple process data object

For the invocation of the Simple Process Data object the following rules apply:

– Simple Process Data objects shall not exceed the total length of 240 octets. The attribute Process Data description has to be set accordingly.
– At least one service shall be allowed for the access of the object.
– At least one AR shall have access to the object.
– The access rights have to be set accordingly to the allowed service(s).

6.2.1.2.3 Combined process data class specification

6.2.1.2.3.1 Template

A Combined Process Data object is always related to two real objects as shown in Figure 14.

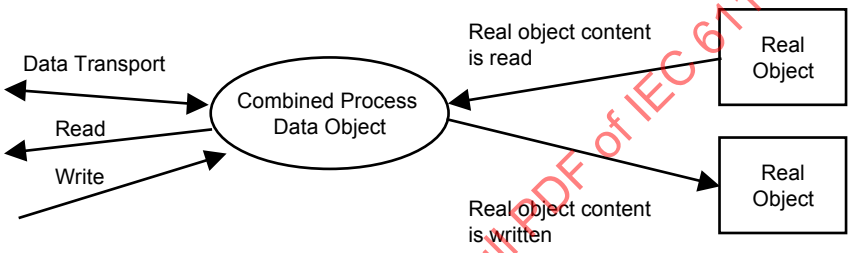


Figure 14 – Relation of a combined process data object to the real objects

A Combined Process Data object is described by the following template:

DP ASE:			**Process Data ASE**
CLASS:			**Combined Process Data**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attributes:	Identifier
2.	(m)	Attribute:	Read Process Data Description
3.	(m)	Attribute:	Read Partial Access
4.	(c)	Constraint:	Access Rights MS2= Right to Read or Right to Read/Write
4.1.	(m)	Attribute:	List of SCL Read
4.1.1.	(m)	Attribute:	SCL
5.	(m)	Attribute:	Write Process Data Description
6.	(m)	Attribute:	Write Partial Access
7.	(c)	Constraint:	Access Rights MS2 = Right to Write or Right to Read/Write
7.1.	(m)	Attribute:	List of SCL Write
7.1.1.	(m)	Attribute:	SCL
8.	(m)	Attribute:	Access Rights MS1
9.	(m)	Attribute:	Access Rights MS2
10.	(o)	Attribute:	Process Data Name
11.	(o)	Attribute:	Local Detail
SERVICES:			
1.	(o)	OpsService:	Data Transport
2.	(o)	OpsService:	Read
3.	(o)	OpsService:	Write

6.2.1.2.3.2 Attributes

Identifier
This key attribute is a triplet composed of API, Slot Number, and Index to define to which Application Process Identifier, module and index within the specified module the Combined Process Data object belongs. This Identifier shall not be used by another Action object, Function Invocation object, Load Region object or Process Data object.

Attribute Type: Unsigned8, Unsigned8, Unsigned8

Allowed triplet values: [0 to 254, 0 to 254, 0 to 254]

Read Process Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. of the Process Data ASE. For data description the template given in IEC 61158-5-10, 5.1 is used.

Read Partial Access
This Boolean attribute defines whether a partial read access to the Combined Process Data object is supported or not. Partial access equal TRUE means that the content of the Combined Process Data object is read from the first octet up to the octet defined by the length given in the service. The entire object will be read if the length given in the service is greater than the length of the object. Partial access equal FALSE means that the content of the Combined Process Data object can only be read entirely. That implies that the length given in the service shall be equal to or larger than the length of the object.

Attribute Type: Boolean

List of SCL Read
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one octet value for read access protection to this Combined Process Data object. The rules for accepting or rejecting a read access are described in 6.2.1.3.

Attribute Type: Unsigned8

Write Process Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Write Partial Access
This Boolean attribute defines whether a partial write access to the Combined Process Data object is supported or not. Partial access equal TRUE means that the content of the Combined Process Data object is written from the first octet up to the octet defined by the length given in the service. The length given in the service shall be less or equal the value of the attribute length. Partial access equal FALSE means that the content of the Combined Process Data object can only be written entirely. That implies that the length given in the service shall be equal to the length of the object.

Attribute Type: Boolean

List of SCL Write
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one octet value for write access protection to this Combined Process Data object. The rules for accepting or rejecting a write access are described in 6.2.1.3.

Attribute Type: Unsigned8

Access Rights MS1
This attribute defines the type of access rights defined for a MS1 AR for the Combined Process Data object.

The allowed values are shown in Table 5.

Table 5 – Access Rights MS1

Value	Meaning
0	Right to Write
1	Right to Read
2	Right to Read/Write
3	No Access

Access Rights MS2
This attribute defines the type of access rights defined for a MS2 AR for the Combined Process Data object.

The allowed values are shown in Table 6.

Table 6 – Access Rights MS2

Value	Meaning
0	Right to Write
1	Right to Read
2	Right to Read/Write
3	No Access

Process Data Name
This optional attribute defines the name of the Process Data object.

Attribute Type: Visible String(32)

Local Detail
The attribute local detail contains additional local information about the Simple Process Data object.
Attribute Type: Octet String

6.2.1.2.3.3 Invocation of the combined process data object

For the invocation of the Combined Process Data object the following rules apply:

– Combined Process Data objects shall not exceed the total length of 240 octets for both Read Process Data and Write Process Data. The attributes Read Data Description and Write Data Description have to be set accordingly.
– At least one service shall be allowed for the access of the object.
– At least one AR shall have access to the object.
– The access rights have to be set accordingly to the allowed optional service(s).

6.2.1.3 Access protection on process data objects

The Process Data ASE in combination with the Context ASE provides also access protection. This is not a protection against intentional misuse of the communication facilities of a field device but helps to protect a system for accidental erroneous use of Process Data.

The protection is accomplished by two mechanisms:

– the Initiate service will set up a security level for this CR.

– the Process Data object will have a list of SCL values for the read and write service. An access will be accepted only if the SCL of the initiate service matches (see Table 7) with an element of the SCL list for this Process Data object.

This allows the User to select a level for read and write objects. Thus a service device can use a SCL which allows access to almost all objects. A text display may have only read access to specific objects.

Table 7 – SCL matching rules

SCL in Initiate.ind	SCL in Process Data object	Condition	Result
= 0	= 0	-	access allowed
≠ 0	= 0	-	access allowed
= 0	≠ 0	-	access denied
≠ 0	≠ 0	SCL in Initiate.ind not contained in List of SCL at Process Data object	access denied
≠ 0	≠ 0	SCL in Initiate.ind contained in List of SCL at Process Data object	access allowed

6.2.1.4 Process data service specification

6.2.1.4.1 Read

This confirmed service may be used to read the value of a Process Data object. This service shall only be used in conjunction with the MS1 or MS2 AR. Table 8 shows the parameters of the service.

Table 8 – Read

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Index	M	M(=)		
Length	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Length			M	M(=)
Data			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Process Data object in a specific slot (typically a module).

Index
The parameter Index is used in the destination device for addressing the desired Process Data object.

Length
The parameter Length indicates the number of octets of a Process Data object that has to be read.

Result(+)
This selection type parameter indicates that the service request succeeded.

Data
The parameter Data contains the value of the object which has been read and consists of the number of octets indicated in the Length of the response. This parameter shall be composed of the data types defined in Clause 5.

Result(-)
This selection type parameter indicates that the service request failed.

Error Decode
This parameter selects one of the following Error schemes: DPV1, FMS, Profile_Specific.

Error Code 1
In case of Error Decode equal DPV1 the Error Code 1 assumes one of the following values:
read error, module failure, version conflict, feature not supported, user specific, invalid index, invalid slot, type conflict, invalid area, state conflict, access denied, invalid range, invalid parameter, invalid type, read constrain conflict, resource busy, resource unavailable.

Error Code 2
The parameter Error Code 2 is user specific.

6.2.1.4.2 Write

This confirmed service may be used to write the value of a Process Data object. This service shall only be used in conjunction with the MS1 or MS2 AR. Table 9 shows the parameters of the service.

Table 9 – Write

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Index	M	M(=)		
Length	M	M(=)		
Data	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Length			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Process Data object in a specific slot (typically a module).

Index
The parameter Index is used in the destination device for addressing the desired Process Data object.

Length
The parameter Length indicates the number of octets of a Process Data object which has to be written.

Data
The parameter Data contains the value of the Process Data object which has to be written and consists of the number of octets indicated in the Length of the request. This parameter has to be composed of the data types defined in Clause 5.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Decode
This parameter selects one of the following Error schemes: DPV1, FMS, Profile_Specific.

Error Code 1
In case of Error Decode equal DPV1 the Error Code 1 assumes one of the following values:
write error, module failure, version conflict, feature not supported, user specific, invalid index, write length error, invalid slot, type conflict, invalid area, state conflict, access

denied, invalid range, invalid parameter, invalid type, write constrain conflict, resource busy, resource unavailable.

Error Code 2
The parameter Error Code 2 is user specific.

6.2.1.4.3 Data transport

This service is used to write and read the values of a single Combined Process Data object. This service may be used only via MS2 AR. Table 10 shows the parameters of the service.

With this service the data of a Process Data object are transferred

– from the DP-master to the DP-slave,
– from the DP-slave to the DP-master (Class 2), or
– from the DP-master to the DP-slave and back.

Table 10 – Data transport

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Index	M	M(=)		
Length	M	M(=)		
Data	C	C(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Length			M	M(=)
Data			C	C(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired slot (typically a module).

Index
The parameter Index is used in the destination device to address the desired data.

Length
In the Request the parameter Length indicates the number of octets of the parameter "Data" that are transferred from the Client to the Server. In the Response the parameter

Length indicates the number of octets of the parameter "Data" that are transferred from the Server to the Client.

Data
The parameter Data contains the values transferred from the Client to the Server and vice versa and consists of the number of octets indicated in the parameter Length. This parameter shall be composed of the data types defined in Clause 5.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Decode
This parameter selects one of the following Error schemes: DPV1, FMS, Profile_Specific.

Error Code 1
In case of Error Decode equal DPV1 the Error Code 1 assumes one of the following values:
read error, write error, module failure, version conflict, feature not supported, user specific, invalid index, write length error, invalid slot, type conflict, invalid area, state conflict, access denied, invalid range, invalid parameter, invalid type, read constrain conflict, write constrain conflict, resource busy, resource unavailable.

Error Code 2
The parameter Error Code 2 is user specific.

6.2.2 I/O data ASE

6.2.2.1 Overview

In the DP Application Layer environment, the default application process of a DP-slave may contain I/O Data objects that shall be cyclically conveyed over the network. Therefore the I/O Data ASE defines attributes of the I/O Data objects and provides a set of services used to get and set the buffer to transport their values. The I/O Data objects are separated in an Input Data object and an Output Data object.

An Input Data object or an Output Data object shall be partitioned in Input Data or Output Data Elements which represent the I/O Data of the modules. The remote Input Data subscribed to in combination with the Output Data represent the real output data of the DP-slave.

Additional services are provided to read acyclically the values of the I/O Data objects and to indicate new values for the Input Data object and the Output Data object. Furthermore services are provided to indicate the presence and to get the content of new subscribed Input Data.

The I/O Data objects are implicitly addressed through the related services. The granularity of input or output data in a Server/Publisher is according to the correspondent MS0 Configuration Elements object. A Client is always related to the whole Input Data and/or Output Data of a Server. A Subscriber may be related to the whole Input Data or a part of the Input Data of a Publisher according to the attribute values of the DXB-Linktable object.

The I/O Data ASE uses the Client/Server and Publisher/Subscriber access model. The Client/Server model is characterized by a Client application conveying the value for the Output Data object to the Server's buffer. The Server application fetches this value by a Get Output service. The receipt of a new value is indicated by the New Output service. The Server application conveys the value of its Input Data object via the Set Input service to the Server's buffer as soon as one of the real object values changes. The Client application gets this value

by the Get Input service. The transmission of the Server buffer to the Client buffer and vice versa is done asynchronously to the Get Output and Set Input services.

The Publisher/Subscriber model is characterized by DP-slaves acting as Publisher which publish the values of the Input Data objects and Subscribers which subscribe one or all values of the Input Data object of one or several Publisher (DP-slaves). The Publisher application conveys the value of its Input Data object (Simple or Extended Input Data object) via the Set Input service to the Server's buffer as soon as one of the real object values changes. The Client (DP-master (Class 1)) causes the Publisher to publish the values of the Input Data object so that these values are available at the Client side and at every Subscriber on the network. The Subscriber subscribes to as many Input Data objects as the attribute values of the DXB-Linktable or DXB-Subscribertable indicate. The receipt of new values of subscribed Input Data is indicated by the New Publisher Data service at the Subscriber. The Subscriber application gets these values by the Get Publisher Data service.

It is possible to update only parts of the values of the I/O Data objects according to the given granularity regarding the structure of the objects. Furthermore, it is also possible that a DP-slave has only either an Input Data object or an Output Data object.

Most DP-slaves support only Octet or Word structures for the Input Data object and the Output Data object. This type of devices use the Simple Input and Output Data object. However, complex devices use Floating Point values or Doubleword structures or even combinations of several data types. The extended Input/Output Data object reflect the requirements of these enhanced devices.

The values of the I/O Data objects of all or a group of DP-slaves within a given network may be synchronized by its assigned DP-master with help of the Global Control service.

The synchronization of the values of the Input Data objects is achieved by the Freeze Command and the synchronization of the values of the Output Data objects is achieved by the Sync Command. This synchronization mechanism has no effect on the reception of the subscribed Input Data.

With each Freeze Command the current value of the Input Data object in the MS0 buffer is frozen in the addressed DP-slaves. If the Server application now updates the value of the Input Data object by the Set Input service then this value will not be issued to the buffer until the next Freeze Command occurs. That means, that the addressed DP-slaves will convey the "frozen" values of the Input Data objects. The synchronization of the Input Data objects will be switched off by a Freeze Command with the value Unfreeze.

With the first Sync Command the current value of the Output Data object in the MS0 buffer is frozen in the addressed DP-slaves. With each further Sync Command the previously transmitted values of the Output Data objects will be updated to the MS0 AR buffer for access with the Get Output service. The synchronization of the Output Data objects will be switched off by a Sync Command with the value Unsync. A detailed description is given in 6.2.8.3.2.

The values of the subscribed Input Data objects will not be synchronized with the values of the Output Data object in the Subscriber.

The DP-master (Class 1) sends information about its own operation mode to its assigned DP-slaves for synchronization. This will be done at every change of the operation mode of the DP-master (Class 1) or occurs in a time interval which can be parameterized. If the DP-master's state is CLEAR then the values of the Output Data object shall be set to a value corresponding to a safe state.

The values of the I/O Data objects of all or a group of DP-slaves within a given network may be synchronized by the assigned DP-master (Class 1) by a special synchronization mechanism (Isochronous Mode).

The synchronization of the values of the Input Data objects and the synchronization of the values of the Output Data objects is achieved by the synchronization message (SYNCH) and fixed time settings related to the synchronization message within the involved DP-slaves and the DP-master (Class 1).

The values of the subscribed Input Data objects in the Subscriber supporting the Isochronous Mode may also be synchronized by the Isochronous Mode.

The DP-master (Class 1) communicates via the MS0 AR with all Input Data and Output Data objects of the assigned DP-slaves simultaneously. Additionally the Client application may communicate with an Input Data object, an Output Data object, an Input Data Element or an Output Data Element of a single DP-slave.

The DP-master (Class 2) communicates via the MS0 AR with the Input Data and Output Data object of a single DP-slave.

A DP-slave, configured as Subscriber, communicates with the Input Data object(s) of one or more DP-slaves, configured as Publisher and to which he is subscribed to via the MS0 AR.

The mapping of the subscribed Input Data to all or parts of the Output Data Elements of the DP-slave may be done application specific or may be accomplished by the application of the DP-slave with the DXB-Subscribertable in conjunction with the configuration data.

The formal model of the I/O Data ASE is presented next, followed by a description of its services.

6.2.2.2 I/O data class specification

6.2.2.2.1 Simple input data class specification

6.2.2.2.1.1 Template

The Simple Input Data object is described by the following template:

DP ASE:			**I/O Data ASE**
CLASS:			**Simple Input Data**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of Input Data Elements
2.1.	(m)	Attribute:	Slot Number
2.2.	(m)	Attribute:	Simple Input Data Description
2.2.1.	(m)	Attribute:	Length
2.2.2.	(m)	Attribute:	Format
2.2.3.	(m)	Attribute:	Consistency
2.2.4.	(o)	Attribute:	Name
SERVICES:			
1.	(m)	OpsService:	Set Input
2.	(o)	OpsService:	Read Input
3.	(m)	OpsService:	Get Input
4.	(o)	OpsService:	New Input
5.	(m)	OpsService:	Get Publisher Data
6.	(o)	OpsService:	New Publisher Data
7.	(m)	OpsService:	Global Control
8.	(o)	OpsService:	SYNCH
9.	(o)	OpsService:	SYNCH Delayed
10.	(o)	OpsService:	DX Finished
11.	(o)	OpsService:	SYNCH Event

6.2.2.2.1.2 Attributes

Implicit
The attribute Implicit indicates that the Simple Input Data object is implicitly addressed by the services.

List of Input Data Elements
One Input Data Element is composed of the following list elements:

Slot Number
This attribute defines to which module the Input Data Element belongs.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Simple Input Data Description
This attribute consists of the following attributes.

Length
The attribute Length defines the number of octets of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

Attribute Type: Unsigned8

Allowed values: 1 to 128

Format
This attribute defines the format of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 11.

Table 11 – Format (simple input data description)

Value	Meaning
0	Octet
1	Word

Consistency
This attribute defines the Consistency of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 12.

Table 12 – Consistency (simple input data description)

Value	Meaning
0	Consistency over the defined Format of the Input Data Element
1	Consistency over the whole Input Data Element

Name
This attribute defines the name of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

Attribute Type: Visible String(32)

6.2.2.2.1.3 Invocation of the simple input data object

For the invocation of the Simple Input Data object the following rules apply:

— Simple Input Data objects shall not exceed the total length of 244 octets. The attributes List of Input Data Elements and Length have to be set accordingly.

— Only one Simple Input Data object shall be invoked in a DP-slave.

— A Simple Input Data object shall only be invoked if no Extended Input Data object (see 6.2.2.2.3) has been invoked.

6.2.2.2.2 Simple output data class specification

6.2.2.2.2.1 Template

The Simple Output Data object is described by the following template:

DP ASE:			**I/O Data ASE**
CLASS:			**Simple Output Data**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of Output Data Elements
2.1.	(m)	Attribute:	Slot Number
2.2.	(m)	Attribute:	Simple Output Data Description
2.2.1.	(m)	Attribute:	Length
2.2.2.	(m)	Attribute:	Format
2.2.3.	(m)	Attribute:	Consistency
2.2.4.	(o)	Attribute:	Name
SERVICES:			
1.	(m)	OpsService:	Set Output
2.	(o)	OpsService:	Read Output
3.	(m)	OpsService:	Get Output
4.	(o)	OpsService:	New Output
5.	(m)	OpsService:	Global Control
6.	(o)	OpsService:	SYNCH
7.	(o)	OpsService:	SYNCH Delayed
8.	(o)	OpsService:	DX Finished
9.	(o)	OpsService:	SYNCH Event

6.2.2.2.2.2 Attributes

Implicit
The attribute Implicit indicates that the Simple Output Data object is implicitly addressed by the services.

List of Output Data Elements
One Output Data Element is composed of the following list elements:

Slot Number
This attribute defines to which module the Output Data Element belongs.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Simple Output Data Description
This attribute consists of the following attributes.

Length
The attribute Length defines the number of octets of the Output Data Element. The value of this attribute is determined by the corresponding configuration data.

Attribute Type: Unsigned8

Allowed values: 1 to 128

Format
This attribute defines the format of the Output Data Element. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 13.

Table 13 – Format (simple output data)

Value	Meaning
0	Octet
1	Word

Consistency
This attribute defines the Consistency of the Output Data Element. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 14.

Table 14 – Consistency (simple output data)

Value	Meaning
0	Consistency over the defined Format of the Output Data Element
1	Consistency over the whole Output Data Element

Name
This optional attribute defines the name of the Output Data Element.

Attribute Type: Visible String(32)

6.2.2.2.2.3 Invocation of the simple output data object

For the invocation of the Simple Output Data object the following rules apply:

— Simple Output Data objects shall not exceed the total length of 244 octets. The attributes List of Output Data Elements and Length have to be set accordingly.

— Only one Simple Output Data object shall be invoked in a DP-slave.

— A Simple Output Data object shall only be invoked if no Extended Output Data object (see 6.2.2.2.4) has been invoked.

6.2.2.2.3 Extended input data class specification

6.2.2.2.3.1 Template

The Extended Input Data object is described by the following template:

DP ASE: **I/O Data ASE**
CLASS: **Extended Input Data**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of Input Data Elements
2.1.	(m)	Attribute:	Slot Number
2.2.	(m)	Attribute:	Selector of Data Type
2.2.1.	(s)	Attribute:	Extended Input Data Description
2.2.2.	(s)	Attribute:	Simple Input Data Description
2.2.2.1.	(m)	Attribute:	Length
2.2.2.2.	(m)	Attribute:	Format
2.2.2.3.	(m)	Attribute:	Consistency
2.3.	(o)	Attribute:	Name

SERVICES:

1.	(m)	OpsService:	Set Input
2.	(o)	OpsService:	Read Input
3.	(m)	OpsService:	Get Input
4.	(o)	OpsService:	New Input
5.	(m)	OpsService:	Get Publisher Data
6.	(o)	OpsService:	New Publisher Data
7.	(m)	OpsService:	Global Control
8.	(o)	OpsService:	SYNCH
9.	(o)	OpsService:	SYNCH Delayed
10.	(o)	OpsService:	DX Finished
11.	(o)	OpsService:	SYNCH Event

6.2.2.2.3.2 Attributes

Implicit

The attribute Implicit indicates that the Extended Input Data object is implicitly addressed by the services.

List of Input Data Elements
One Input Data Element is composed of the following list elements:

Slot Number
This attribute defines to which module the Input Data Element belongs.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Selector of Data Type
This attribute defines whether this module is described according to the Extended Input Data Description or according to the Simple Input Data Description. In case of an Extended Input Data Description, consistency is over the whole Input Data Element.

Extended Input Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Simple Input Data Description
This attribute consists of the following attributes.

Length
The attribute Length defines the number of octets of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

Attribute Type: Unsigned8

Allowed values: 1 to 128

Format
This attribute defines the format of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 15.

Table 15 – Format (extended input data)

Value	Meaning
0	Octet
1	Word

Consistency
This attribute defines the Consistency of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 16.

Table 16 – Consistency (extended input data)

Value	Meaning
0	Consistency over the defined Format of the Input Data Element
1	Consistency over the whole Input Data Element

Name
This optional attribute defines the name of the Input Data Element. The value of this attribute is determined by the corresponding configuration data.

Attribute Type: Visible String(32)

6.2.2.2.3.3 Invocation of the extended input data object

For the invocation of the Extended Input Data object the following rules apply:

— Extended Input Data objects shall not exceed the total length of 244 octets. The attributes List of Input Data Elements, Input Data Description respectively Format and Length have to be set accordingly.

— Only one Extended Input Data object shall be invoked in a DP-slave.

— An Extended Input Data object shall only be invoked if no Simple Input Data object has been invoked.

6.2.2.2.4 Extended output data class specification

The Extended Output Data object is described by the following template:

DP ASE: **I/O Data ASE**
CLASS: **Extended Output Data**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of Output Data Elements
2.1.	(m)	Attribute:	Slot Number
2.2.	(m)	Attribute:	Selector of Data Type
2.2.1.	(s)	Attribute:	Extended Output Data Description
2.2.2.	(s)	Attribute:	Simple Output Data Description
2.2.2.1.	(m)	Attribute:	Length
2.2.2.2.	(m)	Attribute:	Format
2.2.2.3.	(m)	Attribute:	Consistency
2.3.	(o)	Attribute:	Name

SERVICES:

1.	(m)	OpsService:	Set Output
2.	(o)	OpsService:	Read Output
3.	(m)	OpsService:	Get Output
4.	(o)	OpsService:	New Output
5.	(m)	OpsService:	Global Control
6.	(o)	OpsService:	SYNCH
7.	(o)	OpsService:	SYNCH Delayed
8.	(o)	OpsService:	DX Finished
9.	(o)	OpsService:	SYNCH Event

6.2.2.2.4.1 Attributes

Implicit
The attribute Implicit indicates that the Extended Output Data object is implicitly addressed by the services.

List of Output Data Elements
One Output Data Element is composed of the following list elements:

Slot Number
This attribute defines to which module the Output Data Element belongs.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Selector of Data Type
This attribute defines whether this module is described according to the Extended Output Data Description or according to the Simple Output Data Description. In case of an Extended Output Data Description, consistency is over the whole Output Data Element.

Extended Output Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Simple Output Data Description
This attribute consists of the following attributes.

Length
The attribute Length defines the number of octets of the Output Data Element. The value of this attribute is determined by the corresponding configuration data. The value of this attribute is determined by the corresponding configuration data.

Attribute Type: Unsigned8

Allowed values: 1 to 128

Format
This attribute defines the format of the Output Data Element. The value of this attribute is determined by the corresponding configuration data. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 17.

Table 17 – Format (extended output data)

Value	Meaning
0	Octet
1	Word

Consistency
This attribute defines the Consistency of the Output Data Element. The value of this attribute is determined by the corresponding configuration data. The value of this attribute is determined by the corresponding configuration data.

The allowed values are shown in Table 18.

Table 18 – Consistency (extended output data)

Value	Meaning
0	Consistency over the defined Format of the Output Data Element
1	Consistency over the whole Output Data Element

Name
This optional attribute defines the name of the Output Data Element. The value of this attribute is determined by the corresponding configuration data. The value of this attribute is determined by the corresponding configuration data.

Attribute Type: Visible String(32)

6.2.2.2.4.2 Invocation of the extended output data object

For the invocation of the Extended Output Data object the following rules apply:

– Extended Output Data objects shall not exceed the total length of 244 octets. The attributes List of Output Data Elements, Output Data Description respectively Format and Length have to be set accordingly.
– Only one Extended Output Data object shall be invoked in a DP-slave.
– An Extended Output Data object shall only be invoked if no Simple Output Data object has been invoked.

6.2.2.3 I/O data service specification

6.2.2.3.1 Set input

This service shall be used by a DP-slave to pass the value of a Simple Input Data object or an Extended Input Data object to the DP Application Layer. This service shall only be used in conjunction with the MS0 AR.

By this service the application process of the DP-slave updates the local buffer. According to the cyclic buffer to buffer transportation characteristics of the used application relationship the following behavior is possible:

– The Set Input requests come faster than the contents of the buffers are transported over the network to the Client and Subscribers. In this case not each value of the Input Data object provided with the Set Input service is conveyed over the network. Only the latest value is transported, others are locally overwritten.
– The Set Input requests are issued slower than the contents of the buffers are transported over the network to the Client and Subscribers. In this case each value of the Input Data object provided with the Set Input service is conveyed more than once over the network.
– If the Set Input requests are issued synchronous with the transport of the contents of the buffers each value of the Input Data object provided with the Set Input service is conveyed over the network.

Table 19 shows the parameters of the service.

Table 19 – Set Input

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Input Data	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Input Data
This parameter contains the value of the Input Data object.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

6.2.2.3.2 Read input

By this service any DP-master (Class 2) can read the value of the Input Data object of a DP-slave. The service will only deliver a positive result if the DP-slave is already in the data exchange mode with its assigned DP-master (Class 1). This service shall only be used in conjunction with the MS0 AR. Table 20 shows the parameters of the service.

Table 20 – Read Input

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Input Data		M
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Input Data
This parameter contains the requested value of the Input Data object.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the occurred error.

Allowed values: DS, NA, RS, UE, NR, RE

6.2.2.3.3 Get input

This service shall only be used in conjunction with the MS0 AR. It is used by a DP-master (Class 1) to read:

– the values of all Input Data objects of the assigned DP-slaves if the parameter CREP and Slot Number have not been used in the Get Input service request,
– the value of the Input Data object of a single assigned DP-slave if the parameter Slot Number has not been used in the Get Input service request,
– the value of an Input Data Element of a single assigned DP-slave if the parameter CREP and Slot Number have been used in the Get Input service request.

This service may be acknowledged negatively if the DP-master (Class 1) operates in the Isochronous Mode in the following cases:

– if the service request is made not within the allowed window of interactions of the Enhanced Synchronized Isochronous Mode (SE),
– if the service request is made between the service Set Output with the parameter Final set to TRUE and the next SYNCH indication in case of Buffered Synchronized Isochronous Mode (SE),
– if the addressed DP-slave is deactivated (SC).

Table 21 shows the parameters of the service.

Table 21 – Get Input

Parameter name	Req	Cnf
Argument	M	
AREP	M	
CREP	U	
Slot Number	U	
Result(+)		S
AREP		M
CREP		C
Slot Number		C
Input Data		M
Result(-)		S
AREP		M
CREP		C
Slot Number		C
Status		C

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

CREP
If this parameter is present then the Input Data object of a single DP-slave is addressed.

Slot Number
If this parameter is present then an Input Data Element of a single DP-slave is addressed. If this parameter is used the parameter CREP shall also be present.

Allowed values: 1 to 244

Result(+)
This selection type parameter indicates that the service request succeeded.

Input Data
This parameter contains the selected value.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the occurred error.

Allowed values: SC, SE

6.2.2.3.4 New input

The New Input service signals to the application of the DP-master (Class 1) that new values of all Input Data objects of a DP-slave which are related to the MS0 AR are present. This service shall only be used in conjunction with the MS0 AR. Table 22 shows the parameters of the service.

Table 22 – New Input

Parameter name	Ind
Argument	M
AREP	M
CREP	M

Argument
The argument shall convey the service specific parameters of the service indication.

AREP
This parameter is the local identifier of the AR which passed the indication.

CREP
This parameter is the local identifier of a single DP-slave.

6.2.2.3.5 Set output

By this service the Client application process of the DP-master (Class 1) requests the conveyance of:

– the values of all Output Data objects of the assigned DP-slaves if the parameter CREP and Slot Number have not been used in the Set Output service request,

– the value of the Output Data object of a single assigned DP-slave if the parameter Slot Number has not been used in the Set Output service request,

– the value of an Output Data Element of a single assigned DP-slave if the parameter CREP and Slot Number have been used in the Set Output service request.

This service shall only be used in conjunction with the MS0 AR.

According to the cyclic buffer to buffer transportation characteristics the following behavior is possible:

– The Set Output requests come faster than the contents of the buffers are transported over the network. In this case only the values provided with the latest Set Output service are conveyed over the network.

– The Set Output requests issued slower than the contents of the buffers are transported over the network. In this case the values provided with each Set Output service are conveyed more than once over the network.

– If the Set Output requests are issued synchronous with the transport of the contents of the buffers the values provided with each Set Output service are conveyed once over the network.

This service may be acknowledged negatively if the DP-master (Class 1) operates in the Isochronous Mode in the following cases:

– if the service request is made not within the allowed window of interactions of the Enhanced Synchronized Isochronous Mode (SE),

– if the service request is made between the service Set Output with the parameter Final set to TRUE and the next SYNCH indication in case of Buffered Synchronized Isochronous Mode (SE),

– if the addressed DP-slave is deactivated (SC).

Table 23 shows the parameters of the service.

Table 23 – Set Output

Parameter name	Req	Cnf
Argument	M	
AREP	M	
CREP	U	
Slot Number	U	
Output Data	M	
Final	M	
Result(+)		S
AREP		M
CREP		C
Slot Number		C
Result(-)		S
AREP		M
CREP		C
Slot Number		C
Status		C

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

CREP
If this parameter is present then the Output Data object of a single DP-slave is addressed.

Slot Number
If this parameter is present then an Output Data Element of a single DP-slave is addressed. If this parameter is used the parameter CREP shall also be present.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Output Data
This parameter contains the selected value.

Final
This parameter indicates to the DP Application Layer to change the intermediate buffer of the Input Data and of the Output Data of the DP-slaves. The DP Application Layer changes these buffers after the DX Finished indication and before the next SYNCH indication of the Isochronous DP cycle in case of Buffered Synchronized Isochronous Mode.

The allowed values are shown in Table 24.

Table 24 – Final

Value	Meaning
TRUE	change of the intermediate buffer of the Input Data and of the Output Data shall be done
FALSE	change of the intermediate buffer of the Input Data and of the Output Data is not requested

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the occurred error.

Allowed values: SC, SE

6.2.2.3.6 Read output

By this service any DP-master (Class 2) can read acyclic the value of the Output Data object of a DP-slave. The service will only deliver a positive result if the DP-slave is already in the data exchange mode with its assigned DP-master (Class 1). This service shall only be used in conjunction with the MS0 AR.

If the DP-master (Class 1) changes the operation mode to CLEAR the DP-slave shall set the value of the Output Data object to a value corresponding to a safe state (e.g. zero, last or default value). In this case the value received with this service may not correspond to the real value of the Output Data object.

Table 25 shows the parameters of the service.

Table 25 – Read Output

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Output Data		M
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Output Data
This parameter contains the requested value of the Output Data object.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the occurred error.

Allowed values: DS, NA, RS, UE, NR, RE

6.2.2.3.7 Get output

This service shall be used by a DP-slave to read the value for its Simple Output Data object or Extended Output Data object from the DP Application Layer. This service shall only be used in conjunction with the MS0 AR.

Table 26 shows the parameters of the service.

Table 26 – Get Output

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Output Data		M
Clear Flag		M
New Flag		M
Result(-)		S
AREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Output Data
This parameter contains the value for the Output Data object.

Clear Flag
This parameter indicates whether the value for the Output Data object is valid or cleared.

The allowed values are shown in Table 27.

Table 27 – Clear Flag

Value	Meaning
0	value for the Output Data object valid (DP-master in the state OPERATE)
1	value for the Output Data cleared (DP-master in the state CLEAR)

New Flag
This parameter indicates whether a new value for the Output Data object is received or not.

The allowed values are shown in Table 28.

Table 28 – New Flag

Value	Meaning
0	no new value received
1	new value received

Result(-)

This selection type parameter indicates that the service request failed.

6.2.2.3.8 New output

This service indicates to the application of the DP-slave that a new value for its Simple Output Data object or Extended Output Data object is present. Table 29 shows the parameters of the service.

Table 29 – New Output

Parameter name	Ind
Argument	M
AREP	M
Clear Flag	M

Argument
The argument shall convey the service specific parameters of the service indication.

AREP
This parameter is the local identifier of the AR which passed the indication.

Clear Flag
This parameter indicates whether the value for the Output Data object is valid or cleared.

The allowed values are shown in Table 30.

Table 30 – Clear Flag

Value	Meaning
0	value for the Output Data object valid (DP-master in the state OPERATE)
1	value for the Output Data cleared (DP-master in the state CLEAR)

6.2.2.3.9 Global control

This service is used by the DP-master (Class 1) for the synchronization of the values of the Input Data objects (Freeze Command) and the synchronization of the values of the Output Data objects (Sync Command) of a group or all assigned DP-slaves. This service shall only be used in conjunction with the MS0 AR.

If the DP-master (Class 1) operates in the Isochronous Mode with all or a group of its assigned DP-slaves group 8 of the parameter Group Select of the Global Control service is no longer a valid group value for this parameter.

Furthermore this service is used by the DP-master (Class 1) to send information about its own operation mode (Clear Command) to all assigned DP-slaves.

Table 31 shows the parameters of the service.

Table 31 – Global Control

Parameter name	Req	Ind	Cnf
Argument	M	M(=)	
AREP	M	M(=)	
Clear Command		M	
Sync Command	M	M(=)	
Freeze Command	M	M(=)	
Group Select	M	M(=)	
Result(+)			S
AREP			M
Result(-)			S
AREP			M
Status			M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Clear Command
This parameter contains the Clear Command.

The allowed values are shown in Table 32.

Table 32 – Clear Command

Value	Meaning
0	DP-master (Class 1) in the state OPERATE: The values for the Output Data objects are valid.
1	DP-master (Class 1) in the state CLEAR: The values for the Output Data objects shall be set to a value corresponding to a safe state.

Sync Command
This parameter controls the Sync Operation.

The allowed values are shown in Table 33.

Table 33 – Sync Command

Value	Meaning
0	no action
1	Sync
2	Unsync

Freeze Command
This parameter controls the Freeze Operation.

The allowed values are shown in Table 34.

Table 34 – Freeze Command

Value	Meaning
0	no action
1	Freeze
2	Unfreeze

Group Select
This parameter determines which group(s) of assigned DP-slaves shall be addressed. The control command takes effect if the AND operation between the attribute Group Identifier of the ARL object of the DP-slave and the Group Select parameter results in a value different from zero. If the parameter Group Select is zero all assigned DP-slaves are addressed.

If the DP-master (Class 1) operates in the Isochronous Mode group 8 is not allowed for this parameter. The service request shall be confirmed negatively with an appropriate status value (GE).

Result(+)
This selection type parameter indicates that the service request has been transmitted.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the occurred error.

Allowed values: DS, NO, IV, GE

6.2.2.3.10 New publisher data

This service indicates to the application of the DP-slave that a new value from a subscribed Input Data object is present. Table 35 shows the parameters of the service.

Table 35 – New publisher data

Parameter name	Ind
Argument	M
AREP	M
CREP	M

Argument
The argument shall convey the service specific parameters of the service indication.

AREP
This parameter is the local identifier for the desired AR.

CREP
This parameter is the local identifier of the CR (Publisher) which passed the indication.

6.2.2.3.11 Get publisher data

This service shall be used by the DP-slave (Subscriber) application to read the value from a particular subscribed Input Data object from the DP Application Layer. This service shall only be used in conjunction with the MS0 AR.

Table 36 shows the parameters of the service.

Table 36 – Get publisher data

Parameter name	Req	Cnf
Argument	M	
AREP	M	
CREP	M	
Result(+)		S
AREP		M
CREP		M
Data		M
New Flag		M
Result(-)		S
AREP		M
CREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

CREP
This parameter is the local identifier for the desired CR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Data
This parameter contains the value from the subscribed Input Data object.

New Flag
This parameter indicates whether a new value from the subscribed Input Data object is received or not.

The allowed values are shown in Table 37.

Table 37 – New Flag

Value	Meaning
0	no new value received
1	new value received

Result(-)
This selection type parameter indicates that the service request failed.

6.2.2.3.12 SYNCH

The service SYNCH indicates to the application of the DP-master (Class 1) that a new Isochronous DP cycle has been started.

Table 38 shows the parameters of the service.

Table 38 – SYNCH

Parameter name	Ind
Argument	M
AREP	M

Argument
The argument shall convey the service specific parameters of the service indication.

AREP
This parameter is the local identifier of the AR which passed the indication.

6.2.2.3.13 SYNCH delayed

The service SYNCH Delayed shall be passed to the application of the DP-master (Class 1) only if the last synchronization message was not sent within the allowed time $maxT_{SH}$. The value of the time $maxT_{SH}$ is contained in the related MS0 AR entry of the ARL of the DP-master (Class 1). The service also delivers the time difference measured according to the following rules:

- the synchronization message was sent earlier than $|T_{DP} - maxT_{SH}|$
- the synchronization message was sent later than $|T_{DP} + maxT_{SH}|$

Table 39 shows the parameters of the service.

Table 39 – SYNCH Delayed

Parameter name	Ind
Argument	M
AREP	M
Shift	M

Argument
The argument shall convey the service specific parameters of the service indication.

AREP
This parameter is the local identifier of the AR which passed the indication.

Shift
This parameter contains the value of the measured time difference (shift) in units of t_{BIT}.

Attribute Type: Unsigned32

Allowed values: 1 to $2^{32}-1$

6.2.2.3.14 DX finished

The service DX Finished indicates to the application of the DP-master (Class 1) that the cyclic part (DX) of the Isochronous DP cycle is completed.

Table 40 shows the parameters of the service.

Table 40 – DX Finished

Parameter name	Ind
Argument	M
AREP	M

Argument
The argument shall convey the service specific parameters of the service indication.

AREP
This parameter is the local identifier of the AR which passed the indication.

6.2.2.3.15 SYNCH event

The service SYNCH Event indicates to the application of the DP-slave if

– the DP-master (Class 1) and the DP Application Layer of the DP-slave has started to operate in the Isochronous Mode,
– that a new Isochronous DP cycle has been started,
– the DP-master (Class 1) and the DP Application Layer of the DP-slave have stopped to operate in the Isochronous Mode.

Table 41 shows the parameters of the service.

Table 41 – SYNCH Event

Parameter name	Ind
Argument	M
AREP	M
Status	M

Argument
The argument shall convey the service specific parameters of the service indication.

AREP
This parameter is the local identifier of the AR which passed the indication.

Status
This parameter indicates the status of the Isochronous Mode.

The allowed values are shown in Table 42.

Table 42 – Status

Value	Meaning
IsoM Start	start of Isochronous Mode operation
IsoM SYNCH	start of a new Isochronous DP cycle
IsoM Stop	stop of Isochronous Mode operation

6.2.2.4 Behavior of I/O data objects

6.2.2.4.1 General behavior of the output data object

For performance reasons, the value for the Output Data object should be transferred on receipt as fast as possible to the object. The speed and the jitters of this action is a performance parameter of a DP-slave. This parameter determines the synchronization accuracy.

For highly synchronized DP-slaves, this action should be executed always at the same moment of time after the Sync Command.

6.2.2.4.2 Characteristics of a DP system with isochronous mode functionality

6.2.2.4.2.1 Basic behavior of input and output data in the isochronous mode

A DP system operating in the Isochronous Mode provides the following features (specified at data rates above 1 000 kbit/s and isochronous DP cycles between 500 µs and 32 ms):

– all DP-Devices taking part in the Isochronous Mode functionality synchronies their internal clock system with the synchronization message SYNCH with the isochronous DP cycle;

– all DP-Devices taking part in the Isochronous Mode functionality have to be on the same network (Isochronous Mode functionality over multiple networks is not possible; repeaters are allowed);

– the jitter of SYNCH-Event.ind is less than 1 µs (Jitters within DP-slave is assumed as 0);

– in each DP cycle optionally an acyclic service of full length on the MS1 AR may be included (request or response) and be accomplished if the there is enough time to process the service completely before the next Isochronous DP cycle has to be started;

– the operation of a DP-master (Class 2) on the same network for configuration or diagnosis purposes is possible (the time to execute those services should be included in the cycle time).

In a DP system with enabled Isochronous Mode functionality DP-slaves with basic functionality can be used in conjunction with synchronized DP-slaves. The Isochronous Mode functionality tolerates and compensates communication errors like:

– faulty synchronization messages (SYNCH); the synchronization message may be disturbed at the DP-master (Class 1) at the generation or at the conveyance to the DP-slave or at the recognition at the DP-slave;

– the Isochronous Mode is capable to compensate the jitters of the synchronization message produced by the DP-master (Class 1) and if present by network devices like repeaters; this jitter shall be less than 1µs (this includes the jitters within the DP-master (Class 1));

– the deviation of the real Isochronous DP cycle to the ideal, not disturbed Isochronous DP cycle shall not exceed 0,03 %;

– the SYNCH-Event.ind may be missed due to disturbed telegram within the Isochronous DP; up to three subsequently not in time sent or missing synchronization messages are allowed for a DP-master (Class 1) operating in the Isochronous Mode;

– due to disturbances the synchronisation messages may be destroyed at the conveyance to the DP-slave; the clock recovery mechanism in the DP-slave operating in the Isochronous Mode tolerates up to five subsequently not received synchronisation messages.

If a DP system shall operate in the Isochronous Mode the following definitions shall apply.

– The DP-master (Class 1) and the correspondent DP-slaves shall support the Fail Safe mechanism.

– The DP-slave shall operate in the Isochronous Mode only if the parameter Fail Safe and IsoM Req provided by the parameterization are set to TRUE. Additionally the necessary

IsoM Parameter object shall be provided with the Check User Prm or Check Ext User Prm service to the application of the DP-slave to set the Isochronous Mode parameter of the DP-slave.

6.2.2.4.2.2 Telegram sequences of a DP system operating in the isochronous mode

Figure 15 shows the sequences of DP systems operating in the Isochronous Mode. A DP cycle in the Isochronous context is considered from the SYNCH (synchronization message) to the next SYNCH.

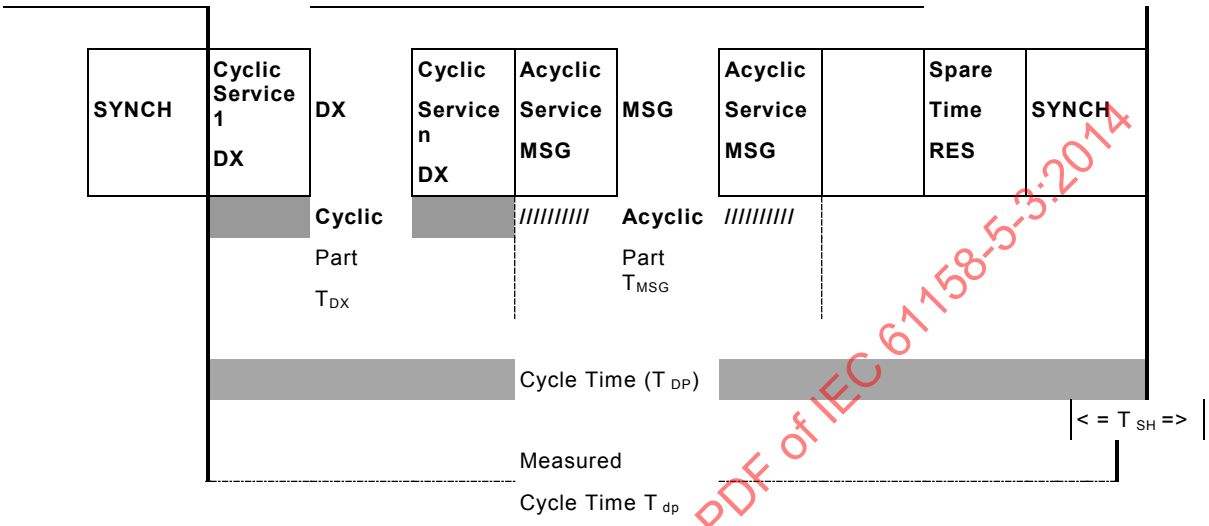


Figure 15 – Sequence of an isochronous DP cycle with one DP-master (class 1)

For the Isochronous Mode the following definitions shall apply.

T_{DX} (Data eXchange Time)
The time T_{DX} may elapse to handle all MS0 communication of one Master (Class 1) of fieldbus DP.

T_{DP} (Data and Parameter exchange Time = Isochronous DP cycle time)
The time T_{DP} may elapse to handle all parts of the Isochronous DP cyclic. The Isochronous DP cycle consists of the following parts:

– SYNCH-Event,
– cyclic services (DX),
– acyclic services (MSG),
– spare time (RES).

T_{MSG} (Message Time)
The times T_{MSG} may elapse to handle all acyclic services on the MS1 AR (T_{MSG1}) and MS2 AR (T_{MSG2}) of fieldbus DP as well as all other DL-services with Service_class = low. These acyclic services shall be executed after the cyclic services. To ensure an Isochronous DP cycle this part shall be limited.

T_{SH} (Shift Time)
This time shall be calculated with $T_{measured\ cycle}$ — T_{DP}. The shift may be caused due to the following reasons:

– delayed sending of the synchronization message (SYNCH) at the DP-master (Class 1),
– inexact calculation of the time T_{DP} at the DP-master (Class 1).

The maximum value of this time (maxT_{SH}) is implementation and configuration dependent. If the time T_{SH} exceeds a certain limit (maxT_{SH}) the application in the DP-master (Class 1) will be informed with the SYNCH Delayed service indicating the measured shift.

Jitters
Jitters may be caused due to the following reasons:

(This definition determines the required parts for a PLL- Minimising Jitters influence, capture drifts and compensate signal losses. It shall be used as a base for quality assurance)

– jitters caused by network components like repeaters or fibre optic interfaces
– transmission rate dependent jitters (jitters < 1 t_{BIT} for the sender, receiver and the network components between)
– implementation dependent reaction time in the sending and receiving station and the network components between

The following changes of T_{DP} are not part of the Jitters definition

– maximum deviation of T_{DP} in the sending and receiving station (jitters < 0,03 % T_{DP})
– losses of clock signaling
– deviation changes due to temperature and ageing (1 ppm within the range of a minute)
– SYNCH is sent too late

RES (Spare time)
This part is defined as the rest of the time between the cyclic part, the acyclic part and the Token handling and the next synchronization message. The duration of the spare time (RES) depends on the services sent in the actual Isochronous DP cycle.

6.2.2.4.2.3 Time relationships of a DP system operating in isochronous mode

Figure 16 shows the time relationships of the applications of the DP-master (Class 1) and 3 DP-slaves operating in the Isochronous Mode.

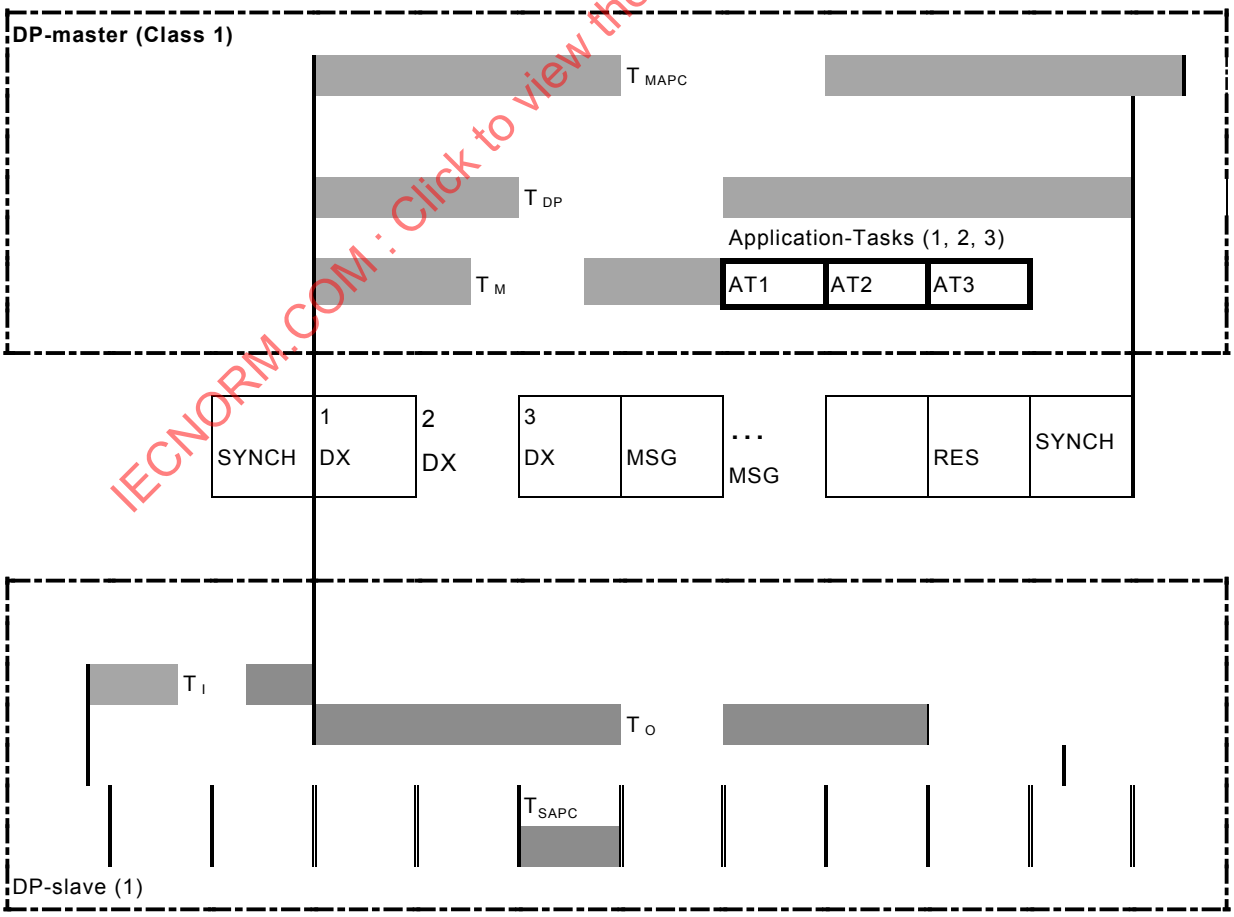


Figure 16 – Additional time relationships in a DP system operating in isochronous mode

For the Isochronous Mode the following time definitions for the application relationships shall be applied.

T_{I_MIN} (Minimum T_I)
This time parameter indicates the minimum time necessary to get and update the Input Data of an individual DP-slave.

T_I (Input Time)
This time is used to trigger the acquisition and update of the Input Data object in a DP-slave operating in Isochronous Mode and may be calculated by the configuration tool. This time is related to the end of the Isochronous DP cycle.

T_{O_MIN} (Minimum T_O)
This time parameter indicates the minimum time that is necessary at the end of the cyclic part of the Isochronous DP cycle (T_{DX}) to get and output the Output Data given in units of T_{BASE_IO} of an individual DP-slave.

T_O (Output Time)
This time is used to trigger the output processing of the Output Data object in a DP-slave operating in Isochronous Mode and may be calculated by the configuration tool. This time is related to the beginning of the Isochronous DP cycle.

T_M (Master Time)
This time is used to shift the processing of the application tasks in the DP-master (Class 1) behind the end of the cyclic part of the Isochronous DP cycle (T_{DX}). This parameter may be set by the application in the DP-master (Class 1) according to $T_M > T_{DX}$. If the time T_M is set to 0 (special case) the DP-master (Class 1) may process input data conveyed a complete Isochronous DP cycle before.

T_{MAPC}, T_{SAPC} (Master Application Cycle Time, Slave Application Cycle Time)
These parameters indicate the time the applications of the DP-master (Class 1) or of the DP-slave need to process an application task completely. The settings of these cycle time parameter may be done device specific. It is allowed to have different settings for the Master Application Cycle Time and Slave Application Cycle Time in each DP-slave using the common time base T_{DP}. The following rules shall apply:

– $T_{MAPC} \geq T_{DP} \geq T_{SAPC}$,
– $T_{MAPC} = n \cdot T_{DP}$ (n = 1 to 14; limited by the slave life sign).

The Time relationships of a DP system, especially of the DP-slave, operating in the Isochronous Mode is described only for the so called compact DP-slave. To make the model also suitable for modular DP-slaves additional aspects have to be added. Especially the time relationships have to be extended and some additional time parameters may be necessary.

At least the following constraints have to be addressed:

– at a modular DP-slave the time parameters depend on the plugged in modules.
– each module may add also a time offset (each module type may have an individual time offset).
– the setting of T_I and T_O may be restricted.
– the device itself may add an additional time offset due to the "transfer" time of the I/O data between the Interface device and the modules.

6.2.2.4.2.4 Example of a DP system with optimized isochronous DP cycle

In the following example the application task 1 and 2 in the DP-master (Class 1) may be processed in each Isochronous DP cycle completely.

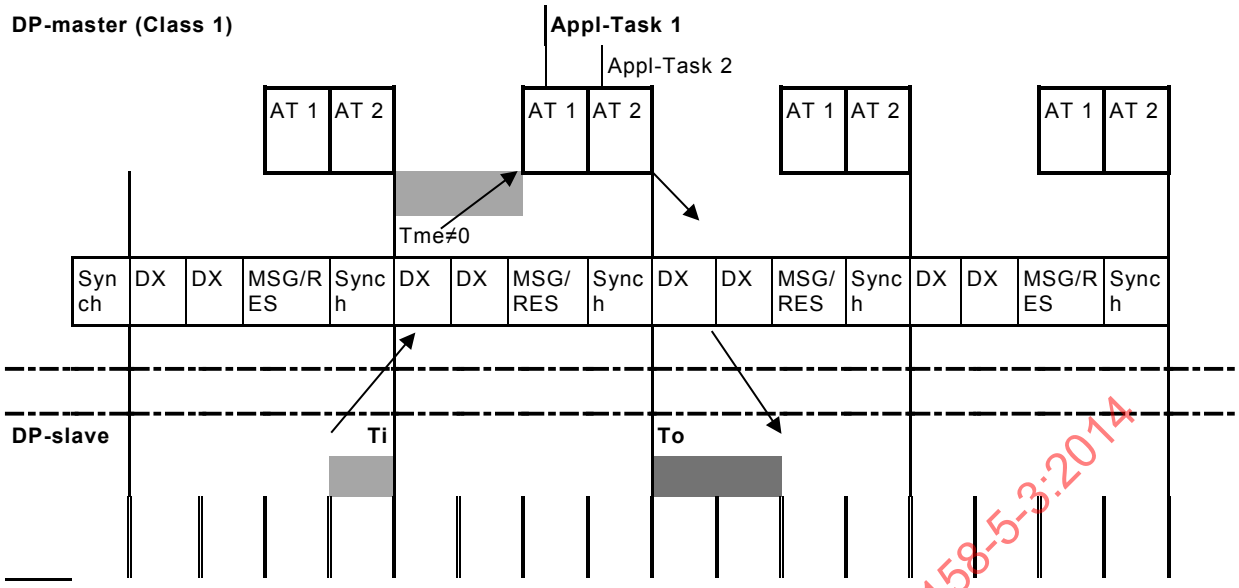


Figure 17 – DP system with optimized isochronous DP cycle

Figure 17 shows a DP System where the sequence of the acquisition and conveyance of the Input Data, processing of the application task and the transfer and output of the Output Data are optimized. This optimization reduces the necessary cycles to perform a complete control loop to less than 3 Isochronous DP cycles and is characterized by the following phases:

– Phase 1: acquisition and update of the value of the Input Data in the DP-slave,

– Phase 2: conveyance of the value of the Input Data to the DP-master (Class 1) and processing of the application task 2 with the actual value of the Input Data and transfer of the new value for the Output Data from the application task 2 to the DP-master (Class 1),

– Phase 3: conveyance of the value for the Output Data to the DP-slave and output of the new Output Data.

This model of synchronization requires enhanced processing capabilities of the DP-master (Class 1) and requires an optimized scheduling in the DP-master (Class 1) and in the DP-slaves.

The dead time in this model of synchronization is given with: $T_D = T_{DP} + T_I + T_O$.

Optimization in the DP-slave (T_I)
The acquisition and update of the value of the Input Data of all involved DP-slaves should be done at the end of each Isochronous DP cycle as close as possible to ensure to provide the most actual values of the Input Data.

Optimization in the DP-slave (T_O)
The output of the value of the Output Data of all involved DP-slaves should be done at the end of the cyclic part (DX) of each Isochronous DP cycle as close as possible to ensure to provide the actual values for the Output Data as soon as possible.

Optimization in the DP-master (Class 1) (T_M)
The processing of the application tasks in the DP-master (Class 1) may be shifted at the end of the cyclic part of the Isochronous DP cycle (T_{DX}) by the parameter T_M. At this time the values of the Input Data of all involved DP-slaves has been update in the DP-master (Class 1). The processing of the application tasks in the DP-master (Class 1) has to be finished within the actual Isochronous DP cycle to provide new values for the Output Data of all involved DP-slaves to be conveyed at the next Isochronous DP cycle.

6.2.2.4.3 Application model of the DP-master (class 1) with isochronous mode functionality

6.2.2.4.3.1 Model variants

Two different models concerning the interface between the DP Application Layer and the application of the DP-master (Class 1) itself are provided by fieldbus DP for the DP-master (Class 1) supporting the Isochronous Mode:

– Buffered Synchronized,

– Enhanced Synchronised.

The DP Application Layer in conjunction with the data-link layer provides the necessary events and messages to synchronies the application of the DP-master (Class 1) with the DP Application Layer for both synchronization mechanism.

6.2.2.4.3.2 Buffered synchronized isochronous mode

The Buffered Synchronized model provides intermediate buffer to store the Input Data from and the Output Data for the DP-slaves during one or more Isochronous DP cycle. During this period the application may access these buffers to read and update their contents at any time. This mechanism requires less processing performance of the application tasks and less synchronization between the application and the DP Application Layer.

At first the application shall read the value of the Input Data with the Get Input service and shall update the value of the Output Data of the DP-slaves with the Set Output service. To inform the DP Application Layer to change the intermediate buffer the application has to set the parameter Final of the Set Output service to TRUE at the update of the last DP-slave. The DP Application Layer exchanges the intermediate buffer at the end of the current Isochronous DP cycle. The application shall wait for the next SYNCH indication to start the next application cycle. The application has to ensure the right sequence of interactions.

Figure 18 illustrates the Buffered Synchronized Isochronous Mode at the DP-master (Class 1).

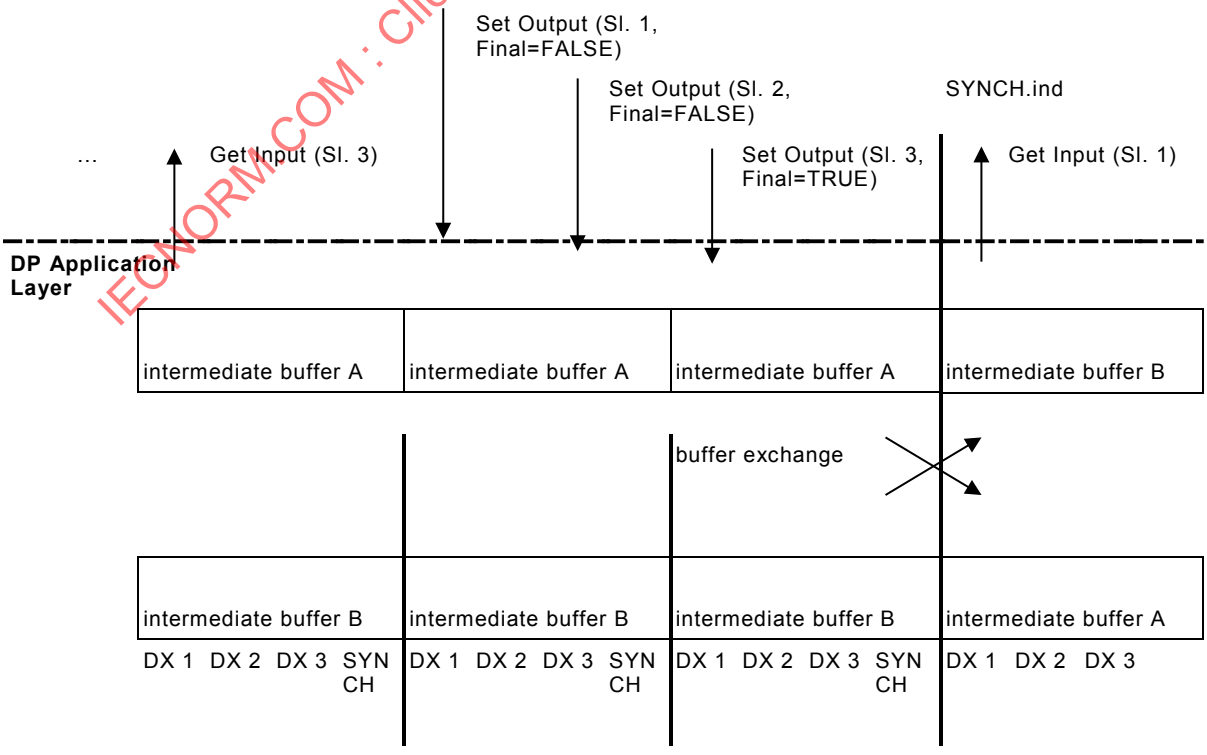


Figure 18 – Buffered synchronized isochronous mode at the DP-master (class 1)

6.2.2.4.3.3 Enhanced synchronized isochronous mode

The Enhanced Synchronized model is characterized by a well-defined sequence of interactions between the application and the DP Application Layer of the DP-master (Class 1). The DP Application Layer provides means to supervise the proper sequence of interactions of the application within the Isochronous DP cycle and informs the application in case of erroneous interactions.

Figure 19 illustrates the Mechanism of the Enhanced Synchronized Isochronous Mode.

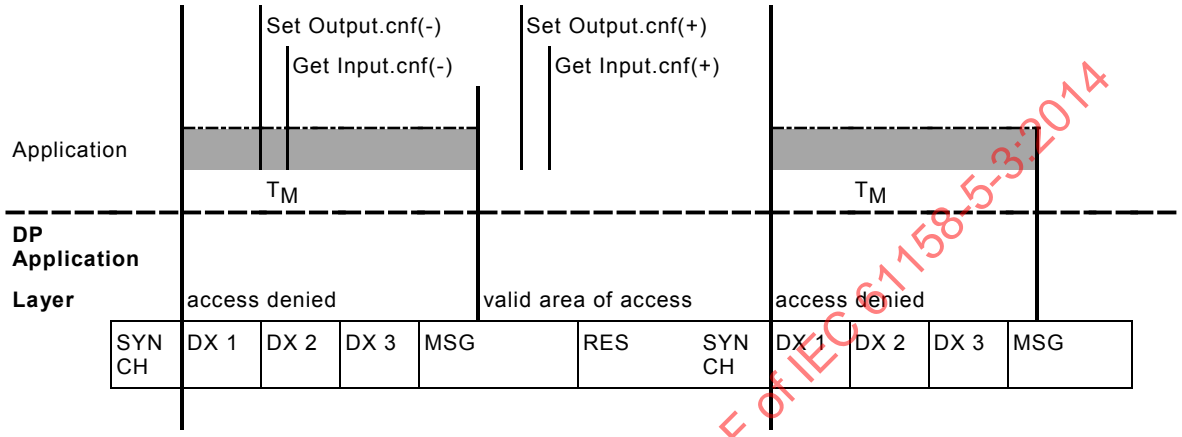


Figure 19 – Enhanced synchronized isochronous mode at the DP-master (class 1)

This mechanism requires enhanced processing performance of the application tasks and a very close synchronization between the application and the DP Application Layer.

6.2.2.4.4 Output data state machine description for isochronous mode

6.2.2.4.4.1 General

The behavior of an Output Data object is defined by the Output Data and PLL state machine in case of Isochronous Mode.

There are several State Machines for Input and Output Data objects. Figure 20 shows the interactions of PLL, Input and Output state machine.

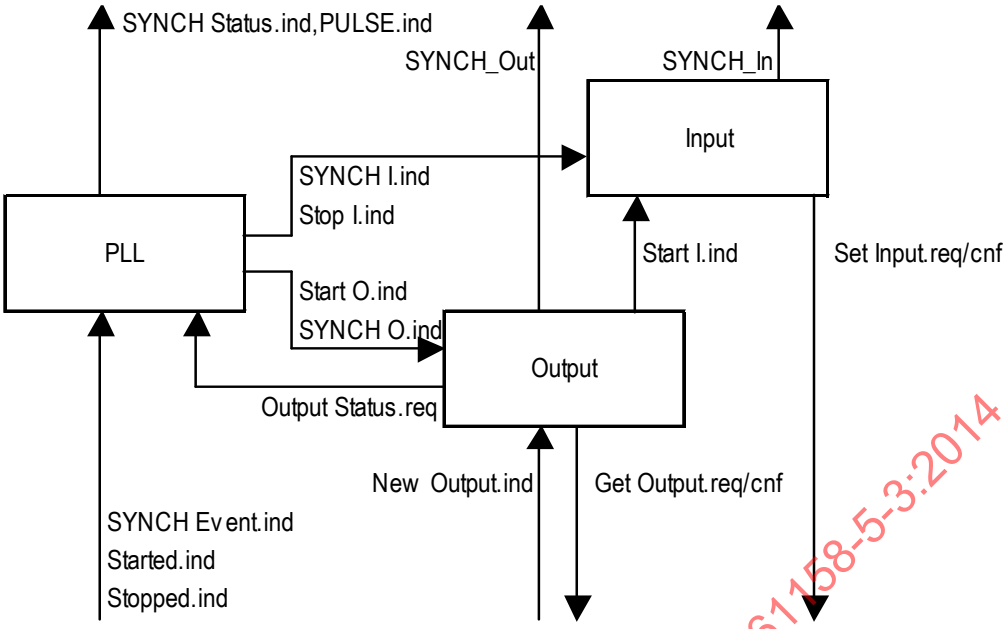


Figure 20 – Input, output and PLL state machine interaction

The Output Data and PLL state machine represents a part of the User functionality and are included to ensure interoperability.

The Output Data and PLL state machine provides events to other parts of the User functionality. The Output Data and PLL state machine have interfaces to the Input Data state machine for co-ordination purposes.

The main functionalities of the Output Data state machine are:

– Transfer the received Output Data to the application on time.
– Check the correct sequence: SYNCH, New Output, read out or the Output Data at To-To_min.
– Check the Masters Sign of Life (MLS).

The main functionalities of the PLL state machine are:

– Jitter compensation of the SYNCH-Event.ind in the DP-slave.
– Drift compensation.
– Generation of a Pulse Scheme for the local application.

The synchronization primitive (SYNCH) is conveyed by a Global Control service sent as multicast with fixed settings of the parameter Group Select and Control Command Freeze and/or Sync.

The synchronization (SYNCH) indicated to the application of the DP-slave may jitter, may be lost at the conveyance to the DP-slave or may be received too late because the DP-master (Class 1) has sent the synchronization message not in time. A PLL (built in hardware or software) shall be used in the DP-slaves to minimize the jitters and phase shifts of the synchronization message and to compensate the loss of synchronization messages and may reduce the possible variation of the synchronization to a value the DP-slave operating in the Isochronous Mode can tolerate. If the variation exceeds a certain limit the application of the DP-slave will leave the Isochronous Mode and will change to a safe state.

The PLL in the DP-slave shall handle the following situations:

– The jitters of the synchronization message at the DP-slave have to be smoothed by the PLL. If the jitters exceed a certain limit, the PLL will recognize a lost clock.

– Synchronization messages lost due to bus disturbances (the multicast may not be retried) have to be compensated.

– Drift: After the start-up the Master will send Synchronization Messages. A DP-slave has to adjust the local time T_{DP} to the Masters T_{DP}. At first SYNCH the Window shall be set to first Window, which is at least T_{PLL_W}*n1+T_{DP}*n2 (n1>1, n2>0,000 3). If the following SYNCH are within the Window, die local T_{DP} will be adjusted and the size of the Window shall be reduced until the window has the size of T_{PLL_W}. The PLL shall enter the normal Operating mode now. Additional Drifts due to temperature change or ageing will be adjusted in the normal operating mode. Drift changes or Jitters should result in a small change of the T_{DP}. To minimize changes in T_{DP} drift changes and Jitters shall be limited.

– Phase shifts due to line delay between the different DP-slaves may be compensated, if the output of the PLL will be shifted in relation to the clock. Each DP-slave and the DP-master (Class 1) have to have an individual time shift to be synchronised with the DP-slave which have the longest line delay.

6.2.2.4.4.2 Primitive definitions

6.2.2.4.4.2.1 Primitives exchanged between the AL and the PLL state machine

Table 43 shows primitive the exchanged between the User and the PLL state machine.

Table 43 – Primitives issued by the AL to the PLL state machine

Primitive Name	Source	Associated parameters	Functions
SYNCH Event.ind	AL	AREP Status	
Started.ind	AL	AREP	
Stopped.ind	AL	AREP	

The parameters used with the primitives are described in the Service Specification of the Output Data object (see 6.2.2.3).

6.2.2.4.4.2.2 Primitives exchanged between the user and the PLL state machine

Table 44 show the primitive exchanged between the User and the PLL state machine.

Table 44 – Primitives issued by the user to the PLL state machine

Primitive Name	Source	Associated parameters	Functions
SYNCH Status.ind	PLL	Status	Reports change of Operation Mode, Warnings and Errors
PULSE.ind	PLL		Signal for Application Process to start a new cycle. This signal can be subdivided from TDP (see Figure 16, TSAPC).

The parameter Status used with the primitive SYNCH Status.ind reports the different operation mode changes, warnings and errors. Table 45 shows the allowed values of Status.

Table 45 – Allowed values of Status

Value	Source	Meaning
PLL_STOP	PLL	PLL is stopped Device has to be resynchronized
PLL_SYNCH_ERR	PLL	PLL detects Synchronization Errors and has to be resynchronized
PLL_SYNCHED	PLL	PLL is synchronized with the Masters SYNCH
OUTPUT_SEQUENCE	Output	Output detects Sequence errors (Violation of the sequence: SYNCH > New Output > SYNCH_Out)
OUTPUT_RESETMLS	Output	Output MLS check detects MLS errors and resynchronizes MLS
OUTPUT_RUNMLS	Output	Output MLS check detects MLS in sequence

6.2.2.4.4.2.3 Primitives exchanged between the user and the input state machine

Table 46 show the primitive exchanged between the User and the Input state machine.

Table 46 – Primitives issued by the user to the input state machine

Primitive Name	Source	Associated parameters	Functions
SYNCH In	Input		Collects the Values of the Input User Data

6.2.2.4.4.2.4 Primitives exchanged between the user and the output state machine

Table 47 shows the primitive exchanged between the User and the Output state machine.

Table 47 – Primitives issued by the user to the output state machine

Primitive Name	Source	Associated parameters	Functions
SYNCH Out	Output		Pass the Output Buffer to the Output User Data

6.2.2.4.4.2.5 Primitives exchanged between the output state machine and the PLL state machine

Table 48 and Table 49 show the primitive exchanged between the Output state machine and the PLL state machine.

Table 48 – Primitives issued by the PLL to the output state machine

Primitive Name	Source	Associated parameters	Functions
Start_O.ind	PLL		Start Output Data processing
SYNCH_O.ind	PLL		Trigger Signal for Output Data Processing

Table 49 – Primitives issued by the output to the PLL state machine

Primitive Name	Source	Associated parameters	Functions
Output_Status.req	Output	Status	Report Status of Output Data processing

The parameter Status used with the primitive Output_Status.req reports the different warnings and errors.

6.2.2.4.4.2.6 Primitives exchanged between the input state machine and the PLL state machine

Table 50 shows the primitive exchanged between the Input state machine and the PLL state machine.

Table 50 – Primitives issued by the PLL to the input state machine

Primitive Name	Source	Associated parameters	Functions
Stop_I.ind	PLL		Stop Input Data processing
SYNCH_I.ind	PLL		Trigger Signal for Input Data Processing

6.2.2.4.4.2.7 Primitives exchanged between the input state machine and the output state machine

Table 51 shows the primitive exchanged between the Input state machine and the Output state machine.

Table 51 – Primitives issued by the output to the input state machine

Primitive Name	Source	Associated parameters	Functions
Start I.ind	Output		Start Output Data processing

6.2.2.4.4.2.8 Primitives exchanged between the AL and the output state machine

Table 52 shows the primitive exchanged between the Output state machine and AL.

Table 52 – Primitives issued by the output state machine to the AL

Primitive Name	Source	Associated parameters	Functions
Get Output.req	Output	AREP	

Table 53 shows the primitive exchanged between the AL and Output state machine.

Table 53 – Primitives issued by the AL to the output state machine

Primitive Name	Source	Associated parameters	Functions
Get Output.cnf	AL	AREP Output Data Clear Flag New Flag	
New Output.ind	AL	AREP Clear Flag	

6.2.2.4.4.2.9 Primitives exchanged between the input state machine and the AL

Table 54 shows the primitive exchanged between the Input state machine and the AL.

Table 54 – Primitives issued by the input state machine to the AL

Primitive Name	Source	Associated parameters	Functions
Set Input.req	Input	AREP Input Data	

Table 55 shows the primitive exchanged between the Input state machine and the AL.

Table 55 – Primitives issued by the AL to the input state machine

Primitive Name	Source	Associated parameters	Functions
Set Input.cnf	AL	AREP	

6.2.2.4.4.3 Local variable definitions of PLL

AREP
This local variable contains the AREP which represents the application relationships.

W
This local variable contains the current value of the window. The window is used to detect a valid SYNCH Event.ind.

Tdp_n
This local variable contains the current value of the adjusted cycle time. The adaptation is necessary due to clock drifts in a DP-master (class 1).

count_error
This local variable contains the current value of the weighted errors in relation to the SYNCH Event.ind within the window.

RUN
This local variable indicates that the Output State Machine is already started.

cnt
This local variable contains the current value of the segment counter.

seg
This local variable contains the current value of the segment time (=Tdp_n/m).

TimerW
This local timer contains the current value of either the time between a SYNCH Event.ind and the beginning of the next window or the time between window start and SYNCH Event.ind. If there is no SYNCH Event.ind within the window, the end of the window substitutes this event. The value of the timer contains the time expired since the last start.

TimerP
This local timer contains the current value of either the time between two PULSE.ind. This is used to subdivide.

6.2.2.4.4.4 Local constant definitions of PLL

E_limit
This local constant limits the number of acceptable synchronization errors during a time interval.

N_error
This local constant is used to weight the synchronization errors and the synchronization success. A factor of 2 means that 2 SYNCH Event.ind are needed to compensate a synchronization error.

m
This local constant is used to subdivide Tdp_n.

6.2.2.4.4.5 Local function

NOTE Some of these functions are also used by the Output and Input state machine.

inc(x)
if (x=LIMIT) then x:=0 else x:=x+1

inc_LS(x)
if (x=LS_LIMIT) then x:=1 else x:=x+1

reduce_Window(W)
Reduces W as long as W>TpIlw. W shall not be smaller than TpIlw.

NOTE This function does not decrement W as linear function but can change in a discontinuous way (keeping W for a while and reducing it by dividing the value with 2).

adjust_Tdp(Tdp_n, error)
Adapt Tdp_n to a change of Tdp. To avoid unnecessary changes due to jitters an error should result in a change of Tdp_n smaller than the error value.

6.2.2.4.4.6 PLL state diagram

Figure 21 shows the state diagram of the PLL state machine.

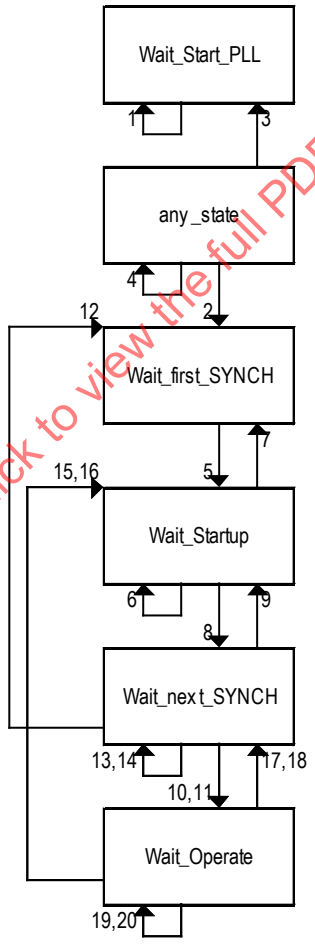


Figure 21 – PLL state diagram

6.2.2.4.4.7 PLL state table

Table 56 shows the states of a PLL state machine.

Table 56 – PLL state table

#	Current State	Event /Condition =>Action	Next State
1	Wait_Start_PLL	SYNCH Event.ind(AREP,Status) => ignore	Wait_Start_PLL
2	any_state	Started.ind(AREP) => W := first_Window Tdp_n := Tdp RUN := 0 count_error := 0	Wait_first_SYNCH
3	any_state	Stopped.ind(AREP) => Status := PLL_STOP SYNCH Status.ind(Status) Stop_I.ind	Wait_Start_PLL
4	any_state	Output Status.req(Status) => SYNCH Status.ind(Status) if (Status = OUTPUT_RESETMLS) Stop_I.ind endif	any_state
5	Wait_first_SYNCH	SYNCH Event.ind(AREP,Status) => start TimerW(Tdp_n-W)	Wait_Startup
6	Wait_Startup	SYNCH Event.ind(AREP,Status) /count_error < e_limit => count_error := count_error + 1	Wait_Startup
7	Wait_Startup	SYNCH Event.ind(AREP,Status) /count_error = e_limit => Status := PLL_SYNCH_ERR W := first_Window Tdp_n := Tdp count_error := 0 start TimerW(Tdp_n-W) SYNCH Status.ind(Status) Stop_I.ind	Wait_first_SYNCH
8	Wait_Startup	TimerW expired => start TimerW(2*W)	Wait_next_SYNCH
9	Wait_next_SYNCH	SYNCH Event.ind(AREP,Status) /W > Tpllw => Tdp_n:= adjust_Tdp(Tdp_n ,(TimerW-W)) W:=reduce_Window(W) start TimerW(Tdp_n-TimerW)	Wait_Startup
10	Wait_next_SYNCH	SYNCH Event.ind(AREP,Status) /W = Tpllw => Tdp_n:= adjust_Tdp(Tdp_n ,(TimerW-W)) count_error := 0start TimerW(Tdp_n-TimerW)	Wait_Operate
11	Wait_next_SYNCH	TimerW expired /count_error < e_limit => count_error := count_error + 1 start TimerW(Tdp-2*W)	Wait_Operate
12	Wait_next_SYNCH	TimerW expired /count_error = e_limit => Status := PLL_SYNCH_ERR W := first_Window Tdp_n := Tdp RUN := 0 count_error := 0 start TimerW(Tdp_n-W) SYNCH Status.ind(Status) Stop_I.ind	Wait_first_SYNCH

#	Current State	Event /Condition =>Action	Next State
13	Wait_next_SYNCH	TimerP expired /cnt < m => cnt:= cnt+1 Start TimerP(Seg) PULSE.ind	Wait_next_SYNCH
14	Wait_next_SYNCH	TimerP expired /cnt = m => PULSE.ind	Wait_next_SYNCH
15	Wait_Operate	SYNCH Event.ind(AREP,Status) /count_error < e_limit => count_error := count_error + 1	Wait_Startup
16	Wait_Operate	SYNCH Event.ind(AREP,Status) /count_error = e_limit => Status := PLL_SYNCH_ERR W := first_Window Tdp_n := Tdp RUN := 0 count_error := 0 start TimerW(Tdp_n-W) SYNCH Status.ind(Status) Stop_I.ind	Wait_Startup
17	Wait_Operate	TimerW expired /RUN = 1 => cnt:=1 Seg = (Tdp_n)/m Start TimerP(W+Delay) Start TimerW(2*W) SYNCH_I.ind SYNCH_O.ind	Wait_next_SYNCH
18	Wait_Operate	TimerW expired /RUN = 0 => Status := PLL_SYNCHED cnt:=1 Seg = (Tdp_n)/m Start TimerP(W+Delay) RUN:= 1 Start TimerW(2*W) Start_O.ind SYNCH_I.ind SYNCH_O.ind SYNCH_Status.ind(Status)	Wait_next_SYNCH
19	Wait_Operate	TimerP expired /cnt < m => cnt:= cnt+1 Start TimerP(Seg) PULSE.ind	Wait_Operate
20	Wait_Operate	TimerP expired /cnt = m => PULSE.ind	Wait_Operate

6.2.2.4.4.8 Local variable definitions of output

AREP
This local variable contains the AREP which represents the application relationships.

wrong_output_update
This local variable contains the current value of the weighted errors of output updates.

MLS_start_counter
This local variable contains the current value of correct MLS at start-up.

MLS_error_counter
This local variable contains the current value of the MLS errors.

TMAPC_counter
This local variable contains the current value of the number of cycles since the last MLS update.

old_MLS
This local variable contains the value of the last MLS.

MLS_State
This local variable contains the value sub-state for MLS calculation.

Possible values are:

- Wait_first_MLS Start-up of MLS
- Wait_next_MLS Wait for next following Start-up MLS
- Wait_MLS Wait for MLS during normal operation

O_Buffer
This local variable contains the values of the Output Data received. It contains an optional variable MLS which contains the live sign. If this variable is not present the State "Check_MLS" has no Action.

TimerTo
This local timer contains the current value of the time between a SYNCH Event.ind and the transfer of the Output_Data to the Application Process.

Timer(To-Tomin)
This local timer contains the current value of the time between a SYNCH Event.ind and the transfer of the Output_Data to the Output State Machine.

6.2.2.4.4.9 Local constant definitions

limit
This local constant limits the number of acceptable output update errors during a time interval.

n
This local constant is used to weight wrong output updates and successful. A factor of 2 means that 2 successful updates are needed to compensate an erroneous update.

MLS_factor
This local constant is used define the number of cycles between increments of MLS.

n_TMAPC
This local constant is used to limit the errors of MLS.

n_MLS
This local constant is used for start-up to have a reasonable number of correct MLS.

6.2.2.4.4.10 Local function

inc(x)
if (x=LIMIT) then x:=0 else x:=x+1

6.2.2.4.4.11 OUTPUT state diagram

Figure 22 shows the state diagram of the OUTPUT state machine.

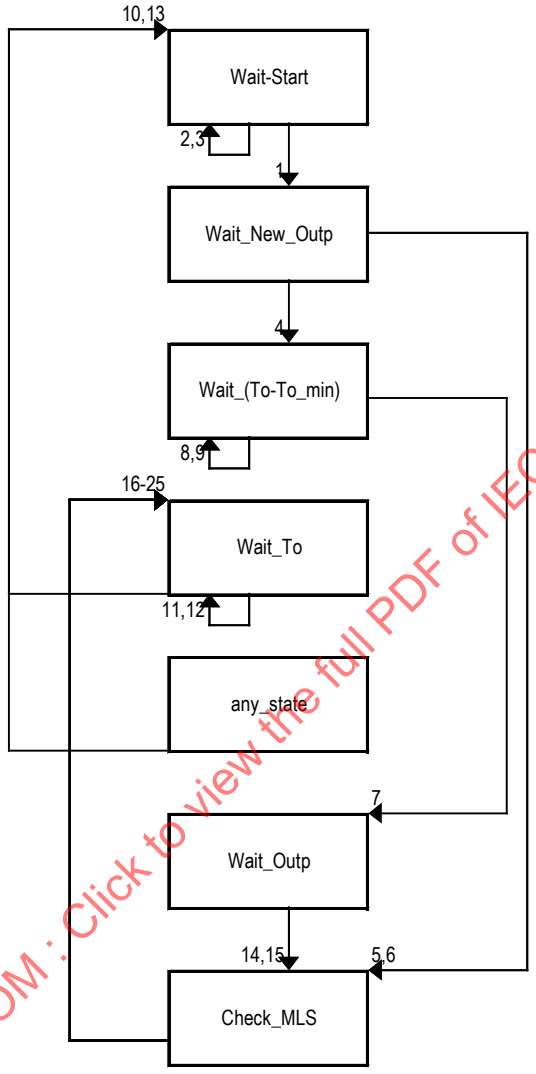


Figure 22 – OUTPUT state diagram

6.2.2.4.4.12 OUTPUT state table

Table 57 shows the states of an OUTPUT state machine.

Table 57 – OUTPUT state table

#	Current State	Event /Condition =>Action	Next State
1	Wait-Start	SYNCH_O.ind => Start Timer(To-To_min) Start TimerTo	Wait_New_Outp
2	Wait-Start	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Wait-Start
3	Wait-Start	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd = limit => Status := OUTPUT_SEQUENCE Output_Status.req (Status)	Wait-Start
4	Wait_New_Outp	New_Output.ind (AREP, Clear_Flag) => if (wrong_output_upd>0) wrong_output_upd= wrong_output_upd-1	Wait_(To-To_min)
5	Wait_New_Outp	Timer(To-To_min) expired /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Check_MLS
6	Wait_New_Outp	Timer(To-To_min) expired /wrong_output_upd = limit => Status := OUTPUT_SEQUENCE Output_Status.req (Status)	Check_MLS
7	Wait_(To-To_min)	Timer(To-To_min) expired => Get_Output.req(AREP)	Wait_Outp
8	Wait_(To-To_min)	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Wait_(To-To_min)
9	Wait_(To-To_min)	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd = limit => Status := OUTPUT_SEQUENCE Output_Status.req (Status)	Wait_(To-To_min)
10	Wait_To	TimerTo expired => SYNCH_Out	Wait-Start
11	Wait_To	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Wait_To
12	Wait_To	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd = limit => Status := OUTPUT_SEQUENCE Output_Status.req (Status)	Wait_To
13	any_state	Start_O.ind=>MLS_State:= Wait_first_MLSO_Buffer:=Nil(with O_Buffer.MLS:=0)old_MLS:=0MLS_error_counter:=0TMAPC_counter:=0MLS_start_counter:=0	Wait-Start
14	Wait_Outp	Get_Output.cnf (AREP, Output_Data, New_Flag, Clear_Flag) /New_Flag && !Clear_Flag => O_Buffer := Output_Data	Check_MLS
15	Wait_Outp	Get_Output.cnf (AREP, Output_Data, New_Flag, Clear_Flag) /!(New_Flag && !Clear_Flag) =>	Check_MLS

#	Current State	Event /Condition =>Action	Next State
16	Check_MLS	/MLS_State = Wait_first_MLS => MLS_State := Wait_next_MLS Status:=OUTPUT_RESETMLS old_MLS:=O_Buffer.MLS Output_Status.req (Status)	Wait_To
17	Check_MLS	/MLS_State = Wait_next_MLS TMAPC_counter < MLS_factor => MLS_State := Wait_next_MLS inc(TMAPC_counter)	Wait_To
18	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter < n_TMAPC => MLS_State := Wait_next_MLS inc_LS(old_MLS) TMAPC_counter:=0 MLS_error_counter := MLS_error_counter + n_error	Wait_To
19	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter >= n_TMAPC => MLS_State := Wait_first_MLS Status:=OUTPUT_RESETMLS TMAPC_counter:=0 MLS_error_counter:=0 old_MLS:=0 MLS_start_counter:=0 Output_Status.req (Status)	Wait_To
20	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS = inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_start_counter < n_MLS => MLS_State := Wait_next_MLS TMAPC_counter:=0 if (MLS_error_counter >0) MLS_error_counter := MLS_error_counter –1 old_MLS:=MLS inc(MLS_start_counter)	Wait_To
21	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS = inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_start_counter = n_MLS => MLS_State := Wait_MLS Status:=OUTPUT_RUNMLS TMAPC_counter:=0 MLS_error_counter:=0 old_MLS:=O_Buffer.MLS MLS_start_counter:=0 Output_Status.req (Status) Start_I.ind	Wait_To
22	Check_MLS	/MLS_State = Wait_MLS TMAPC_counter < MLS_factor => MLS_State := Wait_MLS inc(TMAPC_counter)	Wait_To

#	Current State	Event /Condition =>Action	Next State
23	Check_MLS	/MLS_State = Wait_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter < n_TMAPC => MLS_State := Wait_MLS inc_LS(old_MLS) TMAPC_counter:=0 MLS_error_counter := MLS_error_counter + n_error	Wait_To
24	Check_MLS	/MLS_State = Wait_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter >= n_TMAPC => MLS_State := Wait_first_MLS Status:=OUTPUT_RESETMLS TMAPC_counter:=0 MLS_error_counter:=0 old_MLS:=0 Output_Status.req (Status)	Wait_To
25	Check_MLS	/MLS_State = Wait_MLS O_Buffer.MLS = inc_LS(old_MLS) & TMAPC_counter = MLS_factor => MLS_State := Wait_MLS TMAPC_counter:=0 if (MLS_error_counter >0) MLS_error_counter := MLS_error_counter –1 old_MLS:=O_Buffer.MLS	Wait_To

6.2.2.4.5 Behavior of the input data object

For performance reasons, the value of the Input Data object should be transferred on change as fast as possible to the MS0 buffer for transmission. The speed and the jitters of this action is a performance parameter of a DP-slave. This parameter determines the freeze accuracy.

For highly synchronized DP-slaves, this action should be executed always at the same moment of time before the Freeze Command. This requires that a Freeze Command will be sent cyclically.

6.2.2.4.6 Input data state machine description for isochronous mode

6.2.2.4.6.1 Local variable definitions of input

AREP
This local variable contains the AREP which represents the application relationships.

SLS
This local variable contains the value of the Slave life sign.

I_Buffer
This local variable contains the values of the Input Data fetched from the Application Process. It contains an optional variable SLS which contains the live sign. If SLS is not present the Update of SLS is a NULL Operation.

TimerTix
This local timer contains the current value of the time between a SYNCH Event.ind and the transfer of the Input_Data to the AL.

Timer(TDP-Ti)
This local timer contains the current value of the time between a SYNCH Event.ind and the transfer of the Input_Data to the Input State Machine.

6.2.2.4.6.2 INPUT state diagram

Figure 23 shows the state diagram of the INPUT state machine.

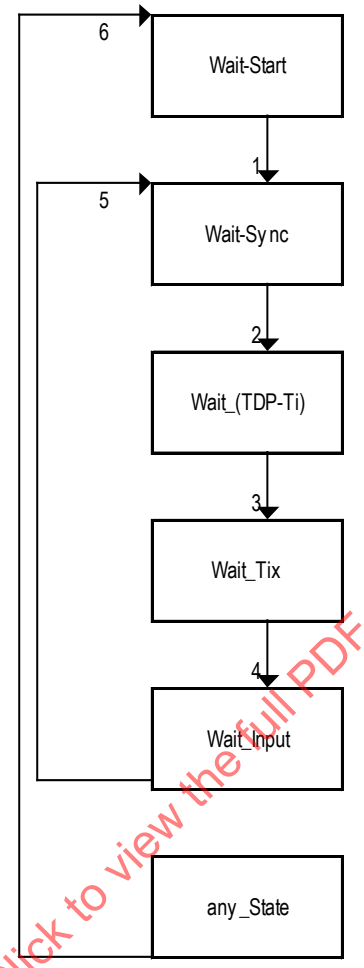


Figure 23 – INPUT state diagram

6.2.2.4.6.3 INPUT state table

Table 58 shows the states of an INPUT state machine.

Table 58 – INPUT state table

#	Current State	Event /Condition =>Action	Next State
1	Wait-Start	**Start_I.ind** =>	Wait-Sync
2	Wait-Sync	**SYNCH_I.ind** => Start Timer(TDP-Ti) Start TimerTix	Wait_(TDP-Ti)
3	Wait_(TDP-Ti)	**Timer(TDP-Ti) expired** => SYNCH_In	Wait_Tix
4	Wait_Tix	**TimerTix expired** => Input_Data:= I_Buffer Input_Data.SLS:= inc_LS(SLS) Set_Input.req (AREP,Input_Data)	Wait_Input
5	Wait_Input	**Set_Input.cnf(AREP)** =>	Wait-Sync
6	any_State	**Stop_I.ind** => SLS:= 0	Wait_Start

6.2.3 Diagnosis ASE

6.2.3.1 Overview

In the DP Application Layer environment, the Diagnosis ASE of a DP-slave contains Diagnosis objects that are used to report the status of the Default Application Process of the DP-slave and the modules which are part of the configuration. The Diagnosis ASE defines attributes of the Diagnosis objects and provides a set of local services used to set and get their values. The following Diagnosis objects are defined:

– Device Related Diagnosis object,
– Status object,
– Module Status Data object,
– DXB-Link Status object,
– Identifier Related Diagnosis object,
– Channel Related Diagnosis object.

The application of a DP-slave may provide the value of one or several Diagnosis objects to the MS0 AR. The DP-slave shall provide only the values of those Diagnosis objects whose values have changed since their last provision. The DP-slave may indicate (with the parameter Ext Diag Overflow) that the values of all Diagnosis objects with a changed value could not be provided with one Set Slave Diag service request. The DP-master (Class 1) shall indicate (with the parameter Ext Diag Overflow) in the Get Slave Diag confirmation that not all received values of the Diagnosis objects from the DP-slave could be provided to the application.

The DP-slave shall indicate (with the parameter Ext Diag Flag) that a relevant diagnosis event has occurred which leads to a change in at least one Diagnosis object (e. g. should be set if one element of the Identifier Related Diagnosis object is set to the value 1). This flag causes an entry in the attribute System Diagnosis of the Master Diag object of the DP-master (Class 1).

The DP-slave may indicate the actual version of the DP-slave with the parameter Revision Number provided by the Set Slave Diag service. The Revision Number may be used by a DP-master (Class 2) to find the related GSD file of the DP-slave.

The Diagnosis model uses the Client/Server access model. The Diagnosis model is based on a buffer to buffer relationship. Thus the Server application updates (sets) the values of the Diagnosis objects in its local buffer for the conveyance. The Client application may get the values of the Diagnosis objects from its local buffer.

The DP-master (Class 1) communicates via the MS0 AR with all Diagnosis objects of the assigned DP-slaves simultaneously. Additionally the Client application may communicate with the Diagnosis objects of a single DP-slave.

The DP-master (Class 2) communicates via the MS0 AR with the Diagnosis objects of a single DP-slave.

Furthermore additional Diagnosis information (attributes of the MS0 AR object) is provided by the DP Application Layer as parameters of the Get Slave Diag service and Read Slave Diag service.

The formal description of the various Diagnosis objects is presented next, followed by a description of its services.

6.2.3.2 Diagnosis class specification

6.2.3.2.1 Device related diagnosis class specification

6.2.3.2.1.1 Template

The Device Related Diagnosis object is described by the following template:

DP ASE: **Diagnosis ASE**
CLASS: **Device Related Diagnosis**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Key Attribute: Implicit
2. (m) Attribute: Device Related Data Description

SERVICES:
1. (m) OpsService: Set Slave Diag
2. (m) OpsService: Get Slave Diag
3. (o) OpsService: Read Slave Diag
4. (o) OpsService: New Slave Diag

6.2.3.2.1.2 Attributes

Implicit
The attribute Implicit indicates that the Device Related Diagnosis object is implicitly addressed by the services.

Device Related Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used. The maximum length of this Device Related Data shall not exceed 62 octets.

6.2.3.2.1.3 Invocation of the device related diagnosis object

For the invocation of the Device Related Diagnosis object the following rules shall be considered:

– Device Related Diagnosis objects shall be used only if the device attribute DPV1 = FALSE.

– Device Related Diagnosis objects shall not exceed the total length of 62 octets. The attribute Device Related Data Description has to be set accordingly.

More than one Device Related Diagnosis object may be invoked in a DP-slave.

6.2.3.2.2 Identifier related diagnosis class specification

6.2.3.2.2.1 Template

This class specifies the state of the modules which are part of the configuration. Each bit is related to a Module (Slot). An element set to FALSE means that no diagnosis of the related module exists or no module is placed. A bit set to TRUE indicates a module failure. If a module failure exists, the values of the Input Data object and/or the Output Data object of this module are set to the default value by the DP-slave.

The Identifier Related Diagnosis object is described by the following template:

DP ASE:			**Diagnosis ASE**
CLASS:			**Identifier Related Diagnosis**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of Identifier Status
2.1.	(m)	Attribute:	Identifier Status
SERVICES:			
1.	(m)	OpsService:	Set Slave Diag
2.	(m)	OpsService:	Get Slave Diag
3.	(o)	OpsService:	Read Slave Diag
4.	(o)	OpsService:	New Slave Diag

6.2.3.2.2.2 Attributes

Implicit
The attribute Implicit indicates that the Identifier Related Diagnosis object is implicitly addressed by the services.

List of Identifier Status
This attribute is composed of the following list elements:

Identifier Status
The elements of this list attribute indicate the state of each module. E.g. if the first module has an error the first element of the List of Identifier Status has the value 1.

The allowed values are shown in Table 59.

Table 59 – Identifier status

Value	Meaning
0	Module in operation.
1	Module not in operation.

6.2.3.2.2.3 Invocation of the identifier related diagnosis object

For the invocation of the Identifier Related Diagnosis object the following rules shall be considered:

– Only one Identifier Related Diagnosis object shall be invoked in a DP-slave.

– According to the maximum number of modules, that are part of the configuration, an Identifier Related Diagnosis object shall not exceed the total length of 32 octets.

6.2.3.2.3 Channel related diagnosis class specification

6.2.3.2.3.1 Template

The Channel Related Diagnosis object is described by the following template:

DP ASE:			**Diagnosis ASE**
CLASS:			**Channel Related Diagnosis**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attributes:	Identifier
2.	(m)	Attribute:	Channel Type
3.	(m)	Attribute:	IO Type
4.	(m)	Attribute:	Error Type
SERVICES:			
1.	(m)	OpsService:	Set Slave Diag
2.	(m)	OpsService:	Get Slave Diag
3.	(o)	OpsService:	Read Slave Diag
4.	(o)	OpsService:	New Slave Diag

6.2.3.2.3.2 Attributes

Identifier
This key attribute is composed of Identifier Number and Channel Number to indicate the source of the Channel Related Diagnosis object.

Allowed values: 0 to 63, 0 to 63

Channel Type
This attribute indicates the channel-size to which the Channel Related Diagnosis object is related.

The allowed values are shown in Table 60.

Table 60 – Channel type

Value	Meaning
0	unspecific
1	1 bit
2	2 bit
3	4 bit
4	octet
5	word
6	2 words

IO Type
This attribute indicates the IO Type of the channel to which the Channel Related Diagnosis object is related.

The allowed values are shown in Table 61.

Table 61 – IO type

Value	Meaning
1	Input channel
2	Output channel
3	Bi-directional channel

Error Type
This attribute indicates the error type of the Channel Related Diagnosis object.

The allowed values are shown in IEC 61158-6-3, 5.3.17.

6.2.3.2.3.3 Invocation of the channel related diagnosis object

For the invocation of the Channel Related Diagnosis object the following rule shall be considered:

– Several Channel Related Diagnosis objects may be invoked in a DP-slave.

6.2.3.2.4 Status class specification

6.2.3.2.4.1 Template

The status model allows the transfer of status messages from the DP-slave to the DP-master (Class 1). The status model is slot based.

A Status object is described by the following template:

DP ASE: **Diagnosis ASE**
CLASS: **Status**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Key Attributes: Identifier
2. (m) Attribute: Status Specifier
3. (o) Attribute: Status Data Description

SERVICES:
1. (m) OpsService: Set Slave Diag
2. (m) OpsService: Get Slave Diag
3. (o) OpsService: Read Slave Diag
4. (o) OpsService: New Slave Diag

6.2.3.2.4.2 Attributes

Identifier
This key attribute is composed of Slot Number and Status Type to indicate the number of the slot and the status type.

Allowed values for Slot Number: 0 to 254

The allowed values for Status Type are shown in Table 62.

Table 62 – Status type

Value	Meaning
1	Status Message
32-126	Manufacturer Specific

Status Specifier
This attribute specifies additional status information.

The allowed values are shown in Table 63.

Table 63 – Status specifier

Value	Meaning
0	no further differentiation
1	status appears
2	status disappears

Status Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

6.2.3.2.4.3 Invocation of the status object

For the invocation of the Status object the following rules apply:

– Status objects shall be used only if the device attribute DPV1 = TRUE
– Several Status objects may be invoked in a DP-slave.
– A Status object shall not exceed the total length of 59 octets. The attribute Status Data Description has to be set accordingly.

6.2.3.2.5 Module status data class specification

6.2.3.2.5.1 Template

This class specifies the status of the modules which are part of the configuration.

The Module Status Data object is described by the following template:

DP ASE: **Diagnosis ASE**
CLASS: **Module Status Data**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Key Attribute: Implicit
2. (m) Attribute: Status Specifier
3. (m) Attribute: List of Module Status
3.1. (m) Attribute: Module Status
SERVICES:
1. (m) OpsService: Set Slave Diag
2. (m) OpsService: Get Slave Diag
3. (o) OpsService: Read Slave Diag
4. (o) OpsService: New Slave Diag

6.2.3.2.5.2 Attributes

Implicit
The attribute Implicit indicates that the Module Status Data object is implicitly addressed by the services.

Status Specifier
This attribute specifies additional status information.

The allowed values are shown in Table 64.

Table 64 – Status specifier

Value	Meaning
0	no further differentiation
1	status appears
2	status disappears

List of Module Status
This attribute is composed of the following list elements:

Module Status
This attribute specifies additional status information for each module. The first element of the list contains the status of the first module (Slot Number 1).

The allowed values are shown in Table 65.

Table 65 – Module status

Value	Meaning
0	data valid
1	data invalid - signal error: the data of the corresponding module are not valid due to an error (inputs not accessible or outputs cannot be set, e.g. due to short circuit or overload)
2	data invalid - wrong module: the data of the corresponding module are not valid, due to a wrong module in place
3	data invalid - no module: the data of the corresponding module are not valid because there is no module in place

6.2.3.2.5.3 Invocation of the module status data object

For the invocation of the Module Status Data object the following rules apply:

– The Module Status Data object shall be used only if the device attribute DPV1 = TRUE.
– Only one Module Status Data object shall be invoked in a DP-slave.
– The Module Status Data object shall not exceed the total length of 59 octets.

6.2.3.2.6 DXB-link status class specification

6.2.3.2.6.1 Template

This class specifies the status of the DXB-Links (Publisher/Subscriber relationships) contained in the DXB-Linktable at the Subscriber.

The DXB-Link Status object shall be used according the following rules:

– the DXB-Link Status object shall be issued at the first time with the Diagnosis information if the DP-slave has received a correct DXB-Linktable and if the DP-slave has entered the data transfer state.
– the DXB-Link Status object shall be omitted if the DXB-Linktable was marked as invalid (e. g. receipt of new parameterisation data without a DXB-Linktable).

The DXB-Link Status object is described by the following template:

DP ASE:			**Diagnosis ASE**
CLASS:			**DXB-Link Status**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	Status Specifier
3.	(m)	Attribute:	List of DXB-Link Status
3.1.	(m)	Attribute:	Publisher Address
3.2.	(m)	Attribute:	Link Status
3.3.	(m)	Attribute:	Link Error
SERVICES:			
1.	(m)	OpsService:	Set Slave Diag
2.	(m)	OpsService:	Get Slave Diag
3.	(o)	OpsService:	Read Slave Diag
4.	(o)	OpsService:	New Slave Diag

6.2.3.2.6.2 Attributes

Implicit
The attribute Implicit indicates that the DXB-Link Status object is implicitly addressed by the services.

Status Specifier
This attribute specifies additional status information.

The allowed values are shown in Table 66.

Table 66 – Status specifier

Value	Meaning
0	no further differentiation
1	status appears
2	status disappears

List of DXB-Link Status
This attribute is composed of the following list elements:

Publisher Address
This attribute indicates the DL-address of the corresponding Publisher.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Link Status
This attribute indicates the status of the related DXB-Link.

The allowed values are shown in Table 67.

Table 67 – Link status

Value	Meaning
TRUE	DXB-Link active; at least one subscription of the related Publisher has been performed during the last watchdog control (T_{WD})
FALSE	DXB-Link not active; no subscription of the related Publisher has been performed during the last watchdog control (T_{WD}); the Subscriber may indicate detailed information about the reason with the attribute Link Error

Link Error
This attribute indicates detailed information about the failure of the related DXB-Link.

The allowed values are shown in Table 68.

Table 68 – Link error

Value	Meaning
0	unspecific
1	wrong length; the length of the Input Data of the Publisher is not in alignment with the value of the attribute Input Data Length Publisher of the related DXB-Link Entry

6.2.3.2.6.3 Invocation of the DXB-link status object

For the invocation of the DXB-Link Status object the following rules apply:

– The DXB-Link Status object shall be used only if the device attribute DPV1 is set to TRUE.
– Only one DXB-Link Status object shall be invoked in a DP-slave.
– The DXB-Link Status object shall not exceed the total length of 59 octets. The attribute List of DXB-Link Status has to be set accordingly.

6.2.3.3 Diagnosis service specification

6.2.3.3.1 Set slave diag

This service shall be used by a DP-slave to pass the values of the Diagnosis objects to the DP Application Layer. The overall transmitted diagnosis information shall not exceed the total length of 238 octets. By this service the Server application process of the DP-slave updates the local buffer. According to the buffer to buffer characteristics of the MS0 AR the following behavior is possible:

– The Set Slave Diag requests come faster than the contents of the buffers are transported over the network. In this case not each value of the Diagnosis objects provided with the Set Slave Diag service is conveyed over the network. Only the latest values are transported, others are locally overwritten.
– The Set Slave Diag requests issued slower than the contents of the buffers are transported over the network. In this case the values of the Diagnosis objects provided with the Set Slave Diag service are conveyed more than once over the network.
– If the Set Slave Diag requests are issued synchronous with the transport of the contents of the buffers each value of the Diagnosis objects provided with the Set Slave Diag service are conveyed over the network.

This service shall only be used in conjunction with the MS0 AR.

Table 69 shows the parameters of the service.

Table 69 – Set Slave Diag

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Ext Diag Overflow	M	
Ext Diag Flag	M	
Ext Diag Data	C	
Device Related Diagnosis Data	U	
List of Identifier Status	U	
Identifier Status	M	
List of Channel Related Diagnosis Data	U	
Identifier Number	M	
Channel Number	M	
Channel Type	M	
IO Type	M	
Error Type	M	
List of Status Data	U	
Slot Number	M	
Status Type	M	
Status Specifier	M	
Status Data	M	
Module Status Data	U	
Status Specifier	M	
List of Module Status	M	
Module Status	M	
DXB-Link Status	U	
Status Specifier	M	
List of DXB-Link Status	M	
Publisher Address	M	
Link Status	M	
Link Error	M	
Revision Number	U	
Result		S
AREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Ext Diag Flag
This parameter indicates that a relevant diagnosis event has occurred which leads to a change in at least one Diagnosis object (e. g. should be set if one element of the Identifier Related Diagnosis object is set to the value 1).

The allowed values are shown in Table 70.

Table 70 – Ext Diag Flag

Value	Meaning
TRUE	The device is in state diagnosis. Thus exist at least one important diagnosis (such as short circuit) item. An entry has to be present in the Diagnosis objects (Device Related Diagnosis objects, Identifier Related Diagnosis object, Channel Related Diagnosis objects, Status objects, Module Status object).
FALSE	The device is NOT in state diagnosis. Thus, no important diagnosis entry exists. If the attribute value is FALSE, a status message (for example, supply voltage OK) may be present in the Diagnosis Data, or there exists no Diagnosis Data.

Ext Diag Overflow
This Boolean parameter indicates that the values of all Diagnosis objects with a changed value could not been provided with this Set Slave Diag service request.

In case of an overflow condition it is only allowed to truncate at Diagnosis object limits.

Allowed values: TRUE, FALSE

Ext Diag Data
This parameter is composed of the following items.

Device Related Diagnosis Data
This parameter contains the value of the Device Related Diagnosis object (this parameter shall be used only if DPV1 = FALSE).

List of Identifier Status
This parameter is composed of the following list elements:

Identifier Status
This parameter contains the value of the attribute Identifier Status of the Identifier Related Diagnosis object.

List of Channel Related Diagnosis Data
This parameter is composed of the following list elements:

Identifier Number
This parameter contains the value of the attribute Identifier Number of one Channel Related Diagnosis object.

Channel Number
This parameter contains the value of the attribute Channel Number of one Channel Related Diagnosis object.

Channel Type
This parameter contains the value of the attribute Channel Type of one Channel Related Diagnosis object.

IO Type
This parameter contains the value of the attribute IO Type of one Channel Related Diagnosis object.

Error Type
This parameter contains the value of one Channel Related Diagnosis object.

List of Status Data
This parameter is composed of the following list elements:

Slot Number
This parameter contains the value of the attribute Slot Number of one Status object.

Status Type
This parameter contains the value of the attribute Status Type for one Status object.

Status Specifier
This parameter contains the value of the attribute Status Specifier of one Status object.

Status Data
This parameter contains the value of one Status object.

Module Status Data
This parameter is composed of the following subparameters:

Status Specifier
This parameter contains the value of the attribute Status Specifier of the Module Status Data object.

List of Module Status
This parameter is composed of the following list elements:

Module Status
This parameter contains the value of the attribute Module Status of the Module Status Data object.

DXB-Link Status
This parameter is composed of the following subparameters:

Status Specifier
This parameter contains the value of the attribute Status Specifier of the DXB-Link Status object.

List of DXB-Link Status
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the attribute Publisher Address of the DXB-Link Status object.

Link Status
This parameter contains the value of the attribute Link Status of the DXB-Link Status object.

Link Error
This parameter contains the value of the attribute Link Error of the DXB-Link Status object.

Revision Number
This parameter contains the actual value of the Revision Number.

Result
This parameter indicates that the service request succeeded.

6.2.3.3.2 Get slave diag

This service may be used by a DP-master (Class 1) to read the previously received values of the Diagnosis objects of one or all assigned DP-slave which are related to the MS0 AR. Additionally the Get Slave Diag service provides the diagnosis information provided by the DP Application Layer of the DP-slave and the DP-master (Class 1). This service shall only be used in conjunction with the MS0 AR. Table 71 shows the parameters of the service.

Table 71 – Get Slave Diag

Parameter name	Req	Cnf
Argument	M	
AREP	M	
CREP	U	
Result(+)		S
AREP		M
CREP		C
List of Diag Data		M
Prm Req		M
Prm Fault		M
Cfg Fault		M
Not Supported		M
Station Not Ready		M
Stat Diag		M
WD On		M
Sync Mode		M
Freeze Mode		M
Master Add		M
Ident Number		M
Deactivated		M
Station Not Existent		M
Invalid Slave Response		M
Master Lock		M
Ext Diag Overflow		M
Ext Diag Flag		M
Ext Diag Data		C
Device Related Diagnosis Data		U
List of Identifier Status		U
Identifier Status		M
List of Channel Related Diagnosis Data		U
Identifier Number		M
Channel Number		M
Channel Type		M
IO Type		M
Error Type		M
List of Status Data		U
Slot Number		M
Status Type		M
Status Specifier		M
Status Data		M
Module Status Data		U
Status Specifier		M
List of Module Status		M

Parameter name	Req	Cnf
Module Status		M
DXB-Link Status		U
Status Specifier		M
List of DXB-Link Status		M
Publisher Address		M
Link Status		M
Link Error		M
Red Diagnosis		U
Red Status		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Red Status2		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Prm Command Ack		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialied		M
TOH started		M
Red Status2		M
Role		M

Parameter name	Req	Cnf
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialised		M
TOH started		M
Revision Number		U
Result(-)		S
AREP		M
CREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

CREP
If this parameter is present then the Diagnosis objects of a single DP-slave are read.

Result(+)
This selection type parameter indicates that the service request succeeded.

List of Diag Data
This parameter is composed of the following list elements:

Prm Req
This parameter contains the value of the attribute Prm Req of the MS0 ARL entry of a DP-slave.

Prm Fault
This parameter contains the value of the attribute Prm Fault of the MS0 ARL entry of a DP-slave.

Cfg Fault
This parameter contains the value of the attribute Cfg Fault of the MS0 ARL entry of a DP-slave.

Not Supported
This parameter contains the value of the attribute Not Supported of the MS0 ARL entry of a DP-slave.

Station Not Ready
This parameter contains the value of the attribute Station Not Ready of the MS0 ARL entry of a DP-slave.

Stat Diag
This parameter contains the value of the attribute Stat Diag of the MS0 ARL entry of a DP-slave.

WD On
This parameter contains the value of the attribute WD On of the MS0 ARL entry of a DP-slave.

Sync Mode
This parameter contains the value of the attribute Sync Mode of the MS0 ARL entry of a DP-slave.

Freeze Mode
This parameter contains the value of the attribute Freeze Mode of the MS0 ARL entry of a DP-slave.

Master Add
This parameter contains the value of the attribute Master Add of the MS0 ARL entry of a DP-slave.

Ident Number
This parameter contains the value of the attribute Ident Number of the MS0 ARL entry of a DP-slave.

Deactivated
This parameter contains the value of the attribute Deactivated of the of the related CRL entry of the correspondent DP-slave of the DP-master (Class 1).

Station Non Existent
This parameter contains the value of the attribute Station Non Existent of the related CRL entry of the correspondent DP-slave of the DP-master (Class 1).

Invalid Slave Response
This parameter contains the value of the attribute Invalid Slave Response of the related CRL entry of the correspondent DP-slave of the DP-master (Class 1).

Master Lock
This parameter indicates whether the DP-slave is locked by a DP-master. If the Master Lock parameter is TRUE the DP-slave is locked by another DP-master. If the parameter is False the DP-slave is not locked by another DP-master (Class 1).

Ext Diag Overflow
This Boolean parameter contains the value of the parameter Ext Diag Overflow of the Set Slave Diag service if not all delivered values of the Diagnosis objects are contained in this service confirmation. This parameter contains the value of the attribute Ext Diag Overflow of the related CRL entry of the correspondent DP-slave of the DP-master (Class 1) if not all delivered values of the Diagnosis objects are contained in this service confirmation due to storage problems.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

NOTE In case of an overflow condition it is allowed to truncate only at Diagnosis object limits.

Ext Diag Flag
This parameter contains the value of the parameter Ext Diag Flag of the Set Slave Diag service of a DP-slave.

Ext Diag Data
This parameter is composed of the following subparameters:

Device Related Diagnosis Data
This parameter contains the value of the Device Related Diagnosis object of a DP-slave.

List of Identifier Status
This parameter is composed of the following list elements:

Identifier Status
This parameter contains the value of the attribute Identifier Status of the Identifier Related Diagnosis object of a DP-slave.

List of Channel Related Diagnosis Data
This parameter is composed of the following list elements:

Identifier Number
This parameter contains the value of the attribute Identifier Number of one Channel Related Diagnosis object of a DP-slave.

Channel Number
This parameter contains the value of the attribute Channel Number of one Channel Related Diagnosis object of a DP-slave.

Channel Type
This parameter contains the value of the attribute Channel Type of one Channel Related Diagnosis object of a DP-slave.

IO Type
This parameter contains the value of the attribute IO Type of one Channel Related Diagnosis object of a DP-slave.

Error Type
This parameter contains the value of the attribute Error Type of one Channel Related Diagnosis object of a DP-slave.

List of Status Data
This parameter is composed of the following list elements:

Slot Number
This parameter contains the value of the attribute Slot Number of one Status object of a DP-slave.

Status Type
This parameter contains the value of the attribute Status Type of one Status object of a DP-slave.

Status Specifier
This parameter contains the value of the attribute Status Specifier of one Status object of a DP-slave.

Status Data
This parameter contains the value of one Status object of a DP-slave.

Module Status Data
This parameter is composed of the following subparameters:

Status Specifier
This parameter contains the value of the attribute Status Specifier of the Module Status Data object of a DP-slave.

List of Module Status
This parameter is composed of the following list elements:

Module Status
This parameter contains the value of the attribute Module Status of the Module Status Data object of a DP-slave.

DXB-Link Status
This parameter is composed of the following subparameters:

Status Specifier
This parameter contains the value of the attribute Status Specifier of the DXB-Link Status object of a DP-slave.

List of DXB-Link Status
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the attribute Publisher Address of the DXB-Link Status object of a DP-slave.

Link Status
This parameter contains the value of the attribute Link Status of the DXB-Link Status object of a DP-slave.

Link Error
This parameter contains the value of the attribute Link Error of the DXB-Link Status object of a DP-slave.

Red Diagnosis
This parameter is composed of the following subparameters:

Red Status
This selection type parameter indicates that the parameter Red Diagnosis contains the diagnosis information of the redundant communication interfaces of a redundant DP-Slave. This parameter is composed of the following subparameters:

Seq Number
This parameter contains the Sequence Number necessary for the unique identification of the received Red Status.

Allowed values: 0 to 31

Function
This parameter is composed of the following subparameters:

Primary Request
This Boolean parameter set to TRUE indicates that a Primary Request was executed at this communication interface. This parameter set to FALSE indicates that no Primary Request was executed at this communication interface.

Allowed values: TRUE, FALSE

MS1 Command
This parameter indicates if a MS1 Command like Start MS1, Stop MS1, Reset MS1 or no MS1 Command was executed at this communication interface.

Allowed values: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
This Boolean parameter set to TRUE indicates that a Check Properties request was executed at this communication interface. This parameter set to FALSE indicates that no Check Properties request was executed at this communication interface.

Allowed values: TRUE, FALSE

Master State Clear
This Boolean parameter set to TRUE indicates that the related DP-Master (Class 1) have set the communication interface in the state Clear. This parameter set to FALSE indicates that the related DP-Master (Class 1) have sent no Master State Clear command.

Allowed values: TRUE, FALSE

Red Status1
This parameter contains the redundancy status of the first communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that this communication interface acts as the primary interface. This parameter set to Backup indicates that this communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of this communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This Boolean parameter set to TRUE indicates that this communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that this communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of this communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Red Status2
This parameter contains the redundancy status of the second (redundant) communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that the other communication interface acts as the primary interface. This parameter set to Backup indicates that the other communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of the other communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This Boolean parameter set to TRUE indicates that the other communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that the other communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of the other communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Prm Command Ack
This selection type parameter indicates that the parameter Red Diagnosis contains the acknowledgment of a previously received Prm Command. This parameter is composed of the following subparameters:

Seq Number
This parameter contains the Sequence Number necessary for the unique identification of the received Prm Command Ack.

Allowed values: 0 to 31

Function
This parameter is composed of the following subparameters:

Primary Request
This Boolean parameter set to TRUE indicates that a Primary Request was executed at this communication interface. This parameter set to FALSE indicates that no Primary Request was executed at this communication interface.

Allowed values: TRUE, FALSE

MS1 Command
This parameter indicates if a MS1 Command like Start MS1, Stop MS1, Reset MS1 or no MS1 Command was executed at this communication interface.

Allowed values: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
This Boolean parameter set to TRUE indicates that a Check Properties request was executed at this communication interface. This parameter set to FALSE indicates that no Check Properties request was executed at this communication interface.

Allowed values: TRUE, FALSE

Master State Clear
This Boolean parameter set to TRUE indicates that the related DP-Master (Class 1) have set the communication interface in the state Clear. This parameter set to FALSE indicates that the related DP-Master (Class 1) have sent no Master State Clear command.

Allowed values: TRUE, FALSE

Red Status1
This parameter contains the redundancy status of the first communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that this communication interface acts as the primary interface. This parameter set to Backup indicates that this communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of this communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This Boolean parameter set to TRUE indicates that this communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that this communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of this communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Red Status2
This parameter contains the redundancy status of the second (redundant) communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that the other communication interface acts as the primary interface. This parameter set to Backup indicates that the other communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment

of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of the other communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This Boolean parameter set to TRUE indicates that the other communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that the other communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of the other communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Revision Number
This parameter contains the actual value of the Revision Number.

Result(-)
This selection type parameter indicates that the service request failed.

6.2.3.3.3 Read slave diag

This service may be used by a DP-master (Class 2) to read acyclic the values of the Diagnosis objects of one DP-slave via a MS0 AR. Additionally the Read Slave Diag service provides the diagnosis information provided by the DP Application Layer of the DP-slave and the DP-master (Class 2).

Table 72 shows the parameters of the service.

Table 72 – Read Slave Diag

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Diag Data		M

Parameter name	Req	Cnf
Prm Req		M
Prm Fault		M
Cfg Fault		M
Not Supported		M
Station Not Ready		M
Stat Diag		M
WD On		M
Sync Mode		M
Freeze Mode		M
Master Add		M
Ident Number		M
Ext Diag Overflow		M
Ext Diag Flag		M
Ext Diag Data		C
Device Related Diagnosis Data		U
List of Identifier Status		U
Identifier Status		M
List of Channel Related Diagnosis Data		U
Identifier Number		M
Channel Number		M
Channel Type		M
IO Type		M
Error Type		M
List of Status Data		U
Slot Number		M
Status Type		M
Status Specifier		M
Status Data		M
Module Status Data		U
Status Specifier		M
List of Module Status		M
Module Status		M
DXB-Link Status		U
Status Specifier		M
List of DXB-Link Status		M
Publisher Address		M
Link Status		M
Link Error		M
Red Diagnosis		U
Red Status		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M

Parameter name	Req	Cnf
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Red Status2		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Prm Command Ack		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialied		M
TOH started		M
Red Status2		M
Role		M
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialised		M
TOH started		M
Revision Number		U
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Diag Data
This parameter is composed of the following list elements:

Prm Req
This parameter contains the value of the attribute Prm Req of the MS0 ARL entry of the selected DP-slave.

Prm Fault
This parameter contains the value of the attribute Prm Fault of the MS0 ARL entry of the selected DP-slave.

Cfg Fault
This parameter contains the value of the attribute Cfg Fault of the MS0 ARL entry of the selected DP-slave.

Not Supported
This parameter contains the value of the attribute Not Supported of the MS0 ARL entry of the selected DP-slave.

Station Not Ready
This parameter contains the value of the attribute Station Not Ready of the MS0 ARL entry of the selected DP-slave.

Stat Diag
This parameter contains the value of the attribute Stat Diag of the MS0 ARL entry of the selected DP-slave.

WD On
This parameter contains the value of the attribute WD On of the MS0 ARL entry of the selected DP-slave.

Sync Mode
This parameter contains the value of the attribute Sync Mode of the MS0 ARL entry of the selected DP-slave.

Freeze Mode
This parameter contains the value of the attribute Freeze Mode of the MS0 ARL entry of the selected DP-slave.

Master Add
This parameter contains the value of the attribute Master Add of the MS0 ARL entry of the selected DP-slave.

Ident Number
This parameter contains the value of the attribute Ident Number of the MS0 ARL entry of the selected DP-slave.

Ext Diag Overflow
This Boolean parameter contains the value of the parameter Ext Diag Overflow of the Set Slave Diag service if not all delivered values of the Diagnosis objects are contained in this service confirmation. This parameter will be set to TRUE by the AP of the DP-master (Class 2) if not all delivered values of the Diagnosis objects are contained in this service confirmation due to storage problems.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

NOTE In case of an overflow condition it is allowed to truncate only at Diagnosis object limits.

Ext Diag Flag
This parameter contains the value of the parameter Ext Diag Flag of the Set Slave Diag service of the selected DP-slave.

Ext Diag Data
This parameter is composed of the following subparameters:

Device Related Diagnosis Data
This parameter contains the value of the Device Related Diagnosis object of the selected DP-slave.

List of Identifier Status
This parameter is composed of the following list elements:

Identifier Status
This parameter contains the value of the attribute Identifier Status of the Identifier Related Diagnosis object of the selected DP-slave.

List of Channel Related Diagnosis Data
This parameter is composed of the following list elements:

Identifier Number
This parameter contains the value of the attribute Identifier Number of one Channel Related Diagnosis object of the selected DP-slave.

Channel Number
This parameter contains the value of the attribute Channel Number of one Channel Related Diagnosis object of the selected DP-slave.

Channel Type
This parameter contains the value of the attribute Channel Type of one Channel Related Diagnosis object of the selected DP-slave.

IO Type
This parameter contains the value of the attribute IO Type of one Channel Related Diagnosis object of the selected DP-slave.

Error Type
This parameter contains the value the attribute Error Type of one Channel Related Diagnosis object of the selected DP-slave.

List of Status Data
This parameter is composed of the following list elements:

Slot Number
This parameter contains the value of the attribute Slot Number of one Status object of the selected DP-slave.

Status Type
This parameter contains the value of the attribute Status Type of one Status object of the selected DP-slave.

Status Specifier
This parameter contains the value of the attribute Status Specifier of one Status object of the selected DP-slave.

Status Data
This parameter contains the value of one Status object of the selected DP-slave.

Module Status Data
This parameter is composed of the following subparameters:

Status Specifier
This parameter contains the value of the attribute Status Specifier of the Module Status Data object of the selected DP-slave.

List of Module Status
This parameter is composed of the following list elements:

Module Status
This parameter contains the value of the attribute Module Status of the Module Status Data object of the selected DP-slave.

DXB-Link Status
This parameter is composed of the following subparameters:

Status Specifier
This parameter contains the value of the attribute Status Specifier of the DXB-Link Status object of a DP-slave.

List of DXB-Link Status
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the attribute Publisher Address of the DXB-Link Status object of a DP-slave.

Link Status
This parameter contains the value of the attribute Link Status of the DXB-Link Status object of a DP-slave.

Link Error
This parameter contains the value of the attribute Link Error of the DXB-Link Status object of a DP-slave.

Red Diagnosis
This parameter is composed of the following subparameters:

Red Status
This selection type parameter indicates that the parameter Red Diagnosis contains the diagnosis information of the redundant communication interfaces of a redundant DP-Slave. This parameter is composed of the following subparameters:

Seq Number
This parameter contains the Sequence Number necessary for the unique identification of the received Red Status.

Allowed values: 0 to 31

Function
This parameter is composed of the following subparameters:

Primary Request
This Boolean parameter set to TRUE indicates that a Primary Request was executed at this communication interface. This parameter set to FALSE indicates that no Primary Request was executed at this communication interface.

Allowed values: TRUE, FALSE

MS1 Command
This parameter indicates if a MS1 Command like Start MS1, Stop MS1, Reset MS1 or no MS1 Command was executed at this communication interface.

Allowed values: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
This Boolean parameter set to TRUE indicates that a Check Properties request was executed at this communication interface. This parameter set to FALSE indicates that no Check Properties request was executed at this communication interface.

Allowed values: TRUE, FALSE

Master State Clear
This Boolean parameter set to TRUE indicates that the related DP-Master (Class 1) have set the communication interface in the state Clear. This parameter set to FALSE indicates that the related DP-Master (Class 1) have sent no Master State Clear command.

Allowed values: TRUE, FALSE

Red Status1
This parameter contains the redundancy status of the first communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that this communication interface acts as the primary interface. This parameter set to Backup indicates that this communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of this communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This Boolean parameter set to TRUE indicates that this communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that this communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of this communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Red Status2
This parameter contains the redundancy status of the second (redundant) communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that the other communication interface acts as the primary interface. This parameter set to Backup indicates that the other communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of the other communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This Boolean parameter set to TRUE indicates that the other communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that the other communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of the other communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Prm Command Ack
This selection type parameter indicates that the parameter Red Diagnosis contains the acknowledgment of a previously received Prm Command. This parameter is composed of the following subparameters:

Seq Number
This parameter contains the Sequence Number necessary for the unique identification of the received Prm Command Ack.

Allowed values: 0 to 31

Function
This parameter is composed of the following subparameters:

Primary Request
This Boolean parameter set to TRUE indicates that a Primary Request was executed at this communication interface. This parameter set to

FALSE indicates that no Primary Request was executed at this communication interface.

Allowed values: TRUE, FALSE

MS1 Command
This parameter indicates if a MS1 Command like Start MS1, Stop MS1, Reset MS1 or no MS1 Command was executed at this communication interface.

Allowed values: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
This Boolean parameter set to TRUE indicates that a Check Properties request was executed at this communication interface. This parameter set to FALSE indicates that no Check Properties request was executed at this communication interface.

Allowed values: TRUE, FALSE

Master State Clear
This Boolean parameter set to TRUE indicates that the related DP-Master (Class 1) have set the communication interface in the state Clear. This parameter set to FALSE indicates that the related DP-Master (Class 1) have sent no Master State Clear command.

Allowed values: TRUE, FALSE

Red Status1
This parameter contains the redundancy status of the first communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that this communication interface acts as the primary interface. This parameter set to Backup indicates that this communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of this communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This Boolean parameter set to TRUE indicates that this communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that this communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of this communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Red Status2
This parameter contains the redundancy status of the second (redundant) communication interface of a redundant DP-Slave. This parameter is composed of the following subparameters:

Role
This parameter set to Primary indicates that the other communication interface acts as the primary interface. This parameter set to Backup indicates that the other communication interface acts as the Backup interface. This parameter set to Power On indicates that the assignment of the roles between the two communication interfaces is not finished yet (e. g. at Power On of the DP-Slave).

Allowed values: Primary, Backup, Power On

HW Defect
This Boolean parameter set to TRUE indicates that an HW Defect of the other communication interface has been detected.

Allowed values: TRUE, FALSE

Data Exchange
This vparameter set to TRUE indicates that the other communication interface is in the data exchange state.

Allowed values: TRUE, FALSE

Safe State
This Boolean parameter set to TRUE indicates that the other communication interface is in the Fail Safe state.

Allowed values: TRUE, FALSE

DL Initialised
This Boolean parameter set to TRUE indicates that the DL parameter of the other communication interface has been loaded.

Allowed values: TRUE, FALSE

TOH started
This Boolean parameter set to TRUE indicates that the Output Hold Timer of this communication interface has been started.

Allowed values: TRUE, FALSE

Revision Number
This parameter contains the value of the Revision Number of the selected DP-slave.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter contains the reason for the failure.

Allowed values: RE, DS, UE, RS, NA, NR, RR

6.2.3.3.4 New slave diag

This service may be used by a DP-master (Class 1) to indicate to the application that new received values of the Diagnosis objects of one assigned DP-slave which is related to the MS0 AR, is present. Additionally this service may be used to indicate that new diagnosis information provided by the DP Application Layer of the DP-slave or the DP-master (Class 1) is present.

Table 73 shows the parameters of the service.

Table 73 – New Slave Diag

Parameter name	Ind
Argument	M
AREP	M
CREP	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the related AR.

CREP
This parameter is the local identifier of the related CR (DP-slave).

6.2.4 Alarm ASE

6.2.4.1 Overview

The alarm model allows the transfer of an alarm from the DP-slave to the assigned DP-master (Class 1) and the explicit acknowledgement of the alarm by the DP-master (Class 1) via the MS1 AR.

Figure 24 shows the alarm interactions between the DP-slave (Alarm Notification) and the DP-master (Alarm Ack).

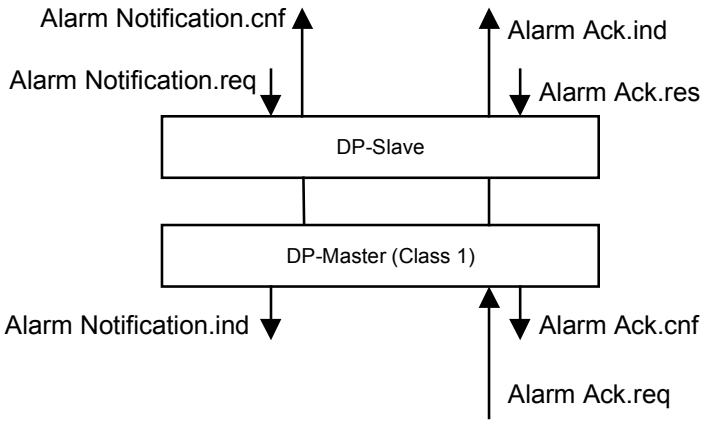


Figure 24 – Treatment of an alarm in the DP system

For the transmission of an alarm the following conditions have to be fulfilled:

– the DP-slave has to be in the data exchange mode,
– the MS1 AR has to be open,
– the corresponding alarm type has to be enabled,
– the limit of active alarms is not exceeded.

In addition to the alarm transmission a Channel Related and/or Identifier Related Diagnosis may be issued which identify the exact source of the alarm.

Two alternatives of the alarm handling (see 6.2.10.3.1) are defined:

– only one alarm of a specific alarm type can be active at the same time (Type Mode),
– several alarms (2 to 32) independent of the type can be active at the same time (Sequence Mode).

The DP-master (Class 1) and the DP-slave shall implement the Type Mode and may additionally implement the Sequence Mode if alarms are supported. Alarms will be queued. The alarm remains active until the DP-master (Class 1) explicitly acknowledges the alarm.

The alarm acknowledgement is either performed via the CR, that conveys the Read and Write service, or optionally via a separate CR on the MS1 AR. With this latter feature, the alarms will not be delayed by read and write services.

The following alarm types are defined. They may be extended specific to the manufacturer:

Diagnosis Alarm
A diagnosis alarm signals an event within a module, for instance overtemperature, short circuit, etc.

Process Alarm
A process alarm signals the occurrence of an event in the connected process, for instance upper limit value exceeded.

Pull Alarm
A slot signals the withdrawal of a module.

Plug Alarm
A slot signals the insertion of a module.

Status Alarm
A status alarm signals a change in the state of a module, for instance run, stop or ready.

Update Alarm
An update alarm signals the change of a parameter in a module e.g. by a local operation or a remote access.

6.2.4.2 Alarm class specification

6.2.4.2.1 Template

An Alarm object is described by the following template:

DP ASE:			**Alarm ASE**
CLASS:			**ALARM DATA**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attributes:	Identifier
2.	(m)	Attribute:	Add Ack
3.	(m)	Attribute:	Alarm Specifier
4.	(o)	Attribute:	Alarm Data Description
SERVICES:			
1.	(m)	OpsService:	Alarm Notification
2.	(m)	OpsService:	Alarm Ack

6.2.4.2.2 Attributes

Identifier
This key attribute is composed of Slot Number and Alarm Type to indicate the number of the slot and the alarm type.

Allowed values for Slot Number: 0 to 254

The allowed values of Alarm Type are shown in Table 74.

Table 74 – Alarm type

Value	Meaning
1	Diagnosis Alarm
2	Process Alarm
3	Pull Alarm
4	Plug Alarm
5	Status Alarm
6	Update Alarm
32-126	Manufacturer Specific

Add Ack
This attribute indicates to the DP-master (Class 1) that this alarm requires an additional User acknowledgement. This can be done for instance by a write service. The alarm remains active after the Alarm Ack.

The allowed values are shown in Table 75.

Table 75 – Add Ack

Value	Meaning
TRUE	additional acknowledgement required
FALSE	no additional acknowledgement required

Alarm Specifier
This attribute specifies additional alarm information.

The allowed values are shown in Table 76.

Table 76 – Alarm specifier

Value	Meaning
0	no further differentiation
1	Alarm appears and the related module is disturbed
2	Alarm disappears and the related module has no further errors
3	Alarm disappears and the related module is still disturbed

Alarm Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

6.2.4.2.3 Invocation of the alarm data object

For the invocation of the Alarm Data object the following rules apply:

– Alarm Data objects shall be used only if the device attribute DPV1 = TRUE,
– Several Alarm Data objects can be invoked in a DP-slave,
– For each module only one Alarm Data object of one type shall be invoked,
– Alarm Data objects shall not exceed the total length of 59 octets. The attribute Alarm Data Description has to be set accordingly.

6.2.4.3 Alarm service specification

6.2.4.3.1 Alarm notification

This service is used to transfer an alarm notification from the DP-slave to the DP-master (Class 1). This service shall only be used in conjunction with the MS1 AR. Table 77 shows the parameters of the service.

Table 77 – Alarm notification

Parameter name	Req	Ind	Cnf
Argument	M	M(=)	
AREP	M	M(=)	
Slot Number	M	M(=)	
Alarm Type	M	M(=)	
Seq Nr	M	M(=)	
Add Ack	M	M(=)	
Alarm Specifier	M	M(=)	
Alarm Data	U	C(=)	
Result(+)			S
AREP			M
Slot Number			M
Alarm Type			M
Seq Nr			M
Result(-)			S
AREP			M
Slot Number			M
Alarm Type			M
Seq Nr			M
Status			M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
This parameter contains the value of the attribute Slot Number of the Alarm Data object.

Alarm Type
This parameter contains the value of the attribute Alarm Type of the Alarm Data object.

Seq Nr
This parameter is used as a unique identifier for the transaction of an alarm notification by the DP Application Layer.

Allowed values: 0 to 31

Add Ack
This parameter contains the value of the attribute Add Ack of the Alarm Data object.

Alarm Specifier
This parameter contains the value of the attribute Alarm Specifier of the Alarm Data object.

Alarm Data
This parameter contains the value of the Alarm Data object.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter contains the reason for the failure.

Allowed values: MSAL1S_Stopped, Not_Enabled, Limit_Expired, Sequence_Nr_Pending, No_Start

6.2.4.3.2 Alarm ack

This service is used by the DP-master (Class 1) to acknowledge the reception of an alarm notification which has been previously received from the DP-slave. This service shall only be used in conjunction with the MS1 AR.

Table 78 shows the parameters of the service.

Table 78 – Alarm Ack

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Alarm Type	M	M(=)		
Seq Nr	M	M(=)		
Result(+)			S	S
AREP			M	M(=)
Slot Number			M	M(=)
Alarm Type			M	M(=)
Seq Nr			M	M(=)
Result(-)			S	S
AREP			M	M(=)
Slot Number			M	M(=)
Alarm Type			M	M(=)
Seq Nr			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
This parameter contains the value of the attribute Slot Number of the of the previously received Alarm Data object.

Alarm Type
This parameter contains the value of the attribute Alarm Type of the previously received Alarm Data object.

Seq Nr
This parameter contains the sequence number of the previously received alarm notification.

Allowed values: 0 to 31

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

6.2.5 Context ASE

6.2.5.1 Overview

The Context ASE provides a set of services to parameterize and configure the Default AP of a DP-slave itself and its modules containing I/O data.

The Application Process of many DP-slaves need parameter (e.g. the range of an analogue signal) from the assigned DP-master (Class 1) to provide the data from the Input Data object and/or for the Output Data object according to the requirements of the AP of the DP-master (Class 1). These DP-slaves provide in the Context ASE a MS0 Parameter object or, for more flexibility, several MS0 Structured Parameter objects. The values for the attributes of these objects will be provided by the DP-master (Class 1) and assigned after validation from the AP of the DP-slave.

The Context ASE provides objects and the related services for the flexible parameterization of a DP-slave. The following objects are defined:

– MS0 User Parameter object (basic parameterization object),
– MS0 Structured User Parameter object,
– DXB-Linktable object,
– DXB-Subscribertable object,
– IsoM Parameter object,
– Time AR Parameter object.

The functionality of a DP-slave application implies the necessary parameterization objects which have to be conveyed at the start-up phase of the DP system from the DP-master (Class 1) to the assigned DP-slaves via the MS0 AR. The application of the DP-slave has to check the parameterization data sent by the DP-master (Class 1). If the check fails the DP-slave may respond with an appropriate entry in the Diagnosis information.

The MS0 Configuration Elements object describes the module structure, the range of input and output data and contains information about the data consistency. The Context ASE defines a set of services used to set and get their values. The application of the DP-slave will compare the configuration data sent by the DP-master (Class 1) with the local configuration of the DP-slave. If they match the MS0 AR to this DP-slave will be established. The real configuration of the DP-slave may be read by a DP-master (Class 2) with the related service.

The DXB-Subscribertable object in conjunction with the MS0 Configuration Elements object provides the information the Subscriber needs to map the output data of the DP-master (Class 1) and/or the subscribed Input Data of one or more Publisher to the data areas of the Output Data Elements of the DP-slave.

The User shall check the contents of the DXB-Subscribertable provided with the Check User Prm service or Check Ext User Prm service and the configuration information provided with the Check Cfg service. In case of a positive result of these checks the User shall extract the necessary information from the DXB-Subscribertable the DP Application Layer needs for subscription. The User shall load this information with the Load CRL DXB Link Entries service.

For this mapping the following rules shall apply:

The data areas of an Output Data Element may be assigned:

– completely to the DP-master (Class 1) (default), or
– partially to the DP-master (Class 1) and partially to one or more Publisher, or
– completely to one or more Publisher.

The lowest level of mapping information contained in the DXB-Subscribertable is one octet, no bit mapping will be supported.

It is allowed that for one or more data areas of an Output Data Element no values are provided from the DP-master (Class 1) or from a Publisher. For these data areas default values (zero) shall be used.

A data area of an Output Data Element shall always be supplied completely either by the DP-master (Class 1) or by a Publisher. It is not allowed to provide values only for a part of a data area of an Output Data Element.

Variations of the data areas (length of the Output Data) which are supplied by the DP-master (Class 1) may be achieved by variations of the Simple or Extended Output Data Description for this Output Data Element.

For those Output Data Elements of a DP-slave where no Output Data are required from the DP-master (Class 1) the attribute Length of the Simple or Extended Output Data Description (no Output Data) shall be set to zero. For these modules no Output Data will be transferred from the DP-master (Class 1).

The data areas of an Output Data Element which are usually assigned to the DP-master (Class 1) should be placed at the beginning of the Output Data Element. In this case no mapping information is necessary in the DXB-Subscribertable (default mapping).

The following checks may be accomplished for each module at the Subscriber depending on the information provided with the Check Cfg service and the information contained in the DXB-Subscribertable.

Case 1:

The length indicated in the Simple or Extended Output Description of a module is set to zero (no Output Data from the DP-master (Class 1)) and the DXB-Subscribertable contains at least one entry for this module then the following rules shall be applied:

— The sum of the length given in the attribute Length Data Area of the entries in the DXB-Subscribertable for the Simple or Extended Output Data Element shall be the length of the Simple or Extended Output Data Element, if there exist an entry for each data area of the Output Data Element.
— Overlapping of the values of the attributes Offset Data Area and Length Data Area of the entries in the DXB-Subscribertable for this Output Data Element is not allowed.
— The DXB-Subscribertable shall not contain any DXB Subscriber Entry of the DP-master (Class 1) (Master Data) for this Output Data Element.
— Not covered data areas of the Output Data Element shall be supplied with default values.

Case 2:

The length indicated in the Simple or Extended Output Description of a module is unequal to zero and smaller than the complete length of the Output Data Element (parts of the Output Data for this Output Data Element will be provided by the DP-master (Class 1)) and the DXB-

Subscribertable contains at least one entry for this module then the following rules shall be applied:

— The sum of the values of the attributes Length Data Area of the DXB Subscriber Entries in the DXB-Subscribertable of the DP-master (Class 1) (Master Data) shall be equal the length given in the Simple or Extended Output Data Description of this Output Data Element.

— If the DXB-Subscribertable contains no entries for the DP-master (Class 1) (Master Data) the values for this Output Data Element from the DP-master (Class 1) shall be mapped to the first data areas of this Output Data Element (default mapping).

— The mapping of the subscribed data of the Publisher shall be done according to the entries in the DXB-Subscribertable.

— Overlapping of the values of the attributes Offset Data Area and Length Data Area of the entries in the DXB-Subscribertable for this Output Data Element is not allowed also not with the data areas supplied by the DP-master (Class 1) via the default mapping.

— Not covered data areas of the Output Data Element shall be supplied with default values.

Case 3:

The length indicated in the Simple or Extended Output Description of a module is equal to the length of the Output Data Element (all Output Data for this Output Data Element will be provided by the DP-master (Class 1)) then the following rule shall be applied:

— The DXB-Subscribertable shall contain no entries for this Output Data Element.

The Context ASE also provides the means to change the address of a DP-slave via the MS0 AR. This may be done by a DP-master (Class 2).

The length of an Input Data Element of a module shall not be influenced by the possible variations of assignments of the data areas of the Output Data Element.

The Context ASE is used to establish or release an association between an individual AP of a DP-master (Class 2) and an individual AP of the DP-slave for acyclic communication. This association exclusively belongs to the Process Data ASE.

The formal description of the Context ASE is presented next, followed by a description of its services.

6.2.5.2 Context class specification

6.2.5.2.1 MS0 user parameter class specification

6.2.5.2.1.1 Template

This class specifies the object for the parameter data that may be related to the DP-slave itself and/or to the modules of the DP-slave.

The MS0 User Parameter object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**MS0 User Parameter**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(o)	Attribute:	User Prm Data
2.1.	(o)	Attribute:	Device Related User Parameter
2.1.1	(m)	Attribute:	Data Description
2.2.	(o)	Attribute:	List of Module Related User Parameter
2.2.1	(m)	Attribute:	Data Description
SERVICES:			

1. (m) OpsService: Check User Prm
2. (m) OpsService: Check User Prm Result

6.2.5.2.1.2 Attributes

Implicit
The attribute Implicit indicates that the User Parameter object is implicitly addressed by the services.

User Prm Data
This attribute is an aggregate of the Device and/or List of Module Related User Parameter and is composed of the following elements.

Device Related User Parameter
This attribute is composed of following elements:

Data Description
This attribute contains either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

List of Module Related User Parameter
This attribute is composed of following list elements (each element corresponds to a module).

Data Description
This attribute contains either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

6.2.5.2.1.3 Invocation of the MS0 user parameter object

For the invocation of the MS0 User Parameter object the following rules apply:

– The MS0 User Parameter object shall not exceed the total length of 237 octets (only 234 if the device attribute DPV1 = TRUE). The attributes Data Description and List of Module Related User Parameter have to be set accordingly.
– The MS0 User Parameter object shall exist only if the device attribute DPV1 is set to TRUE and the parameter Prm Structure provided with the Check User Prm service is set to FALSE or if the device attribute DPV1 is set to FALSE.
– Only one MS0 User Parameter object can be invoked in a DP-slave.

6.2.5.2.2 MS0 structured user parameter class specification

6.2.5.2.2.1 Template

In contrast to the MS0 User Parameter object this class defines separate objects for the device itself and for each of its modules. Furthermore these objects allow the unique identification and assignment.

This structuring should be used by enhanced devices with a high variation of parameters which may not be present in every application. This object is mandatory for DP-slaves which support the Isochronous Mode and for the conveyance of the DXB-Linktable.

The MS0 Structured User Parameter object is described by the following template:

DP ASE: **Context ASE**
CLASS: **MS0 Structured User Parameter**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Key Attribute: Identifier
2. (m) Attribute: Structured User Prm Data

2.1.	(s)	Attribute:	Device Related User Parameter
2.1.1	(m)	Attribute:	Data Description
2.2.	(s)	Attribute:	Module Related User Parameter
2.2.1	(m)	Attribute:	Data Description

SERVICES:

1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.2.2 Attributes

Identifier
This key attribute is composed of Slot Number and Prm Data Type to indicate the number of the slot and the type of parameter data the specified MS0 Structured User Parameter object contains. If the attribute Slot Number contains the value 0 the following Structured Prm Data are Device Related User Parameter, otherwise they are Module Related User Parameter

Attribute Type of Slot Number: Unsigned8

Allowed values for Slot Number: 0 to 244

Attribute Type of Prm Data Type:

The allowed values of Prm Data Type are shown in Table 79.

Table 79 – Prm data type

Value	Meaning
129	Parameter Data
32–128	Manufacturer Specific

Structured User Prm Data
This attribute contains either Device or Module Related User Parameter indicated by the attribute Identifier and is composed of the following elements.

Device Related User Parameter
This attribute is composed of following elements:

Data Description
This attribute contains either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Module Related User Parameter
This attribute is composed of following elements.

Data Description
This attribute contains either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

6.2.5.2.2.3 Invocation of the MS0 structured user parameter object

For the invocation of the MS0 Structured User Parameter object the following rules apply:

– MS0 Structured User Parameter objects shall exist only if the device attribute DPV1 is set to TRUE.

– The MS0 Structured User Parameter objects may be provided with the Check User Prm service if the parameter Prm Structure provided with the Check User Prm service is set to TRUE.

– The description of all MS0 Structured User Parameter objects shall not exceed the total length of 230 octets if the values are provided with the Check User Prm service. The attribute Data Description has to be set accordingly.
– The description of all MS0 Structured User Parameter objects shall not exceed the total length of 240 octets if the values are provided with the Check Ext User Prm service. The attribute Data Description has to be set accordingly.
– Several MS0 Structured User Parameter objects may be invoked in a DP-slave.
– Only one MS0 Structured User Parameter object of the same type shall be invoked for each slot.

6.2.5.2.3 DXB-linktable class specification

6.2.5.2.3.1 Template

This class specifies the entries of the DXB-Linktable. Each entry specifies a Publisher the Subscriber has to subscribe from.

The DXB-Linktable object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**DXB-Linktable**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of DXB Link Entries
2.1.	(m)	Attribute:	Publisher Address
2.2.	(m)	Attribute:	Input Data Length Publisher
2.3.	(m)	Attribute:	Source Offset Publisher
2.4.	(m)	Attribute:	Length
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.3.2 Attributes

Implicit
The attribute Implicit indicates that the DXB-Linktable object is implicitly addressed by the services.

List of DXB Link Entries
This attribute is composed of the following list elements:

Publisher Address
This attribute indicates the DL-address of a DP-slave (Publisher) the Subscriber shall subscribe from. The following rules shall apply for this attribute:

A value between 0 and 125 represents a valid Publisher address and therefore an active DXB-Link.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Input Data Length Publisher
This attribute defines the number of octets of the Simple or Extended Input Data object of the related Publisher.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Source Offset Publisher
This attribute defines the beginning (offset) of the subscription within the Simple or Extended Input Data object of the related Publisher given by the attribute Publisher Address.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Length
This attribute defines the number of octets the Subscriber shall subscribe from the related Publisher starting at the offset given in the attribute Source Offset Publisher.

Attribute Type: Unsigned8

Allowed values: 1 to 244

6.2.5.2.3.3 Invocation of the DXB-linktable object

For the invocation of the DXB-Linktable object the following rules apply:

– The DXB-Linktable object shall exist only if the device attribute DPV1 of the DP-slave is set to TRUE.
– Only one entry for each Publisher shall be contained in the DXB-Linktable.
– The values for the DXB-Linktable object may be provided with the Check User Prm service if the parameter Prm Structure provided with the Check User Prm service is set to TRUE.
– The description of the DXB-Linktable object shall not exceed the total length of 230 octets if the values of the DXB-Linktable object are provided with the Check User Prm service. The attribute List of DXB Link Entries has to be set accordingly.
– The description of the DXB-Linktable object shall not exceed the total length of 240 octets if the values of the DXB-Linktable object are provided with the Check Ext User Prm service. The attribute List of DXB Link Entries has to be set accordingly.
– Only one DXB-Linktable object shall be invoked in a DP-slave.
– The DXB-Linktable object shall only be invoked in a DP-slave if no DXB-Subscribertable object is invoked and vice versa.

6.2.5.2.4 DXB-subscribertable class specification

6.2.5.2.4.1 Template

This class specifies the entries of the DXB-Subscribertable. Each entry specifies the source, a Publisher or the DP-master (Class 1), and the related mapping information needed to transfer the correspondent data to the data areas of an Output Data Element of the related module.

The DXB-Subscribertable object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**DXB-Subscribertable**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of DXB Subscriber Entries
2.1.	(s)	Attribute:	Publisher Address
2.1.1.	(m)	Attribute:	Input Data Length Publisher
2.1.2.	(m)	Attribute:	Source Offset Publisher
2.2.	(s)	Attribute:	Master Data
2.2.1.	(m)	Attribute:	Source Offset Master
2.3.	(m)	Attribute:	Dest Slot Number
2.4.	(m)	Attribute:	Offset Data Area
2.5.	(m)	Attribute:	Length Data Area

SERVICES:

1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.4.2 Attributes

Implicit
The attribute Implicit indicates that the DXB-Subscribertable object is implicitly addressed by the services.

List of DXB Subscriber Entries
This attribute is composed of the following list elements.

Publisher Address
This selection attribute indicates the DL-address of a DP-slave (Publisher) the Subscriber shall subscribe from. The following rules shall apply for this attribute:

A value between 0 and 125 represents a valid Publisher address and therefore an active DXB-Link.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Input Data Length Publisher
This attribute defines the number of octets of the Simple or Extended Input Data object of the related Publisher.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Source Offset Publisher
This attribute defines the beginning (offset) of the subscription within the Simple or Extended Input Data object of the related Publisher given by the attribute Publisher Address.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Master Data
This selection attribute indicates that for this entry the data shall be derived from the related DP-master (Class 1).

Attribute Type: Unsigned8

Allowed values: 128

Source Offset Master
This attribute defines the beginning (offset) within the Output Data of the DP-master (Class 1).

Attribute Type: Unsigned8

Allowed values: 0 to 243

Dest Slot Number
This attribute indicates the Slot Number of the module to which the subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped.

Attribute Type: Unsigned8

Allowed values: 0 to 244

Offset Data Area
This attribute defines the beginning (offset) within the Output Data of the module (data area) to which the subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped.

Attribute Type: Unsigned8

Allowed values: 0 to 244

Length Data Area
This attribute defines the number of octets from the related Publisher the Subscriber shall subscribe to, starting at the offset given in the attribute Source Offset Publisher in case of a Publisher. If the data shall be derived from the DP-master (Class 1) the starting offset is given in the attribute Source Offset Master.

The subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped to the data area of the module given in the attributes Dest Slot Number and Offset Data Area of the related DXB Subscriber Entry.

Attribute Type: Unsigned8

Allowed values: 1 to 244

6.2.5.2.4.3 Invocation of the DXB-subscribertable object

For the invocation of the DXB-Subscribertable object the following rules apply:

– The DXB-Subscribertable object shall exist only if the device attribute DPV1 of the DP-slave is set to TRUE.
– The values for the DXB-Subscribertable object may be provided with the Check User Prm service if the parameter Prm Structure provided with the Check User Prm service is set to TRUE.
– The description of the DXB-Subscribertable object shall not exceed the total length of 230 octets if the values of the DXB-Subscribertable object are provided with the Check User Prm service. The attribute List of DXB Link Entries has to be set accordingly.
– The description of the DXB-Subscribertable object shall not exceed the total length of 240 octets if the values of the DXB-Subscribertable object are provided with the Check Ext User Prm service. The attribute List of DXB Link Entries has to be set accordingly.
– Only one DXB-Subscribertable object shall be invoked in a DP-slave.
– The DXB-Subscribertable object shall only be invoked in a DP-slave if no DXB-Linktable object is invoked and vice versa.

6.2.5.2.5 IsoM parameter class specification

6.2.5.2.5.1 Template

This class specifies the necessary parameter for a DP-slave to operate in the Isochronous Mode.

The IsoM Parameter object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**IsoM Parameter**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	T_{BASE_DP}
3.	(m)	Attribute:	T_{DP}
4.	(m)	Attribute:	T_{MAPC}
5.	(m)	Attribute:	T_{BASE_IO}
6.	(m)	Attribute:	T_I
7.	(m)	Attribute:	T_O
8.	(m)	Attribute:	T_{DX}
9.	(m)	Attribute:	T_{PLL_W}
10.	(m)	Attribute:	T_{PLL_D}
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.5.2 Attributes

Implicit
The attribute Implicit indicates that the IsoM Parameter object is implicitly addressed by the services.

T_{BASE_DP}
This attribute indicates the time base (in units of 1/12 µs) for the time T_{DP}.

Allowed values: 375, 750, 1 500, 3 000, 6 000, 12 000 (the value 1 500 is mandatory for standard Isochronous Mode)

T_{DP}
This attribute indicates the time T_{DP} (in units of T_{BASE_DP}. This time includes all parts of the Isochronous DP cycle:

– synchronization message (SYNCH),
– cyclic services (DX),
– acyclic services (MSG),
– token (TOK),
– spare time (RES).

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

T_{MAPC}
This attribute indicates the Master Application Cycle Time (in units of T_{DP}) the application of the DP-master (Class 1) needs to process an application task completely. From Isochronous point of view there is exactly one user task to control the isochronous part of application.

Allowed values: 1 to 14

T_{BASE_IO}
This attribute indicates the time base (in units of 1/12 µs) for the times T_I and T_O of the DP-slave.

Allowed values: 375, 750, 1 500, 3 000, 6 000, 12 000 (the value 1 500 is mandatory for standard Isochronous Mode)

T_I
This attribute indicates the time T_I (in units of T_{BASE_IO}) for the DP-slave.

Allowed values: 0 (special case), 1 to 2^{16} - 1

T_O
This attribute indicates the time T_O (in units of T_{BASE_IO}) for the DP-slave.

Allowed values: 0 (special case), 1 to 2^{16} - 1

T_{DX}
This attribute indicates the time T_{DX} (in units of 1/12 μs) for the DP-slave.

Allowed values: 0 (special case), 1 to 2^{32} - 1

T_{PLL_W}
This attribute indicates the time T_{PLL_W} (in units of 1/12 μs) for the PLL of the DP-slave.

Allowed values: 1 to 2^{16} - 1 (default value 12)

T_{PLL_D}
This attribute indicates the time T_{PLL_D} (in units of 1/12 μs) for the PLL of the DP-slave.

Attribute Type: Unsigned16

Allowed values: 0 to 2^{16} - 1

6.2.5.2.5.3 Invocation of the IsoM parameter object

For the invocation of the IsoM Parameter object the following rules apply:

– The IsoM Parameter object shall exist only if the device attribute DPV1 of the correspondent DP-slave is set to TRUE.

– The values of the IsoM Parameter object may be provided with the Check User Prm service if the parameter Prm Structure provided with the Check User Prm service is set to TRUE.

– The values of the IsoM Parameter object may be provided with the Check Ext User Prm service.

Only one IsoM Parameter object shall be invoked in a DP-slave.

6.2.5.2.6 Time AR parameter class specification

6.2.5.2.6.1 Template

This class specifies the necessary parameter for a DP-slave to operate as a Time Receiver.

The Time AR Parameter object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**Time AR Parameter**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	Clock Sync Interval
3.	(o)	Attribute:	CS Delay Time
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.6.2 Attributes

Implicit
The attribute Implicit indicates that the Time AR Parameter object is implicitly addressed by the services.

Clock Sync Interval
This attribute contains the value for the attribute clock synchronization interval of the ARL of the DP-slave operating as time receiver.

Allowed values: 1 to 2^{16} - 1; Default value: 1 000

Time Base: 10

CS Delay Time
This attribute contains the value for the attribute CS Delay Time of the ARL of the DP-slave operating as time receiver.

Default value: 0

Type: Network Time Difference

6.2.5.2.6.3 Invocation of the Time AR parameter object

For the invocation of the Time AR Parameter object the following rules have to be obeyed:

– The Time AR Parameter object shall exist only if the device attribute DPV1 of the correspondent DP-slave is set to TRUE.

– The values of the Time AR Parameter object may be provided with the Check User Prm service if the parameter Prm Structure provided with the Check User Prm service is set to TRUE.

– The values of the Time AR Parameter object may be provided with the Check Ext User Prm service.

Only one Time AR Parameter object shall be invoked in a DP-slave.

6.2.5.2.7 MS0 configuration elements class specification

6.2.5.2.7.1 Template

The MS0 Configuration Elements object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**MS0 Configuration Elements**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of Cfg Data
2.1.	(o)	Attribute:	Input Data Reference
2.2.	(o)	Attribute:	Output Data Reference
2.3.	(o)	Attribute:	Manufacturer Specific Data
2.4.	(o)	Attribute:	Module Name
SERVICES:			
1.	(m)	OpsService:	Set Cfg
2.	(m)	OpsService:	Get Cfg
3.	(m)	OpsService:	Check Cfg
4.	(m)	OpsService:	Check Cfg Result

6.2.5.2.7.2 Attributes

Implicit
The attribute Implicit indicates that the MS0 Configuration Elements object is implicitly addressed by the services.

List of Cfg Data
This attribute is an aggregate of the configuration data of a DP-slave which is described by Cfg Identifier or Special Cfg Identifier. It shall contain at least one Cfg Data entry. A Cfg Data entry consists of the following attributes.

Input Data Reference
Reference to an Input Data Element of a Simple or an Extended Input Data object.

Output Data Reference
Reference to an Output Data Element of a Simple or an Extended Output Data object.

Manufacturer Specific Data
This attribute contains a manufacturer specific addition to the Input and/or Output Data Reference attributes. The contents are device specific or module specific.

Attribute Type: Octet string

Module Name
This attribute defines a name of a list element.

Attribute Type: Visible String(32)

6.2.5.2.7.3 Invocation of the MS0 configuration elements object

For the invocation of the MS0 Configuration Elements object the following rules apply:

– Only one MS0 Configuration Elements object shall be invoked in a DP-slave.

– The description of the MS0 Configurations Elements object shall not exceed 244 octets. The attribute List of Cfg Data has to be set accordingly.

6.2.5.2.8 Remanent parameter class specification

6.2.5.2.8.1 Template

A Remanent Parameter object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**Remanent Parameter**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	No Add Change
3.	(m)	Attribute:	New Slave Add
4.	(o)	Attribute:	Rem Para
4.1	(m)	Attribute:	Data Description
SERVICES:			
1.	(m)	OpsService:	Set Slave Add

6.2.5.2.8.2 Attributes

Implicit
The attribute Implicit indicates that a Remanent Parameter object is implicitly addressed by the service.

No Add Change
This attribute specifies whether it is allowed to change the DP-slave address again. After the initial reset the DP-slave DL-address will be set to the default address (126) and the attribute No Add Change will be set to FALSE. To change this attribute from TRUE to FALSE an initial reset of the DP-slave is necessary.

Allowed values: FALSE, TRUE

New Slave Add
This attribute contains the address of the DP-slave which is to be set.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Rem Para
This attribute contains parameters to be stored permanently.

Data Description
This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

6.2.5.2.8.3 Invocation of Rem Para object

For the invocation of the Remanent Parameter object the following rules apply:

– Only one Remanent Parameter object shall be invoked in a DP-slave.
– The Remanent Parameter object shall not exceed 242 octets. The attribute Data Description has to be set accordingly.

6.2.5.2.9 MS2 user parameter class specification

6.2.5.2.9.1 Template

The MS2 User Parameter object is described by the following template:

DP ASE:			**Context ASE**
CLASS:			**MS2 User Parameter**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	Features Supported
2.1.	(m)	Attribute:	Supported Feature
3.	(m)	Attribute:	Profile Features Supported
3.1.	(m)	Attribute:	Supported Profile Feature
4.	(m)	Attribute:	Profile Ident Number
SERVICES:			
1.	(m)	OpsService:	Initiate
2.	(m)	OpsService:	Abort

6.2.5.2.9.2 Attributes

Implicit
The attribute Implicit indicates that the MS2 User Parameter object is implicitly addressed by the services.

Features Supported
This attribute is composed of the following list elements:

Supported Feature
This attribute reflects the actual features on the MS2 AR of a DP-slave. The DP-master (Class 2) and the DP-slave inform each other about the supported service functionality. The DP-slave and the DP-master (Class 2) have the possibility to adjust their functionality according to the requirements. The User application of the DP-master has to decide if the reported functionality is sufficient or not.

The allowed values are shown in Table 80.

Table 80 – Supported feature

Value	Meaning
1	Read and Write service supported
2 to 16	reserved for further features

Profile Features Supported
This attribute is composed of the following list elements:

Supported Profile Feature
This attribute reflects the actual profile features on the MS2 AR of a DP-slave. The DP-master (Class 2) and the DP-slave inform each other about the supported profile features. Regarding the used profile definition, the profile is identified by the Profile Ident Number. The User application of the DP-master has to decide if the reported functionality is sufficient or not.

The allowed values are shown in Table 81.

Table 81 – Supported profile feature

Value	Meaning
1 to 16	feature to be defined by profiles

Profile Ident Number
The attribute Profile Ident Number contains the unique identifier of a profile definition. All devices using the same profile definition have to use the same Profile Ident Number. The Profile Ident Number will be taken from the pool of Ident Numbers for vendor specific or authorized profiles. The value 0 indicates that no profile is supported.

Attribute type: Unsigned16

6.2.5.2.9.3 Invocation of the MS2 user parameter object

For the invocation of the MS2 user parameter object the following rules have to be obeyed:

Only one MS2 user parameter object shall be invoked in a DP-slave.

6.2.5.2.10 Red Context class specification

6.2.5.2.10.1 Template

This class specifies the status of the redundant communication interfaces of a DP-slave and the state of a previous provided switchover command.

The following template describes the Red Context object:

DP ASE: **Context ASE**
CLASS: **Red Context**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Key Attribute: Communication Interface Identifier
2. (m) Attribute: Role
SERVICES:
1. (m) OpsService: Check User Prm
2. (m) OpsService: Check User Prm Result

6.2.5.2.10.2 Attributes

Communication Interface Identifier

This attribute identifies uniquely the communication interface.

Role

This attribute defines the actual role of the communication interface with the values according to Table 82.

Allowed values: Primary, Backup, Power On

Table 82 – Role

Value	Meaning
Primary	At the primary interface there exists no constraints between the communication interface and the ASE's.
Backup	In case of Backup the following items between the communication interface and the ASE's are active: Diagnosis ASE (all services) Context ASE (Check User Prm, Check Ext User Prm, Check Cfg) Process Data ASE (all services, only via MS2 AR) Load Region ASE (all services, only via MS2 AR) Function Invocation ASE (all services, only via MS2 AR)
Power On	In case of Power On the following items between the communication interface and the ASE's are active: Diagnosis ASE (all services) Context ASE (Check User Prm, Check Ext User Prm)

6.2.5.2.10.3 Invocation of the Red Context object

For the invocation of the Red Context object the following rules have to be obeyed.

– The Red Context object shall be used only if the device attribute DPV1 is set to TRUE.

– For each communication interface only one Red Context object shall be invoked.

6.2.5.3 Context service specification

6.2.5.3.1 Check user Prm

The DP Application Layer of the DP-slave notifies the application when a valid parameterization is received from the DP-master because this service requests a validation of the User Prm Data or List of Structured Prm Data. A further Check User Prm indication is only issued after the Check User Prm service is completed with a Check User Prm Result service.

The Check User Prm service shall be acknowledged negatively (the parameter Prm OK of the service Check User Prm Result shall be set to FALSE) by the application of the DP-slave if the following conditions are met:

– the parameter Prm Structure is set to FALSE and the parameterization data (User Prm Data) are not allowed or not acceptable by the application of the DP-slave,

– the parameter Prm Structure is set to TRUE and at least one of the MS0 Structured User Parameter object, the DXB-Linktable object, the DXB-Subscribertable object, or the IsoM Parameter object does not exist,

– the parameter Prm Structure is set to TRUE and at least one value for a MS0 Structured User Parameter object, or for the DXB-Linktable object, or for the DXB-Subscribertable object, or for the IsoM Parameter object is not allowed or not acceptable by the application of the DP-slave.

Table 83 shows the parameters of the service.

Table 83 – Check user Prm

Parameter name	Ind
Argument	M
AREP	M
Prm Structure	M
User Prm Data	C
Device Prm Data	U
Module Prm Data	U
List of Structured Prm Data	C
Structured User Prm Data	U
Device Related User Parameter	U
Device Prm Data	M
Module Related User Parameter	U
Slot Number	M
Module Prm Data	M
DXB-Linktable	U
List of DXB Link Entries	M
Publisher Address	M
Input Data Length Publisher	M
Source Offset Publisher	M
Length	M
DXB-Subscribertable	U
List of DXB Subscriber Entries	M
Publisher Address	C
Input Data Length Publisher	M
Source Offset Publisher	M
Master Data	C
Source Offset Master	M
Dest Slot Number	M
Offset Data Area	M
Length Data Area	M
IsoM Parameter	U
T_{BASE_DP}	M
T_{DP}	M
T_{MAPC}	M
T_{BASE_IO}	M
T_{I}	M
T_{O}	M
T_{DX}	M
T_{PLL_W}	M
T_{PLL_D}	M
Prm Command	U
Seq Number	M
Primary Request	M
MS1 Command	M
Master State Clear	M
Check Properties	M
Primary Request used	M
MS1 Command used	M
Address Change	M
Address Offset	M
Output Hold Time	M
Time AR Parameter	U
Clock Sync Interval	M
CS Delay Time	U

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the AR.

Prm Structure
This parameter indicates which type of parameterization objects are provided by the service Check User Prm. This parameter shall only be evaluated if the device attribute DPV1 is set to TRUE.

The allowed values are shown in Table 84.

Table 84 – Prm structure

Value	Meaning
TRUE	The parameter which follow are related to the following objects: MS0 Structured User Parameter DXB-Linktable IsoM Parameter
FALSE	The parameters which follow are only related to the MS0 User Parameter object.

User Prm Data
This conditional parameter is only present if the parameter Prm Structure contains the value FALSE. It is composed of one or both of the following subparameters.

Device Prm Data
This parameter contains the value for the Device Related User Parameter of the MS0 User Parameter object.

Module Prm Data
This parameter contains the value for the Module Related User Parameter of the MS0 User Parameter object.

List of Structured Prm Data
This conditional parameter is only present if the parameter Prm Structure contains the value TRUE. This parameter is composed of one or more of the following list elements.

Structured User Prm Data
This parameter is composed of one or both of the following subparameters.

Device Related User Parameter
This parameter indicates the presence of Device Related User Parameter.

Device Prm Data
This parameter contains the value for the Device Related User Parameter of the MS0 Structured User Parameter object.

Module Related User Parameter
This parameter indicates the presence of Module Related User Parameter.

Slot Number
This parameter contains the value of the attribute Slot Number of the MS0 Structured User Parameter object.

Module Prm Data
This parameter contains the value for the Module Related User Parameter of the MS0 Structured User Parameter object.

DXB-Linktable
This parameter indicates the presence of the DXB-Linktable.

List of DXB Link Entries
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the attribute Publisher Address of one DXB-Linktable entry of the DXB-Linktable object.

Input Data Length Publisher
This parameter contains the value of the attribute Input Data Length Publisher of one DXB-Linktable entry of the DXB-Linktable object.

Source Offset Publisher
This parameter contains the value of the attribute Source Offset Publisher of one DXB-Linktable entry of the DXB-Linktable object.

Length
This parameter contains the value of the attribute Length of one DXB-Linktable entry of the DXB-Linktable object.

DXB-Subscribertable
This parameter indicates the presence of the DXB-Subscribertable.

List of DXB Subscriber Entries
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the attribute Publisher Address of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Input Data Length Publisher
This parameter contains the value of the attribute Input Data Length Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Source Offset Publisher
This parameter contains the value of the attribute Source Offset Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Master Data
This parameter contains the value of the attribute Master Data of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Source Offset Master
This parameter contains the value of the attribute Source Offset Master of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Dest Slot Number
This parameter contains the value of the attribute Dest Slot Number of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Offset Data Area
This parameter contains the value of the attribute Offset Data Area of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Length Data Area
This parameter contains the value of the attribute Length Data Area of one DXB-Subscribertable entry of the DXB-Subscribertable object.

IsoM Parameter
This parameter indicates the presence of the IsoM Parameter.

TBASE_DP
This parameter contains the value of the attribute T_{BASE_DP} of the IsoM Parameter object.

T_{DP}
This parameter contains the value of the attribute T_{DP} of the IsoM Parameter object.

TMAPC
This parameter contains the value of the attribute T_{MAPC} of the IsoM Parameter object.

TBASE_IO
This parameter contains the value of the attribute T_{BASE_IO} of the IsoM Parameter object.

T_I
This parameter contains the value of the attribute T_I of the IsoM Parameter object.

T_O
This parameter contains the value of the attribute T_O of the IsoM Parameter object.

T_{DX}
This parameter contains the value of the attribute T_{DX} of the IsoM Parameter object.

TPLL_W
This parameter contains the value of the attribute T_{PLL_W} of the IsoM Parameter object.

TPLL_D
This parameter contains the value of the attribute T_{PLL_D} of the IsoM Parameter object.

Prm Command
This parameter indicates the presence of the Prm Command.

Seq Number
This parameter contains the Sequence Number necessary for the unique identification of the received Prm Command.

Allowed values: 0 to 31

Primary Request
If this parameter is set to TRUE the selected communication interface shall operate as the primary one. All MS1 functions are active and will be processed as if redundancy is not activated. Also all diagnosis information including channel related, or other status messages, have to be present.

This parameter set to FALSE means that the role of the communication interface (primary, backup) shall not be changed.

Allowed values: TRUE, FALSE

MS1 Command
This parameter defines the desired command for the MS1 AR of the selected communication interface.

The allowed values are shown in Table 85.

Table 85 – MS1 Command

Value	Meaning
no Action	No Command for the MS1 AR is present.
Stop MS1	This parameter specifies when the MS1 AR of the DP-slave shall be stopped by the DP Application Layer. No MS1 AR services are possible any longer.
Start MS1	This parameter specifies when the MS1 AR of the DP-slave shall be (re)started by the DP Application Layer. After the (re)start all MS1 AR services are possible if the role of the communication interface is primary.
Reset MS1	Means that the MS1 AR shall be stopped and then be started afterwards.

Master State Clear
This parameter indicates the actual state of the MS0 AR of the DP-master (class 1). This parameter set to TRUE means that the MS0 AR of the DP-master (class 1) is in the operation mode CLEAR. This parameter set to FALSE means that the MS0 AR of the DP-master (class 1) is in the operation mode OPERATE.

Allowed values: TRUE, FALSE

Check Properties
This parameter indicates whether the DP-slave shall check the properties provided by the Check User Prm service. The parameter set to TRUE indicates that properties shall be checked (e. g. at the start-up of the DP-slave). This parameter set to FALSE indicates that the check of the properties shall be omitted (e. g. at the data exchange mode of the DP-slave).

Allowed values: TRUE, FALSE

Primary Request used
This parameter indicates whether the communication interface of a redundant DP-slave supports primary requests. The DP-slave shall respond negatively (with the parameter Prm OK of the Check User Prm service set to FALSE) if those requests are not supported. In the other case the Check Properties shall be accepted.

Allowed values: TRUE, FALSE

MS1 Command used
This parameter indicates whether the MS1 Commands Start MS1, Stop MS1 and Reset MS1 are supported by the communication interface of a redundant DP-slave. The DP-slave shall respond negatively (with the parameter Prm OK of the Check User Prm service set to FALSE) if those requests are not supported. In the other case the Check Properties shall be accepted.

Allowed values: TRUE, FALSE

Address Change
This parameter indicates whether address changes are requested with the primary request. TRUE indicates that address changes are requested and FALSE indicates that no address changes are requested. The DP-slave shall respond negatively (with the parameter Prm OK of the Check User Prm service set to FALSE) if those requests are not supported. In the other case the Check Properties shall be accepted.

Allowed values: TRUE, FALSE

Address Offset
This parameter indicates if the address offset of 64 will be used at the start-up of the DP-slave or not. The DP-slave shall respond negatively (with the parameter Prm OK of the Check User Prm service set to FALSE) if the address offset is not supported. In the other case the Check Properties shall be accepted.

Allowed values: TRUE, FALSE

Output Hold Time
This parameter indicates the time the redundant DP-slave shall not change its outputs when it performs a redundancy switchover. When the backup communication interface becomes the primary (and vice versa) the User may use the output values previously transmitted from the old primary communication interface or the User has to wait for the first data exchange after the switchover to write new values to its outputs. A possible switchover from backup to primary will always start the Output_Hold_Time.

Timebase: 10 ms.

Allowed values: 0 to 2^{16} - 1

Time AR Parameter
This parameter indicates the presence of the Time AR Parameter.

Clock Sync Interval
This parameter contains the value for the attribute Clock Sync Interval of the ARL DP-slave object.

CS Delay Time
This parameter contains the value for the attribute CS Delay Time of the ARL DP-slave object.

6.2.5.3.2 Check user Prm result

This service is used in a DP-slave as a reply to the previous issued Check User Prm service. The Default AP of a DP-slave uses this service to provide the result of the validation of the User Prm Data or of the List of Structured Prm Data. This service shall only be used in conjunction with the MS0 AR.

Table 86 shows the parameters of the service.

Table 86 – Check user Prm result

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Prm OK	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the desired AR.

Prm OK
The parameter Prm OK equal TRUE specifies that the User Prm Data or the List of Structured Prm Data conveyed by the previous Check User Prm service are acceptable by the application of the DP-slave. If they are not acceptable for the application of the DP-slave, this parameter shall have the value FALSE. In this case the attributes Prm Req and Prm Fault of the related MS0 ARL Entry will be set to TRUE by the DP Application Layer.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed. The DP-slave shall ignore the DP-masters Parameterization.

Status
This parameter indicates the cause of the failure.

The allowed values are shown in Table 87.

Table 87 – Status values

Value	Meaning
Reset	Reset still in progress
Not Pending	No Validation pending
New Prm	New User Prm Data or List of Structured Prm Data received during validation
Inv Master Add	A different DP-master has parameterized the DP-slave or the DP-master has released the DP-slave (the validation of the previous User Prm Data or List of Structured Prm Data was useless).

6.2.5.3.3 Check Ext user Prm

This service shall only be used if the MS0 AR offers an additional CR for the conveyance of the parameterization data and the device attribute DPV 1 is set to TRUE.

The DP Application Layer of the DP-slave notifies the application if additional parameterization data are received from the DP-master (Class 1). This service requests a validation of the List of Structured Prm Data. A further Check Ext User Prm indication is only issued after the Check Ext User Prm service is completed with a Check Ext User Prm Result service.

The Check Ext User Prm service shall be acknowledged negatively (the parameter Ext Prm OK of the service Check Ext User Prm Result shall be set to FALSE) if at least one parameterization object is unknown or not allowed or acceptable by the application of the DP-slave.

Table 88 shows the parameters of the service.

Table 88 – Check Ext user Prm

Parameter name	Ind
Argument	M
AREP	M
List of Structured Prm Data	M
Structured User Prm Data	U
Device Related User Parameter	U
Device Prm Data	M
Module Related User Parameter	U
Slot Number	M
Module Prm Data	M
DXB-Linktable	U
List of DXB Link Entries	M
Publisher Address	M
Input Data Length Publisher	M
Source Offset Publisher	M
Length	M
DXB-Subscribertable	U
List of DXB Subscriber Entries	M
Publisher Address	C
Input Data Length Publisher	M
Source Offset Publisher	M
Master Data	C
Source Offset Master	M
Dest Slot Number	M
Offset Data Area	M
Length Data Area	M
IsoM Parameter	U
T_{BASE_DP}	M
T_{DP}	M
T_{MAPC}	M
T_{BASE_IO}	M
T_{I}	M
T_{O}	M
T_{DX}	M
T_{PLL_W}	M
T_{PLL_D}	M
Time AR Parameter	U
Clock Sync Interval	M
CS Delay Time	U

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the AR.

List of Structured Prm Data
This parameter is composed of one or more of the following list elements.

Structured User Prm Data
This parameter is composed of one or both of the following subparameters.

Device Related User Parameter
This parameter indicates the presence of Device Related User Parameter.

Device Prm Data
This parameter contains the value for the Device Related User Parameter of the MS0 Structured User Parameter object.

Module Related User Parameter
This parameter indicates the presence of Module Related User Parameter.

Slot Number
This parameter contains the value of the attribute Slot Number of the MS0 Structured User Parameter object.

Module Prm Data
This parameter contains the value for the Module Related User Parameter of the MS0 Structured User Parameter object.

DXB-Linktable
This parameter indicates the presence of the DXB-Linktable.

List of DXB Link Entries
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the attribute Publisher Address of one DXB-Linktable entry of the DXB-Linktable object.

Input Data Length Publisher
This parameter contains the value of the attribute Input Data Length Publisher of one DXB-Linktable entry of the DXB-Linktable object.

Source Offset Publisher
This parameter contains the value of the attribute Source Offset Publisher of one DXB-Linktable entry of the DXB-Linktable object.

Length
This parameter contains the value of the attribute Length of one DXB-Linktable entry of the DXB-Linktable object.

DXB-Subscribertable
This parameter indicates the presence of the DXB-Subscribertable.

List of DXB Subscriber Entries
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the attribute Publisher Address of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Input Data Length Publisher
This parameter contains the value of the attribute Input Data Length Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Source Offset Publisher
This parameter contains the value of the attribute Source Offset Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Master Data
This parameter contains the value of the attribute Master Data of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Source Offset Master
This parameter contains the value of the attribute Source Offset Master of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Dest Slot Number
This parameter contains the value of the attribute Dest Slot Number of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Offset Data Area
This parameter contains the value of the attribute Offset Data Area of one DXB-Subscribertable entry of the DXB-Subscribertable object.

Length Data Area
This parameter contains the value of the attribute Length Data Area of one DXB-Subscribertable entry of the DXB-Subscribertable object.

IsoM Parameter
This parameter indicates the presence of the IsoM Parameter.

TBASE_DP
This parameter contains the value of the attribute T_{BASE_DP} of the IsoM Parameter object.

T_{DP}
This parameter contains the value of the attribute T_{DP} of the IsoM Parameter object.

TMAPC
This parameter contains the value of the attribute T_{MAPC} of the IsoM Parameter object.

TBASE_IO
This parameter contains the value of the attribute T_{BASE_IO} of the IsoM Parameter object.

T_I
This parameter contains the value of the attribute T_I of the IsoM Parameter object.

T_O
This parameter contains the value of the attribute T_O of the IsoM Parameter object.

T_{DX}
This parameter contains the value of the attribute T_{DX} of the IsoM Parameter object.

TPLL_W
This parameter contains the value of the attribute T_{PLL_W} of the IsoM Parameter object.

TPLL_D
This parameter contains the value of the attribute T_{PLL_D} of the IsoM Parameter object.

Time AR Parameter
This parameter indicates the presence of the Time AR Parameter.

Clock Sync Interval
This parameter contains the value for the attribute Clock Sync Interval of the ARL DP-slave object.

CS Delay Time
This parameter contains the value for the attribute CS Delay Time of the ARL DP-slave object.

6.2.5.3.4 Check Ext user Prm result

This service is used in a DP-slave as a reply to the previous issued Check Ext User Prm service. The Default AP of a DP-slave uses this service to provide the result of the validation of the List of Structured Prm Data. This service shall only be used in conjunction with the MS0 AR.

Table 89 shows the parameters of the service.

Table 89 – Check Ext user Prm result

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Ext Prm OK	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the desired AR.

Ext Prm OK
The parameter Ext Prm OK equal TRUE specifies that the List of Structured Prm Data conveyed by the previous Check Ext User Prm service are acceptable by the application of the DP-slave. If the List of Structured Prm Data are not acceptable for the application of the DP-slave, this parameter shall have the value FALSE. In this case the attributes Prm Req and Cfg Fault of the related MS0 ARL Entry will be set to TRUE by the DP Application Layer.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed. The DP-slave shall ignore the DP-masters Parameterization.

Status
This parameter indicates the cause of the failure.

The allowed values are shown in Table 90.

Table 90 – Status values

Value	Meaning
Reset	Reset still in progress
Not Pending	No Validation pending
New Prm	New List of Structured Prm Data received during validation
Inv Master Add	A different DP-master has parameterized the DP-slave or the DP-master has released the DP-slave (the validation of the previous List of Structured Prm Data was useless).

6.2.5.3.5 Check Cfg

The DP Application Layer of the DP-slave notifies the application when Cfg Data (Configuration Data) are received from the DP-master because this service requests a validation of the Cfg Data by the application. A further Check Cfg indication is only issued after the Check Cfg service is completed with a Check Cfg Result service. The DP-slave compares its own configuration data with the Cfg Data received from the DP-master.

The check shall cover the following details:

– module structure,
– range of Input and/or Output Data Element,
– format,
– consistency.

The consistency also affects the DP-slave and the DP-master. The verification of the consistency in the Cfg Data list element shall cause only in the following cases a configuration fault:

– The DP-slave needs consistency for an Input and/or Output Data Element and the DP-master indicates no consistency.
– The DP-slave cannot provide consistency for an Input and/or Output Data Element and the DP-master requires consistency for this Input and/or Output Data Element.

DP-slaves may accept different sets of Cfg Data. In this case the Check Cfg shall be successful if one set of configuration data matches. If the service parameter Check Cfg Mode is set to TRUE the check of the Cfg Data may be done in a different User specific way. For example a temporary not available module may be accepted even if the Cfg Data contains the respective Configuration Identifier.

Table 91 shows the parameters of the service.

Table 91 – Check Cfg

Parameter name	Ind
Argument	—
AREP	M
Check Cfg Mode	C
Cfg Data	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the AR.

Check Cfg Mode
This parameter indicates how the DP-slave application has to check the Cfg Data. This parameter shall only be evaluated if the device attribute DPV1 is set TRUE.

Cfg Data
This parameter contains the Cfg Data to be verified by the application of the DP-slave.

6.2.5.3.6 Check Cfg result

This service is used in a DP-slave as a reply to the previous issued Check Cfg service. The Default AP of the DP-slave uses this service to provide the result of the validation of the Cfg Data. This service shall only be used in conjunction with the MS0 AR.

Table 92 shows the parameters of the service.

Table 92 – Check Cfg result

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Cfg OK	M	
Input Data Len	M	
Output Data Len	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the desired AR.

Cfg OK
The parameter Cfg OK equal TRUE specifies that the Cfg Data conveyed by the previous Check Cfg service are acceptable by the application of the DP-slave. The application of the DP-slave may update the Real Cfg Data of the MS0 AR with the Set Cfg service. If the Cfg Data are not acceptable for the application of the DP-slave, this parameter shall have the value FALSE. In this case the attributes Prm Req and Cfg fault will be set to TRUE by the DP Application Layer.

Input Data Len
This parameter contains the actual length of the Input Data.

Output Data Len
This parameter contains the actual length of the Output Data.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the cause of the failure.

The allowed values are shown in Table 93.

Table 93 – Status values

Value	Meaning
Reset	Reset still in progress
Not Pending	No Validation of Cfg Data pending
New Cfg	New Cfg Data received during validation
Inv Master Add	A different DP-master has parameterized the DP-slave or the DP-master has released the DP-slave (the validation of the previous Cfg Data was useless).

6.2.5.3.7 Set Cfg

This service is used by the application of the DP-slave to pass configuration data to the DP-Application Layer. This service shall only be used in conjunction with the MS0 AR.

Table 94 shows the parameters of the service.

Table 94 – Set Cfg

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Cfg Data	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the desired AR.

Cfg Data
This parameter contains the Cfg Data.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

6.2.5.3.8 Get Cfg

This service allows the DP-master (Class 2) to read the configuration data of the DP-slave via the MS0 AR.

Table 95 shows the parameters of the service.

Table 95 – Get Cfg

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Cfg Data		M
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the desired AR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Cfg Data
This parameter contains the Cfg Data.

Result(-)
This selection type parameter indicates that the service request failed.

Status
The parameter Status indicates the type of failure of the service.

Allowed values: DS, NA, RS, UE, NR, RE

6.2.5.3.9 Set slave add

This service permits a DP-master (Class 2) to change the address of a DP-slave and to pass remanent Slave Data. The Set Slave Add service is only successful if the Ident Number transmitted corresponds with the Ident Number of the DP-slave and the attribute No Add Change (see 6.2.10.3.1) of the DP-slave has the value FALSE. This service shall only be used in conjunction with the MS0 AR.

Table 96 shows the parameters of the service.

Table 96 – Set Slave Add

Parameter name	Req	Ind	Cnf
Argument	M	M(=)	
AREP	M	M(=)	
New Slave Add	M	M(=)	
Ident Number	M	M(=)	
No Add Chg	M	M(=)	
Rem Slave Data	U	C(=)	
Result(+)			S
AREP			M
Result(-)			S
AREP			M
Status			M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the desired AR.

New Slave Add
This parameter contains the value of the attribute New Slave Add of the Remanent Parameter object of the DP-slave which is to be set.

Ident Number
This parameter contains the Ident Number to be checked with the Ident Number of the DP-slave.

No Add Chg
This parameter contains the value of the attribute No Add Change of the Remanent Parameter object. The attribute No Add Change will assume the value of this service parameter in case of success, otherwise No Add Change remains unchanged.

Rem Slave Data
This parameter contains the value of the attribute Rem Para of the Remanent Parameter object.

Result(+)
This selection type parameter indicates that the service request succeeded. This confirmation does not reflect whether the service was successful, i.e. the new values were accepted by the DP-slave. Therefore the DP-master shall check the correct execution of the service by using the Read Slave Diag service with the new DP-slave address.

Result(-)
This selection type parameter indicates that the service request failed.

Status
The parameter Status indicates the type of failure of the service.

Allowed values: DS, NA, RS, RR, UE, RE

6.2.5.3.10 Initiate

This service shall be used by a DP-master (Class 2) to establish a connection to a desired DP-slave. This service shall only be used in conjunction with a MS2 AR.

Table 97 shows the parameters of the service.

Table 97 – Initiate

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Send Timeout	M			
Features Supported	M	M(=)		
Profile Features Supported	M	M(=)		
Profile Ident Number	M	M(=)		
Add Addr Param	M	M(=)		
S-Type	M	M(=)		
S-Len	M	M(=)		
D-Type	M	M(=)		
D-Len	M	M(=)		
S-Addr	M	M(=)		
S-API	M	M(=)		
S-SCL	M	M(=)		
S-Network Address	C	C(=)		
S-MAC Address	C	C(=)		
D-Addr	M	M(=)		
D-API	M	M(=)		
D-SCL	M	M(=)		
D-Network Address	C	C(=)		
D-MAC Address	C	C(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Max Len Data Unit			M	M(=)
Features Supported			M	M(=)
Profile Features Supported			M	M(=)
Profile Ident Number			M	M(=)
Add Addr Param			M	M(=)
S-Type			M	M(=)
S-Len			M	M(=)
D-Type			M	M(=)
D-Len			M	M(=)
S-Addr			M	M(=)
S-API			M	M(=)
S-SCL			M	M(=)
S-Network Address			C	C(=)
S-MAC Address			C	C(=)

Parameter name	Req	Ind	Rsp	Cnf
D-Addr			M	M(=)
D-API			M	M(=)
D-SCL			M	M(=)
D-Network Address			C	C(=)
D-MAC Address			C	C(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Max Len Data Unit
This parameter contains the value of the attribute Max Len Data Unit of the ARL DP-slave object of a DP-slave.

Send Timeout
This parameter defines the control time for monitoring the MS2 AR requested by the DP-master (Class 2). DP-master and DP-slave inform each other about the supported Send Timeout. The DP-slave adjusts its Send Timeout. The DP-master (Class 2) decides whether the reported Send Timeout is sufficient or not.

Features Supported
DP-master and DP-slave inform each other about the supported service functionality. The DP-slave has the possibility to adjust its functionality to the Master's requirements. The User application of the DP-master has to decide if the reported functionality is sufficient or not.

Profile Features Supported
DP-master and DP-slave inform each other about the supported service functionality regarding the used profile definition. The profile is identified by the Profile Ident Number. The meaning of the defined bits is profile or vendor specific.

Profile Ident Number
This parameter identifies a profile definition uniquely. All devices using the same profile definition have to use the same Profile Ident Number. The Profile Ident Number will be taken from the pool of Ident Numbers for vendor specific or authorized profiles. The value 0 indicates that no profile is supported. If the requested profile is supported by the DP-slave, the Profile Ident Number is mirrored in the response. If the requested profile is not supported by the DP-slave, the DP-slave responds negatively or with a Profile Ident Number supported by the DP-slave.

Add Addr Param
Add Addr Param consists of two parts, additional address information of the source and additional address information of the destination. The additional address information contains the parameter API plus an optional Network/MAC address.

D-Addr and S-Addr (API, SCL, Network Address, MAC Address) are related to the direction of the telegrams, which means the source becomes destination and vice versa in the response.

Add Addr Param contains the following subparameters:

S-Type
This subparameter indicates the presence (S-Type = 1) of the optional Network/MAC address in the Add Addr Param of the source.

S-Len
This subparameter indicates the length of the S-Addr subparameter.

D-Type
This subparameter indicates the presence (D-Type = 1) of the optional Network/MAC address in the Add Addr Param of the destination.

D-Len
This subparameter indicates the length of the D-Addr subparameter.

S-Addr
This subparameter contains the additional address information of the source:

S-API
This subparameter identifies the AP of the source.

S-SCL
This subparameter identifies the access level of the source.

S-Network Address
In case of S-Type = 1 this subparameter identifies the network address of the source according to ISO/OSI-Network addresses.

S-MAC Address
In case of S-Type = 1 this subparameter identifies the MAC Address of the source.

D-Addr
This subparameter contains the additional address information of the destination:

D-API
This subparameter identifies the AP of the destination.

D-SCL
This subparameter identifies the access level of the destination.

D-Network Address
In case of D-Type = 1 this subparameter identifies the network address of the destination according to ISO/OSI-Network addresses.

D-MAC Address
In case of D-Type = 1 this subparameter identifies the MAC Address of the destination.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Decode
This parameter selects one of the following Error schemes: DPV1, FMS, Profile_Specific.

Error Code 1
In case of Error Decode equal DPV1 the Error Code 1 assumes one of the following values:
version conflict, feature not supported, user specific, state conflict, invalid range, invalid parameter, invalid type, resource busy, resource unavailable.

Error Code 2
The parameter Error Code 2 is user specific.

6.2.5.3.11 Abort

The DP-master and DP-slave use this service to abort an AR. With the Abort.req the User resets the FSPM. In faulty conditions the FSPM informs the User with an Abort.ind and resets itself. This service shall only be used in conjunction with the MS2 AR.

Table 98 shows the parameters of the service.

Table 98 – Abort

Parameter name	Req	Ind
Argument	M	M
AREP	M	M(=)
Locally Generated		C
Subnet	C	C(=)
Instance	C	C(=)
Reason Code	C	C(=)
Additional Detail		C

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is used as local identifier of the desired AR.

Locally Generated
This parameter only exists in the indication in case of an MS2 AR. It indicates that the Abort has been produced by the remote partner (Locally Generated = FALSE) or in the local station (Locally Generated = TRUE).

Subnet
This parameter is used in case of an MS2 AR to encode information about the location of the source of the Abort.

Instance
This parameter is used in case of an MS2 AR to encode information about the protocol instance that detected a problem causing the Abort.

The allowed values are shown in Table 99.

Table 99 – Instance

Value	Meaning
DL	Data-link layer
MS2	MS2 AR
User	User

Reason Code
The Reason Code indicates the reason for the Abort in case of an MS2 AR.

Allowed Values: UE, RR, RS, NR, DH, LR, RDL, RDH, DS, NA, ABT_SE, ABT_FE, ABT_TO, ABT_RE, ABT_IV, ABT_STO, ABT_IA, ABT_OC.

Additional Detail
This parameter contains detail information in case of an abort which is caused by an inappropriate Initiate (e.g. Send Timeout to small) in case of an MS2 AR. This parameter is only present if the parameter Reason Code has the value ABT_STO and the parameter Instance equals MS2. In this case the parameter additional detail contains the value of the received Send Timeout of the DP-slave.

6.2.5.3.12 MS0 init DP-slave

This service is used to initialize the MS0 AR of a DP-slave.

Table 100 shows the parameters of the service.

Table 100 – MS0 init DP-slave

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result		M
AREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result
This parameter indicates that the service request succeeded.

6.2.5.3.13 MS1 init DP-slave

This service is used to initialize the MS1 AR of a DP-slave.

Table 101 shows the parameters of the service.

Table 101 – MS1 init DP-slave

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result		M
AREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result
This parameter indicates that the service request succeeded.

6.2.5.3.14 MS2 init DP-slave

This service is used to initialize all MS2 AR of a DP-slave.

Table 102 shows the parameters of the service.

Table 102 – MS2 init DP-slave

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result		M
AREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result
This parameter indicates that the service request succeeded.

6.2.5.3.15 DP-slave started

With this service the supported alarm features on this MS1 AR are indicated. The indication occurs after the DP-slave has entered the data exchange mode and the transfer of alarm and status messages is possible from now on.

Table 103 shows the parameters of the service.

Table 103 – DP-slave started

Parameter name	Ind
Argument	M
AREP	M
Actual Enabled Alarms	M
Alarm Sequence	M
Alarm Limit	C

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Actual Enabled Alarms
This parameter indicates which alarm types are supported by the DP-master (Class 1).

Allowed values: Update Alarm enabled / disabled, Manufacturer Specific Alarm enabled / disabled, Diagnosis Alarm enabled / disabled, Process Alarm enabled / disabled, Pull Plug Alarm enabled / disabled.

Alarm Sequence
This Boolean parameter indicates whether only one alarm of a specific Alarm Type can be active at one time (Alarm Sequence=FALSE) or several alarms (2 to 32) of the same or different type can be active at one time (Alarm Sequence=TRUE).

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Alarm Limit
This parameter indicates the maximum number of parallel alarms.

The allowed values are shown in Table 104.

Table 104 – Alarm limit

Value	Meaning
0	1 alarm of each type
1	2 alarms in total
2	4 alarms in total
3	8 alarms in total
4	12 alarms in total
5	16 alarms in total
6	24 alarms in total
7	32 alarms in total

6.2.5.3.16 DP-slave stopped

In case of an MS0 AR, this Application Layer service indicates to the User that the DP-slave has left the data exchange mode and that the MS1 AR is closed. As a consequence the transfer of alarms as well as the processing of Write and Read services is no longer possible.

In case of an MS1 AR, this service indicates to the User that the acyclic communication relationship to the DP-master (Class 1) has been stopped. This means that there is no longer cyclic data exchange, the MS0 AR and MS1 AR of a DP-slave will be set to the initial state. Table 105 shows the parameters of the service.

Table 105 – DP-slave stopped

Parameter name	Ind
Argument	M
AREP	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

6.2.5.3.17 Reset DP-slave

By this service the User of a DP-slave resets the Application Layer.

Table 106 shows the parameters of the service.

Table 106 – Reset DP-slave

Parameter name	Req	Cnf
Argument		

Argument
The argument shall convey the service specific parameters of the service request. Nevertheless, in this service there are no specific parameters.

6.2.5.3.18 DP-slave fault

By this service the Application Layer indicates to the User of a DP-slave that due to fatal errors all ARs have been reset.

Table 107 shows the parameters of the service.

Table 107 – DP-slave fault

Parameter name	Ind
Argument	

Argument
The argument shall convey the service specific parameters of the service request. Nevertheless, in this service there are no specific parameters.

6.2.5.3.19 Application ready DP-slave

By this service the User of a DP-slave indicates to the Application Layer that the application is ready. This service shall only be used in conjunction with the MS0 AR.

Table 108 shows the parameters of the service.

Table 108 – Application ready DP-slave

Parameter name	Req
Argument	M
AREP	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

6.2.5.3.20 Start subscriber

This service is used by the application of the DP-slave to start the subscription of a certain DXB-Link.

Table 109 shows the parameters of the service.

Table 109 – Start subscriber

Parameter name	Req	Cnf
Argument	M	
AREP	M	
CREP	M	
Result(+)		S
AREP		M
CREP		M
Result(-)		S
AREP		M
CREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

CREP
This parameter is the local identifier for the desired CR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

6.2.5.3.21 Stop subscriber

This service is used by the application of the DP-slave to stop the subscription of a certain DXB-Link.

Table 110 shows the parameters of the service.

Table 110 – Stop subscriber

Parameter name	Req	Cnf
Argument	M	
AREP	M	
CREP	M	
Result(+)		S
AREP		M
CREP		M
Result(-)		S
AREP		M
CREP		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

CREP
This parameter is the local identifier for the desired CR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

6.2.5.3.22 Publisher active

This service indicates to the application of the Subscriber (DP-slave) the state of the Publisher of a certain DXB-Link. This service will be issued by the DP Application Layer only if the state of the Publisher has been changed since the last watchdog period.

Table 111 shows the parameters of the service.

Table 111 – Publisher active

Parameter name	Ind
Argument	M
AREP	M
CREP	M
Status	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

CREP
This parameter is the local identifier for the desired CR (Publisher).

Status
This parameter indicates the state of the related Publisher.

The allowed values are shown in Table 112.

Table 112 – Status

Value	Meaning
TRUE	the related Publisher is active: during the last watchdog period the subscribed Input Data object of the related Publisher has been received at least one time
FALSE	the related Publisher is not active; during the last watchdog the subscribed Input Data object of the related Publisher has not been received due to the following problems: the value of the attribute Input Data Length Publisher of the related Publisher of the related CRL entry was not in alignment with the actual length of the Input Data object published by the related Publisher the related Publisher was not in the data exchange mode breakdown of the related Publisher

6.2.5.3.23 Init DP-master CI1

By this service the User initializes the Application Layer of a DP-master (Class 1). The parameters are described in this specification, part 3 and 4.

Table 113 shows the parameters of the service.

Table 113 – Init DP-master CI1

Parameter name	Req	Cnf
Argument	M	
Bus Para	M	
TS	M	
Data rate	M	
TSL	M	
$minT_{SDR}$	M	
$maxT_{SDR}$	M	
TQUI	M	
TSET	M	
TTR	M	
G	M	
HSA	M	
max retry limit	M	
Result		M

Argument
The argument shall convey the service specific parameters of the service request.

Bus Para
This parameter is composed of the following subparameters:

TS
This parameter contains the DL-address of this DP-master (Class 1).

Data rate
This parameter contains the data rate of this DP-master (Class 1).

TSL
This parameter contains the slot time for this DP-master (Class 1).

minT_{SDR}
This parameter contains the smallest station delay time for this DP-master (Class 1).

maxT_{SDR}
This parameter contains the largest station delay time for this DP-master (Class 1).

TQUI
This parameter contains the transmitter fall time for this DP-master (Class 1).

TSET
This parameter contains the setup time for this DP-master (Class 1).

TTR
This parameter contains the target rotation time for this DP-master (Class 1).

G
This parameter contains the GAP update factor for this DP-master (Class 1).

HSA
This parameter contains the highest station (DL-) address for this DP-master (Class 1).

max retry limit
This parameter contains the maximum number of retries for this DP-master (Class 1).

Result
This parameter indicates that the service request succeeded.

6.2.5.3.24 DP-master CI1 started

In case of a MS0 AR, this Application Layer service indicates to the User of a DP-master (Class 1) that the assigned DP-slave is in the data exchange mode.

In case of a MS1 AR, this Application Layer service indicates that the handling of alarm requests for this DP-slave is possible.

Table 114 shows the parameters of the service.

Table 114 – DP-master CI1 started

Parameter name	Ind
Argument	M
AREP	M
Alarm Limit	C

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR (Address of DP-slave).

Alarm Limit
In case of an MS1 AR, this parameter indicates the maximum number of parallel alarms.

The allowed values are shown in Table 115.

Table 115 – Alarm limit

Value	Meaning
0	1 alarm of each type
1	2 alarms in total
2	4 alarms in total
3	8 alarms in total
4	12 alarms in total
5	16 alarms in total
6	24 alarms in total
7	32 alarms in total

6.2.5.3.25 DP-master Cl1 stopped

This Application Layer service indicates to the User of a DP-master (Class 1) that the related AR has been stopped. In case of a MS0 AR, the MS1 AR has been stopped, too. A new start is necessary to restart MS0 AR and MS1 AR.

Table 116 shows the parameters of the service.

Table 116 – DP-master Cl1 stopped

Parameter name	Ind
Argument	M
AREP	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

6.2.5.3.26 Reset DP-master Cl1

By this service the User of a DP-master (Class 1) resets the Application Layer.

Table 117 shows the parameters of the service.

Table 117 – Reset DP-master Cl1

Parameter name	Req	Cnf
Argument	M	
Result		M

Argument
The argument shall convey the service specific parameters of the service request. Nevertheless, in this service there are no specific parameters.

Result
This parameter indicates that the service request succeeded.

6.2.5.3.27 DP-master CI1 fault

By this service the Application Layer indicates to the User of a DP-master (Class 1) that a fatal error has occurred and all ARs have been stopped. Further communication is only possible after a Reset.req of the User.

Table 118 shows the parameters of the service.

Table 118 – DP-master CI1 fault

Parameter name	Ind
Argument	—

Argument
The argument shall convey the service specific parameters of the service request. Nevertheless, in this service there are no specific parameters.

6.2.5.3.28 DP-master CI1 reject

By this service the Application Layer rejects a previous Read/Write.req from the User of a DP-master (Class 1) to the MS1 AR, if the MS1 AR was forced to stop.

Table 119 shows the parameters of the service.

Table 119 – DP-master CI1 reject

Parameter name	Ind
Argument	M
AREP	M
Reason	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR (Address of DP-slave).

Reason
The parameter Status indicates success or failure of the service.

Allowed values: REJ ABORT,REJ LE,REJ PS,REJ SE,REJ IV

6.2.5.3.29 Set mode DP-master CI1

By this service the User of a DP-master (Class 1) can set the operation mode of the MS0 AR. Additionally the User gets information about the actual state of the bus system in case of errors.

Table 120 shows the parameters of the service.

Table 120 – Set mode DP-master CI1

Parameter name	Req	Cnf
Argument	M	
AREP	M	
USIF State	M	
Result(+)		M
AREP		M
Bus Accessible		M
Result(-)		
AREP		M
Bus Accessible		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

USIF State
The Application Layer provides four operation modes for the User:

OFFLINE
Communication to all DP participants (DP-master and DP-slave) is stopped. MAC of local DL is in offline state. The Application Layer waits for a signal to start.

STOP
Changing the operation mode from OFFLINE to STOP the bus parameter set is loaded in the DL by the Application Layer. The DL was activated. The responder services for MM1 AR and MM2 AR can be processed. MS0 and MS1 AR will not be activated (No interactions to DP-slaves).

CLEAR
In this operation mode the DP-master (Class 1) attempts to configure the DP-slaves that are assigned to it, and starts the data exchange mode. The input data of the DP-slaves are read and passed to the User. The User's output data are ignored. The assigned DP-slaves receive a value of 0 or no data for their Output Data objects. (depending on the attribute Fail Safe).

OPERATE
The DP-master (Class 1) is in the data exchange mode with the assigned DP-slaves. The inputs from the DP-slaves are transferred to the User and the outputs from the User are passed to the DP-slaves. If the Application Layer leaves the mode OPERATE, all DP-slaves' outputs change in the safe state by means of the service Global Control (Control Command = Clear Data, Group Select = 0).

Result(+)
This selection type parameter indicates that the service request succeeded.

Bus Accessible
This Boolean parameter indicates whether the DP-master is able to access the bus (TRUE) or not (FALSE).

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Result(-)
This selection type parameter indicates that the service request failed.

6.2.5.3.30 DP-master CI1 mode changed

By this service the Application Layer indicates to the user of a DP-master (Class 1):

– the change of its operation mode from OPERATE to CLEAR if an error occurred and the Error Action Flag is set,

– the change of its operation mode from STOP to OFFLINE if an error occurred while setting the bus parameters.

Table 121 shows the parameters of the service.

Table 121 – DP-master CI1 mode changed

Parameter name	Ind
Argument	M
AREP	M
USIF State	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

USIF State
Allowed values: OFFLINE, CLEAR (see 6.2.5.3.29)

6.2.5.3.31 Load bus Par DP-master CI1

By this service the User of a DP-master (Class 1) can load a new bus parameter set. This service is not allowed in the operation mode OFFLINE. During STOP changes to all bus parameters are possible. In the operation modes CLEAR and OPERATE of the Application Layer it is not allowed to load a bus parameter set with changes to critical parameters DL variables Data_rate and TS.

The changed bus parameters are not stored. Thus, all parameters of this service are lost after changing the operation mode of the Application Layer from STOP to OFFLINE. Changes to be stored can only be done within the bus parameter set (Bus Para) of the Init service.

Table 122 shows the parameters of the service.

Table 122 – Load bus Par DP-master CI1

Parameter name	Req	Cnf
Argument	M	
Bus Para	M	
Result(+)		S
Status		M
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Bus Para
This parameter contains the bus parameter set which shall be loaded. The structure is defined in Master CI1 Init service.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
The parameter Status indicates the reason for success or failure of the service.

Allowed values: NO, IV

6.2.5.3.32 Mark DP-master CI1

By this service the User of a DP-master (Class 1) can operate synchronously with the MS0 AR. After the User has sent this service to the Application Layer, the User will receive an acknowledgement from the Application Layer after all activated DP-slaves were polled at least once.

Table 123 shows the parameters of the service.

Table 123 – Mark DP-master CI1

Parameter name	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Dia		M
Result(-)		S
AREP		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Dia
This parameter indicates (if the value is equal TRUE) whether the DP-master has not executed a data exchange sequence to at least one activated DP-slave.

Allowed values: TRUE, FALSE

Result(-)
This selection type parameter indicates that the service request failed.

Status
The parameter Status indicates success or failure of the service. The value NO will be signaled if the Application Layer is in the operation mode OFFLINE or STOP, or a previous Mark.req is still active.

Allowed values: NO

6.2.5.3.33 Abort DP-master CI1

This service allows to release a MS0, MS1 or MM1 AR of a DP-master (Class 1). The indication may be issued by the MS1 AR if the DP-slave responds incorrectly. In this case the MS0 CRs to that DP-slaves will also be released.

Table 124 shows the parameters of the service.

Table 124 – Abort DP-master CI1

Parameter name	Req	Cnf
Argument	M	M
AREP	M	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

6.2.5.3.34 Read value DP-master Cl1

The User of a DP-master (Class 1) read any of the DL variables with this service.

Table 125 shows the parameters of the service.

Table 125 – Read value DP-master Cl1

Parameter name	Req	Cnf
Argument	M	
Variable	M	
Result		M
Value		M
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Variable
The parameter Variable selects the DL Variable whose current value has to be read, see IEC 61158-3-3 for a list of all possible DL variables.

Result
This parameter indicates that the service request succeeded.

Value
This parameter contains the actual value of the read DL Variable.

Status
The parameter Status indicates success or failure of the service.

Allowed values: OK, NO, IV

6.2.5.3.35 Delete SC DP-master Cl1

By this service the DL statistic counters can be cleared (set to zero) by the User of a DP-master (Class 1).

Table 126 shows the parameters of the service.

Table 126 – Delete SC DP-master Cl1

Parameter name	Req	Cnf
Argument	M	
Address	M	
Result		M

Argument
The argument shall convey the service specific parameters of the service request.

Address
This parameter specifies the DL-address of the statistic counters which have to be cleared.

Attribute Type Unsigend8

Allowed values: 0 to 125

Result
This parameter indicates that the service request succeeded.

6.2.5.3.36 DP-master CI1 event

By this service the events of the service DLM-Event are indicated to the User of a DP-master (Class 1) as an Event.

Table 127 shows the parameters of the service.

Table 127 – DP-master CI1 event

Parameter name	Ind
Argument	M
Event	M
Add Info	M

Argument
The argument shall convey the service specific parameters of the service request.

Event
This parameter can take the values which are allowed for the DLM-Event service.

Add Info
The parameter indicates the added info of the DL.

6.2.5.3.37 Init DP-master CI2

By this service the User initializes the DL-parameter of a DP-master (Class 2).

Table 128 shows the parameters of the service.

Table 128 – Init DP-master CI2

Parameter name	Req	Cnf
Argument	M	
Bus Para	M	
TS	M	
Data rate	M	
TSL	M	
$minT_{SDR}$	M	
$maxT_{SDR}$	M	
TQUI	M	
TSET	M	
TTR	M	
G	M	
HSA	M	
max retry limit	M	
Result		M

Argument
The argument shall convey the service specific parameters of the service request.

Bus Para
This parameter is composed of the following subparameters:

TS
This parameter contains the DL-address of this DP-master (Class 2).

Data rate
This parameter contains the data rate of this DP-master (Class 2).

TSL
This parameter contains the slot time for this DP-master (Class 2).

$minT_{SDR}$
This parameter contains the smallest station delay time for this DP-master (Class 2).

$maxT_{SDR}$
This parameter contains the largest station delay time for this DP-master (Class 2).

TQUI
This parameter contains the transmitter fall time for this DP-master (Class 2).

TSET
This parameter contains the setup time for this DP-master (Class 2).

TTR
This parameter contains the target rotation time for this DP-master (Class 2).

G
This parameter contains the GAP update factor for this DP-master (Class 2).

HSA
This parameter contains the highest station (DL-) address for this DP-master (Class 2).

max retry limit
This parameter contains the maximum number of retries for this DP-master (Class 2).

Result

This parameter indicates that the service request succeeded.

6.2.5.3.38 Reset DP-master CI2

By this service the User of a DP-master (Class 2) resets the Application Layer.

Table 129 shows the parameters of the service.

Table 129 – Reset DP-master CI2

Parameter name	Req	Cnf
Argument		

Argument

The argument shall convey the service specific parameters of the service request. Nevertheless, in this service there are no specific parameters.

6.2.5.3.39 DP-master CI2 fault

By this service the Application Layer indicates a fatal DL error to the User of a DP-master (Class 2).

Table 130 shows the parameters of the service.

Table 130 – DP-master CI2 fault

Parameter name	Ind
Argument	

Argument

The argument shall convey the service specific parameters of the service request. Nevertheless, in this service there are no specific parameters.

6.2.5.3.40 DP-master CI2 reject

By this service the Application Layer rejects a previous Read, Write, Data Transport service from the User of a DP-master (Class 2).

Table 131 shows the parameters of the service.

Table 131 – DP-master CI2 reject

Parameter name	Ind
Argument	M
AREP	M
Reason	M

Argument

The argument shall convey the service specific parameters of the service request.

AREP

This parameter is the local identifier for the desired AR (Address of DP-slave).

Reason
The parameter Status indicates success or failure of the service.

Allowed values: REJ ABORT,REJ LE,REJ PS,REJ SE,REJ IV

6.2.5.3.41 DP-master CI2 closed

By this service the Application Layer informs the User of a DP-master (Class 2) that the MS2 AR has been closed.

Table 132 shows the parameters of the service.

Table 132 – DP-master CI2 closed

Parameter name	Ind
Argument	M
AREP	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

6.2.5.3.42 DP-master CI2 event

By this service the events of the DLM are indicated to the User of a DP-master (Class 2) as an Event.

Table 133 shows the parameters of the service.

Table 133 – DP-master CI2 event

Parameter name	Ind
Argument	M
Event	M
Add Info	M

Argument
The argument shall convey the service specific parameters of the service request.

Event
This parameter can take the values which are allowed for the DLM-Event service.

Add Info
The parameter indicates the added info of the DL.

6.2.6 Management ASE

6.2.6.1 Overview

The Management ASE is used to convey parameters between a DP-master (Class 2) and a DP-master (Class 1). Additionally, the DP-master (Class 2) can read diagnosis information of the DP-master (Class 1) and its assigned DP-slaves. The access of the DP-master (Class 2) to the DP-master (Class 1) is performed according to the Client/Server access model.

6.2.6.2 Management class specification

6.2.6.2.1 Master diag class specification

6.2.6.2.1.1 Template

A Master Diag object is described by the following template:

DP ASE:			**Management ASE**
CLASS:			**Master Diag**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	Master Status
2.1.	(m)	Attribute:	USIF State
2.2.	(m)	Attribute:	Ident Number
2.3.	(m)	Attribute:	Hardware Release DP
2.4.	(m)	Attribute:	Firmware Release DP
2.5.	(m)	Attribute:	Hardware Release User
2.6.	(m)	Attribute:	Firmware Release User
3.	(m)	Attribute:	System Diagnosis
4.	(m)	Attribute:	Data Transfer List
5.	(o)	Attribute:	List of DP-slave Diag
5.1	(m)	Attribute:	DP-slave Address
5.2	(m)	Attribute:	DP-slave Diag
SERVICES:			
1.	(o)	OpsService:	Get Master Diag

6.2.6.2.1.2 Attributes

Implicit
The attribute Implicit indicates that the Master Diag object is implicitly addressed by the service.

Master Status
This parameter is composed of the following items.

USIF State
This attribute describes the state of a DP-master (Class 1).

The allowed values are shown in Table 134.

Table 134 – USIF state

Value	Meaning
0	OFFLINE
1	STOP In this state the DP-master (Class 1) does not execute any services at the MS0 and MS1 AR. The MM1 and MM2 AR can be established.
2	CLEAR The DP-master is operating the MS0 AR. The Output Data are in the safe state. The services related with MS1 AR can be executed. The MM1 and MM2 AR can be established.
3	OPERATE The DP-master is operating the MS0 AR. The Output Data are valid. The services related with MS1 AR can be executed. The MM1 and MM2 AR can be established.

Ident Number
This attribute reflects the Ident Number of the DP-master (Class 1).

Attribute Type: Unsigned16

Hardware Release DP
This attribute specifies the hardware release of the DP implementation of the DP-master (Class 1).

Attribute Type: Unsigned8

Firmware Release DP
This attribute specifies the firmware/software release of the DP implementation of the DP-master (Class 1).

Attribute Type: Unsigned8

Hardware Release User
This attribute specifies the hardware release of the User implementation of the DP-master (Class 1).

Attribute Type: Unsigned8

Firmware Release User
This attribute specifies the firmware/software release of the User implementation of the DP-master (Class 1).

Attribute Type: Unsigned8

System Diagnosis
The System Diagnosis reflects the state of all DP-slaves assigned to the DP-master (Class 1). This state itself is a one-bit-information for each DP-slave which is set to the value 1 if its Ext Diag Flag of the Diagnosis information is set or if it does not respond or if it responds incorrectly so that the MS0 AR has to be re-established. The state of a DP-slave is reset to the value 0 after the MS0 AR is successfully re-established or if the DP-slave is deactivated by the AP of the DP-master (Class 1) or the USIF State of the DP-master (Class 1) is STOP.

Attribute Type: List of 0,1

Data Transfer List
The Data Transfer List reflects the status of the data exchange mode of all DP-slaves assigned to the DP-master (Class 1).

This status itself is a one-bit-information for each DP-slave which is set to the value 1 if the data exchange with the assigned DP-slave was successfully or if between two data exchange requests is only one diagnosis request. All other cases reset the status bit-information of the DP-slave to the value 0.

Attribute Type: List of 0,1

List of DP-slave Diag
This attribute is composed of the following list elements.

DP-slave Address
This attribute specifies the DL-address of the DP-slave.

Attribute Type: Unsigned8

Allowed values: 0 to 125

DP-slave Diag
This attribute reflects the diagnosis information of the DP-slave inclusive the information added by the DP Application Layers for each DP-slave.

6.2.6.2.1.3 Invocation of master diag object

For the invocation of the Master Diag object the following rules apply:

Only one Master Diag object shall be invoked in a DP-master (Class 1).

6.2.6.2.2 Master parameter class specification

6.2.6.2.2.1 Overview specification

The Master Parameter object consists of one Bus Para element and one or more Slave Para elements. The values for the Master Parameter object may be loaded via the MM1 AR.

The Download of the attribute values for Master Parameter object may be done in one sequence or the attribute values for of the Bus Para element and for the Slave Para elements may be loaded separately.

For the download of the attribute values of Master Parameter object the Application Layer has to be in the operation mode STOP. The operation mode of the Application Layer of the DP-master (Class 1) shall be changed with the Act Param service via the MM1 AR in conjunction with the local service Set Mode DP-master CI1.

If the attribute values for Master Parameter object are loaded in one sequence the values have to be stored first in an intermediate buffer. After reception of the Act Param or Act Para Brct service the User has to transfer the values from the intermediate buffer to the attributes of the Master Parameter object. Afterwards the User shall transfer these values accordingly with the Load ARL, Load CRL (parameter Update set to FALSE) and Load Bus Par DP-master CI1 services. The Act Param service shall be confirmed according to the following rules:

– positive, if all load services are confirmed positively,
– negative, if one or more load services are confirmed negatively.

If the attribute values for the Bus Para elements are loaded the values have to be stored first in an intermediate buffer. After reception of the Act Param or Act Para Brct service the User has to transfer the values from the intermediate buffer with the Load Bus Par DP-master CI1 service to the DP Application Layer. The Act Param service shall be confirmed positively, if the previously requested load service are confirmed positively and after the values has been transferred to the attributes of the Master Parameter object.

The Act Param service shall be confirmed negatively, if the previously requested load services are confirmed negatively. In this case the previously transmitted values for the Bus Para elements shall be omitted.

If the attribute values for one Slave Para element are loaded the values have to be stored first in an intermediate buffer. After reception of the Act Param service the User has to transfer the values from the intermediate buffer to the attributes of the Master Parameter object.

Afterwards the User shall transfer these values accordingly with the ARL Slave Update DP-master CI1 and the Load CRL (parameter Update set to TRUE) services. The Act Param service shall be confirmed according to the following rules:

– positive, if all load services are confirmed positively,
– negative, if one or more load services are confirmed negatively.

Care shall be taken by the User because updating of one Slave Para element may interfere with already loaded Bus Para and may change the system behavior.

6.2.6.2.2.2 Template

A Master Parameter object is described by the following template:

DP ASE:			**Management ASE**
CLASS:			**Master Parameter**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	Bus Para
2.1.	(m)	Attribute:	Bus Para Len
2.2.	(m)	Attribute:	TS
2.3.	(m)	Attribute:	Data rate
2.4.	(m)	Attribute:	T_{SL}
2.5.	(m)	Attribute:	min T_{SDR}
2.6.	(m)	Attribute:	max T_{SDR}
2.7.	(m)	Attribute:	T_{QUI}
2.8.	(m)	Attribute:	T_{SET}
2.9.	(m)	Attribute:	T_{TR}
2.10.	(m)	Attribute:	G
2.11.	(m)	Attribute:	HSA
2.12.	(m)	Attribute:	max retry limit
2.13.	(m)	Attribute:	Error Action Flag
2.14.	(o)	Attribute:	Isochronous Mode
2.15.	(o)	Attribute:	IsoM Freeze
2.16.	(o)	Attribute:	IsoM Sync
2.17.	(m)	Attribute:	Min Slave Interval
2.18.	(m)	Attribute:	Poll Timeout
2.19.	(m)	Attribute:	Data Control Time
2.20.	(o)	Attribute:	Alarm Max
2.21.	(m)	Attribute:	Max User Global Control
2.22.	(m)	Attribute:	Master User Data Len
2.23.	(m)	Attribute:	Master Class2 Name
2.24.	(m)	Attribute:	Master User Data
2.25.	(o)	Attribute:	T_{CT}
2.26.	(o)	Attribute:	maxT_{SH}
3.	(o)	Attribute:	List of Slave Para
3.1.	(m)	Attribute:	Slave DL-address
3.2.	(m)	Attribute:	Slave Para Len
3.3.	(o)	Attribute:	Extra Alarm SAP
3.4.	(o)	Attribute:	DPV1 Data Types
3.5.	(o)	Attribute:	DPV1 Supported
3.6.	(m)	Attribute:	Fail Safe
3.7.	(m)	Attribute:	Active
3.8.	(m)	Attribute:	Slave Type
3.9.	(m)	Attribute:	New Prm
3.10.	(m)	Attribute:	Max Diag Data Len
3.11.	(o)	Attribute:	Max Channel Data Length
3.12.	(m)	Attribute:	Diag Upd Delay
3.13.	(o)	Attribute:	Alarm Mode
3.14.	(m)	Attribute:	NA to Abort
3.15.	(m)	Attribute:	Ignore AClr
3.16.	(o)	Attribute:	Publisher Flag
3.17.	(o)	Attribute:	MS1 Timeout
3.18.	(o)	Attribute:	Prm Command supported
3.19.	(m)	Attribute:	Prm Data Len
3.20.	(m)	Attribute:	Prm Data
3.21.	(m)	Attribute:	Cfg Data Len
3.22.	(m)	Attribute:	Cfg Data
3.23.	(m)	Attribute:	Add Tab Len
3.24.	(m)	Attribute:	Add Tab
3.25.	(m)	Attribute:	Slave User Data Len
3.26.	(o)	Attribute:	Slave User Data
3.27.	(o)	Attribute:	Ext User Prm Data Len
3.28.	(o)	Attribute:	Ext User Prm Data
3.29.	(m)	Attribute:	List of assigned CREPs for the MS0 AR
3.29.1.	(m)	Attribute:	CREP

3.30.	(m)	Attribute:	Assigned AREP for the MS1 AR
3.31.	(m)	Attribute:	Assigned AREP for the MS3 AR
4.	(m)	Attribute:	Assigned AREP for the MS0 AR

SERVICES:

1.	(o)	OpsService:	Start Seq
2.	(o)	OpsService:	Download
3.	(o)	OpsService:	Upload
4.	(o)	OpsService:	End Seq
5.	(o)	OpsService:	Act Para Brct
6.	(o)	OpsService:	Act Param

6.2.6.2.2.3 Attributes

Implicit
This attribute indicates that the Master Parameter object is implicitly addressed by the services.

Bus Para
This combined attribute consists of the following items.

Bus Para Len
This attribute contains the length of Bus Para inclusive the length parameter.

Attribute Type: Unsigned16

Allowed values: 66 to 2^{16} - 1

TS
This attribute contains the own DL-address of the DP-master.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Data rate
This attribute contains the code number of the chosen data rate.

Attribute Type: Unsigned8

The allowed values are shown in Table 135.

Table 135 – Data rate

Value	Meaning
0	9,6 kbit/s
1	19,2 kbit/s
2	93,75 kbit/s
3	187,5 kbit/s
4	500 kbit/s
6	1 500 kbit/s
7	3 000 kbit/s
8	6 000 kbit/s
9	12 000 kbit/s
10	31,25 kbit/s
11	45,45 kbit/s

T_{SL}, min T_{SDR}, max T_{SDR}, T_{QUI}, T_{SET}, T_{TR}, G, HSA, max retry limit
These attributes are described in IEC 61158-3-3 and IEC 61158-4-3.

Error Action Flag
This attribute reflects the reaction of the DP-master (Class 1) in case of failed data transfers. This attribute will be evaluated in the operation mode OPERATE and CLEAR of the DP-master (Class 1). If the data transfer to at least one activated DP-slave was impossible for a certain period of time (Data Control Time) and if the Error Action Flag is set, the operation mode will change from OPERATE to CLEAR. If the Error Action Flag is cleared, the DP-master (Class 1) will remain in the operation mode OPERATE despite erroneous data transfers.

If the operation mode of the DP-master (Class 1) shall be changed from CLEAR to OPERATE with the Error Action Flag set to TRUE the data transfer to all activated DP-slaves have to be performed successfully for a certain period of time (Data Control Time)

The allowed values are shown in Table 136.

Table 136 – USIF state

Value	Meaning
0	no change of the operation mode in case of an error
1	change of the operation mode in case of an error

Isochronous Mode
This attribute indicates if the DP-master (Class 1) shall operate in the Isochronous Mode. Two different models concerning the interface between the DP Application Layer and the application of the DP-master (Class 1) itself are provided by fieldbus DP for the DP-master (Class 1), see 6.2.2.4.3.2.

The allowed values are shown in Table 137.

Table 137 – Isochronous mode

Value	Meaning
Not Synchronized	the DP-master (Class 1) shall not operate in the Isochronous Mode
Buffered Synchronized	the DP-master (Class 1) shall operate in the Buffer Synchronized Isochronous Mode
Enhanced Synchronized	the DP-master (Class 1) shall operate in the Enhanced Synchronized Isochronous Mode

IsoM Freeze
This Boolean attribute indicates if the synchronization message (SYNCH) shall be sent with the Control Command Freeze set to TRUE.

Allowed values: TRUE, FALSE

IsoM Sync
This Boolean attribute indicates if the synchronization message (SYNCH) shall be sent with the Control Command Sync set to TRUE.

Allowed values: TRUE, FALSE

Min Slave Interval
This attribute specifies the smallest allowed period of time between two consecutive DP-slave poll cycles. This ensures that the sequence of cyclic service requests from the DP-master (Class 1) can be handled by the DP-slave. This period of time will be complied by the DP-master (Class 1) for every cyclic Master-slave service except for the Global Control service. For the Global Control service the User is responsible for compliance with Min Slave Interval.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 100 μs

Poll Timeout
In the case of MM1 AR this attribute specifies the maximum period of time needed to get the response.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 1

Data Control Time
This attribute specifies the maximum period of time for a Data Exchange with every activated DP-slave. This Time is used for sending out the Global Control cyclically.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 10

Alarm Max
This variable contains the maximum number of alarms per DP-slave, which can be handled by the DP-master. It describes a Master capability. The Master shall be able to handle a minimum number of 7 alarms, because there are existing 7 different alarm classes.

Attribute Type: Unsigned8

Allowed values: 7 to 32

Max User Global Control
This attribute defines the maximum number of Global Control requests, which may be started by the User at the same time. This parameter describes the ability of the DP-master (Class 1). A practicable value for Max User Global Control is 16. For each of the 8 Slave groups one Sync and one Freeze Command may be started at the same time.

Attribute Type: Unsigned8

Allowed values: 1 to 255

Master User Data Len
This attribute contains the length of Master User Data inclusive the length parameter.

Attribute Type: Unsigned16

Allowed values: 34 to 2^{16} - 1

Master Class2 Name
This attribute indicates the name of the DP-master (Class 2) which provided the values for the attributes of this Master Parameter object.

Attribute Type: Visible String(32)

Master User Data
This attribute contains manufacturer specific data which is necessary for the bus parameter set.

Attribute Type: Octet String

T_{CT}
This attribute specifies the time (in units of t_{BIT}) that is necessary to handle a complete Isochronous DP cycle and corresponds to the time T_{DP}.

Allowed values: 1 to 2^{24} - 1

maxT_{SH}
This attribute specifies the maximum allowed time shift (in units of t_{BIT}) of the time T_{DP} measured by the DP-master (Class 1). The application of the DP-master (Class 1) will be informed if the measured Isochronous DP cycle time exceeds this limit by the service SYNCH Delayed indicating the occurred shift.

Allowed values: 1 to 2^{8} - 1

Slave Para
This combined attribute consists of the following items.

Slave DL-address
This attribute reflects the DL-address of the DP-slave. This address is used as an identifier (Area Code) to address the corresponding slave parameter set, e.g. for up- or download purposes.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Slave Para Len
This attribute contains the length of Slave Para inclusive the length parameter. A slave parameter set can be deleted by setting the Slave Para Len to zero.

Attribute Type: Unsigned16

Allowed values: 0 to 2^{16} - 1

Extra Alarm DLSAP
This Boolean attribute selects the Channel for the Alarm Ack service.

– If the attribute is set to FALSE the DP-master (Class 1) acknowledges alarms via DLSAP 51.
– If the attribute is set to TRUE the DP-master (Class 1) acknowledges alarms via DLSAP 50.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

DPV1 Data Types
This Boolean attribute reflects whether the DP-slave uses data types.

- If the attribute is set to FALSE the DP-slave uses only basic types (Octet, Word, Array) for the Data Description.
- If the attribute is set to TRUE the DP-slave uses data types for the Data Description.

Attribute Type: Boolean

DPV1 Supported
This Boolean attribute reflects whether the DP-slave uses the extended functionality.

– If the attribute is set to FALSE the DP-slave uses no extended functionality.
– If the attribute is set to TRUE the DP-slave uses extended functionality.

Attribute Type: Boolean

Fail Safe
This Boolean attribute reflects whether the DP-slave supports the Fail Safe functionality. If the attribute is set to FALSE the DP-slave does not support the Fail Safe functionality. If the attribute is set to TRUE the DP-slave supports the Fail Safe functionality.

Attribute Type: Boolean

Active
This Boolean attribute reflects whether the DP-slave has to be activated. If the attribute Active has the value FALSE, the DP-master (Class 1) stops the DP-slave handling to this assigned DP-slave. In this state it is allowed to change the slave parameter set, for example by a DP-master (Class 2) with the Download service. If the Active-Flag has the value TRUE, the DP-slave handling will be done by the DP-master (Class 1) to this assigned DP-slave. Changing of the slave parameter set in this state is limited (see also the description of the attribute New Prm).

Attribute Type: Boolean

Slave Type
This attribute contains a manufacturer specific type denotation for the DP-slave, see Table 138.

Attribute Type: Unsigned8

The allowed values are shown in Table 138.

Table 138 – Slave type

Value	Meaning
0	DP-slave
1-15	reserved
16-255	manufacturer specific

New Prm
This Boolean attribute reflects whether parameterization data has to be transferred to the DP-slave in the data exchange mode. If this attribute is set to FALSE the Output Data will be transferred to this DP-slave in the data exchange mode. If this attribute is set to TRUE, for one data cycle parameterization data will be transferred instead of Output Data. The consistency of the parameterization data has to be guaranteed by local measures. After transmitting this parameterization data successfully the DP-master (Class 1) sets this attribute to FALSE and the data exchange mode to the DP-slave will be continued.

Attribute Type: Boolean

Max Diag Data Len
This attribute contains the maximum length of Diag Data for the DP-slave.

Attribute Type: Unsigned8

Allowed values: 6 to 244

Max Channel Data Length
This attribute defines the maximum length of a MS1-APDU (Length parameter and 4 Octet Header) for the corresponding DP-slave.

Attribute Type: Unsigned8

Allowed values: 4 to 244

Diag Upd Delay
The attribute reflects the number of Diag service requests performed in the state DIAG2 of the DP-master (Class 1) while Prm Req is set in Diag Data of the DP-slave (for DP-slaves with reduced performance).

Attribute Type: Unsigned8

Allowed values: 0 to 15 (extendible up to 255)

Alarm Mode
This attribute specifies the maximum number of possible active alarms.

Attribute Type: Unsigned8

The allowed values are shown in Table 139.

Table 139 – Alarm mode

Value	Meaning
0	1 alarm of each type
1	2 alarms in total
2	4 alarms in total
3	8 alarms in total
4	12 alarms in total
5	16 alarms in total
6	24 alarms in total
7	32 alarms in total

NA To Abort
This Boolean attribute reflects the behavior of the DP-master (Class 1) if the assigned DP-slave does not respond at the MS0 AR in the DIAG2 state and the data exchange mode.

– If this attribute is set to FALSE the DP-master continues in case of no response (Status = NA) of the assigned DP-slave.
– If this attribute is set to TRUE the DP-master aborts the MS0 AR in case of no response (Status = NA) of assigned the DP-slave (abort means sending an unlock command and to re-establish the MS0 AR).

Attribute Type: Boolean

Ignore AClr
This Boolean attribute in combination with the Error Action Flag allows a DP-slave specific reaction of the DP-master (Class 1) in case of failed data transfers.

– If the attribute is set to FALSE the DP-master (Class 1) processes the auto-clear function as defined in the Error Action Flag.
– If the attribute is set to TRUE the DP-master (Class 1) ignores the auto-clear function as defined in the Error Action Flag.

Attribute Type: Boolean

Publisher Flag
This Boolean attribute indicates if the correspondent DP-slave shall operate as Publisher. If the attribute is set to FALSE the correspondent DP-slave shall not operate as Publisher. If the attribute is set to TRUE the correspondent DP-slave shall be addressed as Publisher by the DP-master (Class 1).

Attribute Type: Boolean

Allowed values: TRUE, FALSE

MS1 Timeout
This attribute contains the value for the Timeout of the acyclic communication on the MS1 AR and specifies the maximum period of time it may take a DP-slave to provide a response. The time contained in this attribute is the maximum time of the assigned DP-slave.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 10

Prm Command supported
This Boolean attribute reflects whether the DP-slave uses PrmCmd within the extended functionality:

– If the attribute is set to FALSE the DP-slave supports no PrmCmd.
– If the attribute is set to TRUE the DP-slave supports PrmCmd.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Prm Data Len
This attribute contains the length of Prm Data inclusive the length parameter.

Attribute Type: Unsigned8

Allowed values: 9 to 246

Prm Data
This attribute reflects the parameterization data sent to the corresponding DP-slave by the DP-master (Class 1). The structure of this attribute is described in 6.2.10.4.2.2.

Cfg Data Len
This attribute contains the length of Cfg Data inclusive the length parameter.

Attribute Type: Unsigned8

Allowed values: 3 to 246

Cfg Data
This attribute reflects the configuration information sent to the DP-slave. The structure of this attribute is described in 6.2.10.4.2.2.

Add Tab Len
This attribute contains the length of Add Tab inclusive the length parameter.

Attribute Type: Unsigned16

Allowed values: 2 to 2^{16} - 31

Add Tab
This attribute contains the address assignment table of the DP-slave. In case of a Programmable Logic Controller, this list contains the Programmable Logic Controller addresses of the decentralized addresses.

Slave User Data Len
This attribute contains the length of the Slave User Data inclusive the length parameter.

Attribute Type: Unsigned16

Allowed values: 2 to 2^{16} - 31

Slave User Data
This attribute contains manufacturer specific data that characterize the DP-slave for the DP-master (Class 1).

Attribute Type: Octet String

Ext User Prm Data Len
This attribute contains the length of Ext Prm Data inclusive the length parameter.

Attribute Type: Unsigned8

Allowed values: 2 to 246

Ext User Prm Data
This attribute reflects the extended user parameterization data sent to the correspondent DP-slave by the DP-master (Class 1). The DP-master (Class 1) will send this data between the parameter data and the configuration information.

List of assigned CREPs for the MS0 AR
This attribute contains the CREPs of the MS0 AR assigned for this DP-slave and is composed of the following list elements:

CREP
This attribute contains one CREP of the MS0 AR.

Assigned AREP for the MS1 AR
This attribute contains the AREP of the MS1 AR assigned for this DP-slave.

Assigned AREP for the MS3 AR
This attribute contains the AREP of the MS3 AR assigned for this DP-slave.

Assigned AREP for the MS0 AR
This attribute contains the AREP of the MS0 AR assigned for all DP-slaves.

6.2.6.2.2.4 Invocation of the master parameter object

For the invocation of the Master Parameter object the following rules apply:

– Only one Master Parameter object can be invoked in a DP-master (Class 1).
– Up to 125 Slave Para elements with a unique value for the attribute Slave DL-address can be contained in a Master Parameter object of a DP-master (Class 1).

6.2.6.3 Management service specification

6.2.6.3.1 Get master diag

With this service a DP-master (Class 2) can read a single element (Master Status, System Diagnosis, Data Transfer List, DP-slave Diag of a single DP-slave) from the Master Diag object of a DP-master (Class 1). This service shall only be used in conjunction with the MM1 AR.

Table 140 shows the parameters of the service.

Table 140 – Get Master Diag

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
MDiag Identifier	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Diagnosis Data			M	M(=)
DP-slave Diag			S	S(=)
System Diagnosis			S	S(=)
Data Transfer List			S	S(=)
Master Status			S	S(=)
Result(-)			S	S(=)
AREP			M	M(=)
Status			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the desired AR.

MDiag Identifier
This parameter specifies the Element of the Master Diag object to be read.

The allowed values are shown in Table 141.

Table 141 – MDiag identifier

Value	Meaning
0-125	DP-slave Diag of the DP-slave with the corresponding DL-address
126	System Diagnosis
127	Master Status
128	Data Transfer List

Result(+)
This selection type parameter indicates that the service request succeeded.

Diagnosis Data
This parameter contains one of the following elements of the Master Diag object as requested.

DP-slave Diag
This parameter contains the Diag Data of the addressed DP-slave.

System Diagnosis
This parameter contains the System Diagnosis of the Master Diag object (see 6.2.6.2.1)

Master Status
This parameter contains the Master Status of the Master Diag object (see 6.2.6.2.1)

Data Transfer List
This parameter contains the Data Transfer List of the Master Diag (see 6.2.6.2.1)

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the failure.

Allowed values: DS, NA, RS, RR, UE, RE, TO, FE, NE, IP, AD, EA, LE

6.2.6.3.2 Start seq

With the Start Seq service the beginning of an Upload or Download sequence is signalized by the DP-master (Class 2) in the DP-master (Class 1). In this case the DP-master (Class 1) can set an access protection for the addressed area until the End Seq service signals the end of the sequence. The addressed data can be transmitted in blocks with the services Upload and Download. This service shall only be used in conjunction with the MM1 AR.

Table 142 shows the parameters of the service.

Table 142 – Start Seq

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M(=)		
Timeout	M	M(=)		
Result(+)			S	S(=)
AREP			M	M
Max Len Data Unit			C	C
Result(-)			S	S(=)
AREP			M	M
Status			M	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the desired AR.

Area Code
The parameter Area Code identifies the element of the Master Parameter object which shall be loaded or read. If the parameter Area Code is set to a value of 255, local access protection for a determined Area Code is not guaranteed for the services following. The local access protection ensures that the User of the DP-master (Class 1) cannot access the partially loaded data during the service sequence.

Attribute Type: Unsigned8

The allowed values are shown in Table 143.

Table 143 – Area code (start seq)

Value	Meaning
0 to 125	Slave Para of the DP-slave with the corresponding DL-address
126	reserved
127	Bus Para
128	reserved
129	statistic counters
130 to 135	reserved
136 to 139	for the transfer of the Master Parameter Set
140 to 254	reserved
255	Start Seq: No local access protection

Timeout
This parameter defines the control time between two successive Upload/Download services. If this time expires the access protection is deactivated. The previously transferred data are invalid.

Attribute Type: Unsigned16

Time Base: 1

Result(+)
This selection type parameter indicates that the service request succeeded.

Max Len Data Unit
The parameter Max Len Data Unit defines the maximum possible length of the parameter Data in the following Upload/Download APDUs.

Attribute Type: Unsigned8

Range: 1 to 240

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the failure.

Allowed values: DS, NA, RS, RR, UE, TO, FE, RE, NE, AD, IP, NI, SE, SC, EA, LE

6.2.6.3.3 Download

This service is used to transfer the values for the attributes of the Master Parameter object from the DP-master (Class 2) to a DP-master (Class 1). The beginning of the Download sequence can be signaled with the service Start Seq. The actual data set is transmitted in blocks with the Download service in the addressed area. After the transfer is completed the sequence shall be terminated with the End Seq service, if the Download sequence was started with the Start Seq service. This service shall only be used in conjunction with a MM1 AR.

Table 144 shows the parameters of the service.

Table 144 – Download

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M		
Add Offset	M	M		
Data	M	M		
Result(+)			S	S(=)
AREP			M	M
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the desired AR.

Area Code
The parameter Area Code identifies the element of the Master Parameter object which shall be loaded.

Attribute Type: Unsigned8

Allowed values: see 6.2.6.3.2

Add Offset
This parameter contains the offset to the begin of the area which is specified by the Area Code.

Attribute Type: Unsigned16

Data
This parameter contains the data to be transferred.

Attribute Type: Octet String

Length: 1 to 240

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the failure.

Allowed values: DS, NA, RS, RR, UE, TO, FE, RE, NE, AD, EA, LE, SC, NI, NC

6.2.6.3.4 Upload

This service is used to transfer the values of the attributes of the Master Parameter object from the DP-master (Class 1) to a DP-master (Class 2). The Upload sequence can be

executed in the same way as the Download sequence. It is possible to transfer single data areas without the services Start Seq and End Seq. In this case the consistency of the transferred data is not guaranteed. This service shall only be used in conjunction with a MM1 AR.

Table 145 shows the parameters of the service.

Table 145 – Upload

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M		
Add Offset	M	M		
Data Len	M	M		
Result(+)			S	S(=)
AREP			M	M
Data			C	C(=)
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the desired AR.

Area Code
The parameter Area Code identifies the element of the Master Parameter object which shall be read

Attribute Type: Unsigned8

Allowed values: see 6.2.6.3.2

Add Offset
This parameter identifies the offset to the begin of the area which is specified by the Area Code.

Attribute Type: Unsigned16

Data Len
The parameter defines the length of the requested data.

Attribute Type: Unsigned8

Allowed values: 1 to 240

Result(+)
This selection type parameter indicates that the service request succeeded.

Data
This parameter contains the requested data.

Attribute Type: Octet String

Length: 1 to 240

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the failure.

Allowed values: DS, NA, RS, RR, UE, TO, FE, RE, NE, EA, LE, NI, SC, AD

6.2.6.3.5 End seq

This service is used to mark the end of a sequence of consistent transfers of data blocks of the Master Parameter object. This service shall only be used in conjunction with a MM1 AR.

Table 146 shows the parameters of the service.

Table 146 – End Seq

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Result(+)			S	S(=)
AREP			M	M
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the desired AR.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the failure.

Allowed values: DS, NA, RS, RR, UE, TO, FE, RE, NI, SE, NE, AD, EA, LE, NC

6.2.6.3.6 Act Para Brct

This service is used to activate the previously transferred values for the Master Parameter object by the DP-master (Class 2). This service may be performed simultaneously to one (single) or many (multicast) DP-master (Class 1). In the operation modes CLEAR and OPERATE of the DP-master (Class 1), the activation of the Bus Para element with changed DL parameters data rate or TS will be rejected. This service is only locally confirmed because a possible change of the data rate shall take effect at all addressed DP-masters (Class 1) at the same time. This service shall only be used in conjunction with a MM2 AR.

Table 147 shows the parameters of the service.

Table 147 – Act Para Brct

Parameter name	Req	Ind	Cnf
Argument	M	M	
AREP	M	M	
Area Code	M	M	
Result(+)			S
AREP			M
Result(-)			S
AREP			M
Status			M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the desired AR.

Area Code
The parameter Area Code identifies the element of the Master Parameter object which shall be activated.

Attribute Type: Unsigned8

The allowed values are shown in Table 148.

Table 148 – Area code (Act Para Brct)

Value	Meaning
0 to 126	not allowed
127	Bus Para
128 to 129	reserved
130 to 135	reserved
136 to 139	for the activation of the Master parameter object
140 to 254	reserved
255	not allowed

In the presence of several DP-masters (Class 2) it shall be guaranteed that the old and the new bus parameter sets are compatible.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the failure.

Allowed values: DS

6.2.6.3.7 Act param

This service is used by the DP-master (Class 2) for the following purposes:

– to activate or to deactivate a parameterized DP-slave,
– to change the operation mode of the DP Application Layer,
– to accept and activate the values for the Bus Para Elements of the Master Parameter object.

If the loaded bus parameter set contains a changed data rate or station address for the DP-master (Class 1) the new bus parameter set shall be activated with the Act Para Brct service. This service shall only be used in conjunction with a MM1 AR.

Table 149 shows the parameters of the service.

Table 149 – Act param

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M(=)		
Activate	M	M(=)		
Result(+)			S	S(=)
AREP			M	M
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier of the desired AR.

Area Code
The parameter Area Code identifies the element of the Master Parameter object which shall be activated/deactivated with the exception of the operation mode.

Attribut Type: Unsigned8

The allowed values are shown in Table 150.

Table 150 – Area code (Act param)

Value	Meaning
0 to 125	Slave Para of the DP-slave with the corresponding DL-address The (Flag/Bit) Active in the slave parameter set of the DP-master (Class 1) is set as specified in the parameter Activate.
126	Reserved
127	Bus Para
128	Operation mode (description, see 6.2.5.3.29)
129	Reserved
130 to 135	reserved
136 to 139	activation of the Master Parameter Set
140 to 255	reserved

Activate
The meaning and the allowed values of this parameter are depending on the Area Code.

Attribute Type: Unsigned8

The allowed values are shown in Table 151.

Table 151 – Activate

Area Code	Allowed values for the parameter Activate	
0 to 125: (Slave Para)	0	(= Deactivate)
	128	(= Activate)
127: (Bus Para)	255	(= Activate)
128: (operation mode)	64	(= STOP)
	128	(= CLEAR)
	192	(= OPERATE)

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter provides detailed information about the failure.

Allowed values: DS, NA, RS, RR, UE, TO, FE, RE, NE, AD, IP, SC, NI, DI, EA, LE

6.2.7 Load region ASE

6.2.7.1 Overview

A Load Region object represents an unstructured memory area whose contents may be uploaded (Pull) or downloaded (Push). Unstructured in this context means that the memory area is represented only as an ordered sequence of octets. No other structure is apparent.

A Load Region object may represent an unnamed volatile memory area, such as that implemented by dynamic computer memory, or a named non-volatile memory object, such as a file. The contents of a Load Region object are referred to as a load image. Load images may contain programs or data. The transfer of the LR Data to or from a Load Region object is performed using the Load Region services.

The Load Region objects shall clearly be separated from the Process Data objects and the Function Invocation objects. No overlapping is allowed.

To maintain integrity, only one download process for a Load Region object is permitted at a time. The Load Region State attribute of the Load Region object is used to indicate whether the Load Region object is empty, being downloaded, or loaded (see the definition of the attribute for a complete list of the states). The Load Region object may be cleared using the Initiate Load service.

The Load Region State attribute also indicates the progress of an upload. Multiple uploads of a Load Region object via different ARs are allowed. This attribute is defined to separate the state of the contents of the Load Region object from the operation of having the contents uploaded.

The Load Region ASE provides services that permit an AP to initiate the upload or download (Pull or Push) of Load Region objects. They identify the Load Region object and whether to Pull or Push as parameters of the initiate load request.

The LR Data may be transferred in segments by using the Pull Segment or the Push Segment service.

LR Data segments are uploaded by the Client issuing Pull Segment requests for them. The AP that contains the LR Data responds by returning the requested segment, indicating with a parameter when the final segment of the LR Data has been returned.

LR Data segments are downloaded by the Client having the AP containing the LR Data image by sending individual segments to the Server and wait for a response that indicates whether the LR Data segment was received. In this case, the Client indicates when the last LR Data segment has been transferred. After the last LR Data segment has been received, the AP that initiated the transfer ends the load process by issuing a Terminate Load request to the remote AP. The Terminate Load request may also be used by an AP if the load process cannot be completed successfully.

To avoid deadlocks at the Server a timeout shall be issued to control the activities of an AR regarding the upload or download procedures.

6.2.7.2 Load region class specification

6.2.7.2.1 Template

The Load Region object is modeled as a Server object. The Load Region object specifies the attributes and services used by Client APs to upload and/or download LR Data. The specification of the Client AP is beyond the scope of this specification.

A Load Region object is described by the following template:

DP ASE:			**Load Region ASE**
CLASS:			Load Region
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Identifier
2	(m)	Attribute:	LR Size
3.	(m)	Attribute:	Access Rights MS1
4.	(m)	Attribute:	Access Rights MS2
5.	(c)	Constraint:	Access Rights MS2 = Right to Pull or Right to Pull/Push
5.1.	(m)	Attribute:	List of SCL Pull
5.1.1.	(m)	Attribute:	SCL
6.	(c)	Constraint:	Access Rights MS2 = Right to Push or Right to Pull/Push
6.1.	(m)	Attribute:	List of SCL Push

6.1.1.	(m)	Attribute:	SCL
7.	(m)	Attribute:	Load Region State
8.	(m)	Attribute:	Actual LR Size
9.	(o)	Attribute:	Identifier of Related Function Invocation
10.	(m)	Attribute:	Max Segment Length
11.	(m)	Attribute:	Max Response Delay
12.	(m)	Attribute:	Uploadcount
13.	(o)	Attribute:	Load Region Additional Information
14.	(o)	Attribute:	Load Region Name
15.	(o)	Attribute:	Local Detail

SERVICES:

1.	(m)	OpsService:	Initiate Load
2.	(o)	OpsService:	Push Segment
3.	(o)	OpsService:	Pull Segment
4.	(m)	OpsService:	Terminate Load

6.2.7.2.2 Attributes

Identifier
This key attribute is a triplet composed of API, Slot Number and LR Index to define to which Application Process Identifier, module and to which index within the specified module the Load Region object belongs. This Identifier shall not be used by another Action object, Load Region object, Function Invocation object or Process Data object.

Attribute Type: Unsigned8, Unsigned8, Unsigned16

Allowed triplet values: [0 to 254, 0 to 254, 0 to 65 199 (65 200 to 65 535 reserved)]

LR Size
This attribute specifies the length of the Load Region object in octets.

Attribute Type: Unsigned32

Allowed values: 1 to 2^{32} - 1

Access Rights MS1
This attribute defines the type of access rights defined for a MS1 AR for the Load Region object.

Attribute Type: Unsigned8

The allowed values are shown in Table 152.

Table 152 – Access rights MS1

Value	Meaning
0	Right to Push
1	Right to Pull
2	Right to Pull/Push
3	No Access

Access Rights MS2
This attribute defines the type of access rights defined for a MS2 AR for the Load Region object.

Attribute Type: Unsigned8

The allowed values are shown in Table 153.

Table 153 – Access rights MS2

Value	Meaning
0	Right to Push
1	Right to Pull
2	Right to Pull/Push
3	No Access

List of SCL Pull
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one value for Pull access protection for this Load Region object. The rules for accepting or rejecting a Pull access are described in 6.2.1.3 according to the read access.

Attribute Type: Unsigned8

List of SCL Push
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one value for Push access protection for this Load Region object. The rules for accepting or rejecting a Push access are described in 6.2.1.3 according to the write access protection.

Attribute Type: Unsigned8

Load Region State
This attribute specifies the state of the Load Region object.

Attribute Type: Unsigned8

The allowed values of this attribute are shown in Table 154.

Table 154 – Load region state

Value	Meaning
0	DOWNLOADABLE
1	DOWNLOADING
2	DOWNLOAD SUCCESS
3	LOADED
4	IN-USE

Load Region State attribute constraints:

– A Load Region object whose contents are stored in non-erasable memory cannot support downloading transfers. Its valid state values are LOADED and IN USE.

– A Load Region object whose contents are stored in erasable memory supports downloading transfers. Its valid state values are DOWNLOADABLE, DOWNLOADING, DOWNLOADING SUCCESS, LOADED and IN USE,

Actual LR Size
This attribute specifies the used length of the Load Region object in octets. A Load Region object whose content is cleared shall have a value of zero for this attribute.

Attribute Type: Unsigned32

Allowed values: 0 to 2^{32} - 1

Identifier of Related Function Invocation
This optional attribute is a triplet composed of API, Slot Number and FI Index to define to which Application Process Identifier, module and to which index within the specified module the related Function Invocation object belongs.

Attribute Type: Unsigned8, Unsigned8, Unsigned16

Allowed triplet values: Allowed values:[0 to 254, 0 to 254, 0 to 65 199 (65 200 to 65 535 reserved)]

Max Segment Length
This attribute specifies the maximum length in octets of each Push Segment or Pull Segment which shall be used for the up- or download of this Load Region object.

Attribute Type: Unsigned8

Allowed values: 1 to 234

Max Response Delay
This attribute specifies the maximum time period (in units of 10) the response of an Initiate Load, Pull Segment, Push Segment or Terminate Load service request may be delayed by the Server due to processing of the request. The value of this attribute shall be less than the value of the attribute MS1 Timeout of the ARL DP-master (Class 1) object.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Uploadcount
This attribute indicates the number of Clients which currently upload the contents of the Load Region object. The value of this attribute has to be zero to download the Load Region object. This attribute is incremented each time a Client uploads the Load Region object in parallel. This attribute is decremented each time a Client finishes the upload of the Load Region object.

Attribute Type: Unsigned8

Allowed values: 0 to 255

Load Region Additional Information
This optional attribute contains additional information about the Load Region object, e.g. permitted data formats.

Attribute Type: Octet String

Load Region Name
This optional attribute defines the name of the Load Region object.

Attribute Type: Visible String(32)

Local Detail
This optional attribute contains additional local information about the Load Region object.

Attribute Type: Octet String

6.2.7.2.3 Invocation of the load region object

For the invocation of the Load Region object the following rules apply.

– Load Region objects shall not exceed the total length of $2^{32} - 1$ octets. The attribute LR Size has to be set accordingly.
– At least the Push Segment or Pull Segment or both services shall be allowed for the access of the Load Region object.
– At least one AR shall have access to the Load Region object.
– Only one Function Invocation shall be related to a Load Region object.
– The access rights shall be set accordingly to the allowed service(s).

6.2.7.3 Load region service specification

6.2.7.3.1 Initiate load

This service is used by an AP to initiate the download or upload of a Load Region. A parameter of the service indicates whether the load image will be transferred into the Load Region object using the Pull Segment service or from the Load Region object using the Push Segment service. This service shall only be used in conjunction with the MS1 or MS2 AR. Table 155 shows the parameter of the service.

Table 155 – Initiate load

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Load Type	M	M(=)		
Load Image Size	C	C(=)		
Intersegment Request Timeout	M	M(=)		
Additional Information	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Actual LR Size			M	M(=)
Max Response Delay			M	M(=)
Max Segment Length			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Load Region object in a specific slot (typically a module).

LR Index
This parameter specifies the index of the Load Region to be downloaded or to be uploaded.

Load Type
This parameter specifies whether the Load Region object has to be downloaded or uploaded.

Allowed values: Push, Pull

Load Image Size
This conditional parameter indicates the number of octets which has to be downloaded to the Load Region object if the parameter Load Type is equal Push. If the parameter Load Type is equal Pull the parameter Load Image Size is not present. This parameter shall have a value of zero if the content of the Load Region object shall be cleared.

Attribute Type: Unsigned32

Allowed values: 0 to 2^{32} - 1

Intersegment Request Timeout
This parameter specifies the timeout period on the Server side between Pull Segment or Push Segment service requests, in units of 100. The default values are shown in Table 156.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Table 156 – Default values for the parameter Intersegment Request Timeout

Baudrate (kbit/s)	≤187,5	500	1 500	3 000	6 000	12 000
Intersegment Request Timeout (sec)	40	20	10	10	10	10

Additional Information
This optional parameter specifies additional information for the subsequent load procedure, e.g. the desired data format.

Result(+)
This selection type parameter indicates that the service request succeeded.

Actual LR Size
This parameter contains the value

– of the attribute Actual LR Size of the Load Region object if the parameter Load Type has had the value Pull in the previous request primitive,

– of the parameter Load Image Size if the parameter Load Type has had the value Push in the previous request primitive.

Attribute Type: Unsigned32

Allowed values: 0 to 2^{32} - 1

Max Response Delay
This parameter contains the value of the attribute Max Response Delay of the Load Region object. The Client AP shall monitor the Initiate Load, Push Segment, Pull Segment or Terminate Load service request and response sequences with this time parameter.

Max Segment Length
This parameter contains the value of the attribute Max Segment Length of the Load Region object.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: state conflict, load image size error, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type.

6.2.7.3.2 Push segment

This optional service is used to transfer LR Data to the Load Region object after the download process has been initiated. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 157 shows the parameter of the service.

Table 157 – Push segment

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Segment Length	M	M(=)		
Segment Number	M	M(=)		
More Follows	M	M(=)		
Data	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Load Region object in a specific slot (typically a module).

LR Index
This parameter specifies the index of the Load Region object to be downloaded.

Segment Length
This parameter indicates the number of octets of the Load Region object which has to be downloaded.

Attribute Type: Unsigned8

Allowed values: 1 to 234

Segment Number
This parameter indicates the number of the segment which is being transferred. Segment numbers are in ascending order without gaps beginning with the value one.

Attribute Type: Unsigned32

Allowed values: 1 to 2^{32} - 1

More Follows
This parameter indicates whether or not any additional segments remain to be transmitted.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Data
The parameter Data contains the segment of the Load Image which has to be downloaded and consists of the number of octets indicated in the parameter Segment Length of the request. This parameter has to be composed of the data type Octet String.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: state conflict, length error, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type, feature not supported.

6.2.7.3.3 Pull segment

This optional service is used to transfer LR Data from the Load Region object after the upload process has been initiated. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 158 shows the parameter of the service.

Table 158 – Pull segment

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Segment Length	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Segment Length			M	M(=)
Segment Number			M	M(=)
More Follows			M	M(=)
Data			C	C(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Load Region object in a specific slot (typically a module).

LR Index
This parameter specifies the index of the Load Region to be uploaded.

Segment Length
This parameter indicates the number of octets of the Load Region object which shall be uploaded with this request.

Attribute Type: Unsigned8

Allowed values: 1 to 234

Result(+)
This selection type parameter indicates that the service request succeeded.

Segment Length
The value of this parameter is equal to the equivalent parameter of the request if the remaining octets to be uploaded are greater or equal to the requested segment length. Otherwise this parameter contains the number of remaining octets.

Attribute Type: Unsigned8

Allowed values: 0 to 234

Segment Number
This parameter indicates the number of the segment which is being transferred. Segment numbers are in ascending order without gaps beginning with the value one.

Attribute Type: Unsigned32

Allowed values: 1 to 2^{32} - 1

More Follows
This Boolean parameter indicates whether or not any additional segments remain to be transmitted.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Data
This conditional parameter is only present if the value of the parameter Segment Length of the response is different from zero and contains the segment of the Load Image which has to be uploaded and consists of the number of octets indicated in the parameter Segment Length of the response. This parameter has to be composed of the data type Octet String.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed value: state conflict, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type, feature not supported.

6.2.7.3.4 Terminate load

This service is used to terminate the upload or download process. It shall be used upon successful completion, or to abort an upload or download in progress.

If the up- or download process is being aborted with the Terminate Load service, then

– in case of a download the contents of the Load Region object shall be deleted (the attribute Actual LR Size shall be set to zero and the attribute Load Region State shall be set to DOWNLOADABLE);
– in case of an upload the value of the attribute Uploadcount of the Load Region object shall be decremented.

This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 159 shows the parameter of the service.

Table 159 – Terminate load

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Load Region object in a specific slot (typically a module).

LR Index
This parameter specifies the index of the Load Region object whose upload or download process shall be terminated.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed value: state conflict, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type.

6.2.7.4 Behavior of the load region object

6.2.7.4.1 Load region state machine description

6.2.7.4.1.1 Overview

The behavior of a Load Region object is defined by the Load Region state machine. Each Load Region object is represented by an own instance of this state machine.

The Load Region state machine represents a part of the User functionality and is included to ensure interoperability. It is assumed that other parts of the User functionality perform the following checks:

– validation of the existence and addressing of the Load Region object,
– validation of the access rights,
– validation of temporary system restrictions for the use of the Load Region object

and generate appropriate Error Codes if necessary.

Two types of Load Region state machines are defined. One represents the behavior of Load Region objects whose contents are stored in erasable memory and are therefore capable to be downloaded. The other one represents the behavior of Load Region objects whose contents are stored in non-erasable memory and are therefore not capable to be downloaded.

The Load Region object may be used by Function Invocation objects. Therefore the Load Region state machines provide an interface to the Function Invocation state machines for co-ordination purposes.

The main functionalities of the Load Region state machine are:

– monitoring of the valid sequence of services executed on the Load Region object,
– to ensure that a download is only executed if no upload is in progress,
– to ensure that an upload or download is only executed if the Load Region object is not used by a Function Invocation object,
– to ensure that an upload is only executed if no download is in progress,
– supervision of the Client activity.

6.2.7.4.1.2 Primitive definitions

6.2.7.4.1.2.1 Primitives exchanged between the user and the load region state machine

Table 160 and Table 161 show the primitives exchanged between the User and the Load Region state machine.

Table 160 – Primitives issued by the user to the Load Region state machine

Primitive Name	Source	Associated parameters	Functions
Abort.ind	User	AREP Locally Generated Subnet Instance Reason Code Additional Detail	—
DP-slave Stopped.ind	User	AREP	—
Initiate Load.ind	User	AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request, Timeout	—
Pull Segment.ind	User	AREP, Slot Number, LR Index, Segment Length	—
Push Segment.ind	User	AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data	—
Terminate Load.ind	User	AREP, Slot Number, LR Index	—

Table 161 – Primitives issued by the Load Region state machine to the user

Primitive Name	Source	Associated parameters	Functions
Initiate Load.rsp(+)	Load Region	AREP, Actual LR Size, Max Response Delay, Max Segment Length	—
Initiate Load.rsp(-)	Load Region	AREP, Error Code	—
Pull Segment.rsp(+)	Load Region	AREP, Segment Length, Segment Number, More Follows, Data	—
Pull Segment.rsp(-)	Load Region	AREP, Error Code	—
Push Segment.rsp(+)	Load Region	AREP	—
Push Segment.rsp(-)	Load Region	AREP, Error Code	—
Terminate Load.rsp(+)	Load Region	AREP	—
Terminate Load.rsp(-)	Load Region	AREP, Error Code	—

The parameters used with the primitives are described in the Service Specification of the Load Region object (see 6.2.7.3).

6.2.7.4.1.2.2 Primitives exchanged between the function invocation and the load region state machine

Table 162 and Table 163 show the primitives exchanged between the Function Invocation and the Load Region state machines.

Table 162 – Primitives issued by the Function Invocation to the Load Region state machine

Primitive Name	Source	Associated parameters	Functions
DeleteInUse.ind	Function Invocation	FI Identifier, LR Identifier	The Function Invocation indicates with this primitive that the Load Region Object is no longer to be used.
SetInUse.ind	Function Invocation	FI Identifier, LR Identifier	The Function Invocation indicates with this primitive that it wants to make use of the Load Region Object.

Table 163 – Primitives issued by the Load Region to the Function Invocation state machine

Primitive Name	Source	Associated parameters	Functions
DeleteInUse.rsp (+)	Load Region	FI Identifier, LR Identifier	The Load Region responds with this primitive to a previous received DeleteInUse.ind.
SetInUse.rsp (+/-)	Load Region	FI Identifier, LR Identifier	The Load Region responds with this primitive to a previous received SetInUse.ind.

6.2.7.4.1.3 State definitions

Table 164 shows the definitions of the states of the Load Region state machine.

Table 164 – Load Region state definitions

State	Description
DOWNLOADABLE	This state indicates that the Load Region object is empty, but is capable of being downloaded.
DOWNLOADING	This state indicates that the download sequence has been initiated.
DOWNLOAD SUCCESS	This transient state indicates that the download sequence has succeeded, but has not yet been terminated.
LOADED	This state indicates that the download sequence has terminated successfully.
IN-USE	This state indicates that the Load Region object is loaded and is currently being used by a Function Invocation object.

6.2.7.4.1.4 Local variable definitions

AREP List

The AREP List consists of a one dimensional list containing AREP(s) which represent application relationships who currently execute a download or upload sequence.

Segment Number List

The Segment Number List consists of a two dimensional list containing AREP(s), also contained in the AREP List, and the actual Segment Number of the sequence executed on this relationship.

Actual Length List

The Actual Length List consists of a two dimensional list containing AREP(s), also contained in the AREP List, and the Actual Length of the amount of data already exchanged in the sequence executed on this relationship.

FI Identifier

This local variable contains the Identifier (API, Slot Number, FI Index) of the Function Invocation object which currently uses this Load Region object.

6.2.7.4.1.5 Timer definitions

Intersegment Request Timer

With the Intersegment Request Timer the Load Region state machine supervises the Client activity during an upload or download sequence.

6.2.7.4.1.6 Function definitions

The following functions contained in Table 165 are used by both Load Region state machines which describe the behavior of a Load Region object stored in erasable memory or in non-erasable memory.

Table 165 – Load Region function table

Function Name	Description
IsServiceAccepted(Load Type, Load Image Size, Additional Information)	This function checks by calling subfunctions if the service is acceptable and match local object conditions. A) If the results of the called subfunctions are as follows: IsLRSizeOK(Load Image Size) = TRUE IsPushAndUpload() = FALSE IsAddInfoOK(Additional Information) = TRUE then it returns TRUE. B) Otherwise it returns FALSE.
IsPushAndUpload()	This function checks whether an Upload sequence is in progress. A) If the attribute Uploadcount is greater than zero then it returns TRUE. B) Otherwise it returns FALSE.
IsLRSizeOK(Load Image Size)	This function checks whether the Load Image Size is matching the attribute LR Size. A) If the Load Image Size is less or equal LR Size then it returns TRUE. B) Otherwise it returns FALSE.
IsAddInfoOK(Additional Information)	This function checks whether the Additional Information is matching the attribute Load Region Additional Information. A) If the Additional Information is compatible to the attribute Load Region Additional Information then it returns TRUE. B) Otherwise it returns FALSE.
StoreSegmentNumber(AREP, Segment Number)	This function increments the Segment Number with one and stores it AREP related.
GetSegmentNumber(AREP)	This function returns the stored AREP related Segment Number.
CheckSegmentNumber(AREP, Segment Number)	This function compares the Segment Number with the stored AREP related Segment Number. A) If the Segment Number is equal to the stored AREP related Segment Number then it returns TRUE. B) Otherwise it returns FALSE.
StoreActualLength(AREP, Actual Length)	This function stores the Actual Length AREP related.
GetActualLength(AREP)	This function returns the stored AREP related Actual Length.

Function Name	Description
GetActualSegLength(AREP, Segment Length)	This function reads the stored AREP related Actual Length by calling the function GetActualLength and calculates the actual Segment Length according to the following equation: A) If the Actual LR Size-Actual Length is greater or equal to Segment Length and less or equal to Max Segment Length then it returns Segment Length. B) If the Actual LR Size-Actual Length is greater or equal to Segment Length and greater than Max Segment Length then it returns Max Segment Length. C) If the Actual LR Size-Actual Length is less than Segment Length and less or equal to Max Segment Length then it returns Actual LR Size-Actual Length. D) If the Actual LR Size-Actual Length is less than Segment Length and greater Max Segment Length then it returns Max Segment Length. Furthermore this function calculates the new Actual Length according to the following equation: Actual Length = Actual Length + returned value. The new Actual Length is stored AREP related by calling the function StoreActualLength.
CheckActualSegLength(AREP, Segment Length)	This function reads the stored AREP related Actual Length by calling the function GetActualLength and checks the Segment Length according to the following conditions: A) If Segment Length is greater than Max Segment Length then it returns FALSE. B) If Segment Length is greater than Actual LR Size-Actual Length then it returns FALSE. C) Otherwise it returns TRUE.
IsLastSegment(AREP)	This function reads the stored AREP related Actual Length by calling the function GetActualLength and calculates whether it is the last segment to transfer according to the following conditions: A) If the Actual Length is equal to Actual LR Size then it returns FALSE. B) Otherwise it returns TRUE.
StoreAREP(AREP)	This function stores the AREP in the AREP List.
CheckAREP(AREP)	This function checks whether the AREP is contained in the AREP List according to the following conditions: A) If the AREP is contained then it returns TRUE. B) Otherwise it returns FALSE.
DeleteAREP(AREP)	This function deletes the AREP in the AREP List.
StartIntersegReqTimer(AREP, T1)	This function starts or restarts the AREP related Intersegment Request Timer.
StopIntersegReqTimer(AREP)	This function stops the AREP related Intersegment Request Timer.
CheckMoreFollows(AREP; More Follows)	This function reads the stored AREP related Actual Length by calling the function GetActualLength and checks whether it is the last segment to receive according to the following conditions: A) If the Actual Length is equal to Actual LR Size then it returns TRUE. B) Otherwise it returns FALSE.
ClearContent	This function deletes the content of the Load Region object.
StoreFIIdentifier(FI Identifier)	This function stores the FI Identifier.
CheckFIIdentifier(FI Identifier)	This function checks the FI Identifier according to the following conditions: A) If the FI Identifier is equal to the stored FI Identifier then it returns TRUE. B) Otherwise it returns FALSE.
DeleteFIIdentifier(FI Identifier)	This function deletes the stored FI Identifier.
StoreData(Data)	This function stores the Data into the content of the Load Region object.
GetData()	This function gets the Data of the content of the Load Region object.

6.2.7.4.2 Load region state diagrams

6.2.7.4.2.1 Load region state diagram for erasable memory

Figure 25 shows the state diagram of a Load Region for erasable memory.

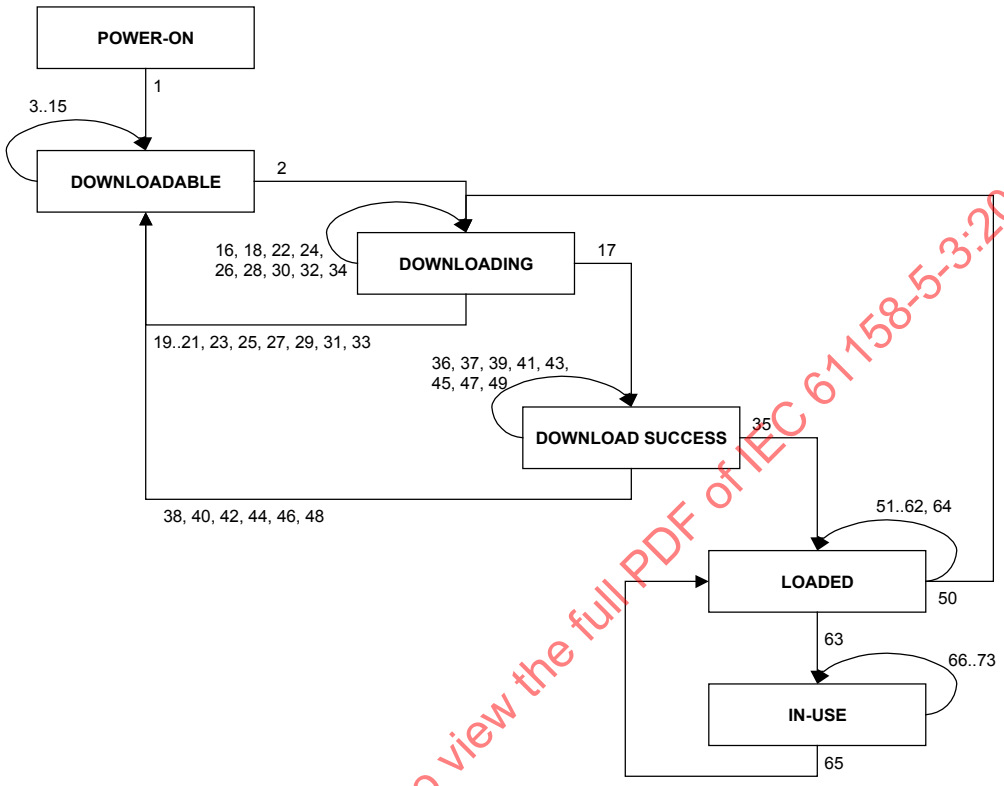


Figure 25 – Load Region state diagram for erasable memory

6.2.7.4.2.2 Load region state diagram for non-erasable memory

Figure 26 shows the state diagram of a Load Region for non-erasable memory.

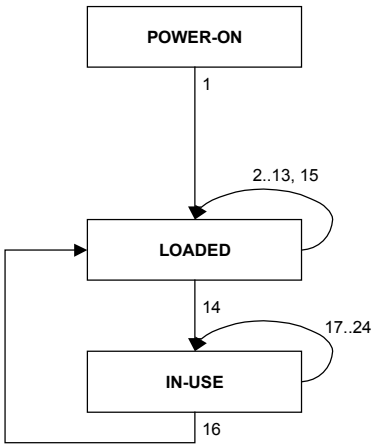


Figure 26 – Load region state diagram for non-erasable memory

6.2.7.4.3 Load region state tables

6.2.7.4.3.1 Load region state table for erasable memory

Table 166 shows the state table of a Load Region for erasable memory.

Table 166 – Load Region state table for erasable memory

#	Current State	Event /Condition =>Action	Next State
1	POWER-ON	=> Uploadcount:=0	DOWNLOADABLE
2	DOWNLOADABLE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) && (IsServiceAccepted(Load Type, Load Image Size, Additional Information)=TRUE) => StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) T1:=Intersegment Request Timeout Actual LR Size:=Load Image Size Load Region State:=DOWNLOADING StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	DOWNLOADING

#	Current State	Event /Condition =>Action	Next State
3	DOWNLOADABLE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) && (IsServiceAccepted(Load Type, Additional Information)=TRUE) => T1:=Intersegment Request Timeout Uploadcount:=Uploadcount+1 StoreAREP(AREP) StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	DOWNLOADABLE

4	DOWNLOADABLE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information)
		/(Load Type=Push) &&
		(IsServiceAccepted(Load Type, Load Image Size, Additional Information)=FALSE)
		=>
		if (IsPushAndUpload()=TRUE)
		Error Code:=state conflict
		elseif(IsLRSizeOK(Load Image Size)=FALSE)
		Error Code:=load image size error
		Endif
		Initiate Load.rsp(-) (AREP, Error Code)
			DOWNLOADABLE

5	DOWNLOADABLE	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
6	DOWNLOADABLE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Segment Length:=0 Segment Number:=1 More Follows:=FALSE StartIntersegReqTimer(AREP, T1) Pull Segment.rsp(+) (AREP, Segment Length, Segment Number, More Follows)	DOWNLOADABLE
7	DOWNLOADABLE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
8	DOWNLOADABLE	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE

#	Current State	Event /Condition =>Action	Next State
9	DOWNLOADABLE	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	DOWNLOADABLE
10	DOWNLOADABLE	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) Uploadcount:=Uploadcount-1 IntersegReqTimerExpired.req(AREP)	DOWNLOADABLE
11	DOWNLOADABLE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	DOWNLOADABLE
12	DOWNLOADABLE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	DOWNLOADABLE
13	DOWNLOADABLE	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	DOWNLOADABLE
14	DOWNLOADABLE	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	DOWNLOADABLE
15	DOWNLOADABLE	SetInUse.ind (FI Identifier, LR Identifier) => SetInUse.rsp(-) (FI Identifier, LR Identifier)	DOWNLOADABLE

#	Current State	Event /Condition =>Action	Next State
16	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=TRUE) && (CheckMoreFollows(AREP; More Follows)=TRUE)&& (More Follows=TRUE) => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) StartIntersegReqTimer(AREP, T1) GetActualSegLength(AREP, Segment Length) StoreData(Data) Push Segment.rsp(+) (AREP)	DOWNLOADING
17	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=TRUE) && (CheckMoreFollows(AREP; More Follows)=TRUE)&& (More Follows=FALSE) => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) StartIntersegReqTimer(AREP, T1) GetActualSegLength(AREP, Segment Length) StoreData(Data) Push Segment.rsp(+) (AREP)	DOWNLOAD SUCCESS
18	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /CheckAREP(AREP)=FALSE => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADING
19	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=FALSE) => Error Code:=invalid parameter DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE

#	Current State	Event /Condition =>Action	Next State
20	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=FALSE) => Error Code:=length error DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
21	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=TRUE) && (CheckMoreFollows(AREP; More Follows)=FALSE) => Error Code:=invalid parameter DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
22	DOWNLOADING	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADING
23	DOWNLOADING	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=TRUE) && (Load Type=Push) => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE

#	Current State	Event /Condition =>Action	Next State
24	DOWNLOADING	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=FALSE) && (Load Type=Push) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADING
25	DOWNLOADING	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
26	DOWNLOADING	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADING
27	DOWNLOADING	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE
28	DOWNLOADING	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOADING
29	DOWNLOADING	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE

#	Current State	Event /Condition =>Action	Next State
30	DOWNLOADING	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	DOWNLOADING
31	DOWNLOADING	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE
32	DOWNLOADING	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	DOWNLOADING
33	DOWNLOADING	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) ClearContent Actual LR Size:=0 IntersegReqTimerExpired.req(AREP)	DOWNLOADABLE
34	DOWNLOADING	SetInUse.ind (FI Identifier, LR Identifier) => SetInUse.rsp(-) (FI Identifier, LR Identifier)	DOWNLOADING
35	DOWNLOAD SUCCESS	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	LOADED
36	DOWNLOAD SUCCESS	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
37	DOWNLOAD SUCCESS	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS

#	Current State	Event /Condition =>Action	Next State
38	DOWNLOAD SUCCESS	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=TRUE) && (Load Type=Push) => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE
39	DOWNLOAD SUCCESS	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=FALSE) && (Load Type=Push) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
40	DOWNLOAD SUCCESS	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
41	DOWNLOAD SUCCESS	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
42	DOWNLOAD SUCCESS	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE

#	Current State	Event /Condition =>Action	Next State
43	DOWNLOAD SUCCESS	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /CheckAREP(AREP)=FALSE => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
44	DOWNLOAD SUCCESS	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE
45	DOWNLOAD SUCCESS	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	DOWNLOAD SUCCESS
46	DOWNLOAD SUCCESS	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE
47	DOWNLOAD SUCCESS	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	DOWNLOAD SUCCESS
48	DOWNLOAD SUCCESS	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) ClearContent Actual LR Size:=0 IntersegReqTimerExpired.req(AREP)	DOWNLOADABLE
49	DOWNLOAD SUCCESS	SetInUse.ind (FI Identifier, LR Identifier) => SetInUse.rsp(-) (FI Identifier, LR Identifier)	DOWNLOAD SUCCESS

#	Current State	Event /Condition =>Action	Next State
50	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) && (IsServiceAccepted(Load Type, Load Image Size, Additional Information)=TRUE) => StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) T1:=Intersegment Request Timeout Actual LR Size:=Load Image Size Load Region State:=DOWNLOADING StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	DOWNLOADING
51	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) && (IsServiceAccepted(Load Type, Additional Information)=TRUE) => T1:=Intersegment Request Timeout Uploadcount:=Uploadcount+1 StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	LOADED

52	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information)
		/(Load Type=Push) &&
		(IsServiceAccepted(Load Type, Load Image Size, Additional Information)=FALSE)
		=>
		if (IsPushAndUpload()=TRUE)
		Error Code:=state conflict
		Elseif(IsLRSizeOK(Load Image Size)=FALSE)
		Error Code:=load image size error
		Endif
		Initiate Load.rsp(-) (AREP, Error Code)
			LOADED

#	Current State	Event /Condition =>Action	Next State
53	LOADED	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	LOADED
54	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) Segment Length:=GetActualSegLength(AREP, Segment Length) More Follows:=IsLastSegment(AREP) StartIntersegReqTimer(AREP, T1) Data:=GetData() Pull Segment.rsp(+) (AREP, Segment Length, Segment Number, More Follows, Data)	LOADED
55	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	LOADED
56	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	LOADED
57	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	LOADED
58	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED
59	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	LOADED

#	Current State	Event /Condition =>Action	Next State
60	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED
61	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	LOADED
62	LOADED	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) Uploadcount:=Uploadcount-1 IntersegReqTimerExpired.req(AREP)	LOADED
63	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount=0 => StoreFIIdentifier(FI Identifier) SetInUse.rsp(+) (FI Identifier, LR Identifier)	IN-USE
64	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount>0 => SetInUse.rsp(-) (FI Identifier, LR Identifier)	LOADED
65	IN-USE	DeleteInUse.ind (FI Identifier, LR Identifier) => DeleteFIIdentifier(FI Identifier) DeleteInUse.rsp(+)(FI Identifier, LR Identifier)	LOADED
66	IN-USE	SetInUse.ind (FI Identifier, LR Identifier) /CheckFIIdentifier(FI Identifier)=FALSE => SetInUse.rsp(-) (FI Identifier, LR Identifier)	IN-USE
67	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	IN-USE
68	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	IN-USE

#	Current State	Event /Condition =>Action	Next State
69	IN-USE	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	IN-USE
70	IN-USE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	IN-USE
71	IN-USE	Terminate Load.ind (AREP, Slot Number, LR Index) => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	IN-USE
72	IN-USE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) =>	IN-USE
73	IN-USE	DP-slave Stopped.ind (AREP) =>	IN-USE

6.2.7.4.4 Load region state table for non-erasable memory

Table 167 shows the state table of a Load Region for non-erasable memory.

Table 167 – Load Region state table for non-erasable memory

#	Current State	Event /Condition =>Action	Next State
1	POWER-ON	=> Uploadcount:=0	LOADED
2	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) => Error Code:= access denied Initiate Load.rsp(-) (AREP, Error Code)	LOADED

#	Current State	Event /Condition =>Action	Next State
3	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) && (IsServiceAccepted(Load Type, Additional Information)=TRUE) => T1:=Intersegment Request Timeout Uploadcount:=Uploadcount+1 StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	LOADED
4	LOADED	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	LOADED
5	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) Segment Length:=GetActualSegLength(AREP, Segment Length) More Follows:=IsLastSegment(AREP) StartIntersegReqTimer(AREP, T1) Data:=GetData() Pull Segment.rsp(+) (AREP, Segment Length, Segment Number, More Follows, Data)	LOADED
6	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	LOADED
7	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	LOADED

#	Current State	Event /Condition =>Action	Next State
8	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	LOADED
9	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED
10	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	LOADED
11	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED
12	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	LOADED
13	LOADED	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) Uploadcount:=Uploadcount-1 IntersegReqTimerExpired.req(AREP)	LOADED
14	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount=0 => StoreFIIdentifier(FI Identifier) SetInUse.rsp(+) (FI Identifier, LR Identifier)	IN-USE
15	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount>0 => SetInUse.rsp(-) (FI Identifier, LR Identifier)	LOADED
16	IN-USE	DeleteInUse.ind (FI Identifier, LR Identifier) => DeleteFIIdentifier(FI Identifier) DeleteInUse.rsp(+)(FI Identifier, LR Identifier)	LOADED

#	Current State	Event /Condition =>Action	Next State
17	IN-USE	SetInUse.ind (FI Identifier, LR Identifier) /CheckFIIdentifier(FI Identifier)=FALSE => SetInUse.rsp(-) (FI Identifier, LR Identifier)	IN-USE
18	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) => Error Code:= access denied Initiate Load.rsp(-) (AREP, Error Code)	IN-USE
19	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	IN-USE
20	IN-USE	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	IN-USE
21	IN-USE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	IN-USE
22	IN-USE	Terminate Load.ind (AREP, Slot Number, LR Index) => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	IN-USE
23	IN-USE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) =>	IN-USE
24	IN-USE	DP-slave Stopped.ind (AREP) =>	IN-USE

6.2.8 Function invocation ASE

6.2.8.1 Overview

The Function Invocation ASE defines two objects, the stateless Action object and the state-oriented Function Invocation object.

Stateless function invocations, referred to as actions, run to completion when invoked and cannot be interrupted. In addition, some atomic actions return a value in response to being invoked, while others do not. Those that do may be used to model software *functions*, and those that do not may be used to model software *procedures*.

State-oriented function invocations, on the other hand, may be controlled during their execution. Services are defined to stop, resume, or reset them once they have been invoked.

They do not return a value in response to being started. If it is necessary to stop a function invocation once it has been started, then it shall be defined as state-oriented.

State-oriented function invocations represent the network view of a specific invocation of a User function. If the User function can be invoked more than once simultaneously, then separate Function Invocation objects are needed to represent each invocation.

State-oriented function invocations may be used to model software processes or user operations that may be started and controlled.

NOTE An example of a state-oriented function invocation that may be used to model a user operation is the "playback" operation of a video recorder. Once it has been started, the playback operation may be stopped (paused) and later resumed.

To support the concept of software process modeling, the definition of functional invocations includes optional attributes that can be used to relate them to Load Region objects.

The Function Invocation objects shall clearly be separated from the Process Data objects and the Load Region objects. No overlapping is allowed.

6.2.8.2 Function invocation model class specifications

6.2.8.2.1 Function invocation class definition

6.2.8.2.1.1 Function invocation formal model

A Function Invocation object is described by the following template:

DP ASE:			**Function Invocation ASE**
CLASS:			Function Invocation
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Identifier
3.	(m)	Attribute:	Access Rights MS1
4.	(m)	Attribute:	Access Rights MS2
4.1.	(m)	Attribute:	List of SCL
4.1.1.	(m)	Attribute:	SCL
7.	(m)	Attribute:	Function Invocation State
8.	(o)	Attribute:	Load Region Object Identifier
8.1	(c)	Attribute:	Load Region Object In Use
9.	(o)	Attribute:	Execution Argument Data Description
10.	(o)	Attribute:	Function Invocation Name
11.	(o)	Attribute:	Local Detail
SERVICES:			
1.	(m)	OpsService:	Start
2.	(o)	OpsService:	Stop
3.	(o)	OpsService:	Resume
4.	(m)	OpsService:	Reset
5.	(m)	OpsService:	Get FI State

6.2.8.2.1.2 Attributes

Identifier
This key attribute is a triplet composed of an API, Slot Number, and FI Index to define to which Application Process Identifier, module and to which index within the specified module the Function Invocation object belongs. This Identifier shall not be used by another Action object, Function Invocation object, Load Region object or Process Data object.

Attribute Type: Unsigned8, Unsigned8, Unsigned16

Allowed triplet values: [0 to 254, 0 to 254, 0 to 65 199 (65 200 to 65 535 reserved)]

Access Rights MS1
This attribute defines the type of access rights defined for a MS1 AR for the Function Invocation object.

Attribute Type: Unsigned8

The allowed values are shown in Table 168.

Table 168 – Access rights MS1

Value	Meaning
2	Right to Start, Stop, Resume, Reset
3	No Access

Access Rights MS2
This attribute defines the type of access rights defined for a MS2 AR for the Function Invocation object.

Attribute Type: Unsigned8

The allowed values are shown in Table 169.

Table 169 – Access rights MS2

Value	Meaning
2	Right to Start, Stop, Resume, Reset
3	No Access

List of SCL
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one value for access protection to this Function Invocation object. The rules for accepting or rejecting an access are described in 6.2.1.3.

Attribute Type: Unsigned8

Function Invocation State
This attribute specifies the state of the Function Invocation object.

Attribute Type: Unsigned8

The allowed values are shown in Table 170.

Table 170 – Function Invocation state

Value	Meaning
0	IDLE
1	STARTING
2	RUNNING
3	STOPPING
4	STOPPED
5	RESUMING
6	UNRUNNABLE
7	RESETTING
8	WAIT-SET-IN-USE
9	WAIT-SET-IN-USE-2
10	WAIT-DEL-IN-USE
11	WAIT-DEL-IN-USE-2
12	EXEC-ERR
13	EXEC-TERM

Load Region Object Identifier
This optional attribute is a triplet composed of API, Slot Number and LR Index to define to which Application Process Identifier, module and to which index within the specified module the related Load Region object belongs.

Attribute Type: Unsigned8, Unsigned8, Unsigned16

Allowed triplet values: [0 to 254, 0 to 254, 0 to 65 199 (65 200 to 65 535 reserved)]

Load Region Object In Use
This Boolean attribute indicates whether the related Load Region object is actual used by this Function Invocation object.

Attribute Type: Boolean

The allowed values are shown in Table 171.

Table 171 – Load Region object in use

Value	Meaning
FALSE	Load Region object not in use
TRUE	Load Region object in use

Execution Argument Data Description
This optional attribute contains the Data Description for the Parameter Execution Argument of the Start service. This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Function Invocation Name
This optional attribute defines the name of the Function Invocation object.

Attribute Type: Visible String(32)

Local Detail
This optional attribute contains additional local information about the Function Invocation object.

Attribute Type: Octet String

6.2.8.2.1.3 Invocation of the function invocation object

For the invocation of the Function Invocation object the following rules apply:

– At least one AR shall have access to the Function Invocation object.
– Only one Load Region object shall be related to a Function Invocation object.
– The access rights shall be set accordingly to the allowed service(s).
– The optional services shall be selected according to the execution mode (cyclic, non-cyclic, …) of the function invocation.

6.2.8.2.2 Action class definition

6.2.8.2.2.1 Action formal model

An Action object is described by the following template:

DP ASE:			**Function Invocation ASE**
CLASS:			Action
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Identifier
3.	(m)	Attribute:	Access Rights MS1
4.	(m)	Attribute:	Access Rights MS2
4.1.	(m)	Attribute:	List of SCL
4.1.1.	(m)	Attribute:	SCL
8.	(o)	Attribute:	Load Region Object Identifier
8.1	(c)	Attribute:	Load Region Object In Use
9.	(o)	Attribute:	Execution Argument Data Description
10.	(o)	Attribute:	Result Argument Data Description
11.	(o)	Attribute:	Action Name
12.	(o)	Attribute:	Local Detail
SERVICES:			
1.	(m)	OpsService:	Call

6.2.8.2.2.2 Attributes

Identifier
This key attribute is a triplet composed of API, Slot Number and FI Index to define to which Application Process Identifier, module and to which index within the specified module the Action object belongs. This Identifier shall not be used by another Action object, Function Invocation object, Load Region object or Process Data object.

Attribute Type: Unsigned8, Unsigned8, Unsigned16

Allowed triplet values: [0 to 254, 0 to 254, 0 to 65 199 (65 200 to 65 535 reserved)]

Access Rights MS1
This attribute defines the type of access rights defined for a MS1 AR for the Action object.

Attribute Type: Unsigned8

The allowed values are shown in Table 172.

Table 172 – Access rights MS1

Value	Meaning
2	Right to Call
3	No Access

Access Rights MS2
This attribute defines the type of access rights defined for a MS2 AR for the Action object.

Attribute Type: Unsigned8

The allowed values are shown in Table 173.

Table 173 – Access rights MS2

Value	Meaning
2	Right to Call
3	No Access

List of SCL
This attribute is composed of the following list elements:

SCL
The attribute SCL contains one value for access protection to this Action object. The rules for accepting or rejecting an access are described in 6.2.1.3.

Attribute Type: Unsigned8

Load Region Object Identifier
This optional attribute is a triplet composed of API, Slot Number and LR Index to define to which Application Process Identifier, module and to which index within the specified module the related Load Region object belongs.

Attribute Type: Unsigned8, Unsigned8, Unsigned16

Allowed triplet values: [0 to 254, 0 to 254, 0 to 65 199 (65 200 to 65 535 reserved)]

Load Region Object In Use
This Boolean attribute indicates whether the related Load Region object is actual used by this Action object.

Attribute Type: Boolean

The allowed values are shown in Table 174.

Table 174 – Load Region object in use

Value	Meaning
FALSE	Load Region object not in use
TRUE	Load Region object in use

Execution Argument Data Description
This optional attribute contains the Data Description for the Parameter Execution Argument of the request of the Call service. This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Result Argument Data Description
This optional attribute contains the Data Description for the Parameter Execution Argument of the response of the Call service. This attribute may contain either a Simple Data Description or an Array Data Description or a Record Data Description. For data description the template given in IEC 61158-5-10, 5.1 is used.

Action Name
This optional attribute defines the name of the Action object.

Attribute Type: Visible String(32)

Local Detail
This optional attribute contains additional local information about the Action object.

Attribute Type: Octet String

6.2.8.2.2.3 Invocation of the action object

For the invocation of the Action object the following rules apply:

— At least one AR shall have access to the Action object.
— Only one Load Region object shall be related to an Action object.
— The access rights shall be set accordingly to the allowed service(s).

6.2.8.3 Function invocation service specification

6.2.8.3.1 Start service

This confirmed service is used to request that a Function Invocation be started. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 175 shows the parameters of the service.

Table 175 – Start

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Execution Argument	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Function Invocation object in a specific slot (typically a module).

FI Index
This parameter specifies the index of the Function Invocation to be started.

Execution Argument
This optional parameter contains additional information to perform the start of the Function Invocation. It shall be present if the attribute Execution Argument Data Description is defined for the addressed Function Invocation object.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: state conflict, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type, resource busy.

6.2.8.3.2 Stop service

This confirmed service is used to stop a Function Invocation retaining its context so that it may be resumed. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 176 shows the parameters of the service.

Table 176 – Stop

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Function Invocation object in a specific slot (typically a module).

FI Index
This parameter specifies the index of the Function Invocation to be stopped.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: state conflict, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type, feature not supported.

6.2.8.3.3 Resume service

This confirmed service is used to resume the execution of a Function Invocation that has been stopped. The execution is resumed using the context saved when the Function Invocation was stopped. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 177 shows the parameters of the service.

Table 177 – Resume

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Execution Argument	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Function Invocation object in a specific slot (typically a module).

FI Index
This parameter specifies the index of the Function Invocation to be resumed.

Execution Argument
This optional parameter contains additional information to resume the execution of the Function Invocation. It shall be present if the attribute Execution Argument Data Description is defined for the addressed Function Invocation object.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: state conflict, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type, feature not supported, resource busy.

6.2.8.3.4 Reset service

This confirmed service is used to reset a Function Invocation with its initial context. Its initial context is defined as the context of the Function Invocation set by its initialization procedures. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 178 shows the parameters of the service.

Table 178 – Reset

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Function Invocation object in a specific slot (typically a module).

FI Index
This parameter specifies the index of the Function Invocation to be reset.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: state conflict, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type.

6.2.8.3.5 Get FI state service

This confirmed service is used to read the actual state of a Function Invocation object. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 179 shows the parameters of the service.

Table 179 – Get FI state

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
FI State			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Function Invocation object in a specific slot (typically a module).

FI Index
This parameter specifies the index of the Function Invocation to be stopped.

Result(+)
This selection type parameter indicates that the service request succeeded.

FI State
This parameter contains the actual state of the Function Invocation object.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type.

6.2.8.3.6 Call service

This confirmed service is used to invoke the execution of an Action object. This service shall only be used in conjunction with the MS1 or MS2 AR.

Table 180 shows the parameters of the service.

Table 180 – Call

Parameter name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Entity Number	U	U(=)		
FI Index	M	M(=)		
Execution Argument	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result Argument			U	U(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Slot Number
The parameter Slot Number is used in the destination device for addressing the desired Action object in a specific slot (typically a module).

Entity Number
The parameter Entity Number is profile specific.

FI Index
This parameter specifies the index of the Action object to be invoked. The available index range is structured as follows:

– Indices reserved for manufacturer specific use (0 - 32 767).
– Indices reserved for PROFIBUS system services (32 768 - 64 999).
– Indices reserved for Identification and Maintenance (I&M) services (65 000 - 65 199).

Execution Argument
This optional parameter contains additional information to perform the invocation of the Action object. It shall be present if the attribute Execution Argument Data Description is defined for the addressed Action object.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result Argument
This optional parameter contains the result of the invocation of the Action object. It shall be present if the attribute Result Argument Data Description is defined for the addressed Action object.

Result(-)
This selection type parameter indicates that the service request failed.

Error Code
This parameter indicates the reason for the failure.

Allowed values: type conflict, access denied, invalid index, invalid slot, module failure, invalid parameter, invalid type.

6.2.8.4 Behavior of the function invocation object

6.2.8.4.1 Function invocation state machine description

6.2.8.4.1.1 General

The behavior of a Function Invocation object is defined by the Function Invocation state machine. Each Function Invocation object is represented by an own instance of this state machine.

The Function Invocation state machine represents a part of the User functionality and is included to ensure interoperability. It is assumed that other parts of the User functionality perform the following checks:

– validation of existence and addressing of the Function Invocation object,
– validation of the access rights,
– validation of temporary system restrictions for the use of the Function Invocation object,
– validation of the correct class of the object

and generate appropriate Error Codes if necessary.

The Function Invocation object may make use of a Load Region object. Therefore the Function Invocation state machine provides an interface to the Load Region state machines for co-ordination purposes.

The main functionalities of the Function Invocation state machine are:

– checking of the validity of the services versus the state of the Function Invocation object,
– to ensure that no parallel services are executed on the Function Invocation object,
– to ensure that a Function Invocation object is only used if the related Load Region object is available,
– supervision of the Client activity.

6.2.8.4.1.2 Primitive definitions

6.2.8.4.1.2.1 Primitives exchanged between the user and the function invocation state machine

Table 181 and Table 182 show the primitive exchanged between the User and the Function Invocation state machine.

Table 181 – Primitives issued by the user to the Function Invocation state machine

Primitive Name	Source	Associated parameters	Functions
Abort.ind	User	AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail	—
Call.ind	User	AREP, Slot Number, FI Index, Execution Argument	—
DP-slave Stopped.ind	User	AREP	—
Reset.ind	User	AREP, Slot Number, FI Index	—
Resume.ind	User	AREP, Slot Number, FI Index, Execution Argument	—
Start.ind	User	AREP, Slot Number, FI Index, Execution Argument	—
Stop.ind	User	AREP, Slot Number, FI Index	—
Get FI State.ind	User	AREP, Slot Number, FI Index	—

Table 182 – Primitives issued by the Function Invocation state machine to the user

Primitive Name	Source	Associated parameters	Functions
Reset.rsp(+)	Function Invocation	AREP	—
Reset.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Resume.rsp(+)	Function Invocation	AREP	—
Resume.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Start.rsp(+)	Function Invocation	AREP	—
Start.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Stop.rsp(+)	Function Invocation	AREP	—
Stop.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Get FI State.rsp(+)	Function Invocation	AREP, FI State	—

The parameters used with the primitives are described in the Service Specification of the Function Invocation object (see 6.2.8.3).

6.2.8.4.1.2.2 Primitives exchanged between the LR and the function invocation state machine

Table 183 and Table 184 show the primitive exchanged between the Load Region state machine and the Function Invocation state machine.

Table 183 – Primitives issued by the Load Region to the Function Invocation state machine

Primitive Name	Source	Associated parameters	Functions
DeleteInUse.cnf(+)	Load Region	FI Identifier, LR Identifier	The Load Region confirms with this primitive positively a previous issued DeleteInUse.req.
SetInUse.cnf(+)	Load Region	FI Identifier, LR Identifier	The Load Region confirms with this primitive positively a previous issued SetInUse.req.
SetInUse.cnf(-)	Load Region	FI Identifier, LR Identifier	The Load Region confirms with this primitive negatively a previous issued SetInUse.req.

Table 184 – Primitives issued by the Function Invocation to the Load Region state machine

Primitive Name	Source	Associated parameters	Functions
DeleteInUse.req	Function Invocation	FI Identifier, LR Identifier	The Function Invocation requests with this primitive the termination of the use of a Load Region object.
SetInUse.req	Function Invocation	FI Identifier, LR Identifier	The Function Invocation requests with this primitive the use of a Load Region object.

6.2.8.4.1.3 State definitions

Table 185 shows the definitions of the states of the Load Region state machine.

Table 185 – Function Invocation state definitions

State	Description
IDLE	This state indicates that the Function Invocation object is not used and can be started.
STARTING	This transient state indicates that the Function Invocation is executing a start procedure.
RUNNING	This state indicates that the object has completed the start-up procedure and executes its normal function.
STOPPING	This transient state indicates that a previous issued Stop is executed.
STOPPED	This state indicates that the Function Invocation is stopped.
RESUMING	This transient state indicates that a Resume command is executed.
UNRUNNABLE	This state indicates that an error occurred.
RESETTING	This transient state indicates that a Reset Command is executed.
WAIT-SET-IN-USE	This transient state indicates that the Function Invocation is executing a start procedure after breakdown of the related Association Relationship.
WAIT-SET-IN-USE-2	This transient state indicates that the Function Invocation is executing a resume procedure after breakdown of the related Association Relationship.
WAIT-DEL-IN-USE	This transient state indicates that the Function Invocation is executing a stop procedure after breakdown of the related Association Relationship.
WAIT-DEL-IN-USE-2	This transient state indicates that the Function Invocation is executing a reset procedure after breakdown of the related Association Relationship.
EXEC-ERR	This transient state indicates that during the execution of a Function Invocation an error has been detected.
EXEC-TERM	This transient state indicates that the execution of a Function Invocation is terminated at the end of the program in case of non-cyclic execution mode.

6.2.8.4.1.4 Local variable definitions

AREP

This local variable contains the AREP which represents the application relationships who currently executes a Start, Stop, Resume or Reset service.

LR Identifier

This local variable contains the Identifier (API, Slot Number, LR Index) of the Load Region object which currently is used by this Function Invocation object.

6.2.8.4.1.5 Local event definitions

Execution Error
This local event is generated if during the execution of a Function Invocation an error has been detected. The execution is terminated and the Function Invocation is set to the state UNRUNNABLE.

Execution Terminated
This local event is generated if the execution of a Function Invocation is terminated at the end of the program in case of non-cyclic execution modes. The Function Invocation is set to the state IDLE.

6.2.8.4.1.6 Function definitions

The functions contained in Table 186 are used by both Function Invocation state machines which describe the behavior of a Function Invocation object.

Table 186 – Function definitions

Function Name	Description
IsServiceAccepted(Execution Argument)	This function checks if the received Execution Argument is compatible to the corresponding attribute of the Function Invocation object. A) If the received Execution Argument is compatible then it returns TRUE. B) Otherwise it returns FALSE.
CheckLRIdentifier (LR Identifier)	This function checks if the received LR Identifier is equal to the corresponding attribute of the Function Invocation object. A) If the received LR Identifier is equal then it returns TRUE. B) Otherwise it returns FALSE.
StoreAREP(AREP)	This function stores the received AREP in the variable AREP.
CheckAREP(AREP)	This function checks if the received AREP is equal to the AREP stored in the variable AREP. A) If the received AREP is equal then it returns TRUE. B) Otherwise it returns FALSE.
GetAREP ()	This function reads the AREP stored in the variable AREP.
GetFIState()	This function reads the actual state from the attribute Function Invocation State of the Function Invocation object.

6.2.8.4.2 Function Invocation state diagram

Figure 27 shows the state diagram of the Function Invocation state machine.

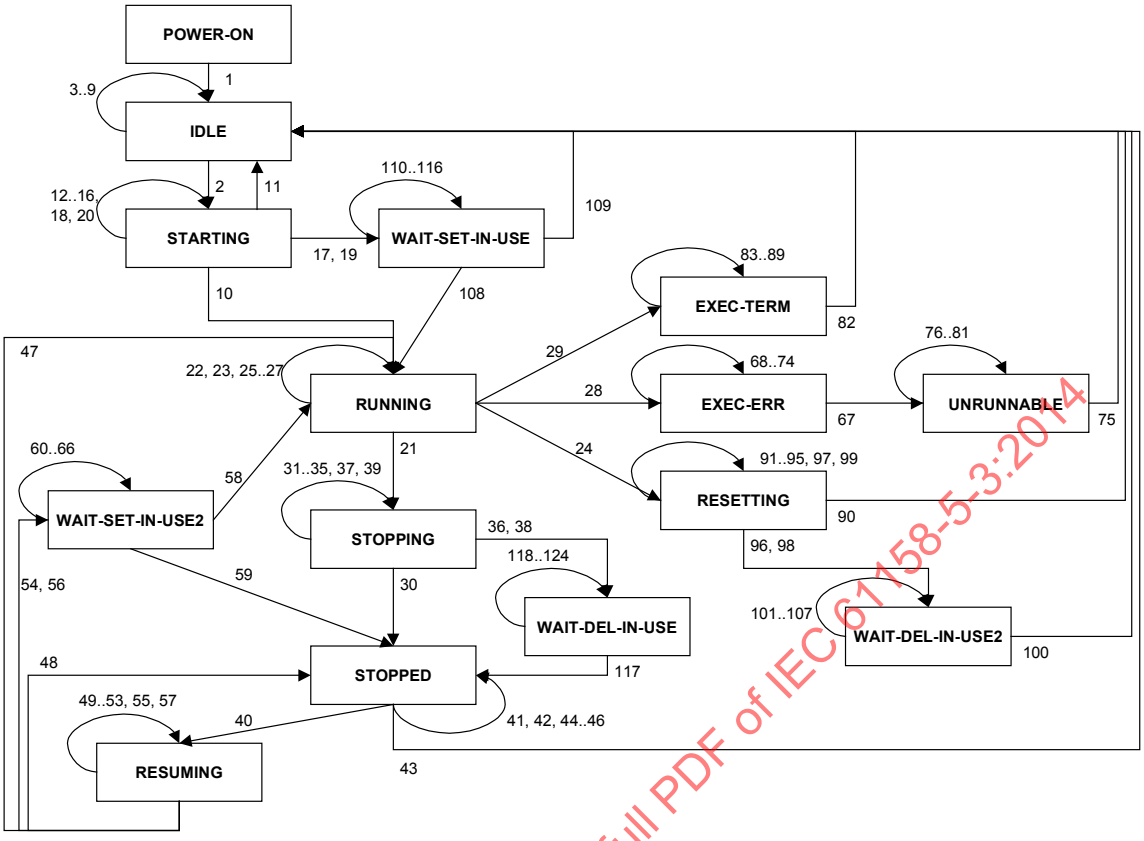


Figure 27 – Function invocation state diagram

6.2.8.4.3 Function Invocation state table

Table 187 shows the states of a Function Invocation state machine.

Table 187 – Function Invocation state table

#	Current State	Event /Condition =>Action	Next State
1	POWER-ON	 =>	IDLE
2	IDLE	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** /(IsServiceAccepted(Execution Argument) = TRUE) => StoreAREP(AREP) SetInUse.req (FI Identifier, LR Identifier)	STARTING
3	IDLE	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** /(IsServiceAccepted(Execution Argument) = FALSE) => Error Code := invalid parameter Start.rsp(-) (AREP, Error Code)	IDLE

#	Current State	Event /Condition =>Action	Next State
4	IDLE	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	IDLE
5	IDLE	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	IDLE
6	IDLE	**Reset.ind (AREP, Slot Number, FI Index)** => Reset.rsp (+) (AREP)	IDLE
7	IDLE	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	IDLE
8	IDLE	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	IDLE
9	IDLE	**DP-slave Stopped.ind (AREP)** =>	IDLE
10	STARTING	**SetInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) => AREP := GetAREP () Start.rsp(+) (AREP)	RUNNING
11	STARTING	**SetInUse.cnf (-) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) => AREP := GetAREP () Error Code := resource busy Start.rsp(-) (AREP, Error Code)	IDLE
12	STARTING	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	STARTING
13	STARTING	**Stop.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	STARTING

#	Current State	Event /Condition =>Action	Next State
14	STARTING	**Resume.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	STARTING
15	STARTING	**Reset.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	STARTING
16	STARTING	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	STARTING
17	STARTING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = TRUE) =>	WAIT-SET-IN-USE
18	STARTING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = FALSE) =>	STARTING
19	STARTING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = TRUE) =>	WAIT-SET-IN-USE
20	STARTING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = FALSE) =>	STARTING
21	RUNNING	**Stop.ind (AREP, Slot Number, FI Index)** => StoreAREP(AREP) DeleteInUse.req (FI Identifier, LR Identifier)	STOPPING
22	RUNNING	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	RUNNING

#	Current State	Event /Condition =>Action	Next State
23	RUNNING	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	RUNNING
24	RUNNING	**Reset.ind (AREP, Slot Number, FI Index)** => DeleteInUse.req (FI Identifier, LR Identifier)	RESETTING
25	RUNNING	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	RUNNING
26	RUNNING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	RUNNING
27	RUNNING	**DP-slave Stopped.ind (AREP)** =>	RUNNING
28	RUNNING	**Execution Error** => DeleteInUse.req (FI Identifier, LR Identifier)	EXEC-ERR
29	RUNNING	**Execution Terminated** => DeleteInUse.req (FI Identifier, LR Identifier)	EXEC-TERM
30	STOPPING	**DeleteInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) => AREP := GetAREP () Stop.rsp(+) (AREP)	STOPPED
31	STOPPING	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	STOPPING
32	STOPPING	**Stop.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	STOPPING

#	Current State	Event /Condition =>Action	Next State
33	STOPPING	**Resume.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	STOPPING
34	STOPPING	**Reset.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	STOPPING
35	STOPPING	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	STOPPING
36	STOPPING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = TRUE) =>	WAIT-DEL-IN-USE
37	STOPPING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = FALSE) =>	STOPPING
38	STOPPING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = TRUE) =>	WAIT-DEL-IN-USE
39	STOPPING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = FALSE) =>	STOPPING
40	STOPPED	**Resume.ind (AREP, Slot Number, FI Index)** => StoreAREP(AREP) SetInUse.req (FI Identifier, LR Identifier)	RESUMING
41	STOPPED	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	STOPPED

#	Current State	Event /Condition =>Action	Next State
42	STOPPED	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	STOPPED
43	STOPPED	**Reset.ind (AREP, Slot Number, FI Index)** => Reset.rsp (+) (AREP)	IDLE
44	STOPPED	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	STOPPED
45	STOPPED	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	STOPPED
46	STOPPED	**DP-slave Stopped.ind (AREP)** =>	STOPPED
47	RESUMING	**SetInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) => AREP := GetAREP () Resume.rsp(+) (AREP)	RUNNING
48	RESUMING	**SetInUse.cnf (-) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) => AREP := GetAREP () Error Code := resource busy Resume.rsp(-) (AREP, Error Code)	STOPPED
49	RESUMING	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	RESUMING
50	RESUMING	**Stop.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	RESUMING

#	Current State	Event /Condition =>Action	Next State
51	RESUMING	**Resume.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	RESUMING
52	RESUMING	**Reset.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	RESUMING
53	RESUMING	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	RESUMING
54	RESUMING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = TRUE) =>	WAIT-SET-IN-USE-2
55	RESUMING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = FALSE) =>	RESUMING
56	RESUMING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = TRUE) =>	WAIT-SET-IN-USE-2
57	RESUMING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = FALSE) =>	RESUMING
58	WAIT-SET-IN-USE-2	**SetInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	RUNNING
59	WAIT-SET-IN-USE-2	**SetInUse.cnf (-) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	STOPPED

#	Current State	Event /Condition =>Action	Next State
60	WAIT-SET-IN-USE-2	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	WAIT-SET-IN-USE-2
61	WAIT-SET-IN-USE-2	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	WAIT-SET-IN-USE-2
62	WAIT-SET-IN-USE-2	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	WAIT-SET-IN-USE-2
63	WAIT-SET-IN-USE-2	**Reset.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	WAIT-SET-IN-USE-2
64	WAIT-SET-IN-USE-2	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	WAIT-SET-IN-USE-2
65	WAIT-SET-IN-USE-2	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	WAIT-SET-IN-USE-2
66	WAIT-SET-IN-USE-2	**DP-slave Stopped.ind (AREP)** =>	WAIT-SET-IN-USE-2
67	EXEC-ERR	**DeleteInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	UNRUNNABLE
68	EXEC-ERR	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	EXEC-ERR
69	EXEC-ERR	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	EXEC-ERR
71	EXEC-ERR	**Reset.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	EXEC-ERR

#	Current State	Event /Condition =>Action	Next State
72	EXEC-ERR	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	EXEC-ERR
73	EXEC-ERR	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	EXEC-ERR
74	EXEC-ERR	**DP-slave Stopped.ind (AREP)** =>	EXEC-ERR
75	UNRUNNABLE	**Reset.ind (AREP, Slot Number, FI Index)** => Reset.rsp (+) (AREP)	IDLE
76	UNRUNNABLE	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	UNRUNNABLE
77	UNRUNNABLE	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	UNRUNNABLE
78	UNRUNNABLE	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	UNRUNNABLE
79	UNRUNNABLE	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	UNRUNNABLE
80	UNRUNNABLE	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	UNRUNNABLE
81	UNRUNNABLE	**DP-slave Stopped.ind (AREP)** =>	UNRUNNABLE
82	EXEC-TERM	**DeleteInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	IDLE

#	Current State	Event /Condition =>Action	Next State
83	EXEC-TERM	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	EXEC-TERM
84	EXEC-TERM	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	EXEC-TERM
85	EXEC-TERM	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	EXEC-TERM
86	EXEC-TERM	**Reset.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	EXEC-TERM
87	EXEC-TERM	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	EXEC-TERM
88	EXEC-TERM	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	EXEC-TERM
89	EXEC-TERM	**DP-slave Stopped.ind (AREP)** =>	EXEC-TERM
90	RESETTING	**DeleteInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) => AREP := GetAREP () Reset.rsp(+) (AREP)	IDLE
91	RESETTING	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	RESETTING
92	RESETTING	**Stop.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	RESETTING

#	Current State	Event /Condition =>Action	Next State
93	RESETTING	**Resume.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	RESETTING
94	RESETTING	**Reset.ind (AREP, Slot Number, FI Index)** /(CheckAREP(AREP) = FALSE) => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	RESETTING
95	RESETTING	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	RESETTING
96	RESETTING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = TRUE) =>	WAIT-DEL-IN-USE-2
97	RESETTING	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** /(CheckAREP(AREP) = FALSE) =>	RESETTING
98	RESETTING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = TRUE) =>	WAIT-DEL-IN-USE-2
99	RESETTING	**DP-slave Stopped.ind (AREP)** /(CheckAREP(AREP) = FALSE) =>	RESETTING
100	WAIT-DEL-IN-USE-2	**DeleteInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	IDLE
101	WAIT-DEL-IN-USE-2	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	WAIT-DEL-IN-USE-2

#	Current State	Event /Condition =>Action	Next State
102	WAIT-DEL-IN-USE-2	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	WAIT-DEL-IN-USE-2
103	WAIT-DEL-IN-USE-2	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	WAIT-DEL-IN-USE-2
104	WAIT-DEL-IN-USE-2	**Reset.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	WAIT-DEL-IN-USE-2
105	WAIT-DEL-IN-USE-2	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	WAIT-DEL-IN-USE-2
106	WAIT-DEL-IN-USE-2	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	WAIT-DEL-IN-USE-2
107	WAIT-DEL-IN-USE-2	**DP-slave Stopped.ind (AREP)** =>	WAIT-DEL-IN-USE-2
108	WAIT-SET-IN-USE	**SetInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	RUNNING
109	WAIT-SET-IN-USE	**SetInUse.cnf (-) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	IDLE
110	WAIT-SET-IN-USE	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	WAIT-SET-IN-USE
111	WAIT-SET-IN-USE	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	WAIT-SET-IN-USE
112	WAIT-SET-IN-USE	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	WAIT-SET-IN-USE

#	Current State	Event /Condition =>Action	Next State
113	WAIT-SET-IN-USE	**Reset.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	WAIT-SET-IN-USE
114	WAIT-SET-IN-USE	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	WAIT-SET-IN-USE
115	WAIT-SET-IN-USE	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	WAIT-SET-IN-USE
116	WAIT-SET-IN-USE	**DP-slave Stopped.ind (AREP)** =>	WAIT-SET-IN-USE
117	WAIT-DEL-IN-USE	**DeleteInUse.cnf (+) (FI Identifier, LR Identifier)** /(CheckLRIdentifier (LR Identifier) = TRUE) =>	STOPPED
118	WAIT-DEL-IN-USE	**Start.ind (AREP, Slot Number, FI Index, Execution Argument)** => Error Code := state conflict Start.rsp(-) (AREP, Error Code)	WAIT-DEL-IN-USE
119	WAIT-DEL-IN-USE	**Stop.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Stop.rsp(-) (AREP, Error Code)	WAIT-DEL-IN-USE
120	WAIT-DEL-IN-USE	**Resume.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Resume.rsp(-) (AREP, Error Code)	WAIT-DEL-IN-USE
121	WAIT-DEL-IN-USE	**Reset.ind (AREP, Slot Number, FI Index)** => Error Code := state conflict Reset.rsp (-) (AREP, Error Code)	WAIT-DEL-IN-USE
122	WAIT-DEL-IN-USE	**Get FI State.ind (AREP, Slot Number, FI Index)** => FI State := GetFIState() Get FI State.rsp (+) (AREP, FI State)	WAIT-DEL-IN-USE
123	WAIT-DEL-IN-USE	**Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail)** =>	WAIT-DEL-IN-USE

#	Current State	Event /Condition =>Action	Next State
124	WAIT-DEL-IN-USE	**DP-slave Stopped.ind (AREP)** =>	WAIT-DEL-IN-USE

6.2.9 Time ASE

6.2.9.1 Overview

The Time ASE specifies the Clock Synchronization mechanism to synchronies the time used within devices of a network. This time could be used for example to add a time value to an alert information and to sequence messages in a time wise order.

A system consists of one or more Time Master and several Time Receiver. Clock synchronization is done by the communication system. Adjustment and management of the clock is task of the application.

Time Master tasks are:
– Sending Time Event and Clock Value within Clock Sync Interval.
– Resolution of Time Master conflicts.
– Report Errors in Clock Synchronisation.

Time Receiver tasks are:
– Monitoring and check of the clock via clock sync interval.
– Control of clock synchronization sequence.
– Interpretation clock value.
– Check status in clock value.
– Adjustment of local time (not part of the Clock Synchronisation specification).

The time object specifies two classes:
– Slave Time Class.
– Link Time Class.

The Slave Time Class comprises the Time Receiver tasks. The Link Time Class comprises Time Master and Time Receiver tasks. But Link Time Class differs from Slave Time Class definition.

The difference is based on a system architecture which provides one or more Time Master in a hierarchical topology and Time Receiver (DP-slaves) which receive a synchronized time value to set a clock within the device, see
Figure 28.

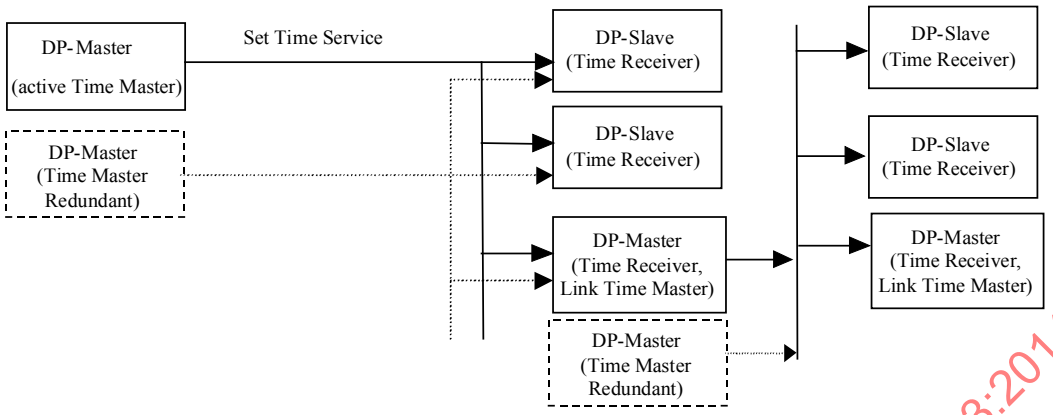


Figure 28 – System architecture

Two services are defined:

– Set Time,

– Sync Interval Violation.

The Set Time service is requested by the DP-master device and the Sync Interval Violation service through DP-slave devices.

Synchronization of time between devices on a fieldbus segment and a Time Master application is provided in parallel with other communication functions. The DP-master (Class 1) operates as the active and if required as the redundant Time Master. If a DP-slave supports the Clock Synchronization mechanism the device attribute DPV1 has to be set to TRUE.

The scheme used is a "backwards time based correction". This results in very few real-time constraints being imposed on a field device. There is no requirement for generating messages at precise instants. Instead, knowledge of when a special Timer event message has been broadcasted is subsequently distributed and used to calculate appropriate clock adjustments.

NOTE The clock synchronization assumes that there are no delays between the invocation of the Set Time service and the begin of the processing of the related service in the data-link layer.

The clock synchronization protocol operates in parallel with the other AL functions. The clock synchronization interval specifies the rate at which the Time Master initiates the clock synchronization sequence. The value of this parameter is a system design issue. It shall be the same for all devices on the network.

The value of the clock synchronization interval is a system design issue. For each station, there is a minimum required clock synchronization interval given by the desired clock accuracy divided by the clock drift rate. The value for the system should be set to the minimum required value for all stations on the network segment. The value should not be set so large that it causes intolerable start-up delays when devices, such as hand-held terminals, are added to a network.

6.2.9.2 Time class specification

6.2.9.2.1 Slave time class specification

6.2.9.2.1.1 Template

DP ASE: **Time ASE**
CLASS: **Slave Time**

CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	Time Data Description
3.	(o)	Attribute:	Local Time Diff
4.	(o)	Attribute:	Time Last Rcvd
5.	(o)	Attribute:	Last Calc Diff
6.	(m)	Attribute:	CS Status
7.	(o)	Attribute:	Summertime

SERVICES:

1.	(m)	OpsService:	Set Time
2.	(m)	OpsService:	Sync Interval Violation

6.2.9.2.1.2 Attributes

Implicit
The attribute Implicit indicates that the Slave Time object is implicitly addressed by the services.

Time Data Description
This attribute contains a Simple Data Description. Only the Data Type Network Time according Clause 5 shall be used.

Local Time Diff
This optional attribute contains the value that needs to be added to the value of the Slave Time object to obtain the local time.

The default value shall be 0 (no local time used).

Attribute Type: Network Time Difference

Time Last Rcvd
This optional attribute contains the parameter Time Value of the last Set Time service.

Attribute Type: Network Time

Time Master Addr
This optional attribute contains the address of the device from whom the previous Set Time service was received.

Attribute Type: Unsigned8

Allowed Value: 0 to 126

Last Calc Diff
This optional attribute contains the difference between the value of the Slave Time object and the parameter Time Value of the Set Time service plus Delay Factor at the receipt of the last Set Time service.

Attribute Type: Network Time Difference

CS Status
This Boolean attribute contains the current status of the synchronization.

Attribute Type: Boolean

The allowed values are shown in Table 188.

Table 188 – CS status

Value	Meaning
FALSE	Synchronized
TRUE	Not Synchronized Time since last reception of a Set Time service is too long (greater than 3 Clock Sync Intervals).

Summertime
This optional Boolean attribute indicates whether the value of the attribute Local Time Diff is based on Summertime or Wintertime.

Attribute Type: Boolean

The allowed values are shown in Table 189.

Table 189 – Summertime

Value	Meaning
FALSE	Wintertime
TRUE	Summertime

6.2.9.2.1.3 Invocation of the slave time object

For the invocation of the Slave Time object the following rules apply:

– Only one Slave Time object shall be invoked in a DP-slave.

6.2.9.2.2 Link time class specification

6.2.9.2.2.1 Template

DP ASE: **Time ASE**
CLASS: **Link Time**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Key Attribute: Implicit
2. (m) Attribute: Time Data Description
3. (o) Attribute: Local Time Diff
4. (o) Attribute: Time Last Rcvd
5. (o) Attribute: Last Calc Diff
6. (m) Attribute: Synchronization Active
7. (o) Attribute: Announcement Hour
8. (o) Attribute: Summertime
9. (o) Attribute: Accuracy

SERVICES:
1. (m) OpsService: Set Time
2. (m) OpsService: Sync Interval Violation

6.2.9.2.2.2 Attributes

Implicit
The attribute Implicit indicates that the Link Time object is implicitly addressed by the services.

Time Data Description
This attribute contains a Simple Data Description. Only the Data Type Network Time according Clause 5 shall be used.

Local Time Diff
This optional attribute contains the value that needs to be added to the value of the Link Time object to obtain the local time. It shall be a multiple of 30 minutes, between 0 and 24 hours.

The default value shall be 0 (no local time used).

Attribute Type: Network Time Difference

Time Last Rcvd
This optional attribute contains the parameter Time Value of the last Set Time service.

Attribute Type: Network Time

Last Calc Diff
This optional attribute contains the difference between the value of the Link Time object and the parameter Time Value of the Set Time service plus Delay Factor at the receipt of the last Set Time service.

Attribute Type: Network Time Difference

Synchronization Active
This Boolean attribute indicates whether the value of the Link Time object is synchronized or not. This attribute is set to 0 if either this station has received a Set Time service with the equivalent Parameter set to 0 or if the time since last reception of a Set Time service is too long (greater than 3 Clock Sync Intervals).

Attribute Type: Boolean

The allowed values are shown in Table 190.

Table 190 – Synchronization active

Value	Meaning
FALSE	Synchronization Failed
TRUE	Synchronization Active

Default value: FALSE

Announcement Hour
This optional Boolean attribute indicates whether a change from Summertime to Wintertime or vice versa shall be performed with the next change of the hour.

Attribute Type: Boolean

The allowed values are shown in Table 191.

Table 191 – Announcement hour

Value	Meaning
FALSE	No change with the next hour change.
TRUE	Change from Summertime to Wintertime or vice versa with the next hour change.

Summertime
This optional Boolean attribute indicates whether the value of the attribute Local Time Diff is based on Summertime or Wintertime.

Attribute Type: Boolean

The allowed values are shown in Table 192.

Table 192 – Summertime

Value	Meaning
FALSE	Wintertime
TRUE	Summertime

Accuracy
This optional attribute indicates the precision of the value of the Link Time object, i.e. which bit positions are relevant. The value will be taken from the Set Time service parameter Accuracy.

Attribute Type: Unsigned8

The allowed values are shown in Table 193.

Table 193 – Accuracy

Value	Meaning
0	< 1 ms
1	< 10 ms
2	< 100 ms
3	< 1 s

Default value: 0

6.2.9.2.2.3 Invocation of the link time object

For the invocation of the Link Time object the following rules apply:

– Only one Link Time object shall be invoked in a Link.

6.2.9.3 Time service specification

6.2.9.3.1 Set time

This confirmed service shall be used to set the value and some attributes of a Slave Time or Link Time object. This service shall be used only in conjunction with the MS3 AR. This AR provides the facility to address a group of or all present devices on a fieldbus segment (multicast/broadcast) and therefore the corresponding Slave Time or Link Time objects will be set simultaneously. Due to this principle no confirmation of reception can be provided at the requesting side.

Table 194 shows the parameters of the Set Time service.

Table 194 – Set time

Parameter name	Req	Ind	Cnf
Argument	M	M(=)	
AREP	M	M	
Time Value	M	M	
Local Time Diff	M	M(=)	
Summertime	M	M(=)	
Accuracy	M	U(=)	
Synchronization Active	M	U(=)	
Announcement Hour	M	U(=)	
Result(+)			S
AREP			M
Result(-)			S
AREP			M
Status			M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

Time Value
This Parameter contains the value for the Link Time or Slave Time object.

Local Time Diff
This attribute contains the value for the attribute Local Time Diff of the Link Time or Slave Time object.

Summertime
This attribute contains the value for the attribute Summertime of the Link Time or Slave Time object.

Accuracy
This attribute contains the value for the attribute Accuracy of the Link Time object and shall be present only in the indication if a Link Time object is present.

Synchronization Active
This attribute contains the value for the attribute Synchronization Active of the Link Time object and shall be present only in the indication if a Link Time object is present.

Announcement Hour
This attribute contains the value of the attribute Announcement Hour of the Link Time object and shall be present only in the indication if a Link Time object is present.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status:
This parameter provides detailed information about the occurred error.

Allowed values: LS, LR, DS, SV, IV

6.2.9.3.2 Sync interval violation

The Sync Interval Violation service signals to the application of the Time Receiver (DP-slave) that no Set Time service has been received during the last 2 Clock Sync Intervals. This service shall be used only in conjunction with the MS3 AR.

Table 195 shows the parameter of the Sync Interval Violation service.

Table 195 – Sync interval violation

Parameter name	Ind
Argument	M
AREP	M

Argument
The argument shall convey the service specific parameters of the service request.

AREP
This parameter is the local identifier for the desired AR.

6.2.10 AR ASE

6.2.10.1 Overview

6.2.10.1.1 General

This ASE specifies the structure of the ARs including the necessary DL parameters for the specific CRs. The AR ASE in each device manages an end point of the AR and all related CRs, and maintains their local context.

An AR is characterized by all of the following basic roles of the AREP used in fieldbus DP:

AR cardinality
– one-to-one

In a one-to-one AR an AP communicates with exactly one remote AP.

– one-to-many

In a one-to-many AR the AP of a DP-master (Class 2) communicates with the APs of a group of DP-masters (Class 1). This comprises the simultaneous communication to the defined group of DP-masters (Class 1). Exactly one AP is addressed per device.

– many-to-many

In a many-to-many AR the APs of a DP-master (Class 1) and optional the APs of one or several DP-master (Class 2) and optional the Default-APs of one or several DP-slave communicate with the Default-APs of a group of DP-slaves. This comprises the sequential as well as the simultaneous communication to the defined group of DP-slaves. Exactly one AP is addressed per device.

Device type relations:
Master-slave (MS) communication
between DP-master (Class 1) and DP-slaves

between DP-master (Class 2) and DP-slaves

Master-master (MM) communication
between DP-master (Class 2) and DP-master (Class 1)

6.2.10.1.2 Communication relationship

6.2.10.1.2.1 General

An AR is mapped to one or more communication relationships (CRs). A CR carries out the exchange of information based on a set of rules.

6.2.10.1.2.2 Communication relationship endpoint

CRs are defined as a set of co-operating Communication Relationship End Points (CREPs). The local context of a CR endpoint is used to control the conveyance of APDUs on the CR. Figure 29 illustrates the Assignment of Communication Relationship to Application Relationship.

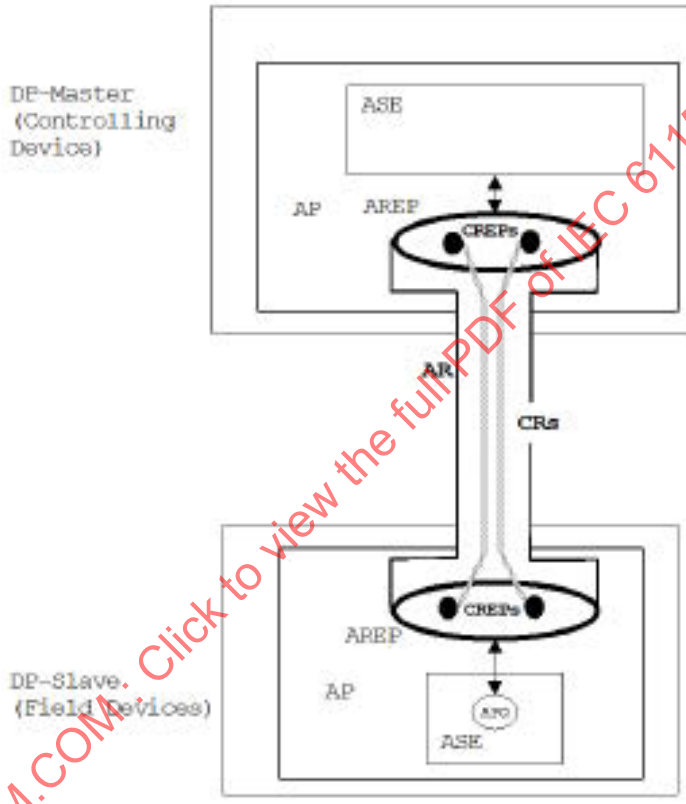


Figure 29 – Assignment of communication relationship to application relationship

6.2.10.1.2.3 Buffer and queues

CREPs may be modeled as a queue or as a buffer. Application data transferred over a queued CREP are delivered in the order received for conveyance. The transfer of application data over a buffered CREP is different. In this case, application data is placed in an application layer buffer for transfer.

When the CREP receives another conveyance request, it replaces the previous contents of the buffer whether or not transmitted. Once application data are written into a buffer for transfer, it is preserved in the buffer until the next application data to be transmitted replaces it. While in the buffer, application data may be read more than once without deleting it from the buffer or changing its contents.

At the receiving end, the operation is similar. The receiving application layer places the received application data into a buffer for access by the different ASEs. When subsequent

application data are received, it overwrites the previous application data in the buffer whether or not it was read by the corresponding ASE. Reading the application data from the buffer is not destructive it does not destroy or change the contents of the buffer, allowing the contents to be read from the buffer one or more times.

6.2.10.1.2.4 Communication relationship endpoint roles

A communication relationship is characterized by all of the following basic roles of the CREP used in fieldbus DP:

CR cardinality

– one-to-one:

  In a one-to-one communication relationship a CREP communicates with exactly one remote CREP.

– one-to-many:

  In a one-to-many relationship one CREP communicates simultaneously with the CREPs of a group of devices. A one-to-many communication relationship does not permit confirmation of the execution of a request.

Connection mode:

connection-oriented

In a connection-oriented application relationship a logical connection is established between application processes. Three different phases are identified in connection-oriented ARs:

– connection establishment phase.

– data transfer phase, and

– connection release phase.

Exchanging data on a connection shall only be possible after a successful connection establishment. In the connection establishment phase a request to establish the connection is announced at the remote application process (e.g. Initiate service). The request to establish the connection-oriented AR contains the information about the context to be used in the data transfer phase. The remote application informs the requesting application process about the acceptance of the context. If the context is accepted both application processes are in the data transfer phase and are able to communicate with each other according to the agreed context. Connection-oriented AR can be monitored. A connection-oriented AR is dissolved by its release (e.g. Abort service).

connectionless

Connectionless application relationships have neither a connection establishment nor a connection release. They are always in the data transfer phase. It is not possible to monitor the connectionless AR.

Transmission types:

cyclic

The transmission type "cyclic" means that after an initial request by the application process the application relationship is established and afterwards the data transfer is executed periodically.

acyclic

The transmission type "acyclic" means that a single transfer is executed after each request by the application process.

both

The transmission type "both" means that a cyclic as well as acyclic transfer is supported.

Transmission characteristics

Buffer to Buffer

The transmission is characterized by a buffer to buffer transfer.

Queue to Queue
The transmission is characterized by a queue to queue transfer.

Buffer to Queue
The transmission is characterized by a buffer to queue transfer.

Queue to Buffer
The transmission is characterized by a queue to buffer transfer.

Service types
Confirmed
Only confirmed services are allowed.

Unconfirmed
Only unconfirmed services are allowed.

Transmission directions:
Bi-directional
APDUs are transmitted in both directions.

Unidirectional
APDUs are transmitted only in one direction.

Service relations:
Client-Server communication
The services are used according to the Client/Server model.

Publisher-Subscriber communication
The services are used according to the Publisher/Subscriber model.

6.2.10.1.2.5 Overview of communication relationships

The DP Application Layer uses the following types of communication relationships:

BBUB (Buffer Buffer Unconfirmed Bidirectional)

This type of communication relationship is characterized by the following role attributes:

– one to one,
– connection-oriented,
– cyclic,
– buffer to buffer,
– unconfirmed,
– bi-directional,
– Client/Server.

BBUU-OM (Buffer Buffer Unconfirmed Unidirectional - One to Many)

This type of communication relationship is characterized by the following role attributes:

– one to many,
– connection-oriented,
– cyclic,
– buffer to buffer,
– unconfirmed,
– unidirectional,
– Publisher/Subscriber.

BQUU (Buffer Queue Unconfirmed Unidirectional)

This type of communication relationship is characterized by the following role attributes:

– one to one,
– connectionless,
– acyclic,
– buffer to queue,
– unconfirmed,
– unidirectional,
– Client/Server.

QQUU-OM (Queue Queue Unconfirmed Unidirectional - One to Many)

This type of communication relationship is characterized by the following role attributes:

– one to many,
– connectionless,
– acyclic,
– queue to queue,
– unconfirmed,
– unidirectional,
– Client/Server.

BBUU (Buffer Buffer Unconfirmed Unidirectional)

This type of communication relationship is characterized by the following role attributes:

– one to one,
– connectionless,
– acyclic,
– buffer to buffer,
– unconfirmed,
– unidirectional,
– Client/Server.

QQCB-CO (Queue Queue Confirmed Bi-directional - Connection-Oriented)

This type of communication relationship is characterized by the following role attributes:

– one to one,
– connection-oriented,
– acyclic,
– queue to queue,
– confirmed,
– bidirectional,
– Client/Server.

QQUU-OO (Queue Queue Unconfirmed Unidirectional - One to One)

This type of communication relationship is characterized by the following role attributes:

– one to one,
– connectionless,
– acyclic,
– queue to queue,
– unconfirmed,
– unidirectional,
– Client/Server.

QBCB (Queue Buffer Confirmed Bi-directional)

This type of communication relationship is characterized by the following role attributes:

– one to one,
– connectionless,
– acyclic,
– queue to buffer,
– confirmed,
– bi-directional,
– Client/Server.

QQCB-CL (Queue Queue Confirmed Bi-directional - Connectionless)

This type of communication relationship is characterized by the following role attributes:

– one to one,
– connectionless,
– acyclic,
– queue to queue,
– confirmed,
– bi-directional,
– Client/Server.

6.2.10.2 Fieldbus ARs

6.2.10.2.1 MS0 application relationship

The MS0 AR reflects the application relationship (see Figure 30) between the application processes of one DP-master (Class 1) and optional the APs of one or several DP-master (Class 2) with the Default AP of all related DP-slaves, additionally the MS0 AR provides the application relationship between the Default-APs of one or several DP-slaves with the Default-APs of a group of DP-slaves for the following purposes:

– cyclic exchange of the I/O data (DP-master (Class 1)) (Client/Server),
– cyclic exchange of the Input Data (DP-slave) (Publisher/Subscriber),
– acyclic data transfer for parameterization, configuration and diagnosis (DP-master (Class 1)),
– acyclic transfer of commands to a set of field devices (DP-master (Class 1)),
– cyclic transfer of synchronization messages to a set of field devices (DP-master (Class 1)),

– acyclic read of the I/O data (DP-master (Class 2)),
– acyclic read of configuration information (DP-master (Class 2)),
– acyclic read of diagnosis information (DP-master (Class 2)),
– acyclic write of remanent parameter (DP-master (Class 2)).

This type of application relationship is characterized by the following role attributes:

– many to many,
– Master-slave communication.

The MS0 AR contains the following CR types:

– BBUB (Data Exchange),
– BBUU-OM (DXB),
– BQUU (Prm, Ext Prm and Cfg),
– BBUU (Slave Diag),
– QQUU-OM (Global Control),
– QBCB (Slave Diag, Read Cfg, Read Input, Read Output, Rem Para).

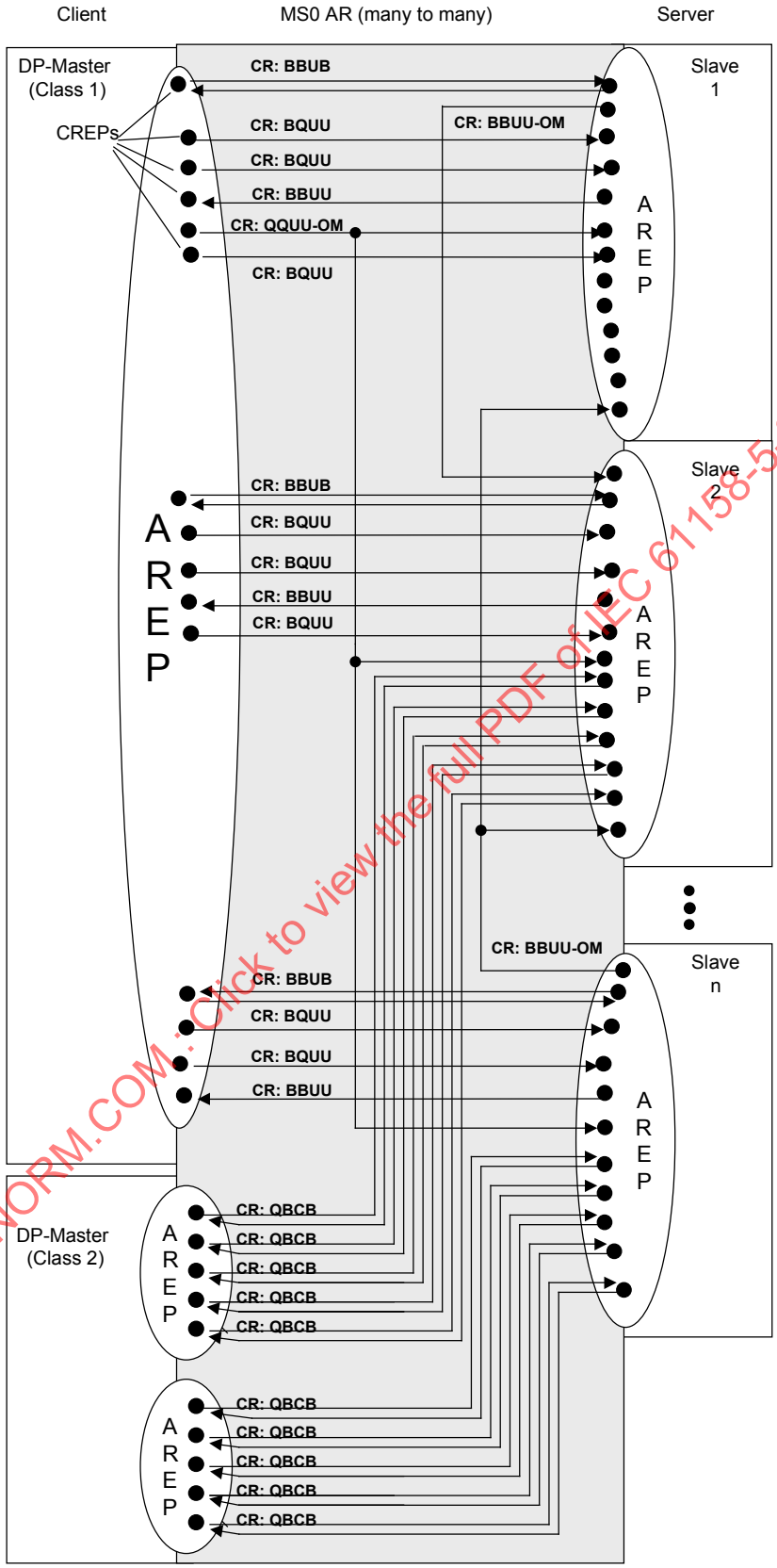


Figure 30 – MS0 application relationship

The DP-Application Layer incorporates buffers to convey the input and output data. These buffers decoupage the communication and data processing. For synchronization purposes two buffers for the Input Data and two buffers for the Outputs are necessary in the DP-Application Layer. If a DP-slave supports no synchronization capabilities only one buffer is needed in each direction. For the data of the subscribed Input Data only one buffer may be provided by the DP-Application Layer. The detailed model is described in the following subclauses. Figure 31 through Figure 34 illustrate the Input and Output Buffer models of a DP-slave.

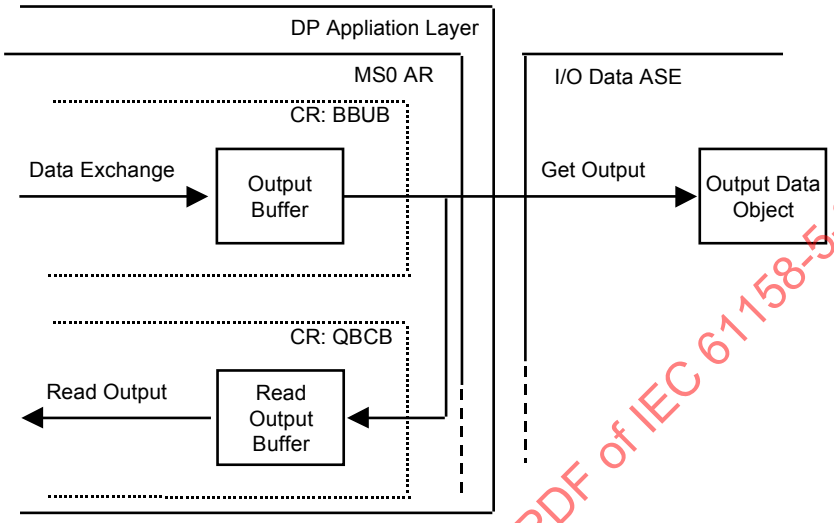


Figure 31 – Output buffer model of a DP-slave without sync functionality

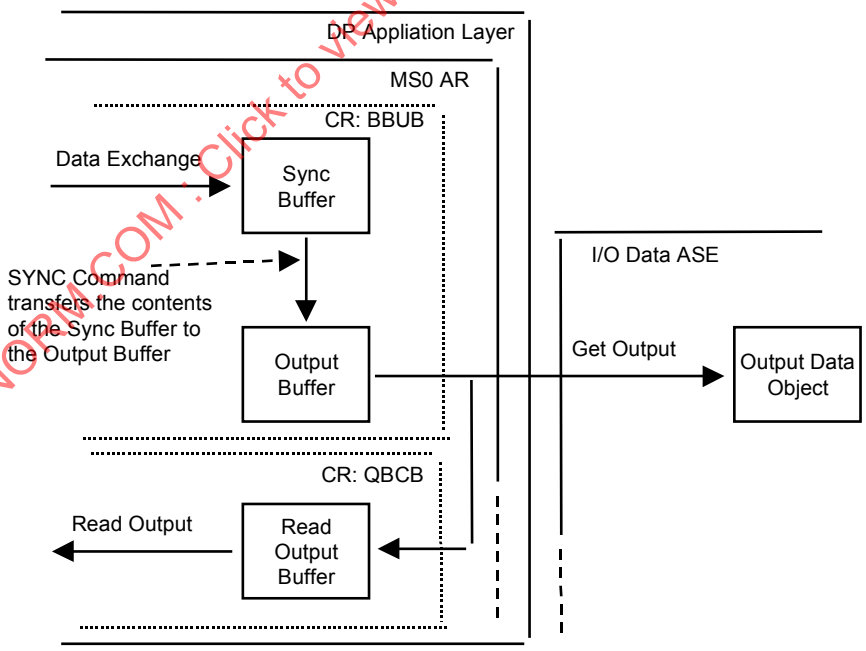


Figure 32 – Output buffer model of a DP-slave with sync functionality

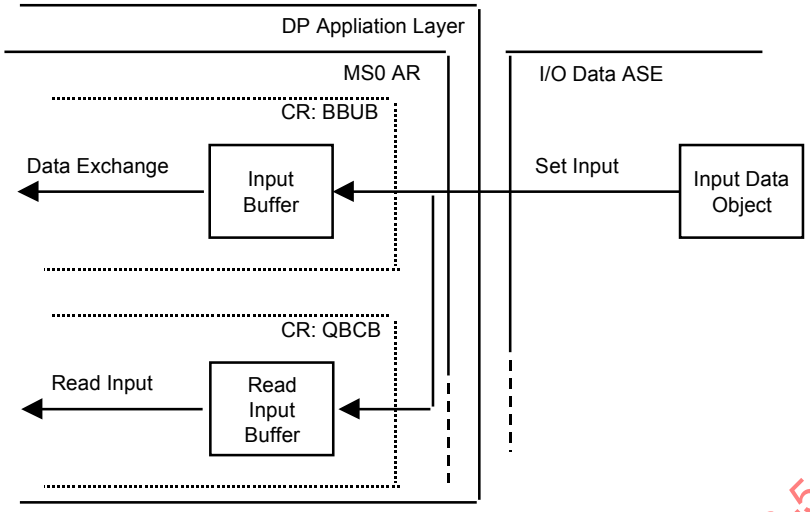


Figure 33 – Input buffer model of a DP-slave without freeze functionality

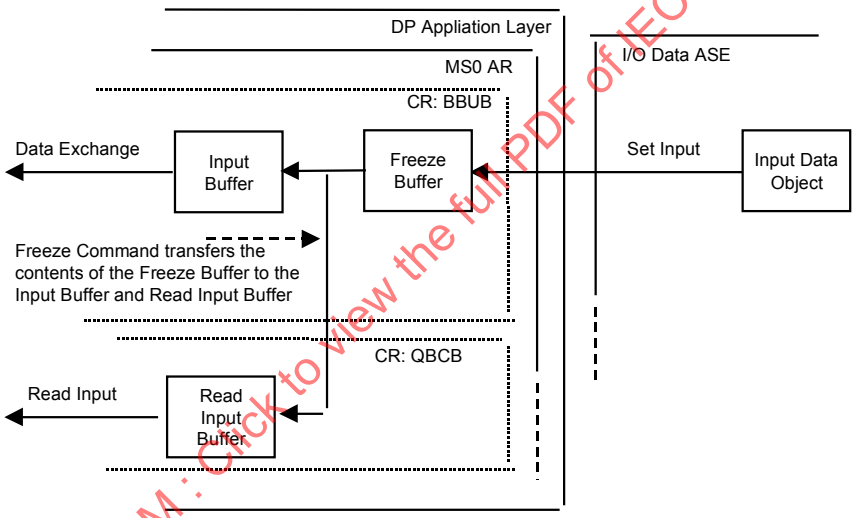


Figure 34 – Input buffer model of a DP-slave with freeze functionality

6.2.10.2.2 MS1 application relationship

The MS1 AR reflects the connection-oriented application relationship (see Figure 35) between the application processes of one DP-master (Class 1) and of one related DP-slave for the following purposes:

– acyclic read and write of variables,
– acyclic transfer of alarms,
– up- and/or download of LR Data,
– invocation of stateless and/or state-oriented functions.

This type of application relationship is characterized by the following role attributes:

– one to one,
– Master-slave communication.

The MS1 AR contains the following CR types:

- QQCB-CO (Process Data, Alarm Ack,)

QQUU-OO (Alarm)

and optionally the following CR type:

QQCB-CO (Alarm Ack)

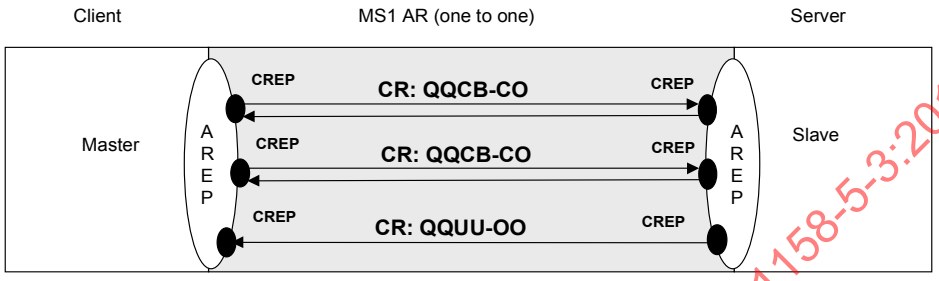


Figure 35 – MS1 application relationship

6.2.10.2.3 MS2 application relationship

6.2.10.2.3.1 General

The MS2 AR reflects the connection-oriented application relationship (see Figure 36) between the application processes of one DP-master (Class 2) and of one related DP-slave for the purpose:

– acyclic read and write of variables,
– up- and/or download of LR Data,
– invocation of stateless and/or state-oriented functions.

This type of application relationship is characterized by the following role attributes:

– one to one,
– Master-slave communication.

The MS2 AR contains the following CR type:

– QQCB-CO (Process Data),
– QQUU-OO (Resource Manager for Context Management).

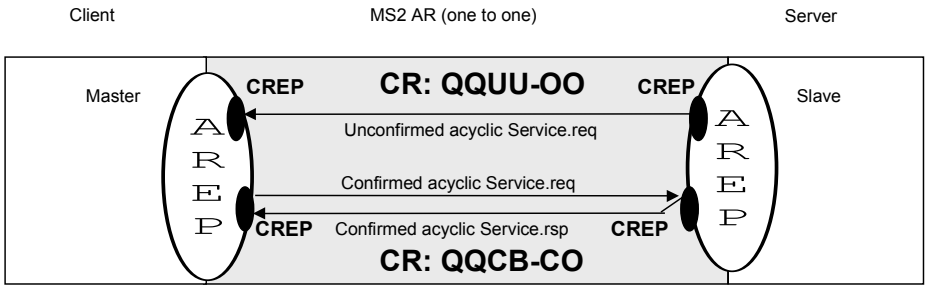


Figure 36 – MS2 application relationship

6.2.10.2.3.2 Communication over multiple interconnected networks

With the Initiate service it is possible to communicate over multiple networks.

The following cases are considered:

– data transfer from one DP-Network to another DP-Network over a link,
– data transfer from a None-DP-Network to a DP-Network over a gateway.

Within such networks each DP-Device including their applications are addressed with the following parameters:

– Network address,
– MAC,
– API, SCL.

The format of these addresses is defined in the Initiate service description. It is necessary to distinguish between DP-Devices which are either used as a link or a communication endpoint. In the case of the link the Initiate-PDU has to be passed to the addressed device. The Network address 0 is reserved for default addressing purposes and therefore shall not be used as a normal network address.

The Figure 37 shows a cut out of an heterogeneous network. Fieldbus DP covers only the marked area.

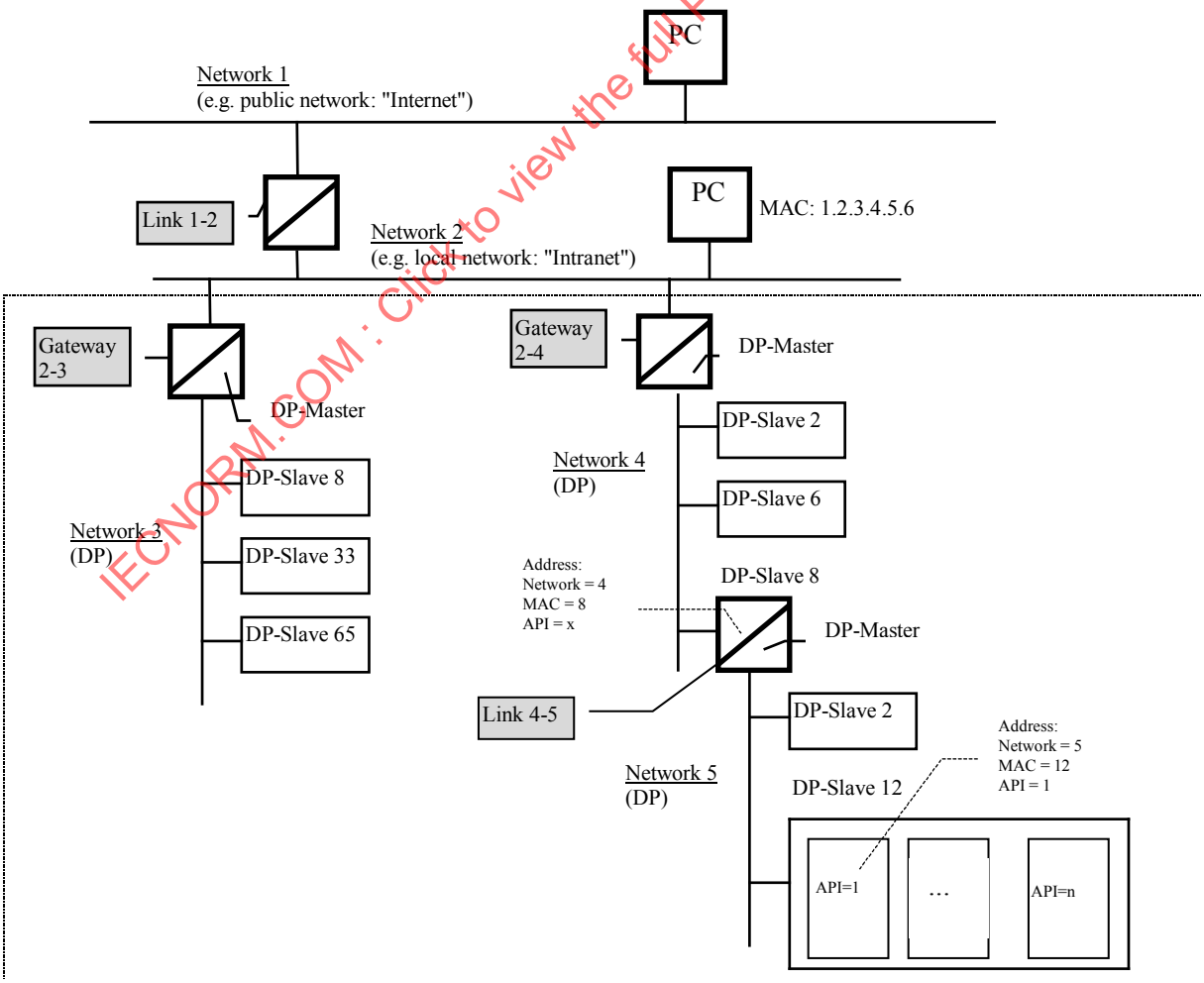


Figure 37 – Example of inter-network communication

The following examples (see Figure 38 through Figure 40) show the exchange of the source- and destination-address parameter of the Initiate.req and the Initiate.res service.

Between the AP 10 of DP-master (Class 2) and AP 1 of the DP-slave a communication relationship will be established. This is done with the Initiate service. After the connection establishment the transfer of data blocks will be realized with the services Read and Write using the parameters Slot Number and Index.

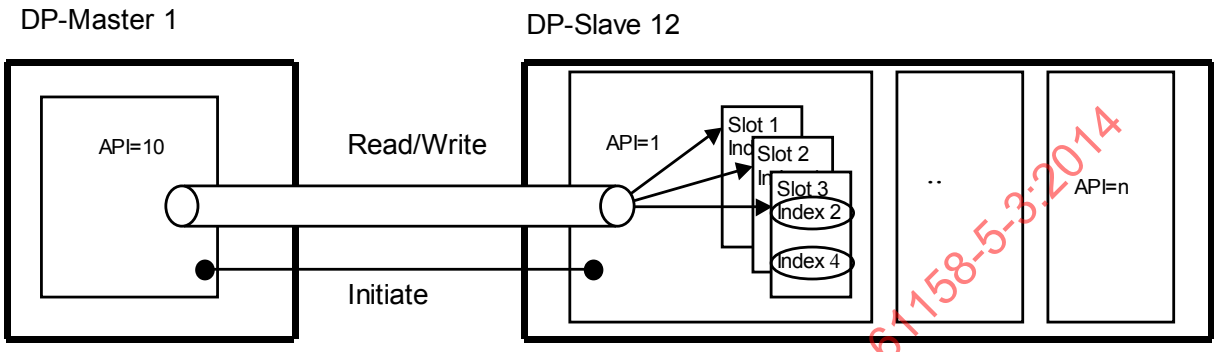


Figure 38 – Example without inter-network addressing

Table 196 shows the parameters of the Initiate service without Inter-Network Addressing.

Table 196 – Parameter of Initiate service without inter-network addressing

Initiate.req (1 -> 12)		Initiate.res (12 -> 1)	
S-Type	0	S-Type	0
S-Len	2	S-Len	2
D-Type	0	D-Type	0
D-Len	2	D-Len	2
S-Addr	API 10, SCL	S-Addr	API 1, SCL
D-Addr	API 1, SCL	D-Addr	API 10, SCL

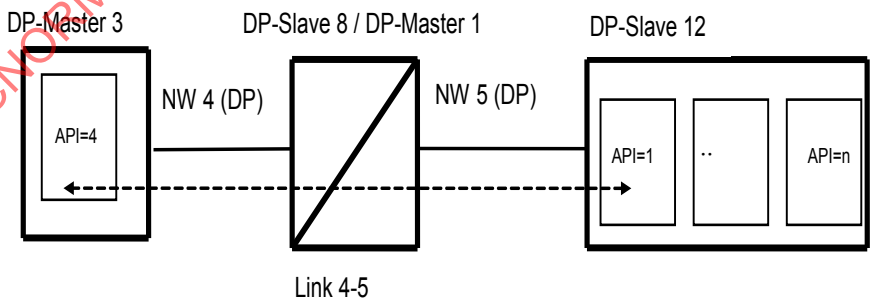


Figure 39 – First example with inter-network addressing

Table 197 shows the parameters of the Initiate service with Inter-Network Addressing.

Table 197 – Parameter of Initiate service with inter-network addressing (first example)

Initiate.req (3 -> 8)		Initiate.res (8 -> 3)	
S-Type	0	S-Type	1
S-Len	2	S-Len	9
D-Type	1	D-Type	0
D-Len	9	D-Len	2
S-Addr	API 4, SCL	S-Addr	API 1, SCL, NW5, 12
D-Addr	API 1, SCL, NW5, 12	D-Addr	API 4, SCL
Initiate.req (1 -> 12)		**Initiate.res (12 -> 1)**	
S-Type	1	S-Type	0
S-Len	9	S-Len	2
D-Type	0	D-Type	1
D-Len	2	D-Len	9
S-Addr	API 4, SCL, NW4, 3	S-Addr	API 1, SCL
D-Addr	API 1, SCL	D-Addr	API 4, SCL, NW4, 3

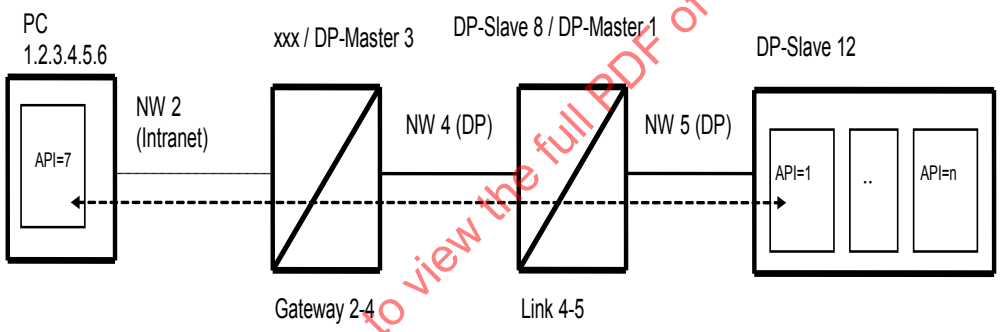


Figure 40 – Second example with inter-network addressing

Table 198 shows the parameters of the Initiate service with Inter-Network Addressing.

Table 198 – Parameter of Initiate service with inter-network addressing (second example)

Initiate-REQ-PDU (3 -> 8)		Initiate-RES-PDU (8 -> 3)	
S-Type	1	S-Type	1
S-Len	14	S-Len	9
D-Type	1	D-Type	1
D-Len	9	D-Len	14
S-Addr	API 7, SCL, NW2, 1.2.3.4.5.6	S-Addr	API 1, SCL, NW5, 12
D-Addr	API 1, SCL, NW5, 12	D-Addr	API 7, SCL, NW2, 1.2.3.4.5.6
Initiate-REQ-PDU (1 -> 12)		**Initiate-RES-PDU (12 -> 1)**	
S-Type	1	S-Type	0
S-Len	14	S-Len	2
D-Type	0	D-Type	1
D-Len	2	D-Len	14
S-Addr	API 7, SCL, NW2, 1.2.3.4.5.6	S-Addr	API 1, SCL
D-Addr	API 1, SCL	D-Addr	API 7, SCL, NW2, 1.2.3.4.5.6

6.2.10.2.4 MS3 application relationship

The MS3 AR reflects the connectionless relationship (see Figure 41) between the application processes of a DP-master and of a set of related DP-slaves for the purpose:

– synchronization of time.

This type of application relationship is characterized by the following role attributes:

– one to many,
– Master-slave communication.

The MS3 AR contains the following CR type:

– QQUU-OM (Time).

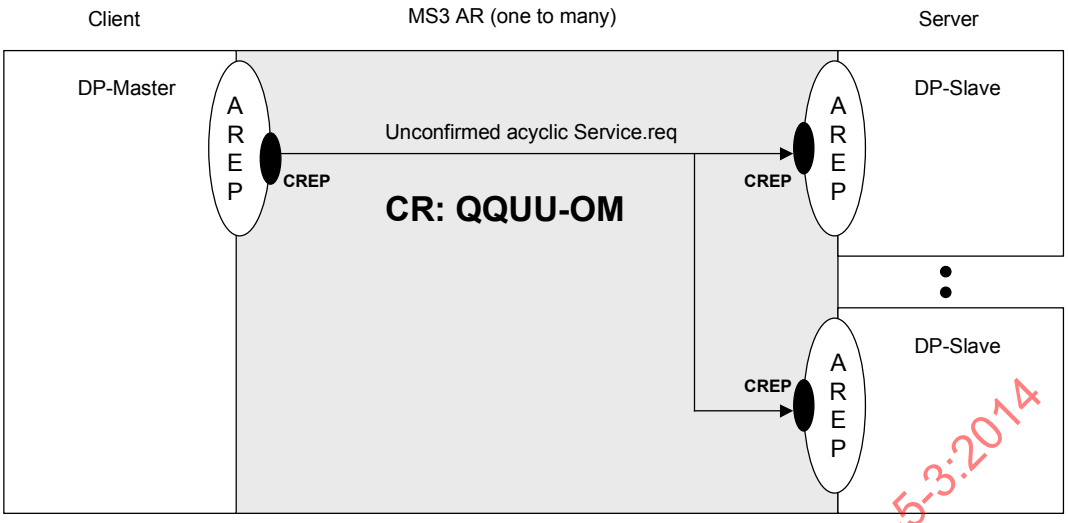


Figure 41 – MS3 application relationship

6.2.10.2.5 MM1 application relationship

The MM1 AR reflects the connectionless relationship (see Figure 42) between the application processes of one configuration device (Master Class 2) and of one related controlling device (Master Class 1) for the following purposes:

– upload and download for configuration information,
– upload of diagnosis information,
– activation of the previously transferred configuration.

This type of application relationship is characterized by the following role attributes:

– one to one,
– Master-master communication.

The MM1 AR contains the following CR type:

– QQCB-CL (MM1(Master Diag/Start Seq/End Seq/Download/Upload/Act Param)).

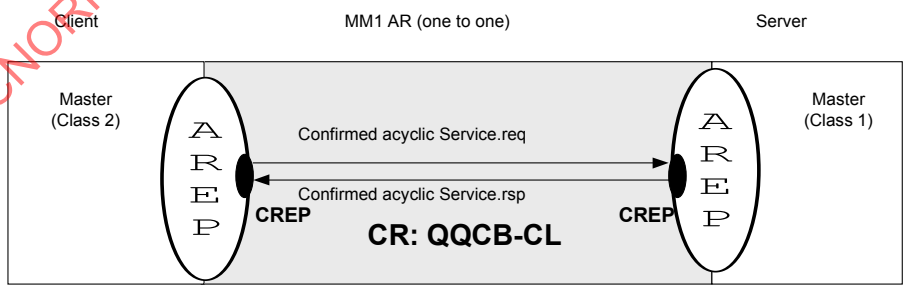


Figure 42 – MM1 application relationship

6.2.10.2.6 MM2 application relationship

The MM2 AR reflects the connectionless relationship (see Figure 43) between the application processes of one configuration device (Master Class 2) and of a set of related controlling devices (Master Class 1) for the purpose:

– activation of the previously transferred configurations

This type of application relationship is characterized by the following role attributes:

– one to many,
– Master-master communication.

The MM2 AR contains the following CR type:

– QQUU-OM (Act Para Brct).

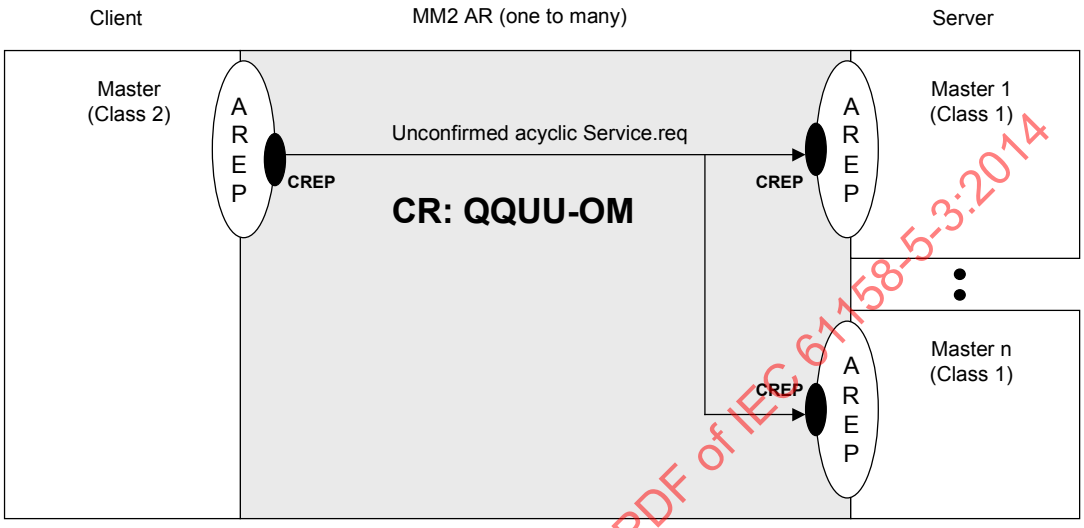


Figure 43 – MM2 application relationship

6.2.10.3 Application relationship list class specification

The ARL Class specification is distinguished between DP-slave, DP-master (Class 1) and DP-master (Class 2).

A static attribute is assigned by a load service and are not changed during operation, whereas a dynamic attribute may set by the load services and may be changed during operation.

6.2.10.3.1 ARL DP-slave class specification

6.2.10.3.1.1 Template

The ARL DP-slave object is described by the following template:

DP ASE: **AR ASE**
CLASS: **ARL DP-slave**
CLASS ID: not used
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Key Attribute:	Implicit
2.	(c)	Constraint:	at least one entry with AR Type = MS2
2.1.	(m)	Attribute:	Min Send Timeout
2.2.	(m)	Attribute:	List of Resource Manager Entries
2.2.1.	(m)	Attribute:	API
2.2.2.	(m)	Attribute:	SCL
2.2.3.	(m)	Attribute:	Max Len Data Unit
2.2.4.	(m)	Attribute:	Send Timeout
2.2.5.	(m)	Attribute:	AREP
2.2.6.	(m)	Attribute:	S_SAP_index
2.2.7.	(m)	Attribute:	In Use
2.3.	(m)	Attribute:	List of S_SAP_indices
2.3.1.	(m)	Attribute:	S_SAP_index
3.	(m)	Attribute:	List of ARL Entries

3.1.	(m)	Attribute:	AREP
3.2.	(m)	Attribute:	AR Type
3.3.	(m)	Attribute:	List of related CRL Entries
3.3.1.	(m)	Attribute:	CREP
3.4.	(c)	Constraint:	AR Type = MS0
3.4.1.	(m)	Attribute:	Prm Req
3.4.2.	(m)	Attribute:	Prm Fault
3.4.3.	(m)	Attribute:	Cfg Fault
3.4.4.	(m)	Attribute:	Not Supported
3.4.5.	(m)	Attribute:	Station Not Ready
3.4.6.	(m)	Attribute:	Stat Diag
3.4.7.	(m)	Attribute:	Master Add
3.4.8.	(m)	Attribute:	Sync Supported
3.4.9.	(m)	Attribute:	Sync Mode
3.4.10.	(m)	Attribute:	Freeze Supported
3.4.11.	(m)	Attribute:	Freeze Mode
3.4.12.	(m)	Attribute:	WD Enabled
3.4.13.	(m)	Attribute:	WD Fact 1
3.4.14.	(m)	Attribute:	WD Fact 2
3.4.15.	(m)	Attribute:	Min TSDR
3.4.16.	(m)	Attribute:	Ident Number
3.4.17.	(m)	Attribute:	Group Identifier
3.4.18.	(m)	Attribute:	DPV1 Supported
3.4.19.	(c)	Constraint:	DPV1 Supported = TRUE
3.4.19.1.	(m)	Attribute:	DPV1 Enabled
3.4.19.2.	(m)	Attribute:	Fail Safe Required
3.4.19.3.	(m)	Attribute:	Fail Safe
3.4.19.4.	(m)	Attribute:	WD Base
3.4.20.	(m)	Attribute:	List of Real Cfg Data
3.4.20.1.	(s)	Attribute:	Cfg Identifier
3.4.20.1.1.	(m)	Attribute:	Type
3.4.20.1.2.	(m)	Attribute:	Format
3.4.20.1.3.	(m)	Attribute:	Consistency
3.4.20.1.4.	(m)	Attribute:	Length
3.4.20.2.	(s)	Attribute:	Special Cfg Identifier
3.4.20.2.1.	(m)	Attribute:	Type
3.4.20.2.2.	(c)	Constraint:	Type = Input or Type = Input-Output
3.4.20.2.2.1.	(m)	Attribute:	Input Length
3.4.20.2.2.2.	(m)	Attribute:	Input Format
3.4.20.2.2.3.	(m)	Attribute:	Input Consistency
3.4.20.2.3.	(c)	Constraint:	Type = Output or Type = Input-Output
3.4.20.2.3.1.	(m)	Attribute:	Output Length
3.4.20.2.3.2.	(m)	Attribute:	Output Format
3.4.20.2.3.3.	(m)	Attribute:	Output Consistency
3.4.20.2.4.	(o)	Attribute:	List of Data Description
3.4.20.2.4.1.	(m)	Attribute:	Data Description
3.4.20.2.5.	(o)	Attribute:	Manufacturer Specific Data
3.4.21.	(o)	Attribute:	No Add Change
3.4.22.	(m)	Attribute:	Publisher Supported
3.4.23.	(m)	Attribute:	Isochronous Mode Supported
3.4.24.	(c)	Constraint:	Isochronous Mode Supported = TRUE
3.4.24.1.	(m)	Attribute:	Isochronous Mode
3.4.25.	(m)	Attribute:	Prm Command Supported
3.5.	(c)	Constraint:	AR Type = MS1
3.5.1	(m)	Attribute:	Pull Plug Alarm supported
3.5.2	(c)	Constraint:	Pull Plug Alarm supported = TRUE
3.5.2.1	(m)	Attribute:	Pull Plug Alarm enabled
3.5.3.	(m)	Attribute:	Process Alarm supported
3.5.4.	(c)	Constraint:	Process Alarm supported = TRUE
3.5.4.1.	(m)	Attribute:	Process Alarm enabled
3.5.5.	(m)	Attribute:	Diagnosis Alarm supported
3.5.6.	(c)	Constraint:	Diagnosis Alarm supported = TRUE
3.5.6.1.	(m)	Attribute:	Diagnosis Alarm enabled
3.5.7.	(m)	Attribute:	Manufacturer Specific Alarm supported
3.5.8.	(c)	Constraint:	Manufacturer Specific Alarm supported = TRUE
3.5.8.1.	(m)	Attribute:	Manufacturer Specific Alarm enabled
3.5.9.	(m)	Attribute:	Status Alarm supported
3.5.10.	(c)	Constraint:	Status Alarm supported = TRUE

3.5.10.1.	(m)	Attribute:	Status Alarm enabled
3.5.11.	(m)	Attribute:	Update Alarm supported
3.5.12.	(c)	Constraint:	Update Alarm supported = TRUE
3.5.12.1.	(m)	Attribute:	Update Alarm enabled
3.5.13.	(m)	Attribute:	Alarm Mode supported
3.5.14.	(c)	Constraint:	Alarm Mode supported = TRUE
3.5.14.1.	(m)	Attribute:	Alarm Mode enabled
3.6.	(c)	Constraint:	AR Type = MS3
3.6.1.	(m)	Attribute:	Clock Sync Interval
3.6.2.	(o)	Attribute:	CS Delay Time
3.7.	(o)	Attribute:	AR Name

SERVICES:

1.	(m)	MgtService:	Load ARL
2.	(o)	MgtService:	Get ARL
3.	(m)	MgtService:	Set Cfg
4.	(o)	MgtService:	Set ARL Isochronous Mode

6.2.10.3.1.2 Attributes

Implicit
The attribute Implicit indicates that the ARL DP-slave object is implicitly addressed by the service.

Min Send Timeout
This conditional and static attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute defines the minimum Send Timeout common for all MS2 ARs supported in this station.

Attribute Type: Unsigned16

Allowed value: 1 to 2^{16} - 1

Time Base: 10

List of Resource Manager Entries
This conditional attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute is common for all MS2 ARs supported in this station and composed of the following list elements (for each supported MS2 AR a complete list of elements is needed):

API
This conditional and static attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. The attribute API specifies the numeric Identifier for the related Application Process of this MS2 AR.

Attribute Type: Unsigned8

Allowed values: 0 to 255 (0 = Default AP)

SCL
This conditional and static attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. The attribute SCL specifies the configured Security Level for the identified Application Process of this MS2 AR.

Attribute Type: Unsigned8

Allowed values: 0 to 255 (0 = no Security Level used)

Max Len Data Unit
This conditional and static attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute reflects the upper limit of the parameter Length of the services of the Process Data ASE for this MS2 AR.

Attribute Type: Unsigned8

Allowed values: 0 to 240

Send Timeout
This conditional and dynamic attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute defines the Send Timeout for this MS2 AR. This attribute is set to the maximum of the value of the attribute Min Send Timeout and the field Send Timeout of the previously received Initiate-REQ-PDU for this MS2 AR. This attribute is not influenced by the Load ARL service.

Attribute Type: Unsigned16

Allowed value: 1 to 2^{16} - 1

Time Base: 10

AREP
This conditional and static attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute specifies the local identifier of this MS2 AR.

Attribute Type: Unsigned8

S_SAP_index
This conditional and dynamic attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute contains the assigned S_SAP_index from the list of S_SAP_indices after reception of an Initiate-REQ-PDU for this MS2 AR. This attribute is not influenced by the Load ARL service.

Attribute Type: Unsigned8

Allowed values: 0 to 48

In Use
This conditional and dynamic attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute specifies the state of this MS2 AR. This attribute is set to TRUE after reception of an Initiate-REQ-PDU and assignment of a S_SAP_index for this MS2 AR. This attribute is set to FALSE after release of this MS2 AR. This attribute is not influenced by the Load ARL service.

Attribute Type: Boolean

Allowed values: FALSE, TRUE

List of S_SAP_indices
This conditional attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute is common for all MS2 ARs supported in this station and composed of the following list elements:

S_SAP_index
This conditional and static attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS2. This attribute specifies an available Responder DLSAP for the MS2 ARs of the DP-slave.

Attribute Type: Unsigned8

Allowed values: 0 to 48

List of ARL Entries
This attribute is composed of the following list elements:

AREP
This static attribute contains the local identifier of this AR.

Attribute Type: Unsigned8

AR Type
This static attribute specifies the type of this AR.

Attribute Type: Unsigned8

The allowed values are shown in Table 199.

Table 199 – AR type

Value	Meaning
0	MS0
1	MS1
2	MS2
3	MS3

List of related CRL Entries
This attribute is composed of the following list elements:

CREP
This static attribute contains the local identifier for the related CR for this AR.

Attribute Type: Unsigned8

AR Type = MS0
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MS0.

Prm Req
This dynamic attribute indicates if the DP-slave has to be reparameterized and reconfigured. This attribute shall be set to TRUE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Prm Fault
This dynamic attribute indicates the result of the check of the parameterization. This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Cfg Fault
This dynamic attribute indicates the result of the check of the configuration data. This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Not Supported
This dynamic attribute indicates that a functionality requested by the DP-master (Class 1) is not supported on the MS0 AR by the DP-slave. This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Station Not Ready
This dynamic attribute indicates the readiness to enter the data exchange mode with the assigned DP-master (Class 1). This attribute shall be set to TRUE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Stat Diag
This dynamic attribute indicates that the DP-master (Class 1) shall fetch permanently diagnosis from the DP-slave until the application indicates readiness for operation with the Application Ready service. This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine and the Application Ready service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Master Add
This dynamic attribute contains the DL-address of the DP-master (Class 1) which has locked the DP-slave. This attribute shall be set to invalid (= 255) with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Unsigned8

Allowed values: 0 to 126, 255

Sync Supported
This static attribute reflects the ability of a DP-slave to participate in the synchronization of the Output Data of all or a group of DP-slaves. As the check whether the Sync ability is supported is done during parameterization phase, errors will be avoided during the data exchange mode. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Table 200 shows the allowed values for this attribute.

Table 200 – Sync supported

Value	Meaning
FALSE	The DP-slave does not support the Sync ability
TRUE	The DP-slave supports the Sync ability

Sync Mode
This dynamic attribute indicates if the Sync mode is activated. This attribute shall always be set to FALSE if the attribute Sync Supported has the value FALSE. If the attribute Sync Supported has the value TRUE the following rules shall apply for the attribute Sync Mode:

– It shall be set to TRUE as soon as the control command Sync is received, which is conveyed by the Global Control service.

– It shall be set to FALSE as soon as the control command Unsync is received which is conveyed by the Global Control service.

This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Freeze Supported
This static attribute reflects the ability of a DP-slave to participate in the synchronization of the Input Data of all or a group of DP-slaves. As the check whether the Freeze ability is supported is done during parameterization phase, errors will be avoided during the data exchange mode. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Table 201 shows the allowed values for this attribute.

Table 201 – Freeze supported

Value	Meaning
FALSE	The DP-slave does not support the Freeze ability
TRUE	The DP-slave supports the Freeze ability

Freeze Mode
This dynamic attribute indicates if the Freeze mode is activated. This attribute shall always be set to FALSE if the attribute Freeze Supported has the value FALSE. If the attribute Freeze Supported has the value TRUE the following rules shall apply for the attribute Freeze Mode:

– It shall be set to TRUE as soon as the control command Freeze is received, which is conveyed by the Global Control service.

– It shall be set to FALSE as soon as the control command Unfreeze is received which is conveyed by the Global Control service.

This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine

Attribute Type: Boolean.

WD Enabled
This dynamic attribute indicates if the watchdog control is activated. This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

WD Fact 1, WD Fact 2
These dynamic attributes contain the values for the monitoring of the DP-master (Class 1) at the MS0 AR (watchdog control (T_{WD})). The watchdog control in a DP-slave takes care that, if the DP-master fails, the outputs will be set in the safe state after the expiration of this time. The time is calculated according to equation:

T_{WD} = Watchdog Time Base × WD Fact 1 × WD Fact 2

If the attribute DPV1 Supported has the value FALSE the Watchdog Time Base shall be set to 10 ms.

If the attribute DPV1 Supported has the value TRUE the Watchdog Time Base shall be set to the value 1if the attribute WD Base has the value TRUE. If the attribute DPV1 Supported has the value TRUE the Watchdog Time Base shall be set to the value 10if the attribute WD Base has the value FALSE. These attributes are influenced by the MS0 AR state machine. These attributes are not influenced by the Load ARL service.

Attribute Type: Unsigned8

Allowed values: 1 to 255

Min TSDR
This dynamic attribute contains the value for the Minimum Station Delay Responder time of the DLL. If this Parameter is equal zero, the previous value of the Min TSDR remains unchanged. This attribute is influenced by the MS0 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Unsigned8

Allowed values: 0 to 255

Ident Number
This static attribute contains the Ident Number of the DP-slave. The DP-slave accepts the parameterization and change of the DL-address only if the transmitted Ident Number in the correspondent service is identical with this attribute. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Unsigned16

Allowed values: 0 to 2^{16} - 1

Group Identifier
This dynamic attribute indicates the group(s) to which this DP-slave has been assigned by the DP-master (Class 1) in the parameterization. The Group Identifier distinguishes 8 groups. If the Group Identifier is set to No Group, the DP-slave is not assigned to any group. This attribute is influenced by the MS0 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: OctetString(1)

The allowed values are shown in Table 202.

Table 202 – Group identifier

Value	Meaning
Bit 0 = 0	Group 1 has no access
Bit 0 = 1	Group 1 has access
Bit 1 = 0	Group 2 has no access
Bit 1 = 1	Group 2 has access
Bit 2 = 0	Group 3 has no access
Bit 2 = 1	Group 3 has access
Bit 3 = 0	Group 4 has no access
Bit 3 = 1	Group 4 has access
Bit 4 = 0	Group 5 has no access
Bit 4 = 1	Group 5 has access
Bit 5 = 0	Group 6 has no access
Bit 5 = 1	Group 6 has access
Bit 6 = 0	Group 7 has no access
Bit 6 = 1	Group 7 has access
Bit 7 = 0	Group 8 has no access
Bit 7 = 1	Group 8 has access

DPV1 Supported
This static attribute indicates whether the DP-slave supports the extended functionality. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

DPV1 Enabled
This dynamic attribute indicates if the extended functionality is activated. This conditional attribute shall only be present if the attribute DPV1 Supported has the value TRUE. The following rules shall apply for the attribute DPV1 Enabled:

– It shall be set to TRUE if the DP-master (Class 1) requests the DPV1 support in the parameterization.

– It shall be set to FALSE if the DP-master (Class 1) requests no DPV1 support in the parameterisation.

This attribute shall be set to FALSE with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Table 203 shows the allowed values for this attribute.

Table 203 – DPV1 enabled

Value	Meaning
FALSE	The extended functionality related with the attribute DPV1 Supported is disabled.
TRUE	The extended functionality related with the attribute DPV1 Supported is enabled.

Fail Safe Required
This static attribute indicates if the DP-slave requires the Fail Safe mode for operation. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Fail Safe
This dynamic attribute indicates if the DP-master (Class 1) supports the Fail Safe mode. If the DP-slave requires the Fail Safe mode and the DP-master (Class 1) does not support this mode the MS0 AR state machine shall set the attribute Prm Fault to TRUE in the MS0 AR entry. This attribute is influenced by the MS0 AR state machine. This attribute is not influenced by the Load ARL service.

Table 204 shows the allowed values for this attribute.

Table 204 – Fail safe

Value	Meaning
FALSE	The DP-master (Class 1) does not support the Fail Safe mode.
TRUE	The DP-master (Class 1) supports the Fail Safe mode.

WD Base
This dynamic attribute contains the Time Base value for the watchdog control according to the parameterization. This attribute is influenced by the MS0 AR state machine. This attribute is not influenced by the Load ARL service.

Table 205 shows the allowed values for this attribute.

Table 205 – WD Base

Value	Meaning
FALSE	The Watchdog Time Base is 10 ms (10).
TRUE	The Watchdog Time Base is 1 ms (1).

List of Real Cfg Data
This dynamic attribute is composed of the following list elements:

Cfg Identifier
This dynamic attribute describes the basic configuration format and shall be set with the Load ARL service or with the Set Cfg service. This attribute is composed of the following elements:

Type
This dynamic attribute indicates the type of the related Input/Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Input, Output, Input-Output

Format
This dynamic attribute indicates the format of the related Input/Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Byte, Word

Consistency
This dynamic attribute indicates the consistency of the related Input/Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Byte, Word, whole length

Length
This dynamic attribute indicates the length of the related Input/Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: 1 to 32

Special Cfg Identifier
This dynamic attribute describes the special configuration format and is composed of the following elements. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Type
This dynamic attribute indicates the type of the related Input/Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Input, Output, Input-Output

Input Length
This dynamic attribute indicates the length of the related Input Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Attribute Type: Unsigned8

Allowed values: 1 to 128

Input Format
This dynamic attribute indicates the format of the related Input Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Byte, Word

Input Consistency
This dynamic attribute indicates the consistency of the related Input Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Byte, Word, whole length

Output Length
This dynamic attribute indicates the length of the related Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Attribute Type: Unsigned8

Allowed values: 1 to 128

Output Format
This dynamic attribute indicates the format of the related Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Byte, Word

Output Consistency
This dynamic attribute indicates the consistency of the related Output Data Element. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Allowed values: Byte, Word, whole length

List of Data Description
This dynamic attribute is composed of the following list elements. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Data Description
This dynamic attribute contains one of the allowed Data Types of the data description of one element of the Extended Input/Output Data object. The allowed Data Types are given in Clause 5. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

Manufacturer Specific Data
This dynamic attribute contains the manufacturer specific data for one Special Cfg Identifier. This attribute shall be set accordingly with the Load ARL service. Furthermore the attribute is influenced by the Set Cfg service.

No Add Change
This dynamic attribute indicates whether it is allowed to change the DP-slave address. To change this attribute itself an initial reset of the DP-slave is necessary. After the initial reset the DP-slave takes the default address 126 and the attribute No Add Change will be set to FALSE. This attribute shall be set accordingly with the Load ARL service. This attribute is further influenced by the MS0 AR state machine.

Table 206 shows the allowed values for this attribute.

Table 206 – No Add change

Value	Meaning
FALSE	Change of the address is allowed.
TRUE	Change of the address is not allowed.

Publisher Supported
This static attribute indicates the capability of a DP-slave to operate as a Publisher. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Isochronous Mode Supported
This static attribute indicates the capability of a DP-slave to operate in the Isochronous Mode. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Isochronous Mode
This dynamic attribute indicates if the DP-slave shall operate in the Isochronous Mode. This attribute shall be set to FALSE with the Load ARL service. After reception of the IsoM Parameter with the Check User Prm or Check Ext User Prm service the User shall set this attribute accordingly with the Set ARL Isochronous Mode service. This shall be done before requesting the services Check User Prm Result or Check Ext User Prm Result.

This conditional attribute shall only be present if the attribute Isochronous Mode Supported is set to TRUE. This attribute is not influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

AR Type = MS1
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MS1. This attribute shall furthermore only be present if the attribute DPV1 Supported is set to TRUE in the MS0 AR entry.

Pull Plug Alarm supported
This static attribute indicates the capability of a DP-slave to send a Pull or a Plug Alarm. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Pull Plug Alarm enabled
This dynamic attribute indicates if the transmission of the Pull Plug Alarm is enabled by the DP-master (Class 1). This conditional attribute shall only be present if the attribute Pull Plug Alarm supported is set to TRUE. This attribute is influenced by the MS1 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Process Alarm supported
This static attribute indicates the capability of a DP-slave to send a Process Alarm. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Process Alarm enabled
This dynamic attribute indicates if the transmission of the Process Alarm is enabled by the DP-master (Class 1). This conditional attribute shall only be present if the attribute Process Alarm supported is set to TRUE. This attribute is influenced by the MS1 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Diagnosis Alarm supported
This static attribute indicates the capability of a DP-slave to send a Diagnosis Alarm. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Diagnosis Alarm enabled
This dynamic attribute indicates if the transmission of the Diagnosis Alarm is enabled by the DP-master (Class 1). This conditional attribute shall only be present if the attribute Diagnosis Alarm supported is set to TRUE. This attribute is influenced by the MS1 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Manufacturer Specific Alarm supported
This static attribute indicates the capability of a DP-slave to send a Manufacturer Specific Alarm. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Manufacturer Specific Alarm enabled
This dynamic attribute indicates if the transmission of the Manufacturer Specific Alarm is enabled by the DP-master (Class 1). This conditional attribute shall only be present if the attribute Manufacturer Specific Alarm supported is set to TRUE. This attribute is influenced by the MS1 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Status Alarm supported
This static attribute indicates the capability of a DP-slave to send a Status Alarm. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Status Alarm enabled
This dynamic attribute indicates if the transmission of the Status Alarm is enabled by the DP-master (Class 1). This conditional attribute shall only be present if the attribute Status Alarm supported is set to TRUE. This attribute is influenced by the MS1 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Update Alarm supported
This static attribute indicates the capability of a DP-slave to send an Update Alarm. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Update Alarm enabled
This dynamic attribute indicates if the transmission of the Update Alarm is enabled by the DP-master (Class 1). This conditional attribute shall only be present if the attribute Update Alarm supported is set to TRUE. This attribute is influenced by the MS1 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Alarm Mode supported
This static attribute indicates the maximum number of possible active alarms in a DP-slave. This attribute shall be set accordingly with the Load ARL service.

Table 207 shows the allowed values for this attribute.

Table 207 – Alarm mode supported

Value	Meaning
0	1 alarm of each type
1	2 alarms in total
2	4 alarms in total
3	8 alarms in total
4	12 alarms in total
5	16 alarms in total
6	24 alarms in total
7	32 alarms in total

Alarm Mode enabled
This dynamic attribute indicates the maximum number of possible active alarms negotiated with the DP-master (Class 1) on this MS1 AR. The negotiation shall be done according to the following rules:

– If the attribute Alarm Mode is set to the value 0 and the desired Alarm Mode of the DP-master (Class 1) contained in the parameterization is set to 0 than this attribute shall be set to 0.

– If the attribute Alarm Mode is set to the value 0 and the desired Alarm Mode of the DP-master (Class 1) contained in the parameterization is set to a value different from 0 than this attribute shall not be set.

– If the attribute Alarm Mode is set to a value different from 0 and the desired Alarm Mode of the DP-master (Class 1) contained in the parameterization is set to 0 than this attribute shall be set to 0.

– If the attribute Alarm Mode is set to a value different from 0 and the desired Alarm Mode of the DP-master (Class 1) contained in the parameterisation is set to a value different from 0 than this attribute shall be set to the lesser value of both.

This attribute is influenced by the MS1 AR state machine. This attribute is not influenced by the Load ARL service.

Attribute Type: Unsigned8

Allowed values: see Table 207

AR Type = MS3
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MS3. This attribute shall furthermore only be present if the attribute DPV1 Supported is set to TRUE in the MS0 AR entry.

Clock Sync Interval
This static attribute contains the value for the clock synchronization interval.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 10

The default value shall be 10 s.

CS Delay Time
This static and optional attribute contains the transmission delay of the network (line delay, repeater delay, ...). It should be included for the calculation of the parameter Time Value of the Set Time service.

Attribute Type: Network Time Difference

The default value shall be 0.

AR Name
This static attribute contains the name of the related ARL Entry. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Visible String (32)

6.2.10.3.1.3 Invocation of ARL DP-slave object

For the invocation of the ARL DP-slave object the following rules apply:

– Only one ARL DP-slave object shall be invoked in a DP-slave.
– Only one MS0 Entry shall be contained in the List of ARL Entries.
– Only one MS1 Entry shall be contained in the List of ARL Entries.
– Only one MS3 Entry shall be contained in the List of ARL Entries.
– No MM1 or MM2 Entry shall be contained in the List of ARL Entries.

6.2.10.3.2 ARL DP-master (class 1) class specification

6.2.10.3.2.1 Template

The ARL DP-master (Class 1) object is described by the following template:

DP ASE:			**AR ASE**
CLASS:			**ARL DP-master (Class 1)**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(c)	Constraint:	at least one entry with AR Type = MS1
2.1.	(m)	Attribute:	Alarm Max
3.	(m)	Attribute:	List of ARL Entries
3.1.	(m)	Attribute:	AREP
3.2.	(m)	Attribute:	AR Type
3.3.	(m)	Attribute:	List of related CRL Entries
3.3.1.	(m)	Attribute:	CREP
3.4.	(c)	Constraint:	AR Type = MS0
3.4.1.	(m)	Attribute:	Error Action Flag
3.4.2.	(m)	Attribute:	Min Slave Interval
3.4.3.	(m)	Attribute:	Data Control Time
3.4.4.	(m)	Attribute:	Max User Global Control
3.4.5.	(m)	Attribute:	Max Diag Data Len
3.4.5.	(m)	Attribute:	Isochronous Mode supp
3.4.6.	(c)	Constraint:	Isochronous Mode supp = TRUE
3.4.6.1.	(m)	Attribute:	Isochronous Mode
3.4.6.2.	(m)	Attribute:	IsoM Freeze
3.4.6.3.	(m)	Attribute:	IsoM Sync
3.4.6.4.	(m)	Attribute:	T_{CT}
3.4.6.5.	(m)	Attribute:	$maxT_{SH}$
3.5.	(c)	Constraint:	AR Type = MS1
3.5.2.	(m)	Attribute:	Extra Alarm SAP
3.5.3.	(m)	Attribute:	Max Channel Data Length
3.5.4.	(m)	Attribute:	Alarm Mode
3.5.5.	(m)	Attribute:	MS1 Timeout
3.6.	(c)	Constraint:	AR Type = MM1
3.6.1.	(m)	Attribute:	Poll Timeout
3.6.2.	(m)	Attribute:	Ident Number
3.7.	(c)	Constraint:	AR Type = MS3
3.7.1.	(m)	Attribute:	Time Device Type
3.7.2.	(c)	Constraint:	Time Device Type = Time Receiver
3.7.2.1.	(m)	Attribute:	Clock Sync Interval

3.7.2.2.	(o)	Attribute:	CS Delay Time
3.8.	(o)	Attribute:	AR Name

SERVICES:

1.	(m)	MgtService:	Load ARL
2.	(o)	MgtService:	Get ARL
3.	(o)	MgtService:	ARL Slave Update DP-master Cl1

6.2.10.3.2.2 Attributes

Implicit
The attribute Implicit indicates that the ARL DP-slave object is implicitly addressed by the service.

Alarm Max
This conditional and static attribute shall only be present if the List of ARL Entries contains at least one entry with the attribute AR Type set to MS1. It indicates the maximum number of alarms per DP-slave, which can be handled by the DP-master (Class 1). It describes a Master capability and is common for all MS1 ARs. The Master shall be able to handle a minimum number of 7 alarms, because there are existing 7 different alarm types. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned8

Allowed values: 7 to 32

List of ARL Entries
This attribute is composed of the following list elements:

AREP
This static attribute contains the local identifier of this AR.

AR Type
This static attribute specifies the type of this AR.

Allowed values: MS0, MS1, MS3, MM1, MM2

List of related CRL Entries
This attribute is composed of the following list elements:

CREP
This static attribute contains the local identifier for the related CR for this AR.

AR Type = MS0
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MS0.

Error Action Flag
This static attribute indicates how the reaction of the DP-master (Class 1) in case of failed data transfers in the operation mode OPERATE shall be. The reaction shall be performed according to the following rules:

– If the data transfer to at least one selected (dependent on the attribute Ignore AClr in the List of CRL Entries) DP-slave was erroneous and the Error Action Flag is set to TRUE, the operation mode will change from OPERATE to CLEAR.

– If the Error Action Flag is set to FALSE, the DP-master (Class 1) will remain in the operation mode OPERATE despite erroneous data transfers.

Furthermore this attribute indicates how the transition from the operation mode CLEAR to the operation mode OPERATE shall be performed:

– If the Error Action Flag set to TRUE the transition shall only be performed if the data transfer to the selected DP-slaves has been successfully executed.

– If the Error Action Flag set to FALSE the transition shall be performed independently of a successfully executed data transfer to all activated DP-slaves.

This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Boolean

Allowed values: TRUE, FALSE

Min Slave Interval
This static attribute indicates the smallest allowed period of time between two consecutive DP-slave poll cycles. This ensures that the sequence of cyclic service requests from the DP-master (Class 1) can be handled by the DP-slave. This period of time will be complied by the DP-master (Class 1) at the beginning of each cyclic data transfer. This time interval is not guaranteed for the User Global Control services and for acyclic MS1 AR requests. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 100 μs

Data Control Time
This static attribute indicates the period of time in which a Global Control shall be sent indicating the operation mode of the DP-master (Class 1) and at least one successful interaction shall be executed with every activated DP-slave. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 10

Max User Global Control
This static attribute indicates the maximum number of Global Control requests, which may be submitted by the User before the Global Control confirmations have been received. This attribute describes the ability of the DP-master (Class 1). A practicable value for Max User Global Control is 16. For each of the 8 Slave groups one Sync and one Freeze Command may be started at the same time. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned8

Allowed values: 1 to 255

Max Diag Data Len
This static attribute indicates the maximum length of Diag Data of all assigned DP-slaves. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned8

Allowed values: 6 to 244

Isochronous Mode supp
This static attribute indicates the capability of the DP-master (Class 1) concerning the Isochronous Mode. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Boolean

The allowed values are shown Table 208.

Table 208 – Isochronous mode supp

Value	Meaning
FALSE	Isochronous Mode not supported by the DP-master (Class 1)
TRUE	Isochronous Mode supported by the DP-master (Class 1)

Isochronous Mode
This attribute indicates if the DP-master (Class 1) shall operate in the Isochronous Mode. Two different models concerning the interface between the DP Application Layer and the application of the DP-master (Class 1) itself are provided by fieldbus DP for the DP-master (Class 1), see 6.2.2.4.3. This attribute shall be set accordingly with the Load ARL service.

The allowed values are shown in Table 209.

Table 209 – Isochronous mode

Value	Meaning
Not Synchronized	the DP-master (Class 1) shall not operate in the Isochronous Mode
Buffered Synchronized	the DP-master (Class 1) shall operate in the Buffer Synchronized Isochronous Mode
Enhanced Synchronized	the DP-master (Class 1) shall operate in the Enhanced Synchronized Isochronous Mode

IsoM Freeze
This attribute indicates if the synchronization message (SYNCH) shall be sent with the Control Command Freeze set to TRUE. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Boolean

Allowed values: TRUE, FALSE

IsoM Sync
This attribute indicates if the synchronization message (SYNCH) shall be sent with the Control Command Sync set to TRUE. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Boolean

Allowed values: TRUE, FALSE

T_{CT}
This attribute specifies the time (in units of t_{BIT}) that is necessary to handle a complete Isochronous DP cycle and corresponds to the time T_{DP}. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned32

Allowed values: 1 to $2^{24} - 1$

maxT_{SH}
This attribute specifies the maximum allowed shift (in units of t_{BIT}) of the time T_{DP} measured by the DP-master (Class 1). The application of the DP-master (Class 1) will be informed if the measured Isochronous DP cycle time exceeds this limit by the service SYNCH Delayed indicating the difference. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned8

Allowed values: 1 to 2^8 - 1

AR Type = MS1
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MS1.

Extra Alarm SAP
This static attribute selects the CR for the Alarm Ack service. If the attribute is set to FALSE the DP-master (Class 1) acknowledges alarms via the CR for the read write services. If the attribute is set to TRUE the DP-master (Class 1) acknowledges alarms via a different CR. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Boolean

Allowed values: TRUE, FALSE

Max Channel Data Length
This static attribute indicates the maximum length of the MS1 APDUs (Length parameter and 4 Byte Header) of the corresponding DP-slave. This attribute shall be set accordingly with the Load ARL service.

Attribute Type : Unsigned8

Allowed values: 4 to 244

Alarm Mode
This static attribute indicates the maximum number of possible active alarms of the corresponding DP-slave. This attribute shall be set accordingly with the Load ARL service.

Table 210 shows the allowed values for this attribute.

Table 210 – Alarm mode

Value	Meaning
0	1 alarm of each type
1	2 alarms in total
2	4 alarms in total
3	8 alarms in total
4	12 alarms in total
5	16 alarms in total
6	24 alarms in total
7	32 alarms in total

MS1 Timeout
This attribute contains the value for the Timeout of the acyclic communication on the MS1 AR and specifies the maximum period of time it may take a DP-slave to provide a response. The time contained in this attribute is the maximum time of the assigned DP-slaves.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 10

AR Type = MM1
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MM1.

Poll Timeout
This static attribute indicates the maximum period of time in which the requester shall fetch the response. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 1

Ident Number
This static attribute contains the Ident Number of the DP-master (Class 1). This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

AR Type = MS3
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MS3.

Time Device Type
This static attribute indicates the Device Type concerning the Time ASE. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Unsigned8

The allowed values are shown in Table 211.

Table 211 – Time device type

Value	Meaning
0	Time Receiver
1	Time Master

Clock Sync Interval
This static and conditional attribute contains the value for the clock synchronization interval. It shall only be present if the attribute Time Device Type has the value Time Receiver.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 10

The default value shall be 10 s.

CS Delay Time
This static, conditional and optional attribute contains the transmission delay of the network (line delay, repeater delay, …). It should be included for the calculation of the parameter Time Value of the Set Time service. It shall only be present if the attribute Time Device Type has the value Time Receiver.

Attribute Type: Network Time Difference

The default value shall be 0.

AR Name
This static attribute contains the name of the related ARL Entry. This attribute shall be set accordingly with the Load ARL service.

6.2.10.3.2.3 Invocation of ARL DP-master (class 1) object

For the invocation of the ARL DP-master (Class 1) object the following rules apply:

- Only one ARL DP-master (Class 1) object shall be invoked in a DP-master (Class 1).
- Only one MS0 Entry shall be contained in the List of ARL Entries.
- Only one MS3 Entry shall be contained in the List of ARL Entries.
- Only one MM1 Entry shall be contained in the List of ARL Entries.
- Only one MM2 Entry shall be contained in the List of ARL Entries.
- No MS2 Entry shall be contained in the List of ARL Entries.

6.2.10.3.3 ARL DP-master (class 2) class specification

6.2.10.3.3.1 Template

The ARL DP-master (Class 2) object is described by the following template:

DP ASE:			**AR ASE**
CLASS:			**ARL DP-master (Class 2)**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of ARL Entries
2.1.	(m)	Attribute:	AREP
2.2.	(m)	Attribute:	AR Type
2.3.	(m)	Attribute:	List of related CRL Entries
2.3.1.	(m)	Attribute:	CREP
2.4.	(c)	Constraint:	AR Type = MM1
2.4.1	(m)	Attribute:	Poll Timeout
2.5	(o)	Attribute:	AR Name
SERVICES:			
1.	(m)	MgtService:	Load ARL
2.	(o)	MgtService:	Get ARL

6.2.10.3.3.2 Attributes

Implicit
The attribute Implicit indicates that the ARL DP-master (Class 2) object is implicitly addressed by the service.

List of ARL Entries
This attribute is composed of the following list elements:

AREP
This static attribute contains the local identifier of this AR.

AR Type
This static attribute specifies the type of this AR.

Allowed values: MS0, MS2, MM1, MM2

List of related CRL Entries
This attribute is composed of the following list elements:

CREP
This static attribute contains the local identifier for the related CR for this AR.

AR Type = MM1
This constraint reflects the attributes for this AR if the value of the attribute AR Type is set to MM1.

Poll Timeout
This static attribute indicates the maximum period of time in which the responder shall provide the response. This attribute shall be set accordingly with the Load ARL service.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

Time Base: 1

AR Name
This static attribute contains the name of the related ARL Entry. This attribute shall be set accordingly with the Load ARL service.

6.2.10.3.3.3 Invocation of ARL DP-master (class 2) object

For the invocation of the ARL DP-master (Class 2) object the following rules apply:

– Only one ARL DP-master (Class 2) object shall be invoked in a DP-master (Class 2).
– Only one MM1 Entry shall be contained in the List of ARL Entries.
– Only one MM2 Entry shall be contained in the List of ARL Entries.
– No MS1, MS3 Entry shall be contained in the List of ARL Entries.

6.2.10.4 Communication relationship list class specification

6.2.10.4.1 CRL DP-slave class specification

6.2.10.4.1.1 Template

The CRL DP-slave object is described by the following template:

DP ASE:			**AR ASE**
CLASS:			**CRL DP-slave**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of CRL Entries
2.1.	(m)	Attribute:	CREP
2.2.	(m)	Attribute:	CR Type
2.3.	(m)	Attribute:	Related AREP
2.4.	(m)	Attribute:	S_SAP_index
2.5.	(m)	Attribute:	D_addr
2.6.	(m)	Attribute:	Service_activate
2.7.	(m)	Attribute:	Role_in_Service
2.8.	(c)	Constraint:	Service_activate = SRD or MSRD
2.8.1	(m)	Attribute:	Indication_mode
2.9.	(m)	Attribute:	Max_DLSDU_length_req_low
2.10.	(m)	Attribute:	Max_DLSDU_length_req_high
2.11.	(m)	Attribute:	Max_DLSDU_length_ind_low
2.12.	(m)	Attribute:	Max_DLSDU_length_ind_high
2.13.	(c)	Constraint:	CR Type = BBUU-OM & Role_in_Service = Initiator or Both
2.13.1.	(m)	Attribute:	Publisher Address
2.13.2.	(m)	Attribute:	Input Data Length Publisher
2.13.3.	(m)	Attribute:	Source Offset Publisher
2.13.4.	(m)	Attribute:	Length
2.14.	(o)	Attribute:	CR Name

SERVICES:

1. (m) MgtService: Load CRL
2. (o) MgtService: Get CRL
3. (o) MgtService: Load CRL DXB Link Entries

6.2.10.4.1.2 Attributes

Implicit
The attribute Implicit indicates that the CRL DP-slave object is implicitly addressed by the service.

List of CRL Entries
This attribute is composed of the following list elements:

CREP
This static attribute contains the local identifier of this CR. This attribute shall be set accordingly with the Load CRL service.

CR Type
This static attribute indicates the type of this CR. This attribute is dependent of the AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service. For each AR Type the correspondent CRL entries shall be set.

Table 212 shows the allowed values for this attribute.

Related AREP
This static attribute contains the local identifier for the related AR for this CR. This attribute shall be set accordingly with the Load CRL service.

S_SAP_index
This static attribute contains the identifier for the related Service Access Point of the local data-link layer. The value for this attribute has to be set correspondent to the attributes CR Type of this CR and to the attribute AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service. Table 212 shows the allowed values for this attribute.

Table 212 – S_SAP_index

AR Type	CR Type	S_SAP_index	Remark
MS0	BBUB	NIL	Mandatory for the MS0 AR
MS0	BBUU-OM	NIL	Optional for the MS0 AR
MS0	BQUU	61, 62	Mandatory for the MS0 AR
MS0	BQUU	53	Optional for the MS0 AR
MS0	BBUU	60	Mandatory for the MS0 AR
MS0	QQUU-OM	58	Mandatory for the MS0 AR
MS0	QBCB	55, 56, 57, 59, 60	Mandatory for the MS0 AR
MS1	QQCB-CO	50	Optional for the MS1 AR
MS1	QQCB-CO	51	Mandatory for the MS1 AR
MS1	QQUU-CO	60	Mandatory for the MS1 AR
MS2	QQCB-CO	0 to 48	Mandatory for each MS2 AR
MS2	QQUU-CO	49	Mandatory for the MS2 AR
MS3	QQUU-OM	CS	Mandatory for the MS3 AR

D_addr
This dynamic attribute contains the DL-address of the remote DP-master or Publisher. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and S_SAP_index of this CR and to the attribute AR Type of the related AR. Table 213 shows the allowed values for this attribute used with the Load CRL

service. Furthermore this attribute is influenced by the AR state machine dependent of the attribute CR Type of this CR and of the attribute AR Type of the related AR. Table 213 shows the allowed values for this attribute for the modification of the AR state machine.

Table 213 – D_addr

AR Type	CR Type	S_SAP_index	D_addr as loaded with the Load CRL service	D_addr modified by the AR state machine
MS0	BBUB	NIL	None	0 to 125
MS0	BBUU-OM	NIL	None	All
MS0	BQUU	61, 62, 53	All	-
MS0	BBUU	60	All	-
MS0	QQUU-OM	58	None	0 to 125
MS0	QBCB	55, 59, 60	All	-
MS0	QBCB	56, 57	None	All
MS1	QQCB-CO	50, 51	None	0 to 125
MS1	QQUU-CO	60	All	-
MS2	QQCB-CO	0 to 48	None	0 to 125
MS2	QQUU-CO	49	All	-
MS3	QQUU-OM	CS	All	-

None	means that the local Service Access Point shall not be activated.
All	means that the local Service Access Point shall be activated with no access protection.
0 to 125	means that the local Service Access Point shall be activated with the corresponding access protection.
-	means that the AR state machines shall not modify the attribute D_addr.

Service_activate

This static attribute indicates the DL services which shall be activated for the S_SAP_index of this CRL entry.

The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and S_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 214 shows the allowed values for this attribute.

Table 214 – Service_activate

AR Type	CR Type	S_SAP_index	Service_activate
MS0	BBUB	NIL	(M)SRD
MS0	BBUU-OM	NIL	MSRD
MS0	BQUU	61, 62, 53	SRD
MS0	BBUU	60	SRD
MS0	QQUU-OM	58	SDN
MS0	QBCB	55, 56, 57, 59, 60	SRD
MS1	QQCB-CO	50	SRD
MS1	QQCB-CO	51	SRD
MS1	QQUU-CO	60	SRD
MS2	QQCB-CO	0 to 48	SRD
MS2	QQUU-CO	49	SRD
MS3	QQUU-OM	CS	CS Time Event CS Clock Value

Role_in_service
This dynamic attribute indicates the role which shall be activated for the S_SAP_index of this CRL entry. The following values are specified:

– Initiator: The station initiates the respective service exclusively.
– Responder: The station response to the service exclusively.
– Subscriber: The station receives the reply DLPDU of this service exclusively.
– Both: The station initiates the service and response to the service.
– Responder/Subscriber: The station responds to the service and receives the reply DLPDU of this service.

The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR.

Table 215 shows the allowed values for this attribute.

Table 215 – Role_in_service

AR Type	CR Type	S_SAP_index	Role_in_service as loaded with the Load CRL service	Role_in_service modified by the AR state machine
MS0	BBUB	NIL	None	Responder
MS0	BBUU-OM	NIL	None	Responder, Subscriber, Responder/Subscriber
MS0	BQUU	61, 62, 53	Responder	-
MS0	BBUU	60	Responder	-
MS0	QQUU-OM	58	None	Responder
MS0	QBCB	55, 59, 60	Responder	-
MS0	QBCB	56, 57	None	Responder
MS1	QQCB-CO	50, 51	None	Responder
MS1	QQUU-CO	60	Responder	-
MS2	QQCB-CO	0 to 48	None	Responder
MS2	QQUU-CO	49	Responder	-
MS3	QQUU-OM	CS	Responder	-

None means that the local Service Access Point shall not be activated

- means that the AR state machines shall not modify the attribute Role_in_service.

Indication_mode

This static attribute shall only be present if the attribute Service_activate of this CRL entry has the value SRD or MSRD. It indicates the Indication_mode which the DLL shall use to convey the indications of the (M)SRD service. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and S_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 216 shows the allowed values for this attribute.

Table 216 – Indication_mode

AR Type	CR Type	S_SAP_index	Role_in_service	Indication_mode
MS0	BBUB	NIL	Responder	All
MS0	BBUU-OM	NIL	Subscriber	—
MS0	BBUU-OM	NIL	Responder, Responder/Subscriber	All
MS0	BQUU	61, 62, 53	Responder	All
MS0	BBUU	60	Responder	All
MS0	QBCB	55, 56, 57, 59	Responder	Data
MS0	QBCB	60	Responder	All
MS1	QQCB-CO	50, 51	Responder	Data
MS1	QQUU-CO	60	Responder	All
MS2	QQCB-CO	0 to 48	Responder	Data
MS2	QQUU-CO	49	Respondr	Data

- means that the attribute Indication_mode is not present for this Role_in_service.

Max_DLSDU_length_req_low
This static attribute indicates the maximum length for the DLSDU with low priority as requester. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and S_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 217 shows the allowed values for this attribute.

Table 217 – Max_DLSDU_length_req_low

AR Type	CR Type	S_SAP_index	Role_in_service	Max_DLSDU_length_req_low
MS0	BBUB	NIL	Responder	0 to 244
MS0	BBUU-OM	NIL	Subscriber	0
MS0	BBUU-OM	NIL	Responder, Responder/Subscriber	0 to 244
MS0	BQUU	61, 62, 53	Responder	0
MS0	BBUU	60	Responder	6 to 244
MS0	QBCB	55	Responder	0
MS0	QBCB	56, 57, 59	Responder	0 to 244
MS0	QQUU-OM	58	Responder	0
MS1	QQCB-CO	50	Responder	4
MS1	QQCB-CO	51	Responder	4 to 244
MS1	QQUU-CO	60	Responder	6 to 244
MS2	QQCB-CO	0 to 48	Responder	48 to 244
MS2	QQUU-CO	49	Responder	4
MS3	QQUU-OM	CS	Responder	0

Max_DLSDU_length_req_high
This static attribute indicates the maximum length for the DLSDU with high priority as requester. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and S_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 218 shows the allowed values for this attribute.

Table 218 – Max_DLSDU_length_req_high

AR Type	CR Type	S_SAP_index	Role_in_service	Max_DLSDU_length_req_high
MS0	BBUB	NIL	Responder	0 to 244
MS0	BBUU-OM	NIL	Subscriber	0
MS0	BBUU-OM	NIL	Responder, Responder/Subscriber	0 to 244
MS0	BQUU	61, 62, 53	Responder	0
MS0	BBUU	60	Responder	0
MS0	QBCB	55, 56, 57, 59	Responder	0
MS0	QQUU-OM	58	Responder	0
MS1	QQCB-CO	50, 51	Responder	0
MS1	QQUU-CO	60	Responder	0
MS2	QQCB-CO	0 to 48	Responder	0
MS2	QQUU-CO	49	Responder	0
MS3	QQUU-CO	CS	Responder	0

Max_DLSDU_length_ind_low
This static attribute indicates the maximum length for the DLSDU with low priority for indication. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and S_SAP_index and Role_in_service of this CR and to the attribute AR Type of the related AR.

Table 219 shows the allowed values for this attribute.

Table 219 – Max_DLSDU_length_ind_low

AR Type	CR Type	S_SAP_index	Role_in_service	Max_DLSDU_length_ind_low
MS0	BBUB	NIL	Responder	0
MS0	BBUU-OM	NIL	Subscriber, Responder/Subscriber	244
MS0	BBUU-OM	NIL	Responder	0
MS0	BQUU	61, 62, 53	Responder	0
MS0	BBUU	60	Responder	0
MS0	QBCB	55	Responder	0
MS0	QBCB	56, 57, 59	Responder	0
MS0	QQUU-OM	58	Responder	0
MS1	QQCB-CO	50	Responder	4
MS1	QQCB-CO	51	Responder	4 to 244
MS1	QQUU-CO	60	Responder	0
MS2	QQCB-CO	0 to 48	Responder	4 to 244
MS2	QQUU-CO	49	Responder	48 to 64
MS3	QQUU-OM	CS	Responder	18

Max_DLSDU_length_ind_high
This static attribute indicates the maximum length for the DLSDU with high priority for indication. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and S_SAP_index and Role_in_service of this CR and to the attribute AR Type of the related AR.

Table 220 shows the allowed values for this attribute.

Table 220 – Max_DLSDU_length_ind_high

AR Type	CR Type	S_SAP_index	Role_in_service	Max_DLSDU_length_ind_high
MS0	BBUB	NIL	Responder	0 to 244
MS0	BBUU-OM	NIL	Subscriber, Responder/Subscriber	244
MS0	BBUU-OM	NIL	Responder	0
MS0	BQUU	61	Responder	7 to 244
MS0	BQUU	62	Responder	1 to 244
MS0	BQUU	53	Responder	5 to 244
MS0	BBUU	60	Responder	0
MS0	QBCB	55	Responder	4 to 244
MS0	QBCB	56, 57, 59	Responder	0
MS0	QQUU-OM	58	Responder	2
MS1	QQCB-CO	50, 51	Responder	0
MS1	QQUU-CO	60	Responder	0
MS2	QQCB-CO	0 to 48	Responder	0
MS2	QQUU-CO	49	Responder	0
MS3	QQUU-OM	CS	Responder	0

CR Type = BBUU-OM & Role_in_Service = Initiator or Both
This constraint reflects the attributes for this CR if the value of the attribute CR Type is set to BBUU-OM and the attribute Role_in_Service is set to Subscriber or Responder/Subscriber.

Publisher Address
This dynamic attribute indicates the DL-address of a DP-slave (Publisher) the Subscriber shall subscribe from. The following rules shall apply for this attribute:

A value between 0 and 125 represents a valid Publisher address and therefore an active DXB-Link. The value 127 indicates that this DXB-Link Entry is no longer active.

This attribute shall be set to the value 127 with the Load CRL service. After reception of the DXB-Linktable with the Check User Prm or Check Ext User Prm service, the User shall set this attribute accordingly with the Load CRL DXB Link Entries service.

The MS0 AR state machine set shall set this attribute to the value 127 if the DP-slave acting as Subscriber leaves the data exchange mode.

Attribute Type: Unsigned8

Allowed values: 0 to 125, 127

Input Data Length Publisher
This static attribute indicates the number of octets of the Simple or Extended Input Data object of the related Publisher. This attribute shall only be evaluated by the MS0 AR state machine if the value of the attribute Publisher Address is not equal to 127.

This attribute shall be set with the Load CRL service. After reception of the DXB-Linktable with the Check User Prm or Check Ext User Prm service, the User shall set this attribute accordingly with the Load CRL DXB Link Entries service.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Source Offset Publisher
This static attribute indicates the beginning (offset) of the subscription within the value of the Simple or Extended Input Data object of the related Publisher given with the attribute Publisher Address. This attribute shall only be evaluated by the MS0 AR state machine if the value of the attribute Publisher Address is not equal to 127.

This attribute shall be set with the Load CRL service. After reception of the DXB-Linktable with the Check User Prm or Check Ext User Prm service, the User shall set this attribute accordingly with the Load CRL DXB Link Entries service.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Length
This static attribute indicates the number of octets the Subscriber shall subscribe from the related Publisher starting at the offset given in the attribute Source Offset Publisher. This attribute shall only be evaluated by the MS0 AR state machine if the value of the attribute Publisher Address is not equal to 127.

This attribute shall be set with the Load CRL service. After reception of the DXB-Linktable with the Check User Prm or Check Ext User Prm service, the User shall set this attribute accordingly with the Load CRL DXB Link Entries service.

Attribute Type: Unsigned8

Allowed values: 1 to 244

CR Name
This static attribute contains the name of the related CRL Entry. This attribute shall be set accordingly with the Load CRL service.

6.2.10.4.1.3 Invocation of CRL DP-slave object

For the invocation of the CRL DP-slave object the following rule has to be considered:

Only one CRL DP-slave object shall be invoked in a DP-slave.

6.2.10.4.2 CRL DP-master (class 1) class specification

6.2.10.4.2.1 Template

The CRL DP-master (Class 1) object is described by the following template:

DP ASE:			**AR ASE**
CLASS:			**CRL DP-master (Class 1)**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of CRL Entries
2.1.	(m)	Attribute:	CREP
2.2.	(m)	Attribute:	CR Type
2.3.	(m)	Attribute:	Related AREP
2.4.	(m)	Attribute:	S_SAP_index
2.5.	(m)	Attribute:	D_SAP_index
2.6.	(m)	Attribute:	D_addr
2.7.	(m)	Attribute:	Service_activate
2.8.	(m)	Attribute:	Role_in_service
2.9.	(c)	Constraint:	Service_activate = SRD or MSRD
2.9.1.	(m)	Attribute:	Indication_mode
2.10.	(m)	Attribute:	Max_DLSDU_length_req_low
2.11.	(m)	Attribute:	Max_DLSDU_length_req_high
2.12.	(m)	Attribute:	Max_DLSDU_length_ind_low

2.13. (m) Attribute: Max_DLSDU_length_ind_high
2.14. (c) Constraint: CR Type = BQUU & D_SAP_index = 61
2.14.1. (m) Attribute: Active
2.14.2. (m) Attribute: New Prm
2.14.3. (m) Attribute: Prm Data
2.14.3.1. (m) Attribute: Lock
2.14.3.2. (m) Attribute: Sync
2.14.3.3. (m) Attribute: Freeze
2.14.3.4. (m) Attribute: WD On
2.14.3.5. (m) Attribute: WD Fact 1
2.14.3.6. (m) Attribute: WD Fact 2
2.14.3.7. (m) Attribute: Min Tsdr
2.14.3.8. (m) Attribute: Ident Number
2.14.3.9. (m) Attribute: Group Ident
2.14.3.10. (o) Attribute: DPV1 Enable
2.14.3.11. (o) Attribute: Fail Safe
2.14.3.12. (o) Attribute: Enable Publisher
2.14.3.13. (o) Attribute: WD Base
2.14.3.14. (o) Attribute: Enable Pull Plug Alarm
2.14.3.15. (o) Attribute: Enable Process Alarm
2.14.3.16. (o) Attribute: Enable Diagnosis Alarm
2.14.3.17. (o) Attribute: Enable Manufacturer Specific Alarm
2.14.3.18. (o) Attribute: Enable Status Alarm
2.14.3.19. (o) Attribute: Enable Update Alarm
2.14.3.20. (o) Attribute: Check Cfg Mode
2.14.3.21. (o) Attribute: Alarm Mode
2.14.3.22. (m) Attribute: Prm Structure
2.14.3.23. (c) Constraint: Prm Structure = FALSE
2.14.3.23.1 (m) Attribute: User Prm Data
2.14.3.23.1.1. (o) Attribute: Device Prm Data
2.14.3.23.1.2. (o) Attribute: Module Prm Data
2.14.3.24. (c) Constraint: Prm Structure = TRUE
2.14.3.24.1. (m) Attribute: List of Structured Prm Data
2.14.3.24.1.1. (o) Attribute: Structured User Prm Data
2.14.3.24.1.1.1. (o) Attribute: Device Related User Parameter
2.14.3.24.1.1.1.1. (m) Attribute: Device Prm Data
2.14.3.24.1.1.2. (o) Attribute: Module Related User Parameter
2.14.3.24.1.1.2.1. (m) Attribute: Slot Number
2.14.3.24.1.1.2.2. (m) Attribute: Module Prm Data
2.14.3.24.1.2. (o) Attribute: List of DXB Link Entries
2.14.3.24.1.2.1. (m) Attribute: Publisher Address
2.14.3.24.1.2.2. (m) Attribute: Input Data Length Publisher
2.14.3.24.1.2.3. (m) Attribute: Source Offset Publisher
2.14.3.24.1.2.4. (m) Attribute: Length
2.14.3.24.1.3. (o) Attribute: IsoM Parameter
2.14.3.24.1.3.1. (m) Attribute: T_{BASE_DP}
2.14.3.24.1.3.2. (m) Attribute: T_{DP}
2.14.3.24.1.3.3. (m) Attribute: T_{MAPC}
2.14.3.24.1.3.4. (m) Attribute: T_{BASE_IO}
2.14.3.24.1.3.5. (m) Attribute: T_{I}
2.14.3.24.1.3.6. (m) Attribute: T_{O}
2.14.3.24.1.3.7. (m) Attribute: T_{PLL_W}
2.14.3.24.1.3.8. (m) Attribute: T_{PLL_D}
2.14.3.24.1.4. (o) Attribute: List of DXB Subscriber Entries
2.14.3.24.1.4.1. (s) Attribute: Publisher Address
2.14.3.24.1.4.2.1. (m) Attribute: Input Data Length Publisher
2.14.3.24.1.4.2.2. (m) Attribute: Source Offset Publisher
2.14.3.24.1.4.3. (s) Attribute: Master Data
2.14.3.24.1.4.3.1. (m) Attribute: Source Offset Master
2.14.3.24.1.4.4. (m) Attribute: Dest Slot Number
2.14.3.24.1.4.5. (m) Attribute: Offset Data Area
2.14.3.24.1.4.6. (m) Attribute: Length Data Area
2.14.3.24.1.5. (o) Attribute: Prm Command
2.14.3.24.1.6. (c) Constraint: Prm Command supported = TRUE
2.14.3.24.1.6.1. (m) Attribute: Seq Number
2.14.3.24.1.6.2. (m) Attribute: Primary Request
2.14.3.24.1.6.3. (m) Attribute: MS1 Command
2.14.3.24.1.6.4. (m) Attribute: Master State Clear

2.14.3.24.1.6.5. (m) Attribute: Check Properties
2.14.3.24.1.6.5.1. (m) Attribute: Primary Request used
2.14.3.24.1.6.5.2. (m) Attribute: MS1 Command used
2.14.3.24.1.6.5.3. (m) Attribute: Address Change
2.14.3.24.1.6.5.4. (m) Attribute: Address Offset
2.14.3.24.1.6.6. (m) Attribute: Output Hold Time
2.14.3.24.1.7. (o) Attribute: Time AR Parameter
2.14.3.24.1.7.1. (m) Attribute: Clock Sync Interval
2.14.3.24.1.7.2. (m) Attribute: CS Delay Time
2.14.3.25. (o) Attribute: IsoM Req
2.15. (c) Constraint: CR Type = BQUU & D_SAP_index = 62
2.15.1. (m) Attribute: List of Cfg Data
2.15.1.1. (s) Attribute: Cfg Identifier
2.15.1.1.1. (m) Attribute: Type
2.15.1.1.2. (m) Attribute: Format
2.15.1.1.3. (m) Attribute: Consistency
2.15.1.1.4. (m) Attribute: Length
2.15.1.2. (s) Attribute: Special Cfg Identifier
2.15.1.2.1. (m) Attribute: Type
2.15.1.2.2. (c) Constraint: Type = Input or Input-Output
2.15.1.2.2.1. (m) Attribute: Input Length
2.15.1.2.2.2. (m) Attribute: Input Format
2.15.1.2.2.3. (m) Attribute: Input Consistency
2.15.1.2.3. (c) Constraint: Type = Output or Input-Output
2.15.1.2.3.1. (m) Attribute: Output Length
2.15.1.2.3.2. (m) Attribute: Output Format
2.15.1.2.3.3. (m) Attribute: Output Consistency
2.15.1.2.4. (o) Attribute: List of Data Description
2.15.1.2.4.1. (m) Attribute: Data Description
2.15.1.2.5. (o) Attribute: Manufacturer Specific Data
2.16. (c) Constraint: CR Type = BBUU & D_SAP_index = 60
2.16.1. (m) Attribute: Diag Upd Delay
2.16.2. (m) Attribute: Ext Diag Overflow
2.16.3. (m) Attribute: Deactivated
2.16.4. (m) Attribute: Station Non Existent
2.16.5. (m) Attribute: Invalid Slave Response
2.17. (c) Constraint: CR Type = BBUB & D_SAP_index = NIL
2.17.1. (m) Attribute: Fail Safe
2.17.2. (m) Attribute: DPV1 Supported
2.17.3. (m) Attribute: NA to Abort
2.17.4. (m) Attribute: Ignore ACI
2.17.5. (m) Attribute: Output Len
2.17.6. (m) Attribute: Exp Inp Len
2.17.7. (m) Attribute: Data Transfer Status
2.17.8. (m) Attribute: System Diagnosis Status
2.17.9. (m) Attribute: Publisher Flag
2.18. (c) Constraint: CR Type = BQUU & D_SAP_index = 53
2.18.1. (m) Attribute: List of Structured Prm Data
2.18.1.1. (o) Attribute: Structured User Prm Data
2.18.1.1.1. (o) Attribute: Device Related User Parameter
2.18.1.1.1. (m) Attribute: Device Prm Data
2.18.1.1.2. (o) Attribute: Module Related User Parameter
2.18.1.1.2.1. (m) Attribute: Slot Number
2.18.1.1.2.2. (m) Attribute: Module Prm Data
2.18.1.2. (o) Attribute: List of DXB Link Entries
2.18.1.2.1. (m) Attribute: Publisher Address
2.18.1.2.2. (m) Attribute: Input Data Length Publisher
2.18.1.2.3. (m) Attribute: Source Offset Publisher
2.18.1.2.4. (m) Attribute: Length
2.18.1.3. (o) Attribute: IsoM Parameter
2.18.1.3.1. (m) Attribute: T_{BASE_DP}
2.18.1.3.2. (m) Attribute: T_{DP}
2.18.1.3.3. (m) Attribute: T_{MAPC}
2.18.1.3.4. (m) Attribute: T_{BASE_IO}
2.18.1.3.5. (m) Attribute: T_{I}
2.18.1.3.6. (m) Attribute: T_{O}
2.18.1.3.7. (m) Attribute: T_{PLL_W}
2.18.1.3.8. (m) Attribute: T_{PLL_D}

2.18.1.4. (o) Attribute: List of DXB Subscriber Entries
2.18.1.4.1. (s) Attribute: Publisher Address
2.18.1.4.2.1. (m) Attribute: Input Data Length Publisher
2.18.1.4.2.2. (m) Attribute: Source Offset Publisher
2.18.1.4.3. (s) Attribute: Master Data
2.18.1.4.3.1. (m) Attribute: Source Offset Master
2.18.1.4.4. (m) Attribute: Dest Slot Number
2.18.1.4.5. (m) Attribute: Offset Data Area
2.18.1.4.6. (m) Attribute: Length Data Area
2.18.1.5. (o) Attribute: Time AR Parameter
2.18.1.5.1. (s) Attribute: Clock Sync Interval
2.18.1.5.2. (s) Attribute: CS Delay Time
2.19. (o) Attribute: CR Name

SERVICES:

1. (m) MgtService: Load CRL
2. (o) MgtService: Get CRL
3. (m) MgtService: CRL Slave Activate
4. (m) MgtService: CRL Slave New Prm
5. (m) MgtService: CRL Slave New Prm Data

6.2.10.4.2.2 Attributes

Implicit
The attribute Implicit indicates that the CRL DP-master (Class 1) object is implicitly addressed by the service.

List of CRL Entries
This attribute is composed of the following list elements:

CREP
This static attribute contains the local identifier of this CR. This attribute shall be set accordingly with the Load CRL service.

CR Type
This static attribute indicates the type of this CR. This attribute is dependent on the AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service. For each AR Type the correspondent CRL entries shall be set.

Table 221 shows the allowed values for this attribute.

Related AREP
This static attribute contains the local identifier for the related AR for this CR. This attribute shall be set accordingly with the Load CRL service.

S_SAP_index
This static attribute contains the identifier for the related Service Access Point of the local Data Link Layer. The value for this attribute has to be set correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service.

Table 221 shows the allowed values for this attribute.

Table 221 – S_SAP_index

AR Type	CR Type	S_SAP_index	Remark
MS0	BBUB	NIL	Mandatory for the MS0 AR
MS0	BQUU	62	Mandatory for the MS0 AR
MS0	BBUU	62	Mandatory for the MS0 AR
MS0	QQUU-OM	62	Mandatory for the MS0 AR
MS1	QQCB-CO	51	Mandatory for the MS1 AR
MS1	QQUU-OO	62	Mandatory for the MS1 AR
MS3	QQUU-OM	CS	Mandatory for the MS3 AR
MM1	QQCB-CL	54	Mandatory for the MM1 AR
MM2	QQUU-OM	54	Mandatory for the MM2 AR

D_SAP_index

This static attribute contains the identifier for the related Service Access Point of the remote Data Link Layer. The value for this attribute has to be set correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service. Table 222 shows the allowed values for this attribute.

Table 222 – D_SAP_index

AR Type	CR Type	D_SAP_index	Remark
MS0	BBUB	NIL	Mandatory for the MS0 AR
MS0	BQUU	61, 62	Mandatory for the MS0 AR
MS0	BQUU	53	Optional for the MS0 AR
MS0	BBUU	60	Mandatory for the MS0 AR
MS0	QQUU-OM	58	Mandatory for the MS0 AR
MS1	QQCB-CO	50	Optional for each MS1 AR
MS1	QQCB-CO	51	Mandatory for the MS1 AR
MS1	QQUU-OO	60	Mandatory for the MS1 AR
MS3	QQUU-OM	CS	Mandatory for the MS3 AR
MM1	QQCB-CL	54	Mandatory for the MM1 AR
MM2	QQUU-OM	54	Mandatory for the MM2 AR

D_addr

This dynamic attribute contains the DL-address of the remote communication partner. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR. Table 223 shows the allowed values for this attribute used with the Load CRL service. Furthermore this attribute is influenced by the AR state machine dependent of the attribute CR Type of this CR and of the attribute AR Type of the related AR. Table 223 shows the allowed values for this attribute for the modification of the AR sate machine.

Table 223 – D_addr

AR Type	CR Type	S_SAP_index	Time Device Type (ARL)	D_addr as loaded with the Load CRL service	D_addr modified by the AR state machine
MS0	BBUB	NIL	—	0 to 126	—
MS0	BQUU	62	—	0 to 126	—
MS0	BBUU	62	—	0 to 126	—
MS0	QQUU-OM	62	—	0 to 127	—
MS1	QQCB-CO	51	—	0 to 126	—
MS1	QQUU-CO	62	—	0 to 126	—
MS3	QQUU-OM	CS	Time Master	127	—
MS3	QQUU-OM	CS	Time Receiver	None	All
MM1	QQCB-CL	54	—	All	0 to 125
MM2	QQUU-OM	54	—	All	0 to 125

All	means that the local Service Access Point shall be activated with no access protection
None	means that the local Service Access Point shall not be activated
0 to 127	means that the attribute D_addr of the local Service Access Point shall be set with the corresponding remote DL-address
-	means that the AR state machines shall not modify the attribute D_addr
--	means that the value of this attribute is not relevant

Service_activate
This static attribute indicates the DL services which shall be activated for the S_SAP_index of this CRL entry.

The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR.

Table 224 shows the allowed values for this attribute.

Table 224 – Service_activate

AR Type	CR Type	S_SAP_index	Publisher Flag	Service_activate
MS0	BBUB	NIL	FALSE	SRD
MS0	BBUB	NIL	TRUE	MSRD
MS0	BQUU	62	—	SRD
MS0	BBUU	62	—	SRD
MS0	QQUU-OM	62	—	SDN
MS1	QQCB-CO	51	—	SRD
MS1	QQUU-CO	62	—	SRD
MS3	QQUU-OM	CS	—	CS Time Event CS Clock Value
MM1	QQCB-CL	54	—	SRD
MM2	QQUU-OM	54	—	SDN

Role_in_service
This static attribute indicates the role which shall be activated for the S_SAP_index of this CRL entry. The following values are specified:

– Initiator: The station initiates the respective service exclusively.

– Responder: The station response to the service exclusively.

The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR.

Table 225 shows the allowed values for this attribute.

Table 225 – Role_in_service

AR Type	CR Type	S_SAP_index	Time Device Type (ARL)	Role_in_service
MS0	BBUB	NIL	—	Initiator
MS0	BQUU	62	—	Initiator
MS0	BBUU	62	—	Initiator
MS0	QQUU-OM	62	—	Initiator
MS1	QQCB-CO	51	—	Initiator
MS1	QQUU-CO	62	—	Initiator
MS3	QQUU-OM	CS	Time Master	Initiator
MS3	QQUU-OM	CS	Time Receiver	Responder
MM1	QQCB-CL	54	—	Responder
MM2	QQUU-OM	54	—	Responder

Indication_mode

This conditional and static attribute shall only be present if the attribute Service_activate of this CRL entry has the value SRD or MSRD and if the attribute Role_in_service of this CRL entry has the value Responder. It indicates the Indication_mode which the DLL shall use to convey the indications of the (M)SRD service. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR.

Table 226 shows the allowed values for this attribute.

Table 226 – Indication_mode

AR Type	CR Type	S_SAP_index	Indication_mode
MS0	BBUB	NIL	—
MS0	BQUU	62	—
MS0	BBUU	62	—
MS0	QQUU-OM	62	—
MS1	QQCB-CO	51	—
MS1	QQUU-CO	62	—
MM1	QQCB-CL	54	Data
MM2	QQUU-OM	54	—
- means that the attribute Indication_mode is not present for this CR Type.			

Max_DLSDU_length_req_low

This static attribute indicates the maximum length for the DLSDU with low priority as requester/responder. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 227 shows the allowed values for this attribute.

Table 227 – Max_DLSDU_length_req_low

AR Type	CR Type	D_SAP_index	Time Device Type (ARL)	Max_DLSDU_length_req_low
MS0	BBUB	NIL	—	0
MS0	BQUU	61, 62, 53	—	0
MS0	BBUU	60	—	0
MS0	QQUU-OM	58	—	0
MS1	QQCB-CO	50	—	4
MS1	QQCB-CO	51	—	4 to 244
MS1	QQUU-OO	60	—	0
MS3	QQUU-OM	CS	Time Master	17
MS3	QQUU-OM	CS	Time Receiver	0
MM1	QQCB-CL	54	—	4 to 244
MM2	QQUU-OM	54	—	0

Max_DLSDU_length_req_high
This static attribute indicates the maximum length for the DLSDU with high priority as requester/responder. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 228 shows the allowed values for this attribute.

Table 228 – Max_DLSDU_length_req_high

AR Type	CR Type	D_SAP_index	Max_DLSDU_length_req_high
MS0	BBUB	NIL	0 to 244
MS0	BQUU	61	7 to 244
MS0	BQUU	62	1 to 244
MS0	BQUU	53	5 to 244
MS0	BBUU	60	0
MS0	QQUU-OM	58	2
MS1	QQCB-CO	50, 51	0
MS1	QQUU-OO	60	0
MS3	QQUU-OM	CS	0
MM1	QQCB-CL	54	0
MM2	QQUU-OM	54	0

Max_DLSDU_length_ind_low
This static attribute indicates the maximum length for the DLSDU with low priority for indication. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 229 shows the allowed values for this attribute.

Table 229 – Max_DLSDU_length_ind_low

AR Type	CR Type	D_SAP_index	Time Device Type (ARL)	Max_DLSDU_length_ind_low
MS0	BBUB	NIL	—	0 to 244
MS0	BQUU	61,62, 53	—	0
MS0	BBUU	60	—	6 to 244
MS0	QQUU-OM	58	—	0
MS1	QQCB-CO	50	—	4
MS1	QQCB-CO	51	—	4 to 244
MS1	QQUU-OO	60	—	6 to 244
MS3	QQUU-OM	CS	Time Master	0
MS3	QQUU-OM	CS	Time Receiver	17
MM1	QQCB-CL	54	—	4 to 244
MM2	QQUU-OM	54	—	2

Max_DLSDU_length_ind_high
This static attribute indicates the maximum length for the DLSDU with high priority for indication. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 230 shows the allowed values for this attribute.

Table 230 – Max_DLSDU_length_ind_high

AR Type	CR Type	D_SAP_index	Max_DLSDU_length_ind_high
MS0	BBUB	NIL	0 to 244
MS0	BQUU	61,62, 53	0
MS0	BBUU	60	0
MS0	QQUU-OM	58	0
MS1	QQCB-CO	50, 51	0
MS1	QQUU-OO	60	0
MS3	QQUU-OM	CS	0
MM1	QQCB-CL	54	0
MM2	QQUU-OM	54	0

CR Type = BQUU & D_SAP_index = 61
This constraint reflects the attributes for this CR if the value of the attribute CR Type is set to BQUU and the attribute D_SAP_index is set to 61.

Active
This dynamic attribute indicates the state of interactions to the correspondent DP-slave. The attribute set to FALSE indicates that the DP-master (Class 1) shall stop the interactions to the correspondent DP-slave. In this state it is allowed to change the CRL entries of this DP-slave with the Load CRL service. If this attribute is set to TRUE the DP-master (Class 1) shall start the interactions to the correspondent DP-slave. In this state only the attribute Prm Data may be changed with the CRL Slave New Prm Data service (see also the attribute New Prm). This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave Activate service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

New Prm
This dynamic attribute indicates to the MS0 AR state machine to send new parameterization data during the data exchange mode to the correspondent DP-slave. The following rules shall be applied:

If the attribute is set to FALSE the DP-master (Class 1) transfers the input, output data to the correspondent DP-slave. If this attribute is set to TRUE and the attribute DPV1 Supported of the related CRL entry has the value FALSE the DP-master (Class 1) transfers for one DP-slave poll cycle new parameterisation data to the correspondent DP-slave. The MS0 AR state machine sets the attribute to FALSE if the transfer was successful.

This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Prm Data
This dynamic attribute is an aggregate of the parameterization data of the correspondent DP-slave and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Lock
This dynamic attribute is used by the MS0 AR state machine to assign or release the correspondent DP-slave. This attribute is set to LOCK if the attribute Active of this related CRL entry is set to TRUE. This attribute is set to UNLOCK if the attribute Active of this related CRL entry is set to FALSE, or due to failures during the data exchange mode. This attribute shall be set to LOCK with the Load CRL service. This attribute is further influenced by the MS0 AR state machine. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: LOCK, UNLOCK

Sync
This dynamic attribute indicates to the correspondent DP-slave that it shall participate in the synchronization of the Output Data (Sync mode). The DP-slave shall check whether it supports the requested functionality and if the DP-slave does not support this function it shall respond with an error entry in the Slave Diagnosis (Not Supported). This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Table 231 shows the allowed values for this attribute.

Table 231 – Sync

Value	Meaning
FALSE	Sync ability not requested for this DP-slave
TRUE	Sync ability requested for this DP-slave

Freeze
This dynamic attribute indicates to the correspondent DP-slave that it shall participate in the synchronization of the Input Data (Freeze mode). The DP-slave shall check whether it supports the requested functionality and if the DP-slave does

not support this function it shall respond with an error entry in the Slave Diagnosis (Not Supported). This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Table 232 shows the allowed values for this attribute.

Table 232 – Freeze

Value	Meaning
FALSE	Freeze ability not requested for this DP-slave
TRUE	Freeze ability requested for this DP-slave

WD On
This dynamic attribute indicates if the watchdog control shall be activated at the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

WD Fact 1, WD Fact 2
These dynamic attributes contain the values for the monitoring of the DP-master (Class 1) at the MS0 AR (watchdog control (T_{WD})) at the correspondent DP-slave. The watchdog control in a DP-slave takes care that, if the DP-master fails, the outputs will be set in the safe state after the expiration of this time. The time is calculated according to equation:

T_{WD} = Watchdog Time Base × WD Fact 1 × WD Fact 2

If the attribute DPV1 Supported has the value FALSE the Watchdog Time Base shall be set to 10 ms.

If the attribute DPV1 Supported has the value TRUE the Watchdog Time Base shall be set to the value 1if the attribute WD Base has the value TRUE. If the attribute DPV1 Supported has the value TRUE the Watchdog Time Base shall be set to the value 10if the attribute WD Base has the value FALSE. These attributes shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: 1 to 255

Min TSDR
This dynamic attribute contains the value for the Minimum Station Delay Responder time of the DLL of the correspondent DP-slave. If this attribute is equal zero, the previous value of the Min TSDR remains unchanged. These attributes shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: 0 to 255

Ident Number
This dynamic attribute contains the Ident Number of the correspondent DP-slave. The DP-slave accepts the parameterization only if the transmitted Ident Number is identical with the attribute Ident Number of the MS0 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Attribute Type: Unsigned16

Allowed values: 0 to 2^{16} - 1

Group Ident
This dynamic attribute indicates the group(s) to which the correspondent DP-slave shall be assigned by the DP-master (Class 1) in the parameterization. The Group Ident distinguishes 8 groups. If the Group Ident is set to No Group, the DP-slave is not assigned to any group. The DP-slave stores the value of this attribute in the attribute Group Identifier of the MS0 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: Group 1 to Group 8, No Group and combinations

DPV1 Enabled
This optional and static indicates if the extended functionality shall be activated at the correspondent DP-slave. This attribute shall only be present if the attribute DPV1 Supported in the related MS0 AR entry in the List of ARL Entries of the DP-slave has the value TRUE. This attribute shall be set accordingly with the Load CRL service.

Table 233 shows the allowed values for this attribute.

Table 233 – DPV1 enabled

Value	Meaning
FALSE	The extended functionality related with the attribute DPV1 Supported is disabled.
TRUE	The extended functionality related with the attribute DPV1 Supported is enabled.

Fail Safe
This optional and static attribute indicates to the correspondent DP-slave if the DP-master (Class 1) supports the Fail Safe mode. If the DP-slave requires the Fail Safe mode and the DP-master (Class 1) does not support this mode the DP-slave shall respond with an error entry in the Slave Diagnosis (Prm Fault). The DP-slave stores the value of this attribute in the attribute Fail Safe of the MS0 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Table 234 shows the allowed values for this attribute.

Table 234 – Fail safe

Value	Meaning
FALSE	The DP-master (Class 1) does not support the Fail Safe mode.
TRUE	The DP-master (Class 1) supports the Fail Safe mode.

Enable Publisher
This optional and static attribute indicates to the correspondent DP-slave that it shall operate as Publisher. The DP-slave shall check whether it supports the requested functionality. If the DP-slave does not support this function it shall respond with an appropriate entry in the Diagnosis information (Prm Fault). This attribute shall be set accordingly with the Load CRL service.

Table 235 shows the allowed values for this attribute.

Table 235 – Enable publisher

Value	Meaning
FALSE	Publisher ability not requested for this DP-slave
TRUE	Publisher ability requested for this DP-slave

WD Base
This optional and static attribute contains the Time Base value for the watchdog control according to the parameterization for the correspondent DP-slave. The DP-slave stores the value of this attribute in the attribute WD Base of the MS0 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Table 236 shows the allowed values for this attribute.

Table 236 – WD Base

Value	Meaning
FALSE	The Watchdog Time Base is 10 ms (10).
TRUE	The Watchdog Time Base is 1 ms (1).

Enable Pull Plug Alarm
This optional and static attribute indicates if the transmission of the Pull Plug Alarm shall be enabled at the correspondent DP-slave. The DP-slave stores the value of this attribute in the attribute Pull Plug Alarm enabled of the MS1 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: TRUE, FALSE

Enable Process Alarm
This optional and static attribute indicates if the transmission of the Process Alarm shall be enabled at the correspondent DP-slave. The DP-slave stores the value of this attribute in the attribute Process Alarm enabled of the MS1 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Enable Diagnosis Alarm
This optional and static attribute indicates if the transmission of the Diagnosis Alarm shall be enabled at the correspondent DP-slave. The DP-slave stores the value of this attribute in the attribute Diagnosis Alarm enabled of the MS1 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Enable Manufacturer Specific Alarm
This optional and static attribute indicates if the transmission of the Manufacturer Specific Alarm shall be enabled at the correspondent DP-slave. The DP-slave stores the value of this attribute in the attribute Manufacturer Specific Alarm enabled of the MS1 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Enable Status Alarm
This optional and static attribute indicates if the transmission of the Status Alarm shall be enabled at the correspondent DP-slave. The DP-slave stores the value of this attribute in the attribute Status Alarm enabled of the MS1 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Enable Update Alarm
This optional and static attribute indicates if the transmission of the Update Alarm shall be enabled at the correspondent DP-slave. The DP-slave stores the value of this attribute in the attribute Update Alarm enabled of the MS1 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Check Cfg Mode
This optional and static attribute indicates how the validation of the Cfg Data shall be performed in the correspondent DP-slave. The value of this attribute is passed to the application of the DP-slave with the Check Cfg service and influences the check of the Cfg Data (see 6.2.5.3.5). This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Alarm Mode
This optional and static attribute indicates the maximum number of possible active alarms between the DP-master (Class 1) and this DP-slave. The DP-slave stores the value of this attribute in the attribute Alarm Mode enabled of the MS1 AR entry in the List of ARL Entries of the DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Table 237 shows the allowed values for this attribute.

Table 237 – Alarm mode

Value	Meaning
0	1 alarm of each type
1	2 alarms in total
2	4 alarms in total
3	8 alarms in total
4	12 alarms in total
5	16 alarms in total
6	24 alarms in total
7	32 alarms in total

Prm Structure
This static attribute indicates which type of parameterization objects shall be provided by the Check User Prm service at the correspondent DP-slave. The value of this attribute is passed to the application of the DP-slave with the Check User Prm service. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Prm Structure = FALSE
This constraint reflects that the parameters which follow are only related to the MS0 User Parameter object.

User Prm Data
This dynamic attribute contains the value(s) for the MS0 User Parameter object (either Device Related User Parameter or a List of Module Related User Parameter or both) of the correspondent DP-slave and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Device Prm Data
This optional and dynamic attribute contains the value for the Device Related User Parameter of the MS0 User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Module Prm Data
This optional and dynamic attribute contains the value for the Module Related User Parameter of the MS0 User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Prm Structure = TRUE
This constraint reflects that the parameters which follow are related to the MS0 Structured User Parameter object and/or the DXB-Linktable object and/or the IsoM Parameter object.

List of Structured Prm Data
This dynamic attribute is composed of one or more of the following elements.

Structured User Prm Data
This optional and dynamic attribute is composed of one or both of the following elements.

Device Related User Parameter
This static attribute indicates the presence of Device Related User Parameter. This attribute shall be set accordingly with the Load CRL service.

Device Prm Data
This static attribute contains the value for the Device Related User Parameter of the MS0 Structured User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Module Related User Parameter
This static attribute indicates the presence of Module Related User Parameter. This attribute shall be set accordingly with the Load CRL service.

Slot Number
This static attribute contains the value for the attribute Slot Number of the MS0 Structured User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Module Prm Data
This static attribute contains the value for the attribute Module Prm Data of the MS0 Structured User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

List of DXB Link Entries
This optional and static attribute indicates the presence of the DXB-Linktable Entries and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

Publisher Address
This static attribute contains the value for the attribute Publisher Address of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Input Data Length Publisher
This static attribute contains the value for the attribute Input Data Length Publisher of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 244

Source Offset Publisher
This static attribute contains the value for the attribute Source Offset Publisher of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Length
This static attribute contains the value for the attribute Length of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 1 to 244

IsoM Parameter
This optional and static attribute indicates the presence of the IsoM Parameter and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

T_{BASE_DP}
This static attribute contains the value for the attribute T_{BASE_DP} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 375, 750, 1 500, 3 000, 6 000, 12 000 (the value 1 500 is mandatory for standard Isochronous Mode)

T_{DP}
This static attribute contains the value for the attribute T_{DP} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

T_{MAPC}
This static attribute contains the value for the attribute T_{MAPC} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 14

T_{BASE_IO}
This static attribute contains the value for the attribute T_{BASE_IO} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 375, 750, 1 500, 3 000, 6 000, 12 000 (the value 1 500 is mandatory for standard Isochronous Mode)

T_I
This static attribute contains the value for the attribute T_I of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned16

Allowed values: 0 (special case), 1 to 2^{16} - 1

T_O
This static attribute contains the value for the attribute T_O of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned16

Allowed values: 0 (special case), 1 to 2^{16} - 1

T_{DX}
This static attribute contains the value for the attribute T_{DX} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned32

Allowed values: 1 to 2^{32} - 1

T_{PLL_W}
This static attribute contains the value for the attribute T_{PLL_W} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned16

Allowed values: 1 to 2^{16} - 1

T_{PLL_D}
This static attribute contains the value for the attribute T_{PLL_D} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned16

Allowed values: 0 to 2^{16} - 1

List of DXB Subscriber Entries
This optional and static attribute indicates the presence of the DXB-Subscribertable Entries and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

Publisher Address
This static and selection attribute contains the value for the attribute Publisher Address of one DXB-Subscribertable entry of the DXB-Subscribertable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Input Data Length Publisher
This static attribute contains the value for the attribute Input Data Length Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 244

Source Offset Publisher
This static attribute contains the value for the attribute Source Offset Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Master Data
This static and selection attribute indicates that for this entry the data shall be derived from the related DP-master (Class 1). This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 128

Source Offset Master
This static attribute defines the beginning (offset) within the Output Data of the DP-master (Class 1). This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Dest Slot Number
This static attribute indicates the Slot Number of the module to which the subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master

(Class 1) shall be mapped. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 244

Offset Data Area
This static attribute defines the beginning (offset) within the Output Data of the module (data area) to which the subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 244

Length Data Area
This attribute defines the number of octets the Subscriber shall subscribe from the related Publisher starting at the offset given in the attribute Source Offset Publisher in case of a Publisher. If the data shall be derived from the DP-master (Class 1) the starting offset is given in the attribute Source Offset Master.

The subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped to the data area of the module given in the attributes Dest Slot Number and Offset Data Area of the related DXB Subscriber Entry.

This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Prm Command
This optional and dynamic attribute indicates the presence of the Prm Command and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Seq Number
This static attribute contains the Seq Number of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: 0 to 31

Primary Request
This static attribute contains the value for the parameter Primary Request of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: TRUE, FALSE

MS1 Command
This static attribute contains the value for the parameter MS1 Command of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: no Action, Stop MS1, Start MS1, Reset MS1

Master State Clear
This static attribute contains the value for the parameter Master State Clear of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: TRUE, FALSE

Check Properties
This static attribute contains the value for the parameter Check Properties of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: TRUE, FALSE

Primary Request used
This static attribute contains the value for the parameter Primary Request used of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: TRUE, FALSE

MS1 Command used
This static attribute contains the value for the parameter MS1 Command used of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: TRUE, FALSE

Address Change
This static attribute contains the value for the parameter Address Change of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: TRUE, FALSE

Address Offset
This static attribute contains the value for the parameter Address Offset of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: TRUE, FALSE

Output Hold Time
This static attribute contains the value for the parameter Output Hold Time of the related Prm Command for the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: 0 to 2^{16} - 1

Time AR Parameter
This optional and dynamic attribute indicates the presence of the Time AR Parameter and consists of the following attributes. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Clock Sync Interval
This static attribute contains the value for the attribute clock synchronization interval of the ARL of the DP-slave operating as time receiver.

Allowed values: 1 to 2^{16} - 1; Default value: 1 000

Time Base: 10

CS Delay Time
This static and optional attribute contains the value for the attribute CS Delay Time of the ARL of the DP-slave operating as time receiver.

Default value: 0

Type: Network Time Difference

IsoM Req
This conditional and dynamic attribute indicates to the DP Application Layer of the correspondent DP-slave to operate in the Isochronous Mode. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

CR Type = BQUU & D_SAP_index = 62
This constraint reflects the attributes for this CR if the value of the attribute CR Type is set to BQUU and the attribute D_SAP_index is set to 62.

List of Cfg Data
This dynamic attribute is composed of the following list elements:

Cfg Identifier
This selection attribute describes the basic configuration format and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

Type
This attribute indicates the type of the related Input/Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Input, Output, Input-Output

Format
This attribute indicates the format of the related Input/Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Octet, Word

Consistency
This attribute indicates the consistency of the related Input/Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Octet, Word, whole length

Length
This attribute indicates the length of the related Input/Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 32

Special Cfg Identifier
This selection attribute describes the special configuration format and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

Type
This attribute indicates the type of the related Input/Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Input, Output, Input-Output

Input Length
This attribute indicates the length of the related Input Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 128

Input Format
This attribute indicates the format of the related Input Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Octet, Word

Input Consistency
This attribute indicates the consistency of the related Input Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Octet, Word, whole length

Output Length
This attribute indicates the length of the related Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 128

Output Format
This attribute indicates the format of the related Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Octet, Word

Output Consistency
This attribute indicates the consistency of the related Output Data Element. This attribute shall be set accordingly with the Load CRL service.

Allowed values: Octet, Word, whole length

List of Data Description
This attribute contains one of the allowed Data Types of the data description of one element of the Extended Input/Output Data object. The allowed Data Types are given in Clause 5. This attribute shall be set accordingly with the Load CRL service.

Manufacturer Specific Data
This attribute contains the manufacturer specific data for one Special Cfg Identifier. This attribute shall be set accordingly with the Load CRL service.

CR Type = BBUU & D_SAP_index = 60
This constraint reflects the attributes for this CR if the value of the attribute CR Type is set to BBUU and the attribute D_SAP_index is set to 60.

Diag Upd Delay
The static attribute indicates the number of Diag service requests performed in the state DIAG2 of the DP-master (Class 1) while Prm Req is set in Diag Data of the correspondent DP-slave. This allows DP-slaves with reduced performance to update the Diag Data during the start-up phase of the DP system. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 0 to 15 (extendible up to 255)

Ext Diag Overflow
This dynamic Boolean attribute shall be set to TRUE if the values of all Diagnosis objects with a changed value could not be provided with one Set Slave Diag service request by the DP-slave or if the DP-master (Class 1) could not provide all received values of the Diagnosis objects from the DP-slave to the application due to storage problems. This attribute shall be set accordingly with the Load CRL service. This attribute is further influenced by the MS0 ARL state machine of the DP-master (Class 1).

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Deactivated
This dynamic attribute indicates the state of a DP-slave which is assigned to the DP-master (Class 1). The attribute shall be set to TRUE if the DP-slave is deactivated by the AP of the DP-master (Class 1) or the USIF State of the DP-master (Class 1) is STOP. This attribute shall be set accordingly with the Load CRL service. This attribute is further influenced by the MS0 AR state machine of the DP-master (Class 1).

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Station Non Existent
This dynamic attribute indicates the status of an assigned DP-slave. The attribute shall be set to TRUE if the correspondent DP-slave does not respond to any request of the DP-master (Class 1). The attribute shall be set to FALSE if the correspondent DP-slave responds to the requests of the DP-master (Class 1). This attribute shall be set accordingly with the Load CRL service. This attribute is further influenced by the MS0 AR state machine of the DP-master (Class 1).

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Invalid Slave Response
This dynamic attribute indicates the status of an assigned DP-slave. The attribute shall be set to TRUE if the correspondent DP-slave does not respond correctly to a request of the DP-master (Class 1) (e. g. the length of the received Input Data is not in alignment with the expected length). The attribute shall be set to FALSE if the correspondent DP-slave responds correctly. This attribute shall be set accordingly with the Load CRL service. This attribute is further influenced by the MS0 AR state machine of the DP-master (Class 1).

Attribute Type: Boolean

Allowed values: TRUE, FALSE

CR Type = BBUB & D_SAP_index = NIL
This constraint reflects the attributes for this CR if the value of the attribute CR Type is set to BBUB and the attribute D_SAP_index is set to NIL.

Fail Safe
This static attribute indicates if the correspondent DP-slave supports the Fail Safe functionality. If the attribute is set to FALSE the DP-master (Class 1) shall send the value 0 for the Output Data object for this DP-slave in the operation mode CLEAR. If the attribute is set to TRUE the following rules shall apply for the DP-master (Class 1) and the DP-slave:

The DP-master (Class 1) shall convey no values (means Output Data length = 0) for the Output Data object for this DP-slave in the operation mode CLEAR.

The DP-slave shall trigger the application to set the values of the Output Data object in a safe state after receiving no values for the Output Data object during the data exchange mode.

This attribute shall be set accordingly with the Load CRL service.

Table 238 shows the allowed values for this attribute.

Table 238 – Fail safe

Value	Meaning
FALSE	Fail Safe functionality not supported by the DP-slave
TRUE	Fail Safe functionality supported by the DP-slave

DPV1 Supported
This static attribute indicates if the correspondent DP-slave supports the extended functionality. If the attribute is set to FALSE the DP-slave supports no extended functionality. If the attribute is set to TRUE the DP-slave supports the extended functionality. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

NA to Abort
This static attribute indicates the reaction of the DP-master (Class 1) if the correspondent DP-slave does not respond at the MS0 AR in the DIAG2 state and the data exchange mode. If this attribute is set to FALSE the DP-master (Class 1) continues with the interactions even the DP-slave does not respond (Status = NA) to a request. If the attribute is set to TRUE the DP-master (Class 1) releases and re-establish the CRs of the MS0 AR to the correspondent DP-slave if it does not respond (Status = NA) to a request. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Ignore ACIr
This static attribute in combination with the attribute Error Action Flag in the MS0 AR entry of the List of ARL Entries of the DP-master (Class 1) allows a DP-slave specific reaction of the DP-master (Class 1) in case of failed data transfers. If the attribute is set to FALSE the DP-master (Class 1) processes the auto-clear function according to the value of the attribute Error Action Flag. If the attribute is set to TRUE the DP-master (Class 1) ignores the auto-clear function according to the value of the attribute Error Action Flag. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Output Len
This static attribute indicates the length of the data of the Output Data object of the correspondent DP-slave. The value for this attribute shall be derived from the attribute List of Cfg Data of the related CRL entry. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 244

Exp Inp Len
This static attribute indicates the length of the data of the Input Data object of the correspondent DP-slave expected by the DP-master (Class 1). If the length of the Input Data of the Input Data object does not correspond with the value contained in this attribute the DP-master (Class 1) releases and re-establish the CRs of the MS0 AR to the correspondent DP-slave. The value for this attribute shall be derived from the attribute List of Cfg Data of the related CRL entry. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 244

Data Transfer Status
This dynamic attribute indicates the status of the data exchange mode of the correspondent DP-slave. This attribute shall be set to TRUE by the MS0 AR state machine of the DP-master (Class 1) if the data exchange with the assigned DP-slave was successfully or if between two data exchange requests was only one diagnosis request.

In all other cases the attribute shall be set to FALSE. This attribute shall be set to FALSE with the Load CRL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

System Diagnosis Status
This dynamic attribute indicates the state of the correspondent DP-slave assigned to the DP-master (Class 1). This attribute shall be set to TRUE by the MS0 AR state machine of the DP-master (Class 1) if the Ext Diag Flag of the Diagnosis information of the correspondent the DP-slave is set or if the correspondent the DP-slave does not respond or if the correspondent the DP-slave responds incorrectly so that the MS0 AR has to be re-established.

This attribute shall be set to FALSE by the MS0 AR state machine of the DP-master (Class 1) after the MS0 AR is successfully re-established or if the correspondent DP-slave is deactivated by the AP of the DP-master (Class 1) or the operation mode of the

DP-master (Class 1) is STOP.

This attribute shall be set to FALSE with the Load CRL service. This attribute is further influenced by the MS0 AR state machine.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

Publisher Flag
This static attribute indicates if the correspondent DP-slave shall operate as Publisher. If the attribute is set to FALSE the correspondent DP-slave shall not operate as Publisher and therefore the DP-master (Class 1) shall address this DP-slave with data exchange requests. If the attribute is set to TRUE the correspondent DP-slave shall operate as Publisher and therefore the DP-master (Class 1) shall address this DP-slave with data exchange requests with changed DL Function (DXB requests). This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Boolean

Allowed values: TRUE, FALSE

CR Type = BQUU & D_SAP_index = 53
This constraint reflects the attributes for this CR if the value of the attribute CR Type is set to BQUU and the attribute D_SAP_index is set to 53.

List of Structured Prm Data
This static attribute is composed of one or more of the following elements.

Structured User Prm Data
This optional and static attribute is composed of one or both of the following elements.

Device Related User Parameter
This static attribute indicates the presence of Device Related User Parameter. This attribute shall be set accordingly with the Load CRL service.

Device Prm Data
This static attribute contains the value for the Device Related User Parameter of the MS0 Structured User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Module Related User Parameter
This static attribute indicates the presence of Module Related User Parameter. This attribute shall be set accordingly with the Load CRL service.

Slot Number
This static attribute contains the value for the attribute Slot Number of the MS0 Structured User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Module Prm Data
This static attribute contains the value for the attribute Module Prm Data of the MS0 Structured User Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

List of DXB Link Entries
This optional and static attribute indicates the presence of the DXB-Linktable Entries and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

Publisher Address
This static attribute contains the value for the attribute Publisher Address of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 0 to 125

Input Data Length Publisher
This static attribute contains the value for the attribute Input Data Length Publisher of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 244

Source Offset Publisher
This static attribute contains the value for the attribute Source Offset Publisher of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 0 to 243

Length
This static attribute contains the value for the attribute Length of one DXB-Linktable entry of the DXB-Linktable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 244

IsoM Parameter
This optional and static attribute indicates the presence of the IsoM Parameter and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

T_{BASE_DP}
This static attribute contains the value for the attribute T_{BASE_DP} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 375, 750, 1 500, 3 000, 6 000, 12 000 (the value 1 500 is mandatory for standard Isochronous Mode)

T_{DP}
This static attribute contains the value for the attribute T_{DP} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to $2^{16} - 1$

T_{MAPC}
This static attribute contains the value for the attribute T_{MAPC} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 14

T_{BASE_IO}
This static attribute contains the value for the attribute T_{BASE_IO} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 375, 750, 1 500, 3 000, 6 000, 12 000 (the value 1 500 is mandatory for standard Isochronous Mode)

T_I
This static attribute contains the value for the attribute T_I of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 0 (special case), 1 to 2^{16} - 1

T_O
This static attribute contains the value for the attribute T_O of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 0 (special case), 1 to 2^{16} - 1

T_{DX}
This static attribute contains the value for the attribute T_{DX} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 2^{32} - 1

T_{PLL_W}
This static attribute contains the value for the attribute T_{PLL_W} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 2^{16} - 1

T_{PLL_D}
This static attribute contains the value for the attribute T_{PLL_D} of the IsoM Parameter object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 0 to 2^{16} - 1

List of DXB Subscriber Entries
This optional and static attribute indicates the presence of the DXB-Subscribertable Entries and is composed of the following elements. This attribute shall be set accordingly with the Load CRL service.

Publisher Address
This static and selection attribute contains the value for the attribute Publisher Address of one DXB-Subscribertable entry of the DXB-Subscribertable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 125

Input Data Length Publisher
This static attribute contains the value for the attribute Input Data Length Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Allowed values: 1 to 244

Source Offset Publisher
This static attribute contains the value for the attribute Source Offset Publisher of one DXB-Subscribertable entry of the DXB-Subscribertable object of the correspondent DP-slave. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Master Data
This static and selection attribute indicates that for this entry the data shall be derived from the related DP-master (Class 1). This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 128

Source Offset Master
This static attribute defines the beginning (offset) within the Output Data of the DP-master (Class 1). This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 243

Dest Slot Number
This static attribute indicates the Slot Number of the module to which the subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped. This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 0 to 244

Offset Data Area
This static attribute defines the beginning (offset) within the Output Data of the module (data area) to which the subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped. This attribute shall be set accordingly with the Load CRL service

Attribute Type: Unsigned8

Allowed values: 0 to 244

Length Data Area
This attribute defines the number of octets the Subscriber shall subscribe from the related Publisher starting at the offset given in the attribute Source Offset Publisher in case of a Publisher. If the data shall be derived from the DP-master (Class 1) the starting offset is given in the attribute Source Offset Master.

The subscribed values of the Input Data object of the addressed Publisher or the values of the addressed Output Data of the related DP-master (Class 1) shall be mapped to the data area of the module given in the attributes Dest Slot Number and Offset Data Area of the related DXB Subscriber Entry.

This attribute shall be set accordingly with the Load CRL service.

Attribute Type: Unsigned8

Allowed values: 1 to 244

Time AR Parameter
This optional and dynamic attribute indicates the presence of the Time AR Parameter and consists of the following attributes. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Clock Sync Interval
This static attribute contains the value for the attribute clock synchronisation interval of the ARL of the DP-slave operating as time receiver. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Allowed values: 1 to 2^{16} - 1; Default value: 1 000

Time Base: 10

CS Delay Time
This static attribute contains the value for the attribute CS Delay Time of the ARL of the DP-slave operating as time receiver. This attribute shall be set accordingly with the Load CRL service. This attribute may be set dynamically by the CRL Slave New Prm Data service.

Default value: 0

Type: Network Time Difference

CR Name
This static attribute contains the name of the related CRL Entry. This attribute shall be set accordingly with the Load CRL service.

6.2.10.4.2.3 Invocation of CRL DP-master (class 1) object

For the invocation of the CRL DP-master (Class 1) object the following rule has to be considered:

Only one CRL DP-master (Class 1) object shall be invoked in a DP-master (Class 1).

6.2.10.4.3 CRL DP-master (class 2) class specification

6.2.10.4.3.1 Template

The CRL DP-master (Class 2) object is described by the following template:

DP ASE:			**AR ASE**
CLASS:			**CRL DP-master (Class 2)**
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Key Attribute:	Implicit
2.	(m)	Attribute:	List of CRL Entries
2.1.	(m)	Attribute:	CREP

2.2.	(m)	Attribute:	CR Type
2.3.	(m)	Attribute:	Related AREP
2.4.	(m)	Attribute:	S_SAP_index
2.5.	(m)	Attribute:	D_SAP_index
2.6.	(m)	Attribute:	D_addr
2.8.	(m)	Attribute:	Service_activate
2.9.	(m)	Attribute:	Role_in_service
2.10.	(m)	Attribute:	Max_DLSDU_length_req_low
2.11.	(m)	Attribute:	Max_DLSDU_length_req_high
2.12.	(m)	Attribute:	Max_DLSDU_length_ind_low
2.13.	(m)	Attribute:	Max_DLSDU_length_ind_high
2.14.	(o)	Attribute:	CR Name

SERVICES:

1.	(m)	MgtService:	Load CRL
2.	(o)	MgtService:	Get CRL

6.2.10.4.3.2 Attributes

Implicit
The attribute Implicit indicates that the CRL DP-master (Class 2) object is implicitly addressed by the service.

List of CRL Entries
This attribute is composed of the following list elements:

CREP
This static attribute contains the local identifier of this CR. This attribute shall be set accordingly with the Load CRL service.

CR Type
This static attribute indicates the type of this CR. This attribute is dependent of the AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service. For each AR Type the correspondent CRL entries shall be set.

Table 239 shows the allowed values for this attribute.

Related AREP
This static attribute contains the local identifier for the related AR for this CR. This attribute shall be set accordingly with the Load CRL service.

S_SAP_index
This static attribute contains the identifier for the related Service Access Point of the local Data Link Layer. The value for this attribute has to be set correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service.

Table 239 shows the allowed values for this attribute.

Table 239 – S_SAP_index

AR Type	CR Type	S_SAP_index	Remark
MS0	QBCB	62	mandatory for the MS0 AR
MS2	QQCB-CO	50	mandatory for the MS2 AR
MS2	QQUU-OO	50	mandatory for the MS2 AR
MM1	QQCB-CL	54	mandatory for the MM1 AR
MM2	QQUU-OM	54	mandatory for the MM2 AR

D_SAP_index
This static attribute contains the identifier for the related Service Access Point of the remote Data Link Layer. The value for this attribute has to be set correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR. This attribute shall be set accordingly with the Load CRL service.

Table 240 shows the allowed values for this attribute.

Table 240 – D_SAP_index

AR Type	CR Type	D_SAP_index	Remark
MS0	QBCB	55, 56, 57, 59, 60	mandatory for the MS0 AR
MS2	QQCB-CO	0 to 48	mandatory for the MS2 AR
MS2	QQUU-OO	49	mandatory for the MS2 AR
MM1	QQCB-CL	54	mandatory for the MM1 AR
MM2	QQUU-OM	54	mandatory for the MM2 AR

D_addr
This static attribute contains the DL-address of the remote communication partner. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR.

Table 241 shows the allowed values for this attribute used with the Load CRL service.

Table 241 – D_addr

AR Type	CR Type	S_SAP_index	D_addr as loaded with the Load CRL service
MS0	QBCB	62	0 to 126
MS2	QQCB-CO	50	0 to 126
MS2	QQUU-OO	50	0 to 126
MM1	QQCB-CL	54	0 to 125
MM2	QQUU-OM	54	0 to 125, 127
where 0 to 125, 127	means that the attribute D_addr of the local Service Access Point shall be set with the corresponding remote DL-address		

Service_activate
This static attribute indicates the DL services which shall be activated for the S_SAP_index of this CRL entry.

The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR.

Table 242 shows the allowed values for this attribute.

Table 242 – Service_activate

AR Type	CR Type	S_SAP_index	Service_activate
MS0	QBCB	62	SRD
MS2	QQCB-CO	50	SRD
MS2	QQUU-OO	50	SRD
MM1	QQCB-CL	54	SRD
MM2	QQUU-OM	54	SDN

Role_in_service
This static attribute indicates the role which shall be activated for the S_SAP_index of this CRL entry. The following value is specified:

Initiator: The station initiates the respective service exclusively

The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type of this CR and to the attribute AR Type of the related AR.

Table 243 shows the allowed values for this attribute.

Table 243 – Role_in_service

AR Type	CR Type	S_SAP_index	Role_in_service
MS0	QBCB	62	Initiator
MS2	QQCB-CO	50	Initiator
MS2	QQUU-OO	50	Initiator
MM1	QQCB-CL	54	Initiator
MM2	QQUU-OM	54	Initiator

Max_DLSDU_length_req_low
This static attribute indicates the maximum length for the DLSDU with low priority as requester. The value for this attribute has to be set with the Load CRL service correspondent to the attributes CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 244 shows the allowed values for this attribute.

Table 244 – Max_DLSDU_length_req_low

AR Type	CR Type	D_SAP_index	Max_DLSDU_length_req_low
MS0	QBCB	55, 56, 57, 59, 60	0
MS2	QQCB-CO	0 to 48	4 to 244
MS2	QQUU-OO	49	48 to 64
MM1	QQCB-CL	54	4 to 244
MM2	QQUU-OM	54	2

Max_DLSDU_length_req_high
This static attribute indicates the maximum length for the DLSDU with high priority as requester. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 245 shows the allowed values for this attribute.

Table 245 – Max_DLSDU_length_req_high

AR Type	CR Type	D_SAP_index	Max_DLSDU_length_req_high
MS0	QBCB	55	4 to 244
MS0	QBCB	56, 57, 59, 60	0
MS2	QQCB-CO	0 to 48	0
MS2	QQUU-OO	49	0
MM1	QQCB-CL	54	0
MM2	QQUU-OM	54	0

Max_DLSDU_length_ind_low
This static attribute indicates the maximum length for the DLSDU with low priority for indication. The value for this attribute has to be set with the Load CRL service

correspondent to the attribute CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 246 shows the allowed values for this attribute.

Table 246 – Max_DLSDU_length_ind_low

AR Type	CR Type	D_SAP_index	Max_DLSDU_length_ind_low
MS0	QBCB	55	0
MS0	QBCB	56, 57	0 to 244
MS0	QBCB	59	1 to 244
MS0	QBCB	60	6 to 244
MS2	QQCB-CO	0 to 48	48 to 244
MS2	QQUU-OO	49	4
MM1	QQCB-CL	54	4 to 244
MM2	QQUU-OM	54	0

Max_DLSDU_length_ind_high
This static attribute indicates the maximum length for the DLSDU with high priority for indication. The value for this attribute has to be set with the Load CRL service correspondent to the attribute CR Type and D_SAP_index of this CR and to the attribute AR Type of the related AR.

Table 247 shows the allowed values for this attribute.

Table 247 – Max_DLSDU_length_ind_high

AR Type	CR Type	D_SAP_index	Max_DLSDU_length_ind_high
MS0	QBCB	55, 56, 57, 59, 60	0
MS2	QQCB-CO	0 to 48	0
MS2	QQUU-OO	49	0
MM1	QQCB-CL	54	0
MM2	QQUU-OM	54	0

CR Name
This static attribute contains the name of the related CRL Entry. This attribute shall be set accordingly with the Load CRL service.

6.2.10.4.3.3 Invocation of CRL DP-master (class 2) object

For the invocation of the CRL DP-master (Class 2) object the following rule has to be considered:

– Only one CRL DP-master (Class 2) object shall be invoked in a DP-master (Class 2).

6.2.10.5 AR service specification

6.2.10.5.1 DLL init DP-slave

This service is used to initialize the data link layer of a DP-slave.

Table 248 shows the parameters of the service.

Table 248 – DLL init DP-slave

Parameter name	Req	Cnf
Argument	M	
Bus Para	M	
TS	M	
$minT_{SDR}$	M	
Baudrate	M	
Slot Timer	M	
Result		M

Argument
The argument shall convey the service specific parameters of the service request.

Bus Para
This parameter is composed of the following subparameters:

TS
This parameter contains the DL-address of this DP-slave.

$minT_{SDR}$
This parameter contains the smallest station delay time of this DP-slave.

Baudrate
This parameter contains the baudrate of this DP-slave.

Slot Time
This parameter contains the slot time of this DP-slave.

Result
This parameter indicates that the service request succeeded.

6.2.10.5.2 Load ARL DP-slave

The service given in Table 249 is used to load the ARL of a DP-slave. For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-slave object.

Table 249 – Load ARL DP-slave

Parameter name	Req	Cnf
Argument	M	
Min Send Timeout	C	
List of Resource Manager Entries	C	
API	C	
SCL	C	
Max Len Data Unit	C	
Send Timeout	C	
AREP	C	
S_SAP_index	C	
In Use	C	
List of S_SAP_indices	C	
S_SAP_index	C	
List of ARL Entries	M	
AREP	M	
AR Type	M	
List of related CRL Entries	M	
CREP	M	
Prm Req	C	
Prm Fault	C	
Cfg Fault	C	
Not Supported	C	
Station Not Ready	C	
Stat Diag	C	
Master Add	C	
Sync Supported	C	
Sync Mode	C	
Freeze Supported	C	
Freeze Mode	C	
WD Enabled	C	
WD Fact 1	C	
WD Fact 2	C	
Min TSDR	C	
Ident Number	C	
Group Identifier	C	
DPV1 Supported	C	
DPV1 Enabled	C	
Fail Safe Required	C	
Fail Safe	C	
WD Base	C	
List of Real Cfg Data	M	
Cfg Identifier	S	
Type	M	
Format	M	
Consistency	M	

Parameter name	Req	Cnf
Length	M	
Special Cfg Identifier	S	
Type	M	
Input Length	C	
Input Format	C	
Input Consistency	C	
Output Length	C	
Output Format	C	
Output Consistency	C	
List of Data Description	U	
Data Description	M	
Manufacturer Specific Data	U	
No Add Change	U	
Publisher Supported	C	
Isochronous Mode Supported	C	
Isochronous Mode	C	
Prm Command Supported	C	
Pull Plug Alarm supported	C	
Pull Plug Alarm enabled	C	
Process Alarm supported	C	
Process Alarm enabled	C	
Diagnosis Alarm supported	C	
Diagnosis Alarm enabled	C	
Manufacturer Specific Alarm supported	C	
Manufacturer Specific Alarm enabled	C	
Status Alarm supported	C	
Status Alarm enabled	C	
Update Alarm supported	C	
Update Alarm enabled	C	
Alarm Mode supported	C	
Alarm Mode enabled	C	
Clock Sync Interval	C	
CS Delay Time	C	
AR Name	U	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Min Send Timeout
This parameter contains the value for the equivalent attribute of the ARL entry.

List of Resource Manager Entries
This parameter is composed of the following list elements:

API
This parameter contains the value for the equivalent attribute of the ARL entry.

SCL
This parameter contains the value for the equivalent attribute of the ARL entry.

Max Len Data Unit
This parameter contains the value for the equivalent attribute of the ARL entry.

Send Timeout
This parameter contains the value for the equivalent attribute of the ARL entry.

AREP
This parameter contains the value for the equivalent attribute of the ARL entry.

S_SAP_index
This parameter contains the value for the equivalent attribute of the ARL entry.

In Use
This parameter contains the value for the equivalent attribute of the ARL entry.

List of S_SAP_indices
This parameter is composed of the following list elements:

S_SAP_index
This parameter contains the value for the equivalent attribute of the ARL entry.

List of ARL Entries
This parameter is composed of the following list elements:

AREP
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Type
This parameter contains the value for the equivalent attribute of the ARL entry.

List of related CRL Entries
This attribute is composed of the following list elements:

CREP
This parameter contains the value for the equivalent attribute of the ARL entry.

Prm Req
This parameter contains the value for the equivalent attribute of the ARL entry.

Prm Fault
This parameter contains the value for the equivalent attribute of the ARL entry.

Cfg Fault
This parameter contains the value for the equivalent attribute of the ARL entry.

Not Supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Station Not Ready
This parameter contains the value for the equivalent attribute of the ARL entry.

Stat Diag
This parameter contains the value for the equivalent attribute of the ARL entry.

Master Add
This parameter contains the value for the equivalent attribute of the ARL entry.

Sync Supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Sync Mode
This parameter contains the value for the equivalent attribute of the ARL entry.

Freeze Supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Freeze Mode
This parameter contains the value for the equivalent attribute of the ARL entry.

WD Enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

WD Fact 1, WD Fact 2
These parameters contain the value for the equivalent attribute of the ARL entry.

Min TSDR
This parameter contains the value for the equivalent attribute of the ARL entry.

Ident Number
This parameter contains the value for the equivalent attribute of the ARL entry.

Group Identifier
This parameter contains the value for the equivalent attribute of the ARL entry.

DPV1 Supported
This parameter contains the value for the equivalent attribute of the ARL entry.

DPV1 Enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Fail Safe Required
This parameter contains the value for the equivalent attribute of the ARL entry.

Fail Safe
This parameter contains the value for the equivalent attribute of the ARL entry.

WD Base
This parameter contains the value for the equivalent attribute of the ARL entry.

List of Real Cfg Data
This parameter is composed of the following list elements:

Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value for the equivalent attribute of the ARL entry.

Format
This parameter contains the value for the equivalent attribute of the ARL entry.

Consistency
This parameter contains the value for the equivalent attribute of the ARL entry.

Length
This parameter contains the value for the equivalent attribute of the ARL entry.

Special Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value for the equivalent attribute of the ARL entry.

Input Length
This parameter contains the value for the equivalent attribute of the ARL entry.

Input Format
This parameter contains the value for the equivalent attribute of the ARL entry.

Input Consistency
This parameter contains the value for the equivalent attribute of the ARL entry.

Output Length
This parameter contains the value for the equivalent attribute of the ARL entry.

Output Format
This parameter contains the value for the equivalent attribute of the ARL entry.

Output Consistency
This parameter contains the value for the equivalent attribute of the ARL entry.

List of Data Description
This attribute is composed of the following list elements:

Data Description
This parameter contains the value for the equivalent attribute of the ARL entry.

Manufacturer Specific Data
This parameter contains the value for the equivalent attribute of the ARL entry.

No Add Change
This parameter contains the value for the equivalent attribute of the ARL entry.

Publisher Supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Isochronous Mode Supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Isochronous Mode
This parameter contains the value for the equivalent attribute of the ARL entry.

Prm Command Supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Pull Plug Alarm supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Pull Plug Alarm enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Process Alarm supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Process Alarm enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Diagnosis Alarm supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Diagnosis Alarm enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Manufacturer Specific Alarm supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Manufacturer Specific Alarm enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Status Alarm supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Status Alarm enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Update Alarm supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Update Alarm enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Alarm Mode supported
This parameter contains the value for the equivalent attribute of the ARL entry.

Alarm Mode enabled
This parameter contains the value for the equivalent attribute of the ARL entry.

Clock Sync Interval
This parameter contains the value for the equivalent attribute of the ARL entry.

CS Delay Time
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Name
This parameter contains the value for the equivalent attribute of the ARL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.3 Get ARL DP-slave

The service given in Table 250 is used to get the ARL of a DP-slave. For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-slave object.

Table 250 – Get ARL DP-slave

Parameter name	Req	Cnf
Argument	M	
Result(+)		S
Min Send Timeout		C
List of Resource Manager Entries		C
API		C
SCL		C
Max Len Data Unit		C
Send Timeout		C
AREP		C
S_SAP_index		C
In Use		C
List of S_SAP_indices		C
S_SAP_index		C
List of ARL Entries		M
AREP		M
AR Type		M
List of related CRL Entries		M
CREP		M
Prm Req		C
Prm Fault		C
Cfg Fault		C
Not Supported		C
Station Not Ready		C
Stat Diag		C
Master Add		C
Sync Supported		C
Sync Mode		C
Freeze Supported		C
Freeze Mode		C
WD Enabled		C
WD Fact 1		C
WD Fact 2		C
Min TSDR		C
Ident Number		C
Group Identifier		C
DPV1 Supported		C
DPV1 Enabled		C
Fail Safe Required		C

Parameter name	Req	Cnf
Fail Safe		C
WD Base		C
List of Real Cfg Data		M
Cfg Identifier		S
Type		M
Format		M
Consistency		M
Length		M
Special Cfg Identifier		S
Type		M
Input Length		C
Input Format		C
Input Consistency		C
Output Length		C
Output Format		C
Output Consistency		C
List of Data Description		U
Data Description		M
Manufacturer Specific Data		U
No Add Change		U
Publisher Supported		C
Isochronous Mode Supported		C
Isochronous Mode		C
Prm Command Supported		C
Pull Plug Alarm supported		C
Pull Plug Alarm enabled		C
Process Alarm supported		C
Process Alarm enabled		C
Diagnosis Alarm supported		C
Diagnosis Alarm enabled		C
Manufacturer Specific Alarm supported		C
Manufacturer Specific Alarm enabled		C
Status Alarm supported		C
Status Alarm enabled		C
Update Alarm supported		C
Update Alarm enabled		C
Alarm Mode supported		C
Alarm Mode enabled		C
Clock Sync Interval		C
CS Delay Time		C
AR Name		U
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Result(+)
This selection type parameter indicates that the service request succeeded.

Min Send Timeout
This parameter contains the value of the equivalent attribute of the ARL entry.

List of Resource Manager Entries
This parameter is composed of the following list elements:

API
This parameter contains the value of the equivalent attribute of the ARL entry.

SCL
This parameter contains the value of the equivalent attribute of the ARL entry.

Max Len Data Unit
This parameter contains the value of the equivalent attribute of the ARL entry.

Send Timeout
This parameter contains the value of the equivalent attribute of the ARL entry.

AREP
This parameter contains the value of the equivalent attribute of the ARL entry.

S_SAP_index
This parameter contains the value of the equivalent attribute of the ARL entry.

In Use
This parameter contains the value of the equivalent attribute of the ARL entry.

List of S_SAP_indices
This parameter is composed of the following list elements:

S_SAP_index
This parameter contains the value of the equivalent attribute of the ARL entry.

List of ARL Entries
This parameter is composed of the following list elements:

AREP
This parameter contains the value of the equivalent attribute of the ARL entry.

AR Type
This parameter contains the value of the equivalent attribute of the ARL entry.

List of related CRL Entries
This attribute is composed of the following list elements:

CREP
This parameter contains the value of the equivalent attribute of the ARL entry.

Prm Req
This parameter contains the value of the equivalent attribute of the ARL entry.

Prm Fault
This parameter contains the value of the equivalent attribute of the ARL entry.

Cfg Fault
This parameter contains the value of the equivalent attribute of the ARL entry.

Not Supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Station Not Ready
This parameter contains the value of the equivalent attribute of the ARL entry.

Stat Diag
This parameter contains the value of the equivalent attribute of the ARL entry.

Master Add
This parameter contains the value of the equivalent attribute of the ARL entry.

Sync Supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Sync Mode
This parameter contains the value of the equivalent attribute of the ARL entry.

Freeze Supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Freeze Mode
This parameter contains the value of the equivalent attribute of the ARL entry.

WD Enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

WD Fact 1, WD Fact 2
These parameters contain the value of the equivalent attribute of the ARL entry.

Min TSDR
This parameter contains the value of the equivalent attribute of the ARL entry.

Ident Number
This parameter contains the value of the equivalent attribute of the ARL entry.

Group Identifier
This parameter contains the value of the equivalent attribute of the ARL entry.

DPV1 Supported
This parameter contains the value of the equivalent attribute of the ARL entry.

DPV1 Enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Fail Safe Required
This parameter contains the value of the equivalent attribute of the ARL entry.

Fail Safe
This parameter contains the value of the equivalent attribute of the ARL entry.

WD Base
This parameter contains the value of the equivalent attribute of the ARL entry.

List of Real Cfg Data
This attribute is composed of the following list elements:

Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value of the equivalent attribute of the ARL entry.

Format
This parameter contains the value of the equivalent attribute of the ARL entry.

Consistency
This parameter contains the value of the equivalent attribute of the ARL entry.

Length
This parameter contains the value of the equivalent attribute of the ARL entry.

Special Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value of the equivalent attribute of the ARL entry.

Input Length
This parameter contains the value of the equivalent attribute of the ARL entry.

Input Format
This parameter contains the value of the equivalent attribute of the ARL entry.

Input Consistency
This parameter contains the value of the equivalent attribute of the ARL entry.

Output Length
This parameter contains the value of the equivalent attribute of the ARL entry.

Output Format
This parameter contains the value of the equivalent attribute of the ARL entry.

Output Consistency
This parameter contains the value of the equivalent attribute of the ARL entry.

List of Data Description
This attribute is composed of the following list elements:

Data Description
This parameter contains the value of the equivalent attribute of the ARL entry.

Manufacturer Specific Data
This parameter contains the value of the equivalent attribute of the ARL entry.

No Add Change
This parameter contains the value of the equivalent attribute of the ARL entry.

Publisher Supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Isochronous Mode Supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Isochronous Mode
This parameter contains the value of the equivalent attribute of the ARL entry.

Prm Command Supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Pull Plug Alarm supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Pull Plug Alarm enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Process Alarm supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Process Alarm enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Diagnosis Alarm supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Diagnosis Alarm enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Manufacturer Specific Alarm supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Manufacturer Specific Alarm enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Status Alarm supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Status Alarm enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Update Alarm supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Update Alarm enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Alarm Mode supported
This parameter contains the value of the equivalent attribute of the ARL entry.

Alarm Mode enabled
This parameter contains the value of the equivalent attribute of the ARL entry.

Clock Sync Interval
This parameter contains the value of the equivalent attribute of the ARL entry.

CS Delay Time
This parameter contains the value of the equivalent attribute of the ARL entry.

AR Name
This parameter contains the value of the equivalent attribute of the ARL entry.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.4 Set ARL isochronous mode

The service given in Table 251 is used to set the value of the attribute Isochronous Mode of the ARL of the DP-slave.

Table 251 – Set ARL isochronous mode

Parameter name	Req	Cnf
Argument	M	
Isochronous Mode	M	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Isochronous Mode
This parameter contains the value for the equivalent attribute of the ARL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.5 Load ARL DP-master CI1

The service given in Table 252 is used to load the ARL of a DP-master (Class 1). For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-master (Class 1) object.

Table 252 – Load ARL DP-master CI1

Parameter name	Req	Cnf
Argument	M	
Alarm Max	C	
List of ARL Entries	M	
AREP	M	
AR Type	M	
List of related CRL Entries	M	
CREP	M	
Error Action Flag	C	
Min Slave Interval	C	
Data Control Time	C	
Max User Global Control	C	
Max Diag Data Len	C	
Isochronous Mode supp	C	
Isochronous Mode	C	
IsoM Freeze	C	
IsoM Sync	C	
T_{CT}	C	
$maxT_{SH}$	C	
Extra Alarm SAP	C	
Max Channel Data Length	C	
Alarm Mode	C	
MS1 Timeout	C	
Poll Timeout	M	
Ident Number	M	
Time Device Type	C	
Clock Sync Interval	C	
CS Delay Time	C	
AR Name	U	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Alarm Max
This parameter contains the value for the equivalent attribute of the ARL entry.

List of ARL Entries
This parameter is composed of the following list elements:

AREP
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Type
This parameter contains the value for the equivalent attribute of the ARL entry.

List of related CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value for the equivalent attribute of the ARL entry.

Error Action Flag
This parameter contains the value for the equivalent attribute of the ARL entry.

Min Slave Interval
This parameter contains the value for the equivalent attribute of the ARL entry.

Data Control Time
This parameter contains the value for the equivalent attribute of the ARL entry.

Max User Global Control
This parameter contains the value for the equivalent attribute of the ARL entry.

Max Diag Data Len
This parameter contains the value for the equivalent attribute of the ARL entry.

Isochronous Mode supp
This parameter contains the value for the equivalent attribute of the ARL entry.

Isochronous Mode
This parameter contains the value for the equivalent attribute of the ARL entry.

IsoM Freeze
This parameter contains the value for the equivalent attribute of the ARL entry.

IsoM Synch
This parameter contains the value for the equivalent attribute of the ARL entry.

T_{CT}
This parameter contains the value for the equivalent attribute of the ARL entry.

maxT_{SH}
This parameter contains the value for the equivalent attribute of the ARL entry.

Extra Alarm SAP
This parameter contains the value for the equivalent attribute of the ARL entry.

Max Channel Data Length
This parameter contains the value for the equivalent attribute of the ARL entry.

Alarm Mode
This parameter contains the value for the equivalent attribute of the ARL entry.

MS1 Timeout
This parameter contains the value for the equivalent attribute of the ARL entry.

Poll Timeout
This parameter contains the value for the equivalent attribute of the ARL entry.

Ident Number
This parameter contains the value for the equivalent attribute of the ARL entry.

Time Device Type
This parameter contains the value for the equivalent attribute of the ARL entry.

Clock Sync Interval
This parameter contains the value for the equivalent attribute of the ARL entry.

CS Delay Time
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Name
This parameter contains the value for the equivalent attribute of the ARL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.6 Get ARL DP-master Cl1

The service given in Table 253 is used to get the ARL of a DP-master (Class 1). For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-master (Class 1) object. For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-master (Class 2) object.

Table 253 – Get ARL DP-master CI1

Parameter name	Req	Cnf
Argument	M	
Result(+)		S
Alarm Max		C
List of ARL Entries		M
AREP		M
AR Type		M
List of related CRL Entries		M
CREP		M
Error Action Flag		C
Min Slave Interval		C
Data Control Time		C
Max User Global Control		C
Max Diag Data Len		C
Isochronous Mode supp		C
Isochronous Mode		C
IsoM Freeze		C
IsoM Sync		C
T_{CT}		C
$maxT_{SH}$		C
Extra Alarm SAP		C
Max Channel Data Length		C
Alarm Mode		C
MS1 Timeout		C
Poll Timeout		M
Ident Number		M
Time Device Type		C
Clock Sync Interval		C
CS Delay Time		C
AR Name		U
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Result(+)
This selection type parameter indicates that the service request succeeded.

Alarm Max
This parameter contains the value of the equivalent attribute of the ARL entry.

List of ARL Entries
This parameter is composed of the following list elements:

AREP
This parameter contains the value of the equivalent attribute of the ARL entry.

AR Type
This parameter contains the value of the equivalent attribute of the ARL entry.

List of related CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value of the equivalent attribute of the ARL entry.

Error Action Flag
This parameter contains the value of the equivalent attribute of the ARL entry.

Min Slave Interval
This parameter contains the value of the equivalent attribute of the ARL entry.

Data Control Time
This parameter contains the value of the equivalent attribute of the ARL entry.

Max User Global Control
This parameter contains the value of the equivalent attribute of the ARL entry.

Max Diag Data Len
This parameter contains the value of the equivalent attribute of the ARL entry.

Isochronous Mode supp
This parameter contains the value of the equivalent attribute of the ARL entry.

Isochronous Mode
This parameter contains the value of the equivalent attribute of the ARL entry.

IsoM Freeze
This parameter contains the value of the equivalent attribute of the ARL entry.

IsoM Synch
This parameter contains the value of the equivalent attribute of the ARL entry.

T_{CT}
This parameter contains the value of the equivalent attribute of the ARL entry.

maxT_{SH}
This parameter contains the value of the equivalent attribute of the ARL entry.

Extra Alarm SAP
This parameter contains the value of the equivalent attribute of the ARL entry.

Max Channel Data Length
This parameter contains the value of the equivalent attribute of the ARL entry.

Alarm Mode
This parameter contains the value of the equivalent attribute of the ARL entry.

MS1 Timeout
This parameter contains the value of the equivalent attribute of the ARL entry.

Poll Timeout
This parameter contains the value of the equivalent attribute of the ARL entry.

Ident Number
This parameter contains the value of the equivalent attribute of the ARL entry.

Time Device Type
This parameter contains the value of the equivalent attribute of the ARL entry.

Clock Sync Interval
This parameter contains the value of the equivalent attribute of the ARL entry.

CS Delay Time
This parameter contains the value of the equivalent attribute of the ARL entry.

AR Name
This parameter contains the value of the equivalent attribute of the ARL entry.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.7 ARL slave update DP-master Cl1

The service given in Table 254 is used to update the ARL of a DP-master (Class 1) concerning the entries of one DP-slave. For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-master (Class 1) object.

Table 254 – ARL Slave update DP-master Cl1

Parameter name	Req	Cnf
Argument	M	
List of ARL Entries	M	
AREP	M	
AR Type	M	
List of related CRL Entries	M	
CREP	M	
Extra Alarm SAP	C	
Max Channel Data Length	C	
Alarm Mode	C	
MS1 Timeout	C	
Time Device Type	C	
Clock Sync Interval	C	
CS Delay Time	C	
AR Name	U	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

List of ARL Entries
This parameter is composed of the following list elements:

AREP
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Type
This parameter contains the value for the equivalent attribute of the ARL entry.

List of related CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value for the equivalent attribute of the ARL entry.

Extra Alarm SAP
This parameter contains the value for the equivalent attribute of the ARL entry.

Max Channel Data Length
This parameter contains the value for the equivalent attribute of the ARL entry.

Alarm Mode
This parameter contains the value for the equivalent attribute of the ARL entry.

MS1 Timeout
This parameter contains the value for the equivalent attribute of the ARL entry.

Time Device Type
This parameter contains the value for the equivalent attribute of the ARL entry.

Clock Sync Interval
This parameter contains the value for the equivalent attribute of the ARL entry.

CS Delay Time
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Name
This parameter contains the value for the equivalent attribute of the ARL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.8 Load ARL DP-master CI2

The service given in Table 255 is used to load the ARL of a DP-master (Class 2). For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-master (Class 2) object.

Table 255 – Load ARL DP-master CI2

Parameter name	Req	Cnf
Argument	M	
List of ARL Entries	M	
AREP	M	
AR Type	M	
List of related CRL Entries	M	
CREP	M	
Poll Timeout	M	
AR Name	U	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

List of ARL Entries
This parameter is composed of the following list elements:

AREP
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Type
This parameter contains the value for the equivalent attribute of the ARL entry.

List of related CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value for the equivalent attribute of the ARL entry.

Poll Timeout
This parameter contains the value for the equivalent attribute of the ARL entry.

AR Name
This parameter contains the value for the equivalent attribute of the ARL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.9 Get ARL DP-master CI2

The service given in Table 256 is used to get the ARL of a DP-master (Class 2). For the use of the conditional parameters in the request primitive refer to the description of the ARL DP-master (Class 2) object.

Table 256 – Get ARL DP-master CI2

Parameter name	Req	Cnf
Argument	M	
Result(+)		S
List of ARL Entries		M
AREP		M
AR Type		M
List of related CRL Entries		M
CREP		M
Poll Timeout		M
AR Name		U
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Result(+)
This selection type parameter indicates that the service request succeeded.

List of ARL Entries
This parameter is composed of the following list elements:

AREP
This parameter contains the value of the equivalent attribute of the ARL entry.

AR Type
This parameter contains the value of the equivalent attribute of the ARL entry.

List of related CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value of the equivalent attribute of the ARL entry.

Poll Timeout
This parameter contains the value of the equivalent attribute of the ARL entry.

AR Name
This parameter contains the value of the equivalent attribute of the ARL entry.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.10 Load CRL DP-slave

The service given in Table 257 is used to load the CRL of a DP-slave. For the presence of the conditional parameters in the indication primitive refer to the description of the CRL DP-slave object.

Table 257 – Load CRL DP-slave

Parameter name	Req	Cnf
Argument	M	
List of CRL Entries	M	
CREP	M	
CR Type	M	
Related AREP	M	
S_SAP_index	M	
D_addr	M	
Service_activate	M	
Role_in_Service	M	
Indication_mode	M	
Max_DLSDU_length_req_low	M	
Max_DLSDU_length_req_high	M	
Max_DLSDU_length_ind_low	M	
Max_DLSDU_length_ind_high	M	
List of DXB Link Entries		
Publisher Address	M	
Input Data Length Publisher	M	
Source Offset Publisher	M	
Length	M	
CR Name	U	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

List of CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value for the equivalent attribute of the CRL entry.

CR Type
This parameter contains the value for the equivalent attribute of the CRL entry.

Related AREP
This parameter contains the value for the equivalent attribute of the CRL entry.

S_SAP_index
This parameter contains the value for the equivalent attribute of the CRL entry.

D_addr
This parameter contains the value for the equivalent attribute of the CRL entry.

Service_activate
This parameter contains the value for the equivalent attribute of the CRL entry.

Role_in_Service
This parameter contains the value for the equivalent attribute of the CRL entry.

Indication_mode
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_low
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_high
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_low
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_high
This parameter contains the value for the equivalent attribute of the CRL entry.

List of DXB Link Entries
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Length
This parameter contains the value for the equivalent attribute of the CRL entry.

CR Name
This parameter contains the value for the equivalent attribute of the CRL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.11 Load CRL DXB link entries

The service given in Table 258 is used to load the DXB Link Entries for the BBUU-OM CRL entries of a DP-slave. This service is requested by the application of the DP-slave after the reception of the attribute values of its DXB-Linktable object. The DP Application Layer may acknowledge this service negatively if the application has already loaded a DXB-Linktable and the Subscriber State-Machine has been started. This service shall only be used in conjunction with the BBUU-OM CR.

Table 258 – Load CRL DXB link entries

Parameter name	Req	Cnf
Argument	M	
CREP	M	
List of DXB Link Entries	M	
Publisher Address	M	
Input Data Length Publisher	M	
Source Offset Publisher	M	
Length	M	
Result(+)		S
CREP		M
Result(-)		S
CREP		M

Argument
The argument shall convey the service specific parameters of the service request.

CREP
This parameter is used as local identifier of a BBUU-OM CR.

List of DXB Link Entries
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value for the attribute Publisher Address of one BBUU-OM CRL entry.

Input Data Length Publisher
This parameter contains the value for the attribute Input Data Length Publisher of one BBUU-OM CRL entry.

Source Offset Publisher
This parameter contains the value for the attribute Source Offset Publisher of one BBUU-OM CRL entry.

Length
This parameter contains the value for the attribute Length of one DXB-Linktable entry of one BBUU-OM CRL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

6.2.10.5.12 Get CRL DP-slave

The service given in Table 259 is used to get the CRL of a DP-slave. For the presence of the conditional parameters in the indication primitive refer to the description of the CRL DP-slave object.

Table 259 – Get CRL DP-slave

Parameter name	Req	Cnf
Argument	M	
Result(+)		S
List of CRL Entries		M
CREP		M
CR Type		M
Related AREP		M
S_SAP_index		M
D_addr		M
Service_activate		M
Role_in_Service		M
Indication_mode		M
Max_DLSDU_length_req_low		M
Max_DLSDU_length_req_high		M
Max_DLSDU_length_ind_low		M
Max_DLSDU_length_ind_high		M
List of DXB Link Entries		U
Publisher Address		M
Input Data Length Publisher		M
Source Offset Publisher		M
Length		M
CR Name		U
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Result(+)
This selection type parameter indicates that the service request succeeded.

List of CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value of the equivalent attribute of the CRL entry.

CR Type
This parameter contains the value of the equivalent attribute of the CRL entry.

Related AREP
This parameter contains the value of the equivalent attribute of the CRL entry.

S_SAP_index
This parameter contains the value of the equivalent attribute of the CRL entry.

D_addr
This parameter contains the value of the equivalent attribute of the CRL entry.

Service_activate
This parameter contains the value of the equivalent attribute of the CRL entry.

Role_in_Service
This parameter contains the value of the equivalent attribute of the CRL entry.

Indication_mode
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_low
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_high
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_low
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_high
This parameter contains the value of the equivalent attribute of the CRL entry.

List of DXB Link Entries
This parameter is composed of the following list elements:

Publisher Address
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Length
This parameter contains the value of the equivalent attribute of the CRL entry.

CR Name
This parameter contains the value of the equivalent attribute of the CRL entry.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.13 Load CRL DP-master Cl1

The service given in Table 260 is used to load the CRL of a DP-master (Class 1). For the use of the conditional parameters in the request primitive refer to the description of the CRL DP-master (Class 1) object.

Table 260 – Load CRL DP-master CI1

Parameter name	Req	Cnf
Argument	M	
Update	M	
List of CRL Entries	M	
CREP	M	
CR Type	M	
Related AREP	M	
S_SAP_index	M	
D_SAP_index	M	
D_addr	M	
Service_activate	M	
Role_in_service	M	
Indication_mode	M	
Max_DLSDU_length_req_low	M	
Max_DLSDU_length_req_high	M	
Max_DLSDU_length_ind_low	M	
Max_DLSDU_length_ind_high	M	
Active	C	
New Prm	C	
Prm Data	C	
Lock	M	
Sync	M	
Freeze	M	
WD On	M	
WD Fact 1	M	
WD Fact 2	M	
Min Tsdr	M	
Ident Number	M	
Group Ident	M	
DPV1 Enable	C	
Fail Safe	C	
Enable Publisher	C	
WD Base	C	
Enable Pull Plug Alarm	C	
Enable Process Alarm	C	
Enable Diagnosis Alarm	C	
Enable Manufacturer Specific Alarm	C	
Enable Status Alarm	C	
Enable Update Alarm	C	
Check Cfg Mode	C	
Alarm Mode	C	
Prm Structure	M	
User Prm Data	C	
Device Prm Data	U	

Parameter name	Req	Cnf
Module Prm Data	U	
List of Structured Prm Data	C	
Structured User Prm Data	U	
Device Related User Parameter	U	
Device Prm Data	M	
Module Related User Parameter	U	
Slot Number	M	
Module Prm Data	M	
List of DXB Link Entries	U	
Publisher Address	M	
Input Data Length Publisher	M	
Source Offset Publisher	M	
Length	M	
IsoM Parameter	U	
T_{BASE_DP}	M	
TDP	M	
TMAPC	M	
TBASE_IO	M	
T_I	M	
T_O	M	
T_{PLL_W}	M	
T_{PLL_D}	M	
List of DXB Subscriber Entries	U	
Publisher Address	C	
Input Data Length Publisher	M	
Source Offset Publisher	M	
Master Data	C	
Source Offset Master	M	
Dest Slot Number	M	
Offset Data Area	M	
Length Data Area	M	
Prm Command	U	
Seq Number	M	
Primary Request	M	
MS1 Command	M	
Master State Clear	M	
Check Properties	M	
Primary Request used	M	
MS1 Command used	M	
Address Change	M	
Address Offset	M	
Output Hold Time	M	
Time AR Parameter	U	

Parameter name	Req	Cnf
Clock Sync Interval	M	
CS Delay Time	M	
IsoM Req	U	
List of Cfg Data	C	
Cfg Identifier	S	
Type	M	
Format	M	
Consistency	M	
Length	M	
Special Cfg Identifier	S	
Type	M	
Input Length	C	
Input Format	C	
Input Consistency	C	
Output Length	C	
Output Format	C	
Output Consistency	C	
List of Data Description	U	
Data Description	M	
Manufacturer Specific Data	U	
Diag Upd Delay	C	
Ext Diag Overflow	C	
Deactivated	C	
Station Non Existent	C	
Invalid Slave Response	C	
Fail Safe	C	
DPV1 Supported	C	
NA to Abort	C	
Ignore AClr	C	
Output Len	C	
Exp Inp Len	C	
Data Transfer Status	C	
System Diagnosis Status	C	
Publisher Flag	C	
List of Structured Prm Data	C	
Structured User Prm Data	U	
Device Related User Parameter	U	
Device Prm Data	M	
Module Related User Parameter	U	
Slot Number	M	
Module Prm Data	M	
List of DXB Link Entries	U	
Publisher Address	M	
Input Data Length Publisher	M	
Source Offset Publisher	M	

Parameter name	Req	Cnf
Length	M	
IsoM Parameter	U	
T_{BASE_DP}	M	
T_{DP}	M	
T_{MAPC}	M	
T_{BASE_IO}	M	
T_I	M	
T_O	M	
T_{PLL_W}	M	
T_{PLL_D}	M	
List of DXB Subscriber Entries	U	
Publisher Address	C	
Input Data Length Publisher	M	
Source Offset Publisher	M	
Master Data	C	
Source Offset Master	M	
Dest Slot Number	M	
Offset Data Area	M	
Length Data Area	M	
Time AR Parameter	U	
Clock Sync Interval	M	
CS Delay Time	M	
CR Name	U	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Update
This parameter set to FALSE indicates that the complete CRL shall be loaded. This parameter set to TRUE indicates that only the CRL entries of one DP-slave shall be updated.

List of CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value for the equivalent attribute of the CRL entry.

CR Type
This parameter contains the value for the equivalent attribute of the CRL entry.

Related AREP
This parameter contains the value for the equivalent attribute of the CRL entry.

S_SAP_index
This parameter contains the value for the equivalent attribute of the CRL entry.

D_SAP_index
This parameter contains the value for the equivalent attribute of the CRL entry.

D_addr
This parameter contains the value for the equivalent attribute of the CRL entry.

Service_activate
This parameter contains the value for the equivalent attribute of the CRL entry.

Role_in_service
This parameter contains the value for the equivalent attribute of the CRL entry.

Indication_mode
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_low
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_high
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_low
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_high
This parameter contains the value for the equivalent attribute of the CRL entry.

Active
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 61.

New Prm
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 61.

Prm Data
This conditional parameter is composed of the following subparameters. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 61.

Lock
This parameter contains the value for the equivalent attribute of the CRL entry.

Sync
This parameter contains the value for the equivalent attribute of the CRL entry.

Freeze
This parameter contains the value for the equivalent attribute of the CRL entry.

WD On
This parameter contains the value for the equivalent attribute of the CRL entry.

WD Fact 1
This parameter contains the value for the equivalent attribute of the CRL entry.

WD Fact 2
This parameter contains the value for the equivalent attribute of the CRL entry.

Min Tsdr
This parameter contains the value for the equivalent attribute of the CRL entry.

Ident Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Group Ident
This parameter contains the value for the equivalent attribute of the CRL entry.

DPV1 Enable
This parameter contains the value for the equivalent attribute of the CRL entry.

Fail Safe
This parameter contains the value for the equivalent attribute of the CRL entry.

Enable Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

WD Base
This parameter contains the value for the equivalent attribute of the CRL entry.

Enable Pull Plug Alarm
This parameter contains the value for the equivalent attribute of the CRL entry.

Enable Process Alarm
This parameter contains the value for the equivalent attribute of the CRL entry.

Enable Diagnosis Alarm
This parameter contains the value for the equivalent attribute of the CRL entry.

Enable Manufacturer Specific Alarm
This parameter contains the value for the equivalent attribute of the CRL entry.

Enable Status Alarm
This parameter contains the value for the equivalent attribute of the CRL entry.

Enable Update Alarm
This parameter contains the value for the equivalent attribute of the CRL entry.

Check Cfg Mode
This parameter contains the value for the equivalent attribute of the CRL entry.

Alarm Mode
This parameter contains the value for the equivalent attribute of the CRL entry.

Prm Structure
This parameter contains the value for the equivalent attribute of the CRL entry.

User Prm Data
This conditional parameter is only present if the parameter Prm Structure contains the value FALSE. This parameter is composed of one or both of the following subparameters.

Device Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Module Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

List of Structured Prm Data
This conditional parameter is only present if the parameter Prm Structure contains the value TRUE. This parameter is composed of one or more of the following list elements.

Structured User Prm Data
This parameter is composed of one or both of the following subparameters.

Device Related User Parameter
This parameter indicates the presence of Device Related User Parameter.

Device Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Module Related User Parameter
This parameter indicates the presence of Module Related User Parameter.

Slot Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Module Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

List of DXB Link Entries
This parameter indicates the presence of the DXB-Linktable and is composed of the following list elements.

Publisher Address
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Length
This parameter contains the value for the equivalent attribute of the CRL entry.

IsoM Parameter
This parameter indicates the presence of the IsoM Parameter.

T_{BASE_DP}
This parameter contains the value for the equivalent attribute of the CRL entry.

T_{DP}
This parameter contains the value for the equivalent attribute of the CRL entry.

T_{MAPC}
This parameter contains the value for the equivalent attribute of the CRL entry.

T_{BASE_IO}
This parameter contains the value for the equivalent attribute of the CRL entry.

T_I
This parameter contains the value for the equivalent attribute of the CRL entry.

T_O
This parameter contains the value for the equivalent attribute of the CRL entry.

T_{PLL_W}
This parameter contains the value for the equivalent attribute of the CRL entry.

T_{PLL_D}
This parameter contains the value for the equivalent attribute of the CRL entry.

List of DXB Subscriber Entries
This parameter contains the value for the equivalent attribute of the CRL entry.

Publisher Address
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Master Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Source Offset Master
This parameter contains the value for the equivalent attribute of the CRL entry.

Dest Slot Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Offset Data Area
This parameter contains the value for the equivalent attribute of the CRL entry.

Length Data Area
This parameter contains the value for the equivalent attribute of the CRL entry.

Prm Command
This parameter contains the value for the equivalent attribute of the CRL entry.

Seq Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Primary Request
This parameter contains the value for the equivalent attribute of the CRL entry.

MS1 Command
This parameter contains the value for the equivalent attribute of the CRL entry.

Master State Clear
This parameter contains the value for the equivalent attribute of the CRL entry.

Check Properties
This parameter contains the value for the equivalent attribute of the CRL entry.

Primary Request used
This parameter contains the value for the equivalent attribute of the CRL entry.

MS1 Command used
This parameter contains the value for the equivalent attribute of the CRL entry.

Address Change
This parameter contains the value for the equivalent attribute of the CRL entry.

Address Offset
This parameter contains the value for the equivalent attribute of the CRL entry.

Output Hold Time
This parameter contains the value for the equivalent attribute of the CRL entry.

Time AR Parameter
This parameter contains the value for the equivalent attribute of the CRL entry.

Clock Sync Interval
This parameter contains the value for the equivalent attribute of the CRL entry.

CS Delay Time
This parameter contains the value for the equivalent attribute of the CRL entry.

IsoM Req
This parameter contains the value for the equivalent attribute of the CRL entry.

List of Cfg Data
This conditional parameter is composed of the following list elements. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 62.

Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value for the equivalent attribute of the CRL entry.

Format
This parameter contains the value for the equivalent attribute of the CRL entry.

Consistency
This parameter contains the value for the equivalent attribute of the CRL entry.

Length
This parameter contains the value for the equivalent attribute of the CRL entry.

Special Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Length
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Format
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Consistency
This parameter contains the value for the equivalent attribute of the CRL entry.

Output Length
This parameter contains the value for the equivalent attribute of the CRL entry.

Output Format
This parameter contains the value for the equivalent attribute of the CRL entry.

Output Consistency
This parameter contains the value for the equivalent attribute of the CRL entry.

List of Data Description
This parameter is composed of the following list elements:

Data Description
This parameter contains the value for the equivalent attribute of the CRL entry.

Manufacturer Specific Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Diag Upd Delay
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Ext Diag Overflow
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Deactivated
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Station Non Existent
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Invalid Slave Response
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Fail Safe
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

DPV1 Supported
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

NA to Abort
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Ignore ACIr
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Output Len
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Exp Inp Len
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Data Transfer Status
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

System Diagnosis Status
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Publisher Flag
This conditional parameter contains the value for the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

List of Structured Prm Data
This conditional parameter is only present if the parameter CR Type contains the value BQUU and the parameter D_SAP_index contains the value 53. This parameter is composed of one or more of the following list elements.

Structured User Prm Data
This parameter is composed of one or both of the following subparameters.

Device Related User Parameter
This parameter indicates the presence of Device Related User Parameter.

Device Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Module Related User Parameter
This parameter indicates the presence of Module Related User Parameter.

Slot Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Module Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

List of DXB Link Entries
This parameter indicates the presence of the DXB-Linktable and is composed of the following list elements.

Publisher Address
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Length
This parameter contains the value for the equivalent attribute of the CRL entry.

IsoM Parameter
This parameter indicates the presence of the IsoM Parameter.

TBASE_DP
This parameter contains the value for the equivalent attribute of the CRL entry.

T_{DP}
This parameter contains the value for the equivalent attribute of the CRL entry.

TMAPC
This parameter contains the value for the equivalent attribute of the CRL entry.

TBASE_IO
This parameter contains the value for the equivalent attribute of the CRL entry.

T_I
This parameter contains the value for the equivalent attribute of the CRL entry.

T_O
This parameter contains the value for the equivalent attribute of the CRL entry.

TPLL_W
This parameter contains the value for the equivalent attribute of the CRL entry.

TPLL_D
This parameter contains the value for the equivalent attribute of the CRL entry.

List of DXB Subscriber Entries
This parameter contains the value for the equivalent attribute of the CRL entry.

Publisher Address
This parameter contains the value for the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value for the equivalent attribute of the CRL entry.

Master Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Source Offset Master
This parameter contains the value for the equivalent attribute of the CRL entry.

Dest Slot Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Offset Data Area
This parameter contains the value for the equivalent attribute of the CRL entry.

Length Data Area
This parameter contains the value for the equivalent attribute of the CRL entry.

Time AR Parameter
This parameter contains the value for the equivalent attribute of the CRL entry.

Clock Sync Interval
This parameter contains the value for the equivalent attribute of the CRL entry.

CS Delay Time
This parameter contains the value for the equivalent attribute of the CRL entry.

CR Name
This parameter contains the value for the equivalent attribute of the CRL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.14 Get CRL DP-master Cl1

The service given in Table 261 is used to get the CRL of a DP-master (Class 1). For the use of the conditional parameters in the request primitive refer to the description of the CRL DP-master (Class 1) object.

Table 261 – Get CRL DP-master Cl1

Parameter name	Req	Cnf
Argument	M	
Result(+)		S
List of CRL Entries		M
CREP		M
CR Type		M
Related AREP		M
S_SAP_index		M
D_SAP_index		M
D_addr		M
Service_activate		M
Role_in_service		M
Indication_mode		M
Max_DLSDU_length_req_low		M
Max_DLSDU_length_req_high		M
Max_DLSDU_length_ind_low		M
Max_DLSDU_length_ind_high		M
Active		C
New Prm		C
Prm Data		C
Lock		M
Sync		M
Freeze		M
WD On		M
WD Fact 1		M

Parameter name	Req	Cnf
WD Fact 2		M
Min Tsdr		M
Ident Number		M
Group Ident		M
DPV1 Enable		C
Fail Safe		C
Enable Publisher		C
WD Base		C
Enable Pull Plug Alarm		C
Enable Process Alarm		C
Enable Diagnosis Alarm		C
Enable Manufacturer Specific Alarm		C
Enable Status Alarm		C
Enable Update Alarm		C
Check Cfg Mode		C
Alarm Mode		C
Prm Structure		M
User Prm Data		C
Device Prm Data		U
Module Prm Data		U
List of Structured Prm Data		C
Structured User Prm Data		U
Device Related User Parameter		U
Device Prm Data		M
Module Related User Parameter		U
Slot Number		M
Module Prm Data		M
List of DXB Link Entries		U
Publisher Address		M
Input Data Length Publisher		M
Source Offset Publisher		M
Length		M
IsoM Parameter		U
T_{BASE_DP}		M
T_{DP}		M
T_{MAPC}		M
T_{BASE_IO}		M
T_I		M
T_O		M
T_{PLL_W}		M
T_{PLL_D}		M
List of DXB Subscriber Entries		U
Publisher Address		C
Input Data Length Publisher		M
Source Offset Publisher		M

Parameter name	Req	Cnf
Master Data		C
Source Offset Master		M
Dest Slot Number		M
Offset Data Area		M
Length Data Area		M
Prm Command		U
Seq Number		M
Primary Request		M
MS1 Command		M
Master State Clear		M
Check Properties		M
Primary Request used		M
MS1 Command used		M
Address Change		M
Address Offset		M
Output Hold Time		M
Time AR Parameter		U
Clock Sync Interval		M
CS Delay Time		M
IsoM Req		U
List of Cfg Data		C
Cfg Identifier		S
Type		M
Format		M
Consistency		M
Length		M
Special Cfg Identifier		S
Type		M
Input Length		C
Input Format		C
Input Consistency		C
Output Length		C
Output Format		C
Output Consistency		C
List of Data Description		U
Data Description		M
Manufacturer Specific Data		U
Diag Upd Delay		C
Ext Diag Overflow		C
Deactivated		C
Station Non Existent		C
Invalid Slave Response		C
Fail Safe		C
DPV1 Supported		C
NA to Abort		C

Parameter name	Req	Cnf
Ignore AClr		C
Output Len		C
Exp Inp Len		C
Data Transfer Status		C
System Diagnosis Status		C
Publisher Flag		C
List of Structured Prm Data		C
Structured User Prm Data		U
Device Related User Parameter		U
Device Prm Data		M
Module Related User Parameter		U
Slot Number		M
Module Prm Data		M
List of DXB Link Entries		U
Publisher Address		M
Input Data Length Publisher		M
Source Offset Publisher		M
Length		M
IsoM Parameter		U
T_{BASE_DP}		M
T_{DP}		M
T_{MAPC}		M
T_{BASE_IO}		M
T_I		M
T_O		M
T_{PLL_W}		M
T_{PLL_D}		M
List of DXB Subscriber Entries		U
Publisher Address		C
Input Data Length Publisher		M
Source Offset Publisher		M
Master Data		C
Source Offset Master		M
Dest Slot Number		M
Offset Data Area		M
Length Data Area		M
Time AR Parameter		U
Clock Sync Interval		M
CS Delay Time		M
CR Name		U
Result(-)		S
Status		M

Argument

The argument shall convey the service specific parameters of the service request.

Result(+)
This selection type parameter indicates that the service request succeeded.

List of CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value of the equivalent attribute of the CRL entry.

CR Type
This parameter contains the value of the equivalent attribute of the CRL entry.

Related AREP
This parameter contains the value of the equivalent attribute of the CRL entry.

S_SAP_index
This parameter contains the value of the equivalent attribute of the CRL entry.

D_SAP_index
This parameter contains the value of the equivalent attribute of the CRL entry.

D_addr
This parameter contains the value of the equivalent attribute of the CRL entry.

Service_activate
This parameter contains the value of the equivalent attribute of the CRL entry.

Role_in_service
This parameter contains the value of the equivalent attribute of the CRL entry.

Indication_mode
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_low
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_high
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_low
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_high
This parameter contains the value of the equivalent attribute of the CRL entry.

Active
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 61.

New Prm
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 61.

Prm Data
This conditional parameter is composed of the following subparameters. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 61.

Lock
This parameter contains the value of the equivalent attribute of the CRL entry.

Sync
This parameter contains the value of the equivalent attribute of the CRL entry.

Freeze
This parameter contains the value of the equivalent attribute of the CRL entry.

WD On
This parameter contains the value of the equivalent attribute of the CRL entry.

WD Fact 1
This parameter contains the value of the equivalent attribute of the CRL entry.

WD Fact 2
This parameter contains the value of the equivalent attribute of the CRL entry.

Min Tsdr
This parameter contains the value of the equivalent attribute of the CRL entry.

Ident Number
This parameter contains the value of the equivalent attribute of the CRL entry.

Group Ident
This parameter contains the value of the equivalent attribute of the CRL entry.

DPV1 Enable
This parameter contains the value of the equivalent attribute of the CRL entry.

Fail Safe
This parameter contains the value of the equivalent attribute of the CRL entry.

Enable Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

WD Base
This parameter contains the value of the equivalent attribute of the CRL entry.

Enable Pull Plug Alarm
This parameter contains the value of the equivalent attribute of the CRL entry.

Enable Process Alarm
This parameter contains the value of the equivalent attribute of the CRL entry.

Enable Diagnosis Alarm
This parameter contains the value of the equivalent attribute of the CRL entry.

Enable Manufacturer Specific Alarm
This parameter contains the value of the equivalent attribute of the CRL entry.

Enable Status Alarm
This parameter contains the value of the equivalent attribute of the CRL entry.

Enable Update Alarm
This parameter contains the value of the equivalent attribute of the CRL entry.

Check Cfg Mode
This parameter contains the value of the equivalent attribute of the CRL entry.

Alarm Mode
This parameter contains the value of the equivalent attribute of the CRL entry.

Prm Structure
This parameter contains the value of the equivalent attribute of the CRL entry.

User Prm Data
This conditional parameter is only present if the parameter Prm Structure contains the value FALSE. This parameter is composed of one or both of the following subparameters.

Device Prm Data
This parameter contains the value of the equivalent attribute of the CRL entry.

Module Prm Data
This parameter contains the value of the equivalent attribute of the CRL entry.

List of Structured Prm Data
This conditional parameter is only present if the parameter Prm Structure contains the value TRUE. This parameter is composed of one or more of the following list elements.

Structured User Prm Data
This parameter is composed of one or both of the following subparameters.

Device Related User Parameter
This parameter indicates the presence of Device Related User Parameter.

Device Prm Data
This parameter contains the value of the equivalent attribute of the CRL entry.

Module Related User Parameter
This parameter indicates the presence of Module Related User Parameter.

Slot Number
This parameter contains the value of the equivalent attribute of the CRL entry.

Module Prm Data
This parameter contains the value of the equivalent attribute of the CRL entry.

List of DXB Link Entries
This parameter indicates the presence of the DXB-Linktable and is composed of the following list elements.

Publisher Address
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Length
This parameter contains the value of the equivalent attribute of the CRL entry.

IsoM Parameter
This parameter indicates the presence of the IsoM Parameter.

T_{BASE_DP}
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{DP}
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{MAPC}
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{BASE_IO}
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{I}
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{O}
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{PLL_W}
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{PLL_D}
This parameter contains the value of the equivalent attribute of the CRL entry.

List of DXB Subscriber Entries
This parameter contains the value of the equivalent attribute of the CRL entry.

Publisher Address
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Master Data
This parameter contains the value of the equivalent attribute of the CRL entry.

Source Offset Master
This parameter contains the value of the equivalent attribute of the CRL entry.

Dest Slot Number
This parameter contains the value of the equivalent attribute of the CRL entry.

Offset Data Area
This parameter contains the value of the equivalent attribute of the CRL entry.

Length Data Area
This parameter contains the value of the equivalent attribute of the CRL entry.

Prm Command
This parameter contains the value of the equivalent attribute of the CRL entry.

Seq Number
This parameter contains the value of the equivalent attribute of the CRL entry.

Primary Request
This parameter contains the value of the equivalent attribute of the CRL entry.

MS1 Command
This parameter contains the value of the equivalent attribute of the CRL entry.

Master State Clear
This parameter contains the value of the equivalent attribute of the CRL entry.

Check Properties
This parameter contains the value of the equivalent attribute of the CRL entry.

Primary Request used
This parameter contains the value of the equivalent attribute of the CRL entry.

MS1 Command used
This parameter contains the value of the equivalent attribute of the CRL entry.

Address Change
This parameter contains the value of the equivalent attribute of the CRL entry.

Address Offset
This parameter contains the value of the equivalent attribute of the CRL entry.

Output Hold Time
This parameter contains the value of the equivalent attribute of the CRL entry.

Time AR Parameter
This parameter contains the value of the equivalent attribute of the CRL entry.

Clock Sync Interval
This parameter contains the value of the equivalent attribute of the CRL entry.

CS Delay Time
This parameter contains the value of the equivalent attribute of the CRL entry.

IsoM Req
This parameter contains the value of the equivalent attribute of the CRL entry.

List of Cfg Data
This conditional parameter is composed of the following list elements. It shall only be present if the parameter CREP has the value BQUU and the parameter D_SAP_index has the value 62.

Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value of the equivalent attribute of the CRL entry.

Format
This parameter contains the value of the equivalent attribute of the CRL entry.

Consistency
This parameter contains the value of the equivalent attribute of the CRL entry.

Length
This parameter contains the value of the equivalent attribute of the CRL entry.

Special Cfg Identifier
This selection type parameter indicates the type of Cfg Identifier format.

Type
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Length
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Format
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Consistency
This parameter contains the value of the equivalent attribute of the CRL entry.

Output Length
This parameter contains the value of the equivalent attribute of the CRL entry.

Output Format
This parameter contains the value of the equivalent attribute of the CRL entry.

Output Consistency
This parameter contains the value of the equivalent attribute of the CRL entry.

List of Data Description
This parameter is composed of the following list elements:

Data Description
This parameter contains the value of the equivalent attribute of the CRL entry.

Manufacturer Specific Data
This parameter contains the value of the equivalent attribute of the CRL entry.

Diag Upd Delay
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Ext Diag Overflow
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Deactivated
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Station Non Existent
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Invalid Slave Response
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUU and the parameter D_SAP_index has the value 60.

Fail Safe
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

DPV1 Supported
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

NA to Abort
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Ignore ACIr
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Output Len
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Exp Inp Len
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Data Transfer Status
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

System Diagnosis Status
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

Publisher Flag
This conditional parameter contains the value of the equivalent attribute of the CRL entry. It shall only be present if the parameter CREP has the value BBUB and the parameter D_SAP_index has the value NIL.

List of Structured Prm Data
This conditional parameter is only present if the parameter CR Type contains the value BQUU and the parameter D_SAP_index contains the value 53. This parameter is composed of one or more of the following list elements.

Structured User Prm Data
This parameter is composed of one or both of the following subparameters.

Device Related User Parameter
This parameter indicates the presence of Device Related User Parameter.

Device Prm Data
This parameter contains the value of the equivalent attribute of the CRL entry.

Module Related User Parameter
This parameter indicates the presence of Module Related User Parameter.

Slot Number
This parameter contains the value of the equivalent attribute of the CRL entry.

Module Prm Data
This parameter contains the value of the equivalent attribute of the CRL entry.

List of DXB Link Entries
This parameter indicates the presence of the DXB-Linktable and is composed of the following list elements.

Publisher Address
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Length
This parameter contains the value of the equivalent attribute of the CRL entry.

IsoM Parameter
This parameter indicates the presence of the IsoM Parameter.

TBASE_DP
This parameter contains the value of the equivalent attribute of the CRL entry.

T_{DP}
This parameter contains the value of the equivalent attribute of the CRL entry.

TMAPC
This parameter contains the value of the equivalent attribute of the CRL entry.

TBASE_IO
This parameter contains the value of the equivalent attribute of the CRL entry.

T_I
This parameter contains the value of the equivalent attribute of the CRL entry.

T_O
This parameter contains the value of the equivalent attribute of the CRL entry.

TPLL_W
This parameter contains the value of the equivalent attribute of the CRL entry.

TPLL_D
This parameter contains the value of the equivalent attribute of the CRL entry.

List of DXB Subscriber Entries
This parameter contains the value of the equivalent attribute of the CRL entry.

Publisher Address
This parameter contains the value of the equivalent attribute of the CRL entry.

Input Data Length Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Source Offset Publisher
This parameter contains the value of the equivalent attribute of the CRL entry.

Master Data
This parameter contains the value of the equivalent attribute of the CRL entry.

Source Offset Master
This parameter contains the value of the equivalent attribute of the CRL entry.

Dest Slot Number
This parameter contains the value of the equivalent attribute of the CRL entry.

Offset Data Area
This parameter contains the value of the equivalent attribute of the CRL entry.

Length Data Area
This parameter contains the value of the equivalent attribute of the CRL entry.

Time AR Parameter
This parameter contains the value of the equivalent attribute of the CRL entry.

Clock Sync Interval
This parameter contains the value of the equivalent attribute of the CRL entry.

CS Delay Time
This parameter contains the value of the equivalent attribute of the CRL entry.

CR Name
This parameter contains the value of the equivalent attribute of the CRL entry.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.15 CRL slave activate

The service given in Table 262 is used to set the value of the attribute Active of the CRL of the DP-master (Class 1).

Table 262 – CRL Slave activate

Parameter name	Req	Cnf
Argument	M	
Activate	M	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Activate
This parameter contains the value for the equivalent attribute of the CRL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.16 CRL slave new Prm

The service given in Table 263 is used to set the value of the attribute New Prm of the CRL of the DP-master (Class 1).

Table 263 – CRL Slave new Prm

Parameter name	Req	Cnf
Argument	M	
New Prm	M	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

New Prm
This parameter contains the value for the equivalent attribute of the CRL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.17 CRL slave new Prm data

The service given in Table 264 is used to set the values of the attribute Prm Data for DP-slaves with the device attribute DPV1 set to FALSE in the CRL of the DP-master (Class 1). This service is also used to send the Prm Command to a DP-slave with the device attribute DPV1 and Prm Command supported set to TRUE in the CRL of the DP-master (Class 1).

Table 264 – CRL Slave new Prm data

Parameter name	Req	Cnf
Argument	M	
Prm Data	M	
Lock	M	
Sync	M	
Freeze	M	
WD On	M	
WD Fact 1	M	
WD Fact 2	M	
Min Tsdr	M	
Ident Number	M	
Group Ident	M	
User Prm Data	C	
Device Prm Data	U	
Module Prm Data	U	
Prm Command	C	
Seq Number	M	
Primary Request	M	
MS1 Command	M	
Master State Clear	M	
Check Properties	M	
Primary Request used	M	
MS1 Command used	M	
Address Change	M	
Address Offset	M	
Output Hold Time	M	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Prm Data
This parameter is composed of the following subparameters.

Lock
This parameter contains the value for the equivalent attribute of the CRL entry.

Sync
This parameter contains the value for the equivalent attribute of the CRL entry.

Freeze
This parameter contains the value for the equivalent attribute of the CRL entry.

WD On
This parameter contains the value for the equivalent attribute of the CRL entry.

WD Fact 1
This parameter contains the value for the equivalent attribute of the CRL entry.

WD Fact 2
This parameter contains the value for the equivalent attribute of the CRL entry.

Min Tsdr
This parameter contains the value for the equivalent attribute of the CRL entry.

Ident Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Group Ident
This parameter contains the value for the equivalent attribute of the CRL entry.

User Prm Data
This conditional parameter is only present if the parameter Prm Structure contains the value FALSE. This parameter is composed of one or both of the following subparameters.

Device Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Module Prm Data
This parameter contains the value for the equivalent attribute of the CRL entry.

Prm Command
This conditional parameter is composed of the following subparameters.

Seq Number
This parameter contains the value for the equivalent attribute of the CRL entry.

Primary Request
This parameter contains the value for the equivalent attribute of the CRL entry.

MS1 Command
This parameter contains the value for the equivalent attribute of the CRL entry.

Master State Clear
This parameter contains the value for the equivalent attribute of the CRL entry.

Check Properties
This parameter contains the value for the equivalent attribute of the CRL entry.

Primary Request used
This parameter contains the value for the equivalent attribute of the CRL entry.

MS1 Command used
This parameter contains the value for the equivalent attribute of the CRL entry.

Address Change
This parameter contains the value for the equivalent attribute of the CRL entry.

Address Offset
This parameter contains the value for the equivalent attribute of the CRL entry.

Output Hold Time
This parameter contains the value for the equivalent attribute of the CRL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.18 Load CRL DP-master CI2

The service given in Table 265 is used to load the CRL of a DP-master (Class 2). For the use of the conditional parameters in the request primitive refer to the description of the CRL DP-master (Class 1) object.

Table 265 – Load CRL DP-master CI2

Parameter name	Req	Cnf
Argument	M	
List of CRL Entries	M	
CREP	M	
CR Type	M	
Related AREP	M	
S_SAP_index	M	
D_SAP_index	M	
D_addr	M	
Service_activate	M	
Role_in_service	M	
Max_DLSDU_length_req_low	M	
Max_DLSDU_length_req_high	M	
Max_DLSDU_length_ind_low	M	
Max_DLSDU_length_ind_high	M	
CR Name	U	
Result(+)		S
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

List of CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value for the equivalent attribute of the CRL entry.

CR Type
This parameter contains the value for the equivalent attribute of the CRL entry.

Related AREP
This parameter contains the value for the equivalent attribute of the CRL entry.

S_SAP_index
This parameter contains the value for the equivalent attribute of the CRL entry.

D_SAP_index
This parameter contains the value for the equivalent attribute of the CRL entry.

D_addr
This parameter contains the value for the equivalent attribute of the CRL entry.

Service_activate
This parameter contains the value for the equivalent attribute of the CRL entry.

Role_in_service
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_low
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_high
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_low
This parameter contains the value for the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_high
This parameter contains the value for the equivalent attribute of the CRL entry.

CR Name
This parameter contains the value for the equivalent attribute of the CRL entry.

Result(+)
This selection type parameter indicates that the service request succeeded.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.2.10.5.19 Get CRL DP-master CI2

The service given in Table 266 is used to get the CRL of a DP-master (Class 2). For the use of the conditional parameters in the request primitive refer to the description of the CRL DP-master (Class 1) object.

Table 266 – Get CRL DP-master CI2

Parameter name	Req	Cnf
Argument	M	
Result(+)		S
List of CRL Entries		M
CREP		M
CR Type		M
Related AREP		M
S_SAP_index		M
D_SAP_index		M
D_addr		M
Service_activate		M
Role_in_service		M
Max_DLSDU_length_req_low		M
Max_DLSDU_length_req_high		M
Max_DLSDU_length_ind_low		M
Max_DLSDU_length_ind_high		M
CR Name		U
Result(-)		S
Status		M

Argument
The argument shall convey the service specific parameters of the service request.

Result(+)
This selection type parameter indicates that the service request succeeded.

List of CRL Entries
This parameter is composed of the following list elements:

CREP
This parameter contains the value of the equivalent attribute of the CRL entry.

CR Type
This parameter contains the value of the equivalent attribute of the CRL entry.

Related AREP
This parameter contains the value of the equivalent attribute of the CRL entry.

S_SAP_index
This parameter contains the value of the equivalent attribute of the CRL entry.

D_SAP_index
This parameter contains the value of the equivalent attribute of the CRL entry.

D_addr
This parameter contains the value of the equivalent attribute of the CRL entry.

Service_activate
This parameter contains the value of the equivalent attribute of the CRL entry.

Role_in_service
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_low
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_req_high
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_low
This parameter contains the value of the equivalent attribute of the CRL entry.

Max_DLSDU_length_ind_high
This parameter contains the value of the equivalent attribute of the CRL entry.

CR Name
This parameter contains the value of the equivalent attribute of the CRL entry.

Result(-)
This selection type parameter indicates that the service request failed.

Status
This parameter indicates the reason for the failure.

Allowed values: NO, IV

6.3 Summary of FAL classes

This subclause contains a summary of the defined fieldbus AL Classes.

Table 267 shows the fieldbus AL Class Summary.

Table 267 – Fieldbus AL class summary

AL ASE	Class
Process Data	Simple Process Data
	Combined Process Data
I/O Data	Simple Input Data
	Simple Output Data
	Extended Input Data
	Extended Output Data
Diagnosis	Device Related Diagnosis
	Identifier Related Diagnosis
	Channel Related Diagnosis
	Status
	Module Status
Alarm	Alarm

AL ASE	Class
Context	MS0 User Parameter
	MS0 Structured User Parameter
	DXB-Linktable
	DXB-Subscribertable
	Time AR Parameter
	IsoM Parameter
	MS0 Configuration Elements
	Remanent Parameter
	MS2 User Parameter
	Red Context
Load Region	Load Region
Function Invocation	Function Invocation
	Action
Time	Slave Time
	Link Time
Management	Master Diag
	Master Parameter
Application Relationship List	ARL DP-slave
	ARL DP-master (Class 1)
	ARL DP-master (Class 2)
Communication Relationship List	CRL DP-slave
	CRL DP-master (Class 1)
	CRL DP-master (Class 2)

6.4 Permitted FAL services by AREP role

Table 268 gives an overview of the requester and responder roles in service execution. Some services are additionally possible. Requester and responder roles in service execution are given separately.

Table 268 – Assignment of the services to DP-masters and DP-slaves

Service	DP-slave				DP-master (Class 1)				DP-master (Class 2)		
	MS0	MS1	MS2	MS3	MS0	MS1	MS3	MM1/ MM2	MS2	MS0	MM1/ MM2
Read		p_R	p_R			I			I		
Write		p_R	p_R			I			I		
Data Transport			p_R						I		
Set Input	LR										
Read Input	i_(R)									I	
Get Input					LR						
New Input					LI						
Set Output					LR						
Read Output	i_(R)									I	
Get Output	LR										
New Output	LI										
Global Control	R				I						
Get Publisher Data	LR										
New Publisher Data	LI										
Publisher Active	LI										
SYNCH					LI						
SYNCH Delayed					LI						
DX Finished					LI						
SYNCH Event	LI										
Set Slave Diag	LR										
Get Slave Diag					LR						
Read Slave Diag	i_(R)									I	
New Slave Diag					LI						
Alarm Notification		I				R					
Alarm Ack		p_R				I					
Initiate			p_R						I		
Abort			R/I						I/R		
Check User Prm	LI										
Check User Prm Result	LR										
Check Ext User Prm	LI										
Check Ext User Prm Result	LR										
Set Cfg	LR										
Get Cfg	i_(R)									I	
Check Cfg	LI										

Service	DP-slave				DP-master (Class 1)				DP-master (Class 2)		
	MS0	MS1	MS2	MS3	MS0	MS1	MS3	MM1/ MM2	MS2	MS0	MM1/ MM2
Check Cfg Result	LR										
Set Slave Add	R									I	
Set Time				R			I				
Sync Interval Violation				LI							
Get Master Diag								p_R			I
Start Seq								p_R			I
Download								p_R			I
Upload								p_R			I
End Seq								p_R			I
Act Para Brct								R			I
Act Param								p_R			I

I	Initiator of a service
R	Receiver (Responder) of a service
(R)	Receiver (Responder) of a service (but no interaction with the application)
LR	Local service initiated by the application
LI	Local service initiated by the DP-Application Layer
i_	this prefix indicates that the response will be sent immediate, without polling.
p_	this prefix indicates that the response will be sent not immediate, polling required.

Table 269 shows the support of ARs in the different DP-Device Types.

Table 269 – Support of AR types in the different DP-device types

AR Type	DP-slave	DP-master (Class 1)	DP-master (Class 2)
MS0	M	M	O
MS1	O	O	—
MS2	O	—	O
MS3	O	O	O
MM1	—	M	O
MM2	—	O	O

M	AR Type has to be supported mandatory.
O	AR Type is supported optionally.
--	AR Type not supported.

Table 270 gives an overview of the services which shall be supported at the different AREPs respectively CREPs.

Table 270 – Support of services at the different AREPs respectively CREPs

AR Type	D_SAP _index	CR Type	CR Name	Supported service at the AREP/ CREP		
				DP-slave	DP-master (Class 1)	DP-master (Class 2)
MS0	NIL	BBUB	Data Exchange	Set Input (M)	—	—
				Get Output (M)	—	—
				New Output (O)	—	—
				SYNCH Event (O)	—	—
				—	Set Output (M)	—
				—	Get Input (M)	—
				—	New Input (O)	—
				—	SYNCH (O)	—
				—	SYNCH Delayed (O)	—
				—	DX Finished (O)	—
		BBUU-OM	DXB	Get Publisher Data (O)	—	—
				New Publisher Data (O)	—	—
				Publisher Active (O)	—	—
	62	BQUU	Cfg	Set Cfg (M)	—	—
				Check Cfg (M)	—	—
				Check Cfg Result (M)	—	—
	61	BQUU	Prm	Check User Prm (M)	—	—
				Check User Prm Result (M)	—	—
	53	BQUU	Ext Prm	Check Ext User Prm (O)	—	—
				Check Ext User Prm Result (O)	—	—
	60	BBUU	Slave Diag	Set Slave Diag (M)	—	—
				—	Get Slave Diag (M)	—
				—	New Slave Diag (O)	—
	58	QQUU-OM	Global Control	Global Control (M)	Global Control (M)	—
	60	QBCB	Read Diag	Read Slave Diag (M)	—	Read Slave Diag (O)
	59	QBCB	Read Cfg	Get Cfg (M)	—	Get Cfg (O)
	57	QBCB	Read Output	Read Output (M)	—	Read Output (O)
	56	QBCB	Read Input	Read Input (M)	—	Read Input (O)
	55	QBCB	Rem Para	Set Slave Add (O)	—	Set Slave Add (O)
MS1	51	QQOB-CO	Process Data	Read (O)	Read (O)	—
			Alarm Ack	Write (O)	Write (O)	—
				Alarm Ack (O)	Alarm Ack (O)	—
				Initiate Load (O)	Initiate Load (O)	—
				Push Segment (O)	Push Segment (O)	—
				Pull Segment (O)	Pull Segment (O)	—
				Terminate Load (O)	Terminate Load (O)	—
				Start (O)	Start (O)	—
				Stop (O)	Stop (O)	—
				Resume (O)	Resume (O)	—

AR Type	D_SAP _index	CR Type	CR Name	Supported service at the AREP/ CREP		
				DP-slave	DP-master (Class 1)	DP-master (Class 2)
				Reset (O)	Reset (O)	—
				Get FI State (O)	Get FI State (O)	—
				Call (O)	Call (O)	—
	50	QQOB-CO	Alarm Ack	Alarm Ack (O)	Alarm Ack (O)	—
	60	QQUU-OO	Alarm	Alarm Notification (O)	Alarm Notification (O)	—
MS2	49	QQUU-OO	Res Mng	Initiate (O)	—	Initiate (O)
	0 - 48	QQCB-CO	Process Data	Abort (O)	—	Abort (O)
				Read (O)	—	Read (O)
				Write (O)	—	Write (O)
				Data Transport (O)	—	Data Transport (O)
				Initiate Load (O)	—	Initiate Load (O)
				Push Segment (O)	—	Push Segment (O)
				Pull Segment (O)	—	Pull Segment (O)
				Terminate Load (O)		Terminate Load (O)
				Start (O)	—	Start (O)
				Stop (O)	—	Stop (O)
				Resume (O)	—	Resume (O)
				Reset (O)	—	Reset (O)
				Get FI State (O)	—	Get FI State (O)
				Call (O)	—	Call (O)
MS3	CS	QQUU-OM	CS	Set Time (O)	Set Time (O)	—
				Sync Interval Violation(O)	—	—
MM1	54	QBCB-CL	MM	—	Get Master Diag (O)	Get Master Diag (O)
				—	Start Seq (O)	Start Seq (O)
				—	Download (O)	Download (O)
				—	Upload (O)	Upload (O)
				—	End Seq (O)	End Seq (O)
				—	Act Param (O)	Act Param (O)
MM2	54	QQUU-OM	Act Para Brct	—	Act Para Brct (O)	Act Para Brct (O)

M mandatory service at this AREP / CREP

O optional service at this AREP / CREP

– non-existent service at this AREP / CREP

6.5 Conformance classes

Conformance Classes are used to define the minimum functionality of a DP-master. There are no Conformance Classes for DP-slaves. The functionality of the DP-slave determines the necessary conformance class of the DP-master.

For the cyclic data exchange the DP-master (Class 1) has to support the following features:

– min. 2 words consistency,

– min. 32 octet data length,
– DPV1 data types; except: Date, Time-Of-Day, Time-Difference (for the Extended Configuration Identifier Format),
– Fail Safe has to be supported,
– min. 32 octet Diag Data.

For the acyclic communication the DP-master has to support the maximum data length of 244 octets.

Table 271 and Table 272 show the Conformance Classes of DP-master Class1 and Class 2.

Table 271 – Conformance classes DP-master (class 1)

Conformance Class	Functionality MS1
A	Read Write
B	Read Write Alarm Notification Alarm Ack

Table 272 – Conformance classes DP-master (class 2)

Conformance Class	Functionality MS2
A	Read Write Initiate Abort Data Transport

6.6 Application characteristics

6.6.1 Time behavior

The following diagram (Figure 44) shows the theoretical system reaction time of a DP-master (Class 1) in dependence of the connected amount of DP-slaves with 2 octets input- and output-data each.

It is presumed that all configured DP-slaves are in the data exchange mode with no diagnosis message pending. The DP-slaves answer immediately after the minimum Station Delay Timer (minT_{SDR} = 11 t_{BIT}) is expired. At the DP-master (Class 1) T_{SDI} is assumed as 37 bit times.

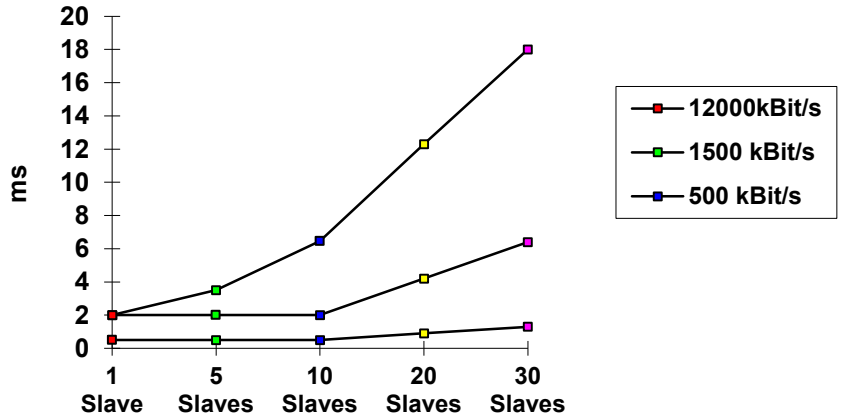


Figure 44 – Cycle time of the DP system

6.6.2 Ident number

The Ident Number is necessary for all DP-Devices except for DP-master (Class 2). For each type of device an individual Ident Number shall be used. It identifies a DP-Device type.

NOTE An Ident Number is not a series number.

If a manufacturer got an Ident Number for one type of device it is allowed to use this number for all produced devices of the same type.

Modular devices of the same type but with differences in the amount of inputs and outputs may use the same Ident Number for each produced DP-Device, if these devices are described in the GSD as a modular device.

Bibliography

IEC 61158-6-10:2014, *Industrial communication networks – Fieldbus specifications – Part 6-10: Application layer protocol specification – Type 10 elements*

IEC 61784-1:2014, *Industrial communication networks – Profiles – Part 1: Fieldbus profiles*

IEC 61784-2:2014, *Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3*

_____

SOMMAIRE

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 5-3: Définition des services de la couche application – Éléments de type 3

AVANT-PROPOS

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

L'attention est attirée sur le fait que l'utilisation du type de protocole associé est restreinte par les détenteurs des droits de propriété intellectuelle. En tout état de cause, l'engagement de renonciation partielle aux droits de propriété intellectuelle pris par les détenteurs de ces droits autorise l'utilisation d'un type de protocole de couche avec les autres protocoles de couche du même type, ou dans des combinaisons avec d'autres types autorisées explicitement par les détenteurs des droits de propriété intellectuelle pour ce type.

NOTE Les combinaisons de types de protocoles sont spécifiées dans la CEI 61784-1 et la CEI 61784-2.

La Norme internationale CEI 61158-5-3 a été établie par le sous-comité 65C: Réseaux industriels, du comité d'études 65 de la CEI: Mesure, commande et automation dans les processus industriels.

Cette troisième édition annule et remplace la deuxième édition, parue en 2010. Cette édition constitue une révision technique.

La principale modification par rapport à l'édition précédente est énumérée ci-dessous:

- Correction de l'orthographe et de la formulation pour une meilleure lisibilité.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
65C/763/FDIS	65C/773/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Une liste de toutes les parties de la série CEI 61158, publiées sous le titre général *Réseaux de communication industriels – Spécifications des bus de terrain*, peut être consultée sur le site web de la CEI.

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

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

IMPORTANT – Le logo *"colour inside"* qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.

INTRODUCTION

La présente partie de la CEI 61158 est l'une d'une série produite pour faciliter l'interconnexion de composants d'un système d'automation. Elle est liée à d'autres normes de la série telle que définie par le modèle de référence des bus de terrain "à trois couches" décrit dans la CEI 61158-1:2013.

Le service application est fourni par le protocole d'application utilisant les services disponibles de la liaison de données ou autre couche immédiatement inférieure. La présente norme définit les caractéristiques de services d'application qui peuvent être exploitées par les applications de bus de terrain et/ou la gestion de système.

Dans toute la série de normes relatives aux bus de terrain, le terme “service” se réfère à la capacité abstraite fournie par une couche du modèle de référence de base de l'Interconnexion des systèmes ouverts (OSI) à la couche immédiatement supérieure. Ainsi, le service de la couche application défini dans la présente norme est un service architectural conceptuel, indépendant des divisions administratives et de mise en œuvre.

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 5-3: Définition des services de la couche application – Éléments de type 3

1 Domaine d'application

1.1 Généralités

La présente norme est l'une d'une série produite pour faciliter l'interconnexion de composants d'un système d'automation. Elle est liée à d'autres normes de la série telle que définie par le modèle de référence des bus de terrain "à trois couches" décrit dans la CEI 61158-1.

La présente sous-partie contient du matériau spécifique au bus de terrain de Type 3.

La couche application de bus de terrain (FAL «Fieldbus Application Layer») fournit aux programmes d'utilisateur un moyen d'accéder à l'environnement de communication du bus de terrain. À cet égard, la FAL peut être vue comme une «fenêtre entre des programmes d'application correspondants».

La présente norme fournit les éléments communs pour les communications de messagerie de base prioritaire et non prioritaire entre des programmes d'application dans un environnement d'automatisation et le matériau spécifique au bus de terrain de Type 3. Le terme "prioritaire" (ou "à temps critique") sert à représenter la présence d'une fenêtre temporelle, dans les limites de laquelle une ou plusieurs actions spécifiées sont tenues d'être parachevées avec un certain niveau défini de certitude. Le manquement à parachever les actions spécifiées dans les limites de la fenêtre temporelle risque d'entraîner la défaillance des applications qui demandent ces actions, avec le risque concomitant pour l'équipement, l'installation et éventuellement pour la vie humaine.

La présente norme définit de manière abstraite le service visible de l'extérieur fourni par les différents types de la couche application de bus de terrain en termes

a) d'un modèle abstrait pour définir des ressources (objets) d'application capables d'être manipulées par les utilisateurs par l'intermédiaire de l'utilisation du service FAL;
b) des actions et événements primitifs du service;
c) des paramètres associés à chaque action primitive et événement primitif, et la forme qu'ils prennent; et
d) l'interrelation entre ces actions et événements, et leurs séquences valides.

Le but de la présente norme est de définir les services fournis à

a) l'utilisateur de FAL à la frontière entre l'utilisateur et la couche application du modèle de référence de bus de terrain; et
b) la gestion des systèmes au niveau de la frontière entre la couche application et la Gestion des systèmes selon le modèle de référence de bus de terrain.

La présente norme spécifie la structure et les services de la couche application des bus de terrain de la CEI, en conformité avec le Modèle de référence de base de l'OSI (ISO/CEI 7498-1) et la Structure de la couche application de l'OSI (ISO/CEI 9545).

Les services et protocoles de la FAL sont fournis par des entités d'application (AE, «Application Entity») de la FAL contenues dans les processus d'application. L'AE de la FAL

se compose d'un jeu d'éléments de service application (ASE, «Application Service Element») orientés objet et d'une entité de gestion de couche (LME, «Layer Management Entity») qui gère l'AE. Les ASE fournissent des services de communication qui fonctionnent sur un jeu de classes d'objets de processus d'application (APO, «Application process object») connexes. L'un des ASE de la FAL est un ASE de gestion qui fournit un jeu commun de services pour la gestion des instances de classes de la FAL.

Bien que ces services spécifient, du point de vue des applications, la manière dont la demande et les réponses sont émises et délivrées, ils n'incluent pas une spécification de ce que les applications qui demandent et qui répondent sont tenues d'en faire. À savoir, les aspects comportementaux des applications ne sont pas spécifiés; seule une définition des demandes et réponses qu'elles peuvent envoyer/recevoir est spécifiée. Cela permet une plus grande flexibilité aux utilisateurs de la FAL pour normaliser un tel comportement d'objet. En plus de ces services, certains services d'appui sont également définis dans la présente norme pour fournir l'accès à la FAL afin de maîtriser certains aspects de son fonctionnement.

1.2 Spécifications

L'objectif principal de la présente norme est de spécifier les caractéristiques des services conceptuels d'une couche application qui sont adaptées à des communications prioritaires et, donc, complètent le Modèle de référence de base de l'OSI en guidant le développement des protocoles de la couche application pour les communications à temps critique.

Un objectif secondaire est de fournir des chemins de migration à partir de protocoles de communications industrielles préexistants. C'est ce dernier objectif qui donne naissance à la diversité des services normalisés comme les divers Types de la CEI 61158, et les protocoles correspondants normalisés dans les sous-parties de la CEI 61158-6.

La présente spécification peut être utilisée comme la base pour les interfaces de programmation d'applications (Application Programming-Interfaces) formelles. Néanmoins, elle n'est pas une interface de programmation formelle et il est nécessaire pour toute interface de ce type de traiter de questions de mise en œuvre qui ne sont pas couvertes par la présente spécification, y compris

a) les tailles et l'ordonnancement des octets pour les divers paramètres de service à plusieurs octets, et

b) la corrélation de primitives appariées "request-confirm" (c'est-à-dire: demande et confirmation) ou "indication-response" (c'est-à-dire: indication et réponse).

1.3 Conformité

La présente norme ni ne spécifie de mises en œuvre individuelles ou de produits individuels ni ne contraint les mises en œuvre d'entités de la couche application au sein des systèmes d'automation industriels.

Il n'y a pas de conformité d'équipement à la présente norme de définition des services de la couche application. Au contraire, la conformité est obtenue par une mise en œuvre de protocoles conformes de la couche application qui satisfont à tout type donné de services de la couche application définis dans la présente partie de la CEI 61158.

2 Références normatives

Les documents suivants sont cités en référence de manière normative, en intégralité ou en partie, dans le présent document et sont indispensables pour son application. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

NOTE Toutes les parties de la série CEI 61158, ainsi que la CEI 61784-1 et la CEI 61784-2 font l'objet d'une maintenance simultanée. Les références croisées à ces documents dans le texte se rapportent par conséquent aux éditions datées dans la présente liste de références normatives.

CEI 61131-1, *Automates programmables – Partie 1: Informations générales*

CEI 61158-1:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 1: Présentation et lignes directrices des séries CEI 61158 et CEI 61784*

CEI 61158-3-3:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 3-3: Définition des services de la couche liaison de données – Eléments de type 3*

CEI 61158-4-3:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 4-3: Spécification du protocole de la couche liaison de données – Eléments de type 3*

CEI 61158-5-10:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 5-10: Définition des services de la couche application – Eléments de type 10*

CEI 61158-6-3:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 6-3: Spécification du protocole de la couche application – Eléments de type 3*

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

ISO/CEI 7498-3, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base: Dénomination et adressage*

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

ISO/IEC 8824-1, *Information Technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation* (disponible en anglais seulement)

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

ISO/CEI 10731, *Technologies de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Conventions pour la définition des services OSI*

3 Termes, définitions, abréviations, symboles et conventions

Pour les besoins du présent document, les termes, définitions, abréviations, symboles et conventions suivants, tels que définis dans l'ISO/CEI 7498-1 s'appliquent:

3.1 Termes et définitions référencés

3.1.1 Termes de l'ISO/CEI 7498-1

a) entité d'application

b) processus d'application

c) unité de données de protocole application

d) élément de service application

e) invocation d'entité d'application

f) invocation de processus d'application

g) transaction d'application

h) système ouvert réel

i) syntaxe de transfert.

3.1.2 Termes de l'ISO/CEI 8822

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 8822 s'appliquent:

a) syntaxe abstraite

b) contexte de présentation.

3.1.3 Termes de l'ISO/CEI 9545

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 9545 s'appliquent:

a) application-association (association d'applications)

b) application-context (contexte d'application)

c) application context name (nom de contexte d'application)

d) application-entity-invocation (invocation d'entité d'application)

e) application-entity-type (type d'entité d'application)

f) application-process-invocation (invocation de processus d'application)

g) application-process-type (type de processus d'application)

h) application-service-element (élément de service application)

i) application control service element (élément de service de contrôle d'application).

3.1.4 Termes de l'ISO/CEI 8824-1

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 8824 s'appliquent:

a) identificateur d'objet

b) type

3.1.5 Termes de la couche liaison de données de bus de terrain

Pour les besoins du présent document, les termes suivants tels que définis dans la CEI 61158-3-3 et dans la CEI 61158-4-3 s'appliquent.

a) DL-time (temps de liaison de données)

b) DL-Scheduling-policy (politique de programmation de liaison de données)

c) DLCEP

d) DLC

e) DL-connection-oriented mode (mode orienté connexion de liaison de données)

f) DLPDU

g) DLSDU

h) DLSAP

i) fixed tag (étiquette fixe)

j) generic tag (étiquette générique)

k) link (liaison)

l) MAC ID (identificateur MAC)

m) network address (adresse réseau)

n) node address (adresse de nœud)

o) node (nœud)

p) tag (indicateur)

q) scheduled (programmé)

r) unscheduled (non programmé).

3.2 Couche application des bus de terrain de Type 3 – termes et définitions spécifiques

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

3.2.1
protection d'accès
limitation de l'utilisation d'un objet d'application à un seul client

3.2.2
table d'affectation d'adresses
mapping du stockage d'objets de données E/S internes du client avec les objets de données d'entrée et de sortie décentralisées

3.2.3
allouer
prendre une ressource dans une zone commune et affecter la ressource en question à l'usage exclusif d'une entité spécifique

3.2.4
application
fonction ou structure de données pour laquelle des données sont consommées ou produites

3.2.5
interopérabilité de la couche application
capacité des entités d'application d'accomplir des opérations coordonnées et coopératives en utilisant les services de la FAL

3.2.6
objets d'application
classes d'objets multiples qui gèrent et assurent un échange de messages pendant le mode exécution à travers le réseau et à l'intérieur de l'appareil de réseau

3.2.7
identificateur de processus d'application
identifie de multiples processus d'application utilisés dans un appareil

3.2.8
objet de processus d'application
composant d'un processus d'application qui est identifiable et accessible par l'intermédiaire d'une relation entre applications de la FAL

Note 1 à l'article: Les définitions d'objet de processus d'application se composent d'un jeu de valeurs pour les attributs de leur classe (voir la définition de "classe d'objets de processus d'application"). Les définitions d'objet de processus d'application sont accessibles à distance à l'aide des services de l'ASE FAL Object Management (Gestion d'objet de la FAL). Les services de Gestion d'objet de la FAL peuvent être utilisés pour charger ou mettre à jour des définitions d'objet, pour lire des définitions d'objet, et pour créer et supprimer de manière dynamique des objets d'application et leurs définitions correspondantes.

3.2.9
classe d'objets de processus d'application
classe d'objets de processus d'application définie en termes du jeu de leurs attributs et services accessibles par le réseau

3.2.10
relation entre applications
association coopérative entre deux ou plusieurs invocations d'entités d'application (application-entity-invocation) à des fins d'échange d'informations et de coordination de leur fonctionnement conjoint.

Note 1 à l'article : Cette relation est activée soit par l'échange d'unités de données de protocole d'application, soit à la suite d'activités de préconfiguration.

3.2.11
élément de service d'application de relation entre applications
application-service-element (élément de service d'application) qui fournit le moyen exclusif d'établir et de faire cesser toutes les relations entre applications

3.2.12
point d'extrémité de relation entre applications
contexte et comportement d'une relation entre applications tels que vus et maintenus par l'un des processus d'application impliqués dans la relation entre applications

Note 1 à l'article: Chaque processus d'application impliqué dans la relation entre applications maintient son propre point d'extrémité de relation entre applications.

3.2.13
attribut
description d'une caractéristique ou fonction visible de l'extérieur d'un objet

Note 1 à l'article: Les attributs d'un objet contiennent de l'information relative à des parties variables d'un objet. Typiquement, ils fournissent des informations de statut ou régissent le fonctionnement d'un objet. Des attributs peuvent aussi avoir une incidence sur le comportement d'un objet. Les attributs se répartissent en attributs de classes et attributs d'instances.

3.2.14
comportement
indication de la façon dont un objet réagit à des événements particuliers

3.2.15
canal
simple liaison physique ou logique d'un objet d'application d'entrée ou de sortie d'un serveur au processus.

3.2.16
diagnostic lié à une voie
informations concernant un élément spécifique d'un objet d'application d'entrée ou de sortie, fournies à des fins de maintenance

EXEMPLE: validité de données

3.2.17
classe
ensemble d'objets, qui représentent tous le même type de composant système

Note 1 à l'article: Une classe est une généralisation d'un objet; un modèle pour définir des variables et des méthodes. Tous les objets dans une classe ont une forme et un comportement identiques, mais contiennent en général des données différentes dans leurs attributs.

3.2.18
attributs de classe
attribut qui est partagé par tous les objets au sein de la même classe

3.2.19
code de classe
identificateur unique attribué à chaque classe d'objets

3.2.20
service spécifique à une classe
service défini par une classe d'objets particulière pour accomplir une fonction requise qui n'est pas accomplie par un service commun

Note 1 à l'article: Un objet spécifique à une classe est unique pour la classe d'objets qui le définit.

3.2.21
client
<vue objet> objet qui utilise les services d'un autre objet (serveur) pour accomplir une tâche

3.2.22
client
<vue communication> initiateur d'un message auquel un serveur réagit

3.2.23
vérification de configuration
comparaison de la structuration attendue de l'objet de données E/S côté client avec la structuration effective de l'objet de données E/S côté serveur dans la phase de démarrage

3.2.24
défaut de configuration
différence inacceptable entre la structuration attendue de l'objet de données E/S et la structuration effective de l'objet de données E/S, telle que détectée par le serveur

3.2.25
identificateur de configuration
représentation d'une partie des données E/S d'un seul module d'entrée et/ou de sortie d'un serveur

3.2.26
connexion
liaison logique entre deux objets d'application qui peuvent se trouver au sein du même appareil ou dans des appareils différents

Note 1 à l'article: Les connexions sont soit point à point, soit multipoint.

Note 2 à l'article: La liaison logique entre puits et source d'attributs et de services en différentes interfaces personnalisées des ASE RT-Auto est appelée "interconnexion". Une distinction est faite entre interconnexion de données et interconnexion d'événements. L'ensemble de la liaison logique et du flot de données entre puits et source d'éléments de données d'automation est appelé "interconnexion de données". L'ensemble de la liaison logique et du flot de données entre puits (méthode) et source (événement) de services opérationnels est appelé "interconnexion d'événements".

3.2.27
consommer
recevoir des données d'un producteur

3.2.28
consommateur
nœud ou puits qui reçoit des données d'un producteur

3.2.29
application consommatrice
application qui consomme des données

3.2.30
cyclique
répétitif d'une manière régulière

3.2.31
cohérence de données
moyen pour une émission et un accès cohérents de l'objet de donnée d'entrée ou de sortie entre client et serveur et au sein du client et du serveur

3.2.32
adresse de DL par défaut
valeur 126 comme valeur initiale pour l'adresse de DL, qu'il est nécessaire de changer (par exemple: attribution d'une adresse de DL par le biais du bus de terrain) avant le fonctionnement avec le maître DP (classe 1) [«DP-master (Class 1)»]

3.2.33
appareil
équipement matériel physique relié à la liaison

Note 1 à l'article: Un appareil peut contenir plus d'un nœud.

3.2.34
informations de diagnostic
toute donnée disponible au niveau du serveur à des fins de maintenance

3.2.35
regroupement d'informations de diagnostic
informations de diagnostic système qui sont regroupées du côté client

3.2.36
maître DP (classe 1)
DP-master (Class 1)
appareil de commande qui commande plusieurs esclaves DP (appareils de terrain)

Note 1 à l'article: Il s'agit habituellement d'un appareil de commande programmable ou d'un système de commande distribué.

3.2.37
maître DP (classe 2)
DP-master (Class 2)
appareil de commande qui gère des données de configuration (jeux de paramètres) et des données de diagnostic d'un maître DP (Classe 1), et qui peut accomplir en plus toutes les capacités de communication d'un maître DP (Classe 1)

3.2.38
esclave DP
DP-slave
appareil de terrain qui peut être affecté à un maître DP (Classe 1) comme fournisseur pour un échange cyclique de données E/S; en outre des fonctions et alarmes acycliques pourraient être fournies

3.2.39
point d'extrémité
l'une des entités en communication impliquées dans une connexion

3.2.40
ingénierie
terme abstrait qui caractérise l'application ou l'appareil client qui a la responsabilité de configurer un système d'automation par le biais d'éléments de données d'interconnexion

3.2.41
erreur
discordance entre une valeur ou un état calculé(e), observé(e) ou mesuré(e) et la valeur ou l'état spécifié(e) ou théoriquement correct(e)

3.2.42
classe d'erreurs
regroupement général pour des définitions d'erreurs connexes et des codes d'erreurs correspondants

3.2.43
code d'erreur
identification d'un type spécifique d'erreur dans une classe d'erreurs

3.2.44
événement
instance d'un changement de conditions

3.2.45
sous-réseau FAL
sous-réseaux constitués d'un ou plusieurs segments de liaison de données, identifiés par un sous-ensemble de l'adresse réseau

Note 1 à l'article: Les sous-réseaux FAL sont autorisés à contenir des ponts, mais pas des routeurs.

3.2.46
freeze
geler
fonction au niveau des esclaves DP pour le transfert simultané de données entre l'objet de données d'entrée et le processus

3.2.47
groupe
<en général> terme général pour un ensemble d'objets. Usages spécifiques:

3.2.48
groupe
<adressage> lors de la description d'une adresse, adresse qui identifie plus d'une entité

3.2.49
groupe
<Type 3> ensemble des esclaves DP qui accomplissent une fonction Freeze (geler) ou Sync (synchroniser)

3.2.50
interface
<général> frontière partagée entre deux unités fonctionnelles, définies par les caractéristiques fonctionnelles, caractéristiques de signal ou autres caractéristiques selon le cas approprié

3.2.51
interface
<FAL> ensemble d'attributs et de services d'une classe de FAL qui représente une vue spécifique sur la classe de FAL

3.2.52
pointeur d'interface
attribut-clé qui adresse sans ambiguïté une instance d'interface d'objets

3.2.53
invocation
acte consistant à utiliser un service ou une autre ressource d'un processus d'application

Note 1 à l'article: Chaque invocation représente un processus distinct de commande qui peut être décrit par son contexte. Une fois que le service s'achève ou que l'utilisation de la ressource est libérée, l'invocation cesse d'exister. Pour des invocations de service, un service qui a été lancé, mais n'est pas encore achevé, s'appelle invocation de service en cours. Également, pour des invocations de service, un "Invoke ID" peut être utilisé pour identifier sans ambiguïté l'invocation de service et la différencier d'autres invocations de service en cours.

3.2.54
données E/S
objet désigné pour être transféré de façon cyclique à des fins de traitement

3.2.55
ident number
type de l'appareil maître DP (Classe 1) ou esclave DP

3.2.56
diagnostic relatif à l'identificateur
informations dédiées aux modules à des fins de maintenance

3.2.57
indice
adresse d'un objet au sein d'un processus d'application

3.2.58
instance
occurrence physique réelle d'un objet au sein d'une classe qui identifie l'un des plusieurs objets au sein de la même classe d'objets

EXEMPLE "California" (La Californie) est une instance de la classe d'objets "state" (état).

Note 1 à l'article: Les termes "objet", "instance" et "instance d'objet" sont utilisés pour se référer à une instance spécifique.

3.2.59
instancié
objet qui a été créé dans un appareil

3.2.60
ID de fabricant
identification de chaque fabrication de produit par un numéro unique

3.2.61
jeu de paramètres de maître
données de configuration et de paramétrisation de tous les esclaves DP qui sont assignés au maître DP correspondant et aux paramètres de bus

3.2.62
membre
élément d'un attribut qui est structuré comme étant un élément d'une matrice

3.2.63
routeur de message
objet au sein d'un nœud qui distribue les demandes de messagerie vers les objets d'application appropriés

3.2.64
méthode
synonyme pour un service opérationnel qui est fourni par l'ASE serveur et invoqué par un client

3.2.65
module
<général> composant matériel ou logiciel d'un appareil physique

3.2.66
module
<Type 3> unité adressable à l'intérieur de l'esclave DP

3.2.67
réseau
ensemble de nœuds reliés par un certain type de support de communication, notamment des répétiteurs intermédiaires, ponts, routeurs et passerelles de couche inférieure

3.2.68
objet
représentation abstraite d'un composant particulier au sein d'un appareil, habituellement un ensemble de données connexes (sous la forme de variables) et de méthodes (procédures) pour effectuer des opérations sur les données en question qui ont une interface et un comportement clairement définis

3.2.69
appareil physique
<général> appareil d'automation ou autre appareil réseau

3.2.70
données de processus
objet(s) déjà prétraité(s) et transféré(s) de façon acyclique à des fins d'informations ou de traitement ultérieur

3.2.71
produire
envoyer des données destinées à être reçues par un consommateur

3.2.72
producteur
nœud qui est chargé d'envoyer des données

3.2.73
fournisseur
source d'une connexion de données

3.2.74
éditeur
rôle d'un point d'extrémité d'AR qui émet des APDU sur le bus de terrain en vue de leur consommation par un ou plusieurs abonnés

Note 1 à l'article: Un éditeur peut ne pas connaître l'identité ou le nombre d'abonnés et il peut éditer ses APDU en utilisant une AR spécialisée.

3.2.75
configuration réelle
structure des données d'entrée et sortie de l'esclave DP, y compris la définition de la cohérence des données

3.2.76
ressource
capacité de traitement ou d'informations d'un sous-système

3.2.77
serveur
<vue de communication> rôle d'un AREP dans lequel il retourne une APDU de réponse de service confirmée au client qui a émis la demande

3.2.78
serveur
<vue d'objet> objet qui fournit des services à un autre objet (client)

3.2.79
service
opération ou fonction accomplie par un objet et/ou une classe d'objets à la demande d'un autre objet et/ou d'une autre classe d'objets

3.2.80
intervalle de temps
adresse d'un module au sein d'un esclave DP

3.2.81
abonné
rôle d'un AREP dans lequel il reçoit des unités APDU produites par un éditeur

3.2.82
sync
fonction au niveau des esclaves DP pour le transfert simultané de données entre l'objet de données de sortie et le processus

3.2.83
cible
nœud d'extrémité auquel une connexion est établie

3.2.84
service non connecté
service de messagerie qui ne repose pas sur l'établissement d'une connexion entre les appareils avant d'autoriser des échanges d'informations

3.3 Symboles et abréviations

AE	Application entity (Entité d'application)
AL	Application Layer (Couche d'application)
ALME	Application Layer Management Entity (Entité de gestion de la couche application)
ALP	Application Layer Protocol (Protocole de la couche d'application)
APO	Application Object (Objet d'application)
AP	Application Process (Processus d'application)
APDU	Application Protocol Data Unit (Unité de données de protocole d'application)
API	Application Process Identifier (Interface de Programmation d'Application)

AR Application Relationship (Relation entre applications)

AREP Application Relationship End Point (Point d'extrémité de relation entre applications)

ASE Application Service Element (Élément de service d'application)

Cnf Confirmation

CR Communication Relationship (Relation de communication)

CREP Communication Relationship End Point (Point d'extrémité de relation de communication)

DL- Préfixe relatif à la couche de liaison de données

DLL Data Link Layer (Couche liaison de données)

DLM Data Link Management (Gestion de liaison de données)

DLSAP Data link Service Access Point (Point d'accès au service liaison de données)

DLSDU DL-service-data-unit (Unité de données de service liaison de données)

DP Decentralised Peripherals (Périphériques décentralisés)

FAL Fieldbus application layer (Couche application de bus de terrain)

ID Identificateur

CEI Commission Électrotechnique Internationale

Ind Indication

ISO Organisation internationale de normalisation

LME Layer Management Entity (Entité de gestion de couche)

OSI Open Systems Interconnect (Interconnexion des systèmes ouverts)

PDev Physical Device (Appareil physique)

PDU Protocol Data Unit (Unité de données de protocole)

PL Physical Layer (Couche physique)

QoS Quality of Service (Qualité de service)

Req Request (Demande)

Rsp Response (Réponse)

SAP Service Access Point (Point d'accès au service)

3.4 Conventions

3.4.1 Vue d'ensemble

La couche FAL est définie comme un jeu d'éléments ASE orientés objet. Chaque ASE est spécifié dans un paragraphe distinct. Chaque spécification d'ASE est constituée de deux parties, à savoir sa spécification de classe et sa spécification de services.

La spécification de classe définit les attributs de la classe. Les attributs sont accessibles à partir d'instances de la classe en utilisant les services d'ASE de gestion d'objets spécifiés à l'Article 5 de la présente norme. La spécification de services définit les services qui sont fournis par l'ASE.

3.4.2 Conventions générales

La présente norme utilise les conventions descriptives données dans l'ISO/CEI 10731.

3.4.3 Conventions pour les définitions de classes

Les définitions de classes sont décrites à l'aide de modèles. Chaque modèle est constitué d'une liste d'attributs de la classe. La forme générale du modèle est montrée ci-dessous:

FAL ASE:			**Nom de l'ASE**
CLASS:			Nom de classe
CLASS ID:			#
PARENT CLASS:			Nom de la classe parent
ATTRIBUTES:			
1	(o)	Attribut-clé:	Identificateur numérique
2	(o)	Attribut-clé:	nom
3	(m)	Attribut:	nom d'attribut(valeurs)
4	(m)	Attribut:	nom d'attribut(valeurs)
4.1	(s)	Attribut:	nom d'attribut(valeurs)
4.2	(s)	Attribut:	nom d'attribut(valeurs)
4.3	(s)	Attribut:	nom d'attribut(valeurs)
5.	(c)	Contrainte:	expression de la contrainte
5.1	(m)	Attribut:	nom d'attribut(valeurs)
5.2	(o)	Attribut:	nom d'attribut(valeurs)
6	(m)	Attribut:	nom d'attribut(valeurs)
6.1	(s)	Attribut:	nom d'attribut(valeurs)
6.2	(s)	Attribut:	nom d'attribut(valeurs)
SERVICES:			
1	(o)	OpsService:	nom du service
2.	(c)	Contrainte:	expression de la contrainte
2.1	(o)	OpsService:	nom du service
3	(m)	MgtService:	nom du service

(1) La rubrique "FAL ASE:" est le nom de l'élément ASE de la couche FAL (FAL ASE) qui fournit les services pour la classe spécifiée.

(2) La rubrique "CLASS:" est le nom de la classe spécifiée. Tous les objets définis à l'aide de ce modèle sont une instance de cette classe. La classe peut être spécifiée par la présente norme ou par un utilisateur de la présente norme.

(3) La rubrique "CLASS ID:" est un numéro qui identifie la classe spécifiée. Ce numéro est unique au sein du FAL ASE qui fournira les services pour cette classe. Lorsqu'il est qualifié par l'identité de son FAL ASE, il identifie sans ambiguïté la classe relevant du domaine d'application de la FAL. La valeur "NULL" indique que la classe ne peut pas être instanciée. Les Class ID (identificateurs de classe) entre 1 et 255 sont réservés par la présente norme pour identifier des classes normalisées. Ils ont été attribués pour maintenir la compatibilité avec des normes nationales existantes. Les CLASS ID entre 256 et 2 048 sont alloués pour identifier les classes définies par l'utilisateur.

(4) La rubrique "PARENT CLASS:" est le nom de la classe parent pour la classe spécifiée. Tous les attributs définis pour la classe parent et hérités par celle-ci sont hérités pour la classe définie, et ils n'ont donc pas à être redéfinis dans le modèle pour cette classe.

NOTE La classe parent "TOP" indique que la classe définie est une définition de classe initiale. La classe parent "TOP" est utilisée comme point de départ à partir duquel toutes les autres classes sont définies. L'usage de "TOP" est réservé pour les classes définies par la présente norme.

(5) L'étiquette "ATTRIBUTES" (ATTRIBUTS) indique que les entrées suivantes sont des attributs définis pour la classe.

a) Chacune des entrées d'attribut contient un numéro de ligne dans la colonne 1, un indicateur obligatoire (m) / facultatif (o) / conditionnel (c) / sélecteur (s) dans la colonne 2, une étiquette de type d'attribut dans la colonne 3, un nom ou une expression conditionnelle dans la colonne 4, et, facultativement, une liste de valeurs énumérées dans la colonne 5. Dans la colonne suivant la liste de valeurs, la valeur par défaut pour l'attribut peut être spécifiée.

b) Les objets sont normalement identifiés par un identificateur numérique et/ou par un nom d'objet. Dans les modèles de classe, ces attributs clés sont définis sous l'attribut clé.

c) Le numéro de ligne définit la séquence et le niveau d'imbrication de la ligne. Chaque niveau d'imbrication est identifié par un point. L'imbrication est utilisée pour spécifier

i) des champs d'un attribut structuré (4.1, 4.2, 4.3),

ii) des attributs conditionnés à un énoncé de contrainte (5). Les attributs peuvent être obligatoires (5.1) ou facultatifs (5.2) si la contrainte est vraie. Tous les attributs facultatifs n'exigent pas des énoncés de contraintes comme le fait l'attribut défini en (5.2).

iii) les champs sélection d'un attribut de type choix (6.1 et 6.2).

(6) L'étiquette "SERVICES" indique que les entrées suivantes sont des services définis pour la classe.

a) Un (m) dans la colonne 2 indique que le service est obligatoire pour la classe, alors qu'un (o) indique qu'il est facultatif. Un (c) dans cette colonne indique que le service est conditionnel. Lorsque tous les services définis pour une classe le sont comme étant facultatifs, il faut en sélectionner au moins un quand une instance de la classe est définie.

b) L'étiquette "OpsService" désigne un service opérationnel (1).

c) L'étiquette "MgtService" désigne un service de gestion (2).

d) Le numéro de ligne définit la séquence et le niveau d'imbrication de la ligne. Chaque niveau d'imbrication est identifié par un point. L'imbrication dans la liste de services sert à spécifier des services conditionnés à un énoncé de contrainte.

3.4.4 Conventions pour les définitions des services

3.4.4.1 Généralités

Le modèle de service, les primitives de service et les diagrammes de temps-séquence utilisés sont des descriptions totalement abstraites; ils ne constituent pas une spécification pour une mise en œuvre.

3.4.4.2 Paramètres du service

Les primitives de service sont utilisées pour représenter les interactions entre utilisateur de service et fournisseur de service (ISO/CEI 10731). Elles acheminent des paramètres qui indiquent des informations disponibles dans l'interaction entre utilisateur et fournisseur. Dans n'importe quelle interface particulière, il n'est pas indispensable d'énoncer tous les paramètres de façon explicite.

Les spécifications de service selon la présente norme utilisent un format de tableau pour décrire les paramètres de composants des primitives du service d'ASE. Les paramètres qui s'appliquent à chaque groupe de primitives de service sont consignés en tableaux. Chaque tableau comporte jusqu'à cinq colonnes pour le/la:

a) Nom de paramètre,

b) primitive "request",

c) primitive "indication",

d) primitive "response", et

e) primitive "confirm".

Un paramètre (ou un composant de celui-ci) est énuméré dans chaque rangée de chaque tableau. Dans les colonnes appropriées de la primitive de service, un code est utilisé pour spécifier le type d'usage du paramètre sur la primitive spécifiée dans la colonne:

M Le paramètre est obligatoire pour la primitive;

U Le paramètre est une option de l'utilisateur et peut ou peut ne pas être fourni, cela dépendant de l'usage dynamique de l'utilisateur du service. Lorsqu'il n'est pas fourni, une valeur par défaut est supposée pour le paramètre;

C Le paramètre est conditionné à d'autres paramètres ou à l'environnement de l'utilisateur du service;

– (blanc/vide) le paramètre n'est jamais présent.

S Le paramètre est un élément sélectionné.

Certaines entrées sont en plus qualifiées par des éléments entre parenthèses. Ceux-ci peuvent être

a) une contrainte spécifique au paramètre:
"(=)" indique que le paramètre équivaut du point de vue de la sémantique au paramètre dans la primitive de service située immédiatement à sa gauche dans le tableau;

b) une indication qu'une certaine note s'applique à l'entrée;
"(n)" indique que la note "n" suivante contient des informations complémentaires relatives au paramètre et à son utilisation.

3.4.4.3 Procédures de service

Les procédures sont définies en termes des

– interactions entre entités d'application par échange d'unités de données de protocole d'application de bus de terrain, et

– interactions entre un fournisseur de services de la couche application et un utilisateur de service de la couche application dans le même système par l'invocation de primitives de service de la couche application.

Ces procédures sont applicables à des instances de communication entre systèmes qui prennent en charge des services de communications à contrainte temporelle au sein de la couche application de bus de terrain.

3.4.5 Vue d'ensemble

La couche FAL est définie comme un jeu d'éléments ASE orientés objet. Chaque ASE est spécifié dans un paragraphe distinct. Chaque spécification d'ASE est constituée de deux parties, à savoir sa spécification de classe et sa spécification de services.

La spécification de classe définit les attributs de la classe. Les attributs sont accessibles à partir d'instances de la classe en utilisant les services d'ASE Object Management (gestion d'objets) spécifiés à l'Article 5 de la présente norme. La spécification de services définit les services qui sont fournis par l'ASE.

3.4.6 Conventions générales

La présente norme utilise les conventions descriptives données dans l'ISO/CEI 10731.

3.4.7 Conventions pour les définitions de classes

Les définitions de classes sont décrites à l'aide de modèles. Chaque modèle est constitué d'une liste d'attributs de la classe. La forme générale du modèle est montrée ci-dessous:

FAL ASE:			**Nom de l'ASE**
CLASS:			Nom de classe
CLASS ID:			#
PARENT CLASS:			Nom de la classe parent
ATTRIBUTES:			
1	(o)	Attribut-clé:	Identificateur numérique
2	(o)	Attribut-clé:	nom
3	(m)	Attribut:	nom d'attribut(valeurs)
4	(m)	Attribut:	nom d'attribut(valeurs)
4.1	(s)	Attribut:	nom d'attribut(valeurs)
4.2	(s)	Attribut:	nom d'attribut(valeurs)
4.3	(s)	Attribut:	nom d'attribut(valeurs)
5.	(c)	Contrainte:	expression de la contrainte
5.1	(m)	Attribut:	nom d'attribut(valeurs)
5.2	(o)	Attribut:	nom d'attribut(valeurs)
6	(m)	Attribut:	nom d'attribut(valeurs)
6.1	(s)	Attribut:	nom d'attribut(valeurs)
6.2	(s)	Attribut:	nom d'attribut(valeurs)
SERVICES:			
1	(o)	OpsService:	nom du service
2.	(c)	Contrainte:	expression de la contrainte
2.1	(o)	OpsService:	nom du service
3	(m)	MgtService:	nom du service

(1) La rubrique "FAL ASE:" est le nom de l'élément ASE de la couche FAL (FAL ASE) qui fournit les services pour la classe spécifiée.

(2) La rubrique "CLASS:" est le nom de la classe spécifiée. Tous les objets définis à l'aide de ce modèle sont une instance de cette classe. La classe peut être spécifiée par la présente norme ou par un utilisateur de la présente norme.

(3) La rubrique "CLASS ID:" est un numéro qui identifie la classe spécifiée. Ce numéro est unique au sein du FAL ASE qui fournira les services pour cette classe. Lorsqu'il est qualifié par l'identité de son FAL ASE, il identifie sans ambiguïté la classe relevant du domaine d'application de la FAL. La valeur "NULL" indique que la classe ne peut pas être instanciée. Les Class ID (identificateurs de classe) entre 1 et 255 sont réservés par la présente norme pour identifier des classes normalisées. Ils ont été attribués pour maintenir la compatibilité avec des normes nationales existantes. Les CLASS ID entre 256 et 2 048 sont alloués pour identifier les classes définies par l'utilisateur.

(4) La rubrique "PARENT CLASS:" est le nom de la classe parent pour la classe spécifiée. Tous les attributs définis pour la classe parent et hérités par celle-ci sont hérités pour la classe définie, et ils n'ont donc pas à être redéfinis dans le modèle pour cette classe.

NOTE La classe parent "TOP" indique que la classe définie est une définition de classe initiale. La classe parent "TOP" est utilisée comme point de départ à partir duquel toutes les autres classes sont définies. L'usage de "TOP" est réservé pour les classes définies par la présente norme.

(5) L'étiquette "ATTRIBUTES" (ATTRIBUTS) indique que les entrées suivantes sont des attributs définis pour la classe.

a) Chacune des entrées d'attribut contient un numéro de ligne dans la colonne 1, un indicateur obligatoire (m) / facultatif (o) / conditionnel (c) / sélecteur (s) dans la colonne 2, une étiquette de type d'attribut dans la colonne 3, un nom ou une expression conditionnelle dans la colonne 4, et, facultativement, une liste de valeurs énumérées dans la colonne 5. Dans la colonne suivant la liste de valeurs, la valeur par défaut pour l'attribut peut être spécifiée.

b) Les objets sont normalement identifiés par un identificateur numérique et/ou par un nom d'objet. Dans les modèles de classe, ces attributs clés sont définis sous l'attribut clé.

c) Le numéro de ligne définit la séquence et le niveau d'imbrication de la ligne. Chaque niveau d'imbrication est identifié par un point. L'imbrication est utilisée pour spécifier

i) des champs d'un attribut structuré (4.1, 4.2, 4.3),

ii) des attributs conditionnés à un énoncé de contrainte (5). Les attributs peuvent être obligatoires (voir 5.1) ou facultatifs (voir 5.2) si la contrainte est vraie. Tous les attributs facultatifs n'exigent pas des énoncés de contraintes comme le fait l'attribut défini en 5.2.

iii) les champs sélection d'un attribut de type choix (voir 6.1 et 6.2).

(6) L'étiquette "SERVICES" indique que les entrées suivantes sont des services définis pour la classe.

a) Un (m) dans la colonne 2 indique que le service est obligatoire pour la classe, alors qu'un (o) indique qu'il est facultatif. Un (c) dans cette colonne indique que le service est conditionnel. Lorsque tous les services définis pour une classe le sont comme étant facultatifs, il faut en sélectionner au moins un quand une instance de la classe est définie.

b) L'étiquette "OpsService" désigne un service opérationnel (1).

c) L'étiquette "MgtService" désigne un service de gestion (2).

d) Le numéro de ligne définit la séquence et le niveau d'imbrication de la ligne. Chaque niveau d'imbrication est identifié par un point. L'imbrication dans la liste de services sert à spécifier des services conditionnés à un énoncé de contrainte.

3.4.8 Conventions pour les définitions des services

3.4.8.1 Généralités

Le modèle de service, les primitives de service et les diagrammes de temps-séquence utilisés sont des descriptions totalement abstraites; ils ne constituent pas une spécification pour une mise en œuvre.

3.4.8.2 Paramètres du service

Les primitives de service sont utilisées pour représenter les interactions entre utilisateur de service et fournisseur de service (ISO/CEI 10731). Elles acheminent des paramètres qui indiquent des informations disponibles dans l'interaction entre utilisateur et fournisseur. Dans n'importe quelle interface particulière, il n'est pas indispensable d'énoncer tous les paramètres de façon explicite.

Les spécifications de service selon la présente norme utilisent un format de tableau pour décrire les paramètres de composants des primitives du service d'ASE. Les paramètres qui s'appliquent à chaque groupe de primitives de service sont consignés en tableaux. Chaque tableau comporte jusqu'à cinq colonnes pour le/la:

1) Nom de paramètre,
2) primitive "request",
3) primitive "indication",
4) primitive "response", et
5) primitive "confirm".

Un paramètre (ou un composant de celui-ci) est énuméré dans chaque rangée de chaque tableau. Dans les colonnes appropriées de la primitive de service, un code est utilisé pour spécifier le type d'usage du paramètre sur la primitive spécifiée dans la colonne:

M Le paramètre est obligatoire pour la primitive;

U Le paramètre est une option de l'utilisateur et peut ou peut ne pas être fourni, cela dépendant de l'usage dynamique de l'utilisateur du service. Lorsqu'il n'est pas fourni, une valeur par défaut est supposée pour le paramètre;

C Le paramètre est conditionné à d'autres paramètres ou à l'environnement de l'utilisateur du service;

– (blanc/vide) le paramètre n'est jamais présent.

S Le paramètre est un élément sélectionné.

Certaines entrées sont en plus qualifiées par des éléments entre parenthèses. Ceux-ci peuvent être

a) une contrainte spécifique au paramètre:
"(=)" indique que le paramètre équivaut du point de vue de la sémantique au paramètre dans la primitive de service située immédiatement à sa gauche dans le tableau;

b) une indication qu'une certaine note s'applique à l'entrée;
"(n)" indique que la note "n" suivante contient des informations complémentaires relatives au paramètre et à son utilisation.

3.4.8.3 Procédures de service

Les procédures sont définies en termes des

– interactions entre entités d'application par échange d'unités de données de protocole d'application de bus de terrain, et

– interactions entre un fournisseur de services de la couche application et un utilisateur de service de la couche application dans le même système par l'invocation de primitives de service de la couche application.

Ces procédures sont applicables à des instances de communication entre systèmes qui prennent en charge des services de communications à contrainte temporelle au sein de la couche application de bus de terrain.

4 Concepts

L'Article 9 de la CEI 61158-1 décrit les concepts des descriptions de service de la couche application et les modèles utilisés dans la présente norme.

5 ASE Data type

Les types de données de bus de terrain spécifient la syntaxe indépendante vis-à-vis de toute machine pour les données d'application acheminées par les services FAL. La couche d'application de bus de terrain prend en charge la définition et le transfert des types de données tant de base que construits.

L'Article 5 de la CEI 61158-5-10 fournit un ensemble de tous les types de données exigés par les services FAL de Type 3 et de Type 10 ainsi que les types de données autorisés à être utilisés par les profils d'applications.

Les règles de codage pour les types de données de la FAL utilisés par le Type 3 sont données dans la CEI 61158-6-3.

6 Spécification de modèle de communication

6.1 Concepts de DP

6.1.1 Exigences utilisateur pour le système DP de bus de terrain

Le système d'automation type est constitué d'un ou plusieurs d'automates programmables connectés par des systèmes E/S à la machine/au processus. Le système E/S est surtout mis en œuvre comme un jeu de connexions à un seul point avec une interface électrique bien définie comme, par exemple, des signaux numériques de 24 V ou des signaux analogiques de 4 mA à 20 mA. La structure globale de l'automate programmable est définie dans la CEI 61131-1 et utilisée à grande échelle.

Pour commuter des systèmes E/S enfichés dans un automate programmable vers des systèmes E/S décentralisés, une liaison série est nécessaire non seulement pour l'échange de données de processus, mais aussi pour émettre des paramètres et données de diagnostic et de configuration.

En outre, plusieurs applications en automation d'usine requièrent de très courts temps de réaction.

Par conséquent, il convient que la conception de l'interface utilisateur dans l'automate programmable pour l'échange de données permette un accès direct à ces variables sans appels de fonctions spécifiques.

En outre, un transfert fiable de données est requis pour les applications industrielles.

L'intégration des appareils de terrain simples exige une architecture de protocole simple pour permettre des mises en œuvre à faible coût.

Pour la prise en charge de l'intégration des appareils de terrain intelligents, des services de communication et des alarmes acycliques sont nécessaires.

Afin d'obtenir un système d'automation constitué des appareils provenant de fabricants différents, la stricte interopérabilité de tous les composants est nécessaire.

En conformité avec la structure de la CEI 61131-1, il convient que la structure décentralisée soit invisible au programme de commande existant.

Pour une communication efficiente entre les esclaves DP, comme requis dans les applications de commande de mouvement, le bus de terrain fournit un mécanisme de communication Éditeur/Abonné.

Le bus de terrain fournit un mécanisme pour la synchronisation de l'application du maître DP (Classe 1) et des esclaves DP avec le cycle de communication pour des applications temps réel hautement synchronisées.

Pour l'horodatage et autres actions avec des exigences de temporisation précise, un mécanisme pour synchroniser des horloges est proposé.

Les appareils de terrain requièrent un mécanisme pour télécharger vers l'amont (Pull, c'est-à-dire tirer) et/ou télécharger vers l'aval (Push, c'est-à-dire pousser) des données LR («LR Data») et un mécanisme pour invoquer les fonctions prédéfinies au sein de l'esclave DP.

Le DP de bus de terrain fournit un mécanisme pour bâtir des esclaves DP redondants, qui peuvent fonctionner dans des systèmes de commande avec plusieurs mises en œuvre du maître.

6.1.2 Caractéristiques des DP de bus de terrain

Le fait de combiner la structure définie dans la CEI 61131-1 avec les DP de bus de terrain procure les avantages suivants:

– Des changements dans la centrale n'induisent que de faibles changements de câblage.
– Les signaux critiques peuvent être transférés sur de courtes distances jusqu'à l'appareil de terrain distant (capteur/actionneur) et sur de longues distances à destination/en provenance de l'automate programmable en passant par des DP de bus de terrain avec une exactitude élevée.
– La paramétrisation et le diagnostic peuvent être réalisés sans câblage supplémentaire.
– La mise en service de sous-parties sans installations supplémentaires.

Le Tableau 1 donne une vue d'ensemble des exigences et des caractéristiques d'un système DP de bus de terrain.

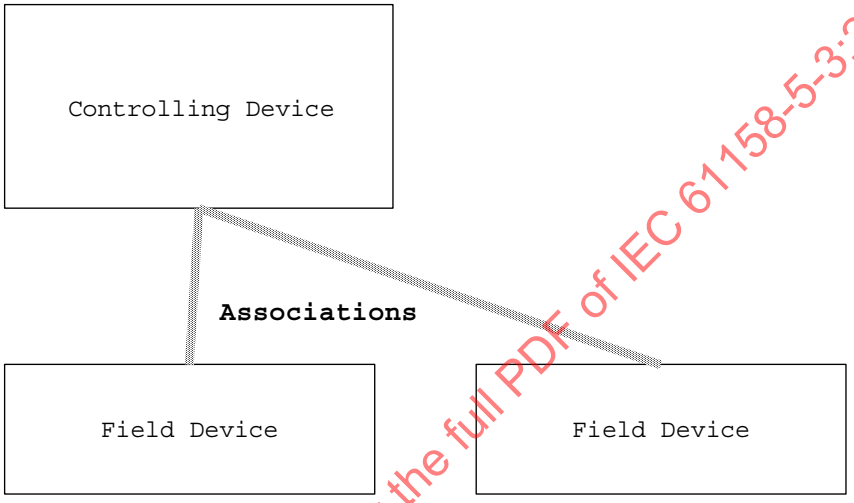
Tableau 1 – Exigences et caractéristiques des DP de bus de terrain

Exigences	Caractéristique de DP de bus de terrain
Court temps de réaction	Échange de plus de 1 000 Entrées et Sorties avec 32 appareils de terrain en moins de 10 ms
Le système d'automation est constitué d'un ou plusieurs automates programmables.	Fonction en monomaître ou multimaître
Appareils de terrain simples	Protocole simple, interface de communications à faible coût
Transfert de données fiable	Distance de Hamming 4 garantie; Mécanisme de reprise sur erreur; Utilisation d'une technologie d'émission éprouvée
Excellent diagnostic	Divers diagnostics dans les Maîtres comme dans les Esclaves
Interface utilisateur avec accès direct aux données d'entrée et de sortie	Interface de tampon pour les données d'entrée et de sortie
Appareils de terrain intelligents	Communication acyclique qui fournit un plan flexible et évolué d'adressage des données au sein des appareils de terrain. La possibilité de transférer une alarme de l'appareil de terrain vers l'automate programmable avec un acquittement explicite
Interopérabilité	Définitions précises et complètes, y compris la définition du comportement du système
Communication efficiente entre esclaves DP pour les applications de commande de mouvement	Le mécanisme de communication Éditeur/Abonné permet la communication directe entre esclaves DP sans interactions de la part de l'utilisateur du maître DP (Classe 1)
Synchronisation de l'application du maître DP (Classe 1) et les esclaves DP	Le mode isochrone fournit un mécanisme pour la synchronisation de l'application du maître DP (Classe 1) et des esclaves DP avec le cycle de communication. La gigue de cycle est inférieure à 1 µs pour des débits de données supérieurs à 1,5 Mbit/s
Mécanisme pour charger des blocs de données non structurés.	Pour charger des blocs de données non structurés, le DP de bus de terrain fournit un mécanisme de Load Region (région de charge)
Invocation de fonctions prédéfinies dans l'esclave DP	Avec le modèle d'invocation de fonctions ("Function Invocation"), les fonctions prédéfinies au sein de l'esclave DP peuvent être invoquées.
Synchronisation d'horloges	Avec la synchronisation du temps (Time Synchronization), un temps réseau (Network Time) uniforme (TUC) avec une exactitude de 1 ms près peut être réalisé.

6.1.3 Modèle de communication de DP

6.1.3.1 Associations

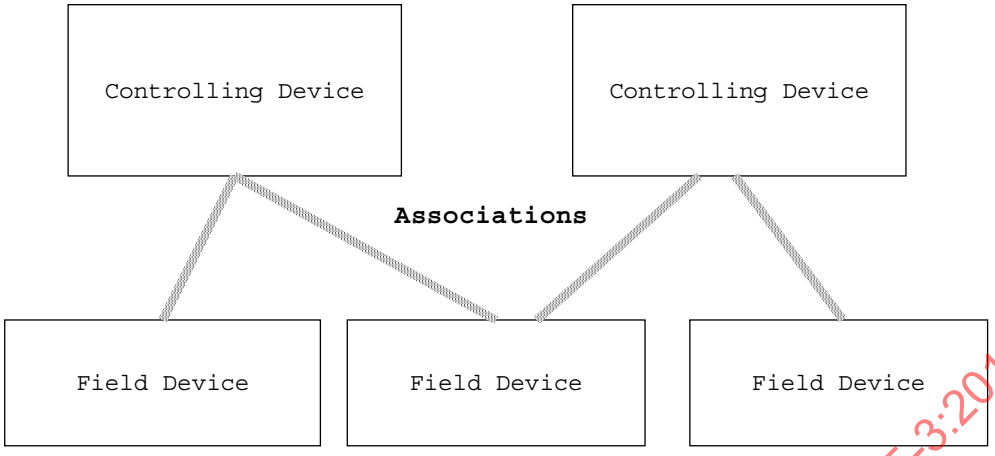
Le modèle de communication de bus de terrain prend en charge la communication des appareils de terrain avec un seul appareil de commande (par exemple: automate programmable ou système de commande distribué) par l'intermédiaire d'associations (voir Figure 1). En outre, la communication entre plusieurs appareils de commande et leurs appareils de terrain assignés est également prise en charge (voir Figure 2). En plus de ces modèles de communications de base, le bus de terrain prend également en charge la communication cyclique entre appareils de terrain (voir Figure 3).



Légende

Anglais	Français
Field Device	Appareil de terrain
Controlling Device	Appareil de commande
Associations	Associations

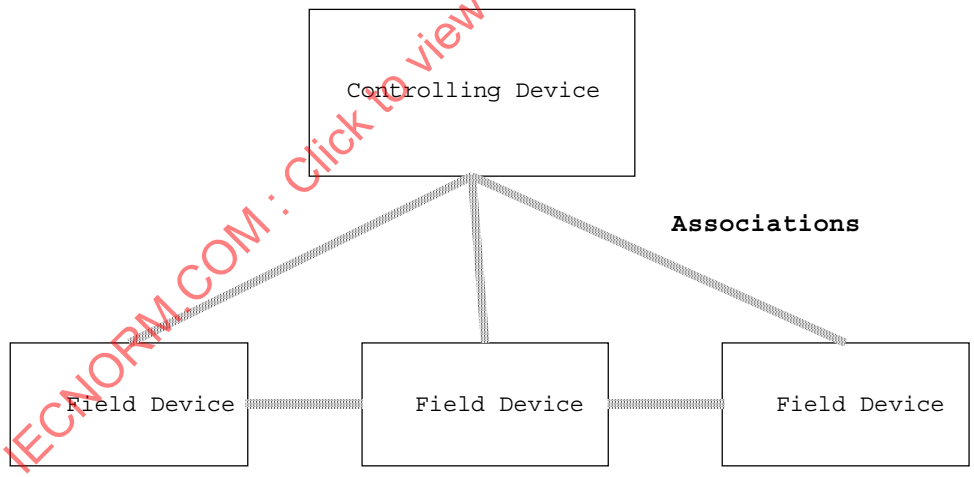
Figure 1 – Exemple de communication DP avec un seul appareil de commande



Légende

Anglais	Français
Field Device	Appareil de terrain
Controlling Device	Appareil de commande
Associations	Associations

Figure 2 – Exemple de communication DP avec plusieurs appareils de commande



Légende

Anglais	Français
Field Device	Appareil de terrain
Controlling Device	Appareil de commande
Associations	Associations

Figure 3 – Exemple de communication DP entre appareils de terrain

6.1.3.2 Types des appareils

6.1.3.2.1 Maître DP (Classe 1)

Le maître DP (Classe 1) est un appareil de commande qui est associé à un ou plusieurs esclaves DP (appareils de terrain). Le maître DP (Classe 1) accomplit une ou plusieurs des fonctionnalités de base suivantes:

– échange cyclique de données E/S avec les esclaves DP connexes
– diagnostic avec les esclaves DP
– configuration d'esclaves DP
– traitement des demandes de configuration et de diagnostic envoyées par un maître DP (Classe 2).

Les fonctionnalités étendues d'un maître DP (Classe 1) sont:

– accès acyclique à des données de processus des esclaves DP,
– traitement d'alarmes provenant d'esclaves DP,
– prise en charge du mode isochrone à des fins de synchronisation avec des esclaves DP,
– prise en charge du mécanisme DXB pour l'échange de données cyclique entre esclaves DP,
– prise en charge du téléchargement vers l'amont et/ou vers l'aval de données LR avec des esclaves DP,
– invocation de fonctions prédéfinies au sein des esclaves DP,
– prise en charge de la synchronisation d'horloge relative aux horloges d'esclaves DP et d'autres maîtres DP.

6.1.3.2.2 Maître DP (Classe 2)

Le maître DP (Classe 2) est un appareil de commande qui gère la fourniture de données de configuration (jeux de paramètres) et la collecte de données de diagnostic à destination/en provenance d'un maître DP (Classe 1) En outre, le maître DP (Classe 2) peut accomplir la totalité des capacités de communication de base et un certain nombre des capacités de communication étendues d'un maître DP (Classe 1) vers un esclave DP.

6.1.3.2.3 Esclave DP

6.1.3.2.3.1 Généralités

L'esclave DP est un appareil de terrain et accomplit les activités suivantes, selon la fonctionnalité.

Les fonctionnalités de base sont:

– échange cyclique de données E/S avec le maître DP assigné,
– diagnostic avec le maître DP assigné,
– traitement de demandes de configuration envoyées par un maître DP.

Les fonctionnalités étendues d'un esclave DP sont:

— fourniture d'un accès acyclique à des données de processus pour des maîtres DP,
— fourniture d'alarmes au maître DP assigné,
— prise en charge du mode isochrone à des fins de synchronisation avec le maître DP (Classe 1),
— échange de données cyclique entre esclaves DP en utilisant la communication Éditeur/Abonné,

— prise en charge du téléchargement vers l'amont et/ou vers l'aval de données LR,

— prise en charge de fonctions prédéfinies qui peuvent être invoquées par le maître DP (Classe 1) ou le maître DP (Classe 2),

— disposition pour synchroniser l'horloge locale avec une horloge de maître DP,

— prise en charge de la redondance d'esclave DP.

Les esclaves DP se composent d'un ou plusieurs modules. Ces modules reflètent habituellement les composants matériels de l'appareil de terrain.

6.1.3.2.3.2 Module

La couche application DP utilise des modules pour refléter la structuration des composants (par exemple: unités matérielles, unités logicielles) à l'intérieur d'un esclave DP. Un module peut avoir une ou plusieurs voies qui représentent la structure réelle des données d'entrée et/ou de sortie. Ces voies peuvent être une subdivision supplémentaire de l'objet Input and/or Output Data.

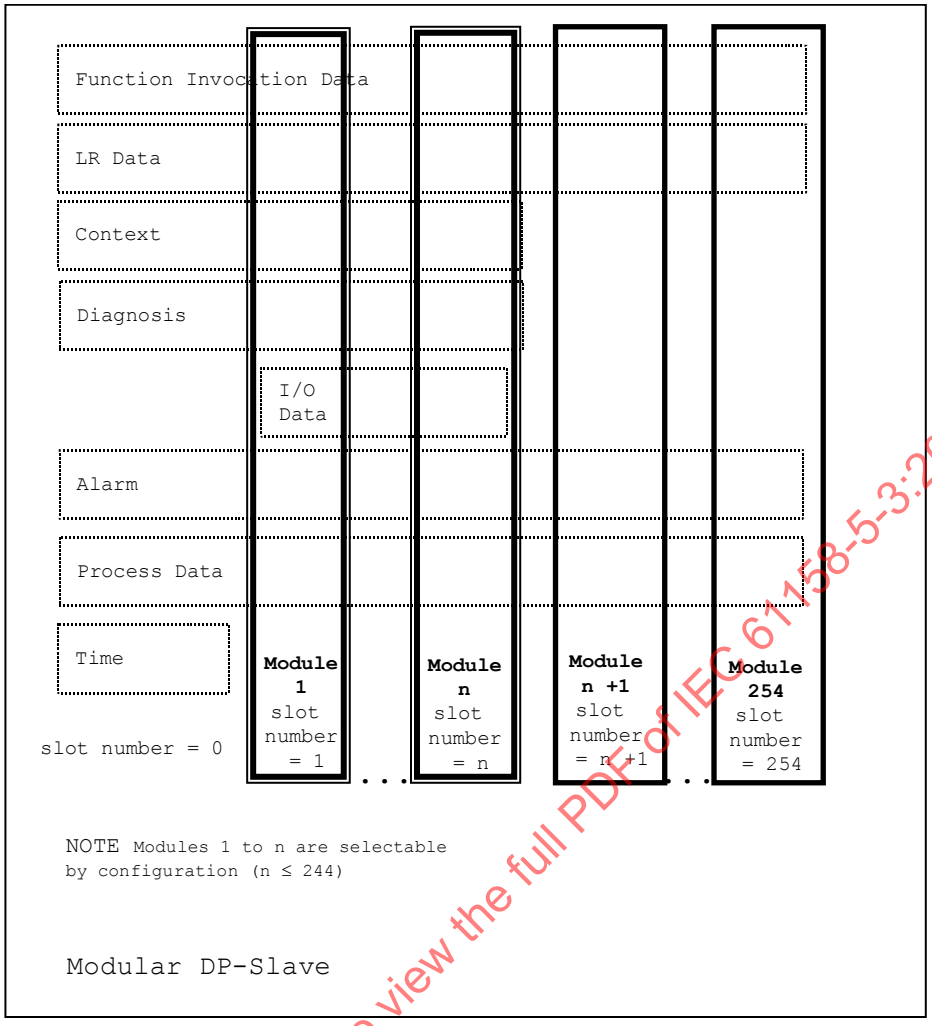
Chaque module est adressé par un numéro de baie (1 à 254). La numérotation ne comporte aucun trou, se fait dans l'ordre croissant et commence par 1. Si une baie n'est pas occupée par un module, une baie vide est enregistrée sous le numéro de baie correspondant dans la configuration. Le numéro de baie "zéro" (Slot = 0) renvoie à l'esclave DP lui-même. Les modules contiennent des données qui sont adressables par indice. L'utilisation concrète d'un numéro de baie et d'un indice pour adresser des données au sein de l'appareil est spécifique à chaque fabricant.

De plus, un identificateur de configuration (Configuration Identifier) doit être attribué à chaque module. La numérotation ne comporte aucun trou, se fait dans l'ordre croissant et commence par 0. Si une baie n'est pas occupée par un module, il faut affecter un Cfg Identifier avec la longueur d'Input and Output Data égale à zéro dans la configuration (baie vide).

Un module fournit un ou plusieurs des éléments de données suivants:

– Diagnosis (Diagnostic),
– I/O Data (Données E/S),
– Alarm (Alarme),
– Process Data (Données de processus),
– LR Data (Données de région de charge),
– Context (Contexte),
– Function Invocation Data (Données d'invocation de fonction).

Dans les esclaves DP modulaires, les modules déterminent la configuration réelle de l'appareil de terrain (voir Figure 4). Le nombre de modules est fixe conformément à la spécification de l'appareil (le nombre maximal de modules est 244).



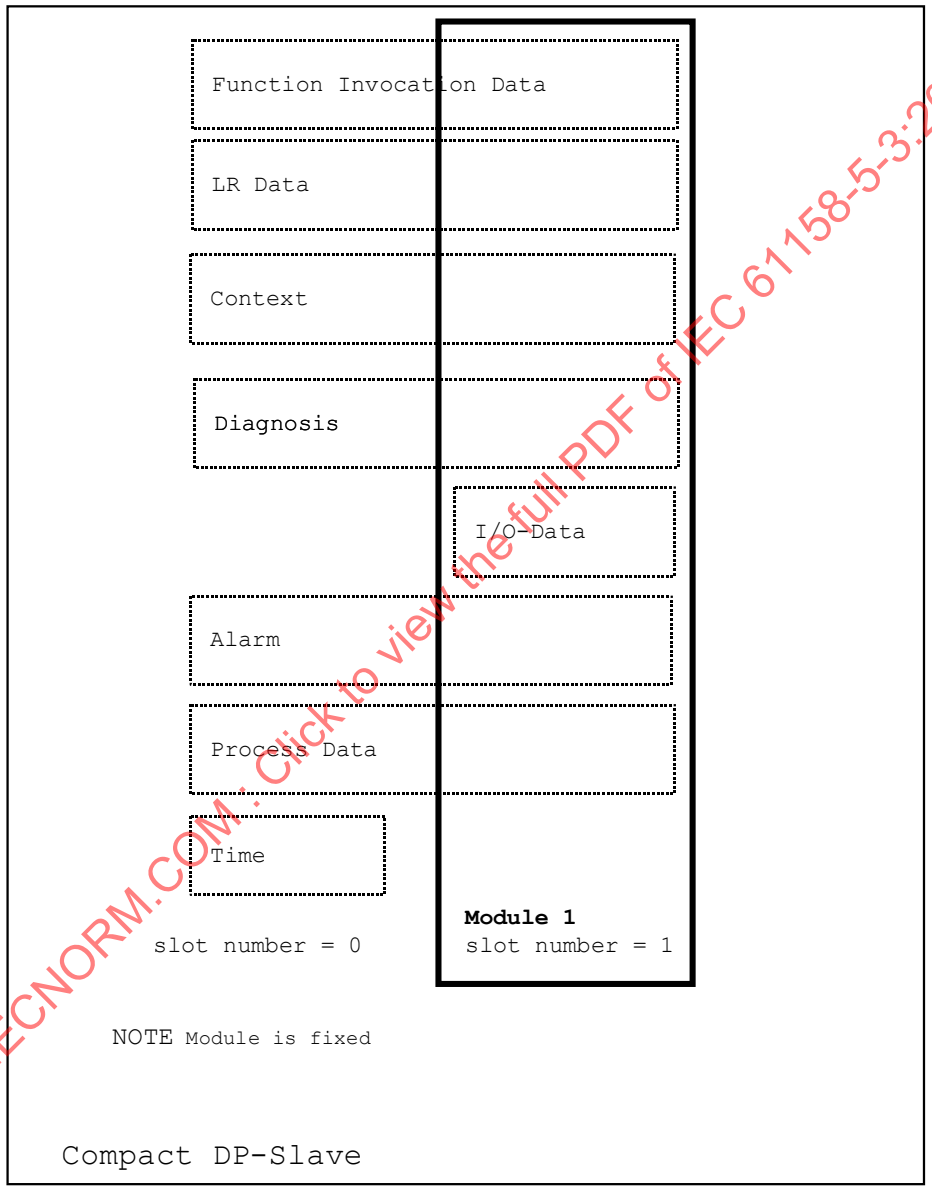
Légende

Anglais	Français
Function Invocation Data	Données d'invocation de fonctions
LR Data	Données de région de charge
Context	Contexte
Diagnosis	Diagnostic
I/O-Data	Données E/S
Alarm	Alarme
Process Data	Données de processus
Time	Temps
slot number = 0	numéro de baie = 0
Module 1 slot number = 1	Module 1 numéro de baie =1
Module 1 slot number = 1	Module 1 Numéro de baie =1
Module n slot number = n	Module n numéro de baie = n
Module n+1 slot number = n+1	Module n+1 numéro de baie = n+1
Module 254 slot number = 254	Module 254 numéro de baie = 254

Anglais	Français
Modular DP-Slave	Esclave DP modulaire
NOTE Modules 1 to n are selectable by configuration (n≤244)	NOTE Les Modules 1 à n sont sélectionnables par configuration n≤244

Figure 4 – Modèle d'esclave DP (esclave DP modulaire)

Dans les esclaves DP compacts eu égard à la spécification de l'appareil, le nombre de modules est fixe et, donc, la configuration aussi (par exemple: un seul module; voir Figure 5).



Légende

Anglais	Français
Function Invocation Data	Données d'invocation de fonctions
LR Data	Données de région de charge
Context	Contexte
Diagnosis	Diagnostic
I/O-Data	Données E/S

Anglais	Français
Alarm	Alarme
Process Data	Données de processus
Time	Temps
slot number = 0	numéro de baie = 0
Module 1 slot number = 1	Module 1 numéro de baie =1
Module 1 slot number = 1	Module 1 numéro de baie =1
NOTE Module is fixed	NOTE Le module est fixe.
Compact DP-Slave	Esclave DP compact

Figure 5 – Modèle d'esclave DP (esclave DP compact)

6.1.3.3 Attributs de l'appareil

Les attributs de l'appareil indiquent qu'outre la fonctionnalité de base, la fonctionnalité étendue est prise en charge en plus.

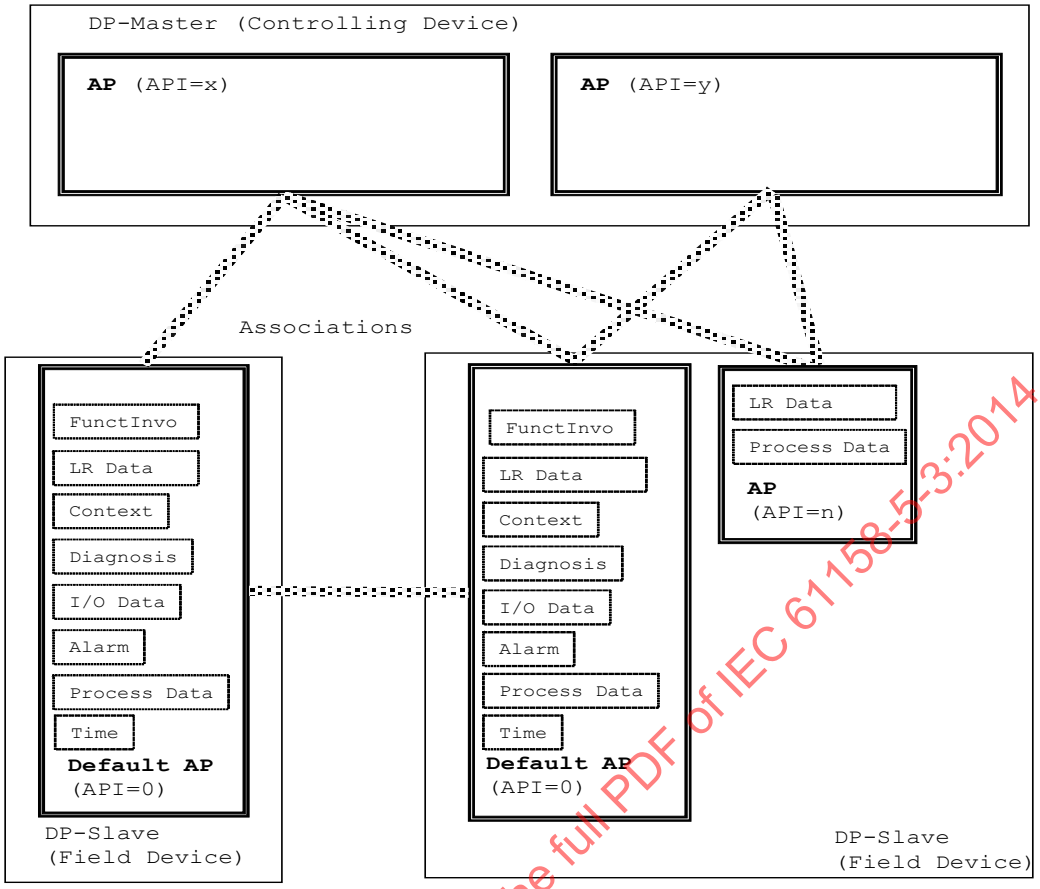
Ces fonctionnalités supplémentaires peuvent impliquer certaines règles pour des structures de données déjà définies. Un attribut de l'appareil indique que l'appareil prend en charge des relations de communication supplémentaires ou un comportement de communication différent.

6.1.3.4 Processus d'application

6.1.3.4.1 Vue d'ensemble

Dans l'environnement de DP de bus de terrain, une application peut être partitionnée et répartie en un certain nombre des appareils sur le réseau. Chacune de ces partitions est appelée "processus d'application" (AP pour «application process»), (voir Figure 6). Un appareil peut avoir plusieurs AP. Dans ce cas, chaque AP individuel est identifié de façon univoque par un identificateur d'AP (API pour «AP Identifier).

Chaque esclave DP doit avoir un AP par défaut (Default AP) avec API = 0. Le Default AP peut être relatif à des données E/S, à des données de diagnostic et d'alarmes, à des données de configuration, à des données de processus et à des LR Data. Tous les autres AP d'esclave DP doivent être relatifs à des données de processus seulement.



Légende

Anglais	Français
DP-Master (Controlling Device)	Maître DP (Appareil de commande)
AP (API=x)	AP (API=x)
AP (API=y)	AP (API=y)
Associations	Associations
FunctInvo	Invocation de fonctions
LR Data	Données de région de charge
Context	Contexte
Diagnosis	Diagnostic
I/O Data	Données E/S
Alarm	Alarme
Process Data	Données de processus
Time	Temps
Default AP	Processus d'application par défaut
(API=0)	(API=0)
DP-Slave	Esclave DP
(Field Device)	(Appareil de terrain)

Figure 6 – Vue d'ensemble de processus d'application

Un AP peut être distribué à plusieurs modules. Chaque module au sein d'un processus d'application est identifié de façon univoque par un numéro de baie (Slot = 1 à 254). Le numéro de baie "zéro" (Slot = 0) renvoie à l'AP lui-même au sein de l'appareil de terrain.

Le Context doit être relatif au Default AP lui-même et à tous ses modules contenant des I/O Data.

Les informations Diagnosis peuvent être seulement relatives au Default AP lui-même et à tous ses modules contenant des I/O Data.

Les I/O Data doivent être relatives à des modules du Default AP et pas à l'AP lui-même. Le nombre de modules avec I/O Data est limité et dépend de la configuration et de la structure d'I/O Data des modules (voir 6.1.3.2.3.2).

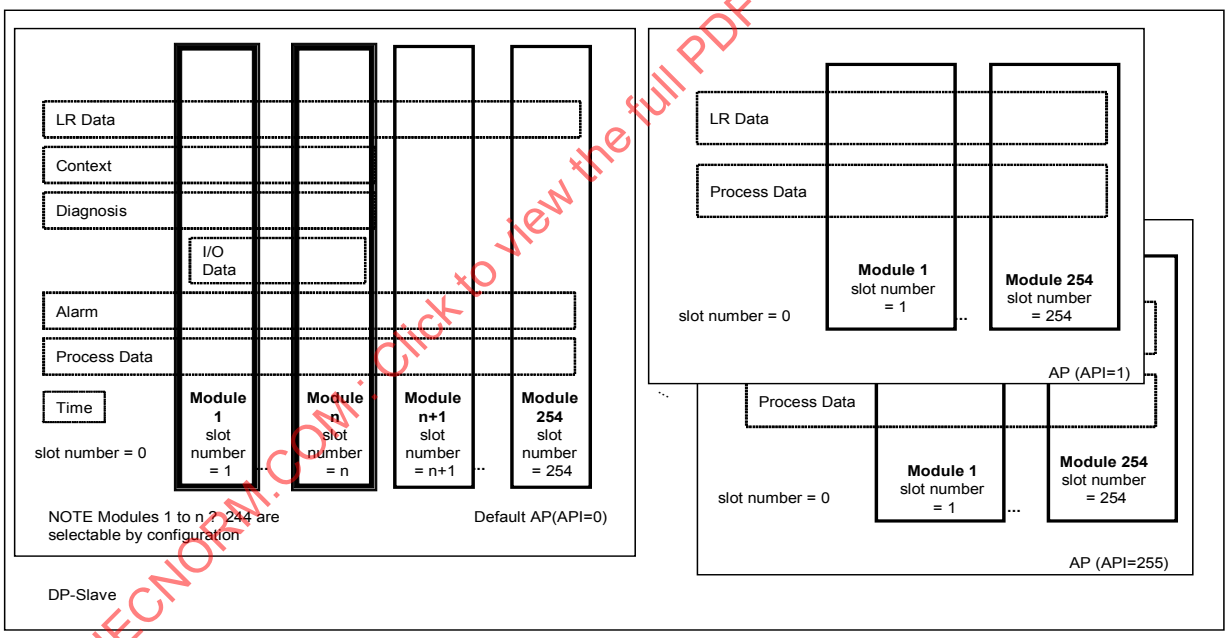
Les données Alarm peuvent être relatives au Default AP lui-même et à tous ses modules.

Les Process Data peuvent être relatives à tous les AP et à tous leurs modules.

Les LR Data peuvent être relatives à tous les AP et à tous leurs modules.

Le temps Time doit être relatif au Default AP lui-même et pas à ses modules.

La Figure 7 montre la relation entre les AP, les éléments de données et les modules d'un esclave DP.



Légende

Anglais	Français
LR Data	Données de région de charge
Context	Contexte
Diagnosis	Diagnostic
I/O Data	Données E/S
Alarm	Alarm
Process Data	Données de processus
Time	Temps
(Field Device)	(Appareil de terrain)

Anglais	Français
Module 1 slot number = 1	Module 1 Numéro de baie =1
Module n slot number = n	Module n numéro de baie = n
Module n+1 slot number = n+1	Module n+1 numéro de baie = n+1
Module 254 slot number = 254	Module 254 numéro de baie = 254
NOTE Modules 1 to n ≤244are selectable by configuration (n≤244)	Note Les Modules 1 à n≤244 sont sélectionnables par configuration
Default AP (API=0)	Processus d'application par défaut (API=0)
DP-Slave	Esclave DP
slot number = 0	numéro de baie = 0
Module 1 slot number = 1	Module 1 numéro de baie =1
AP (API=255)	AP (API=255)
DP-Slave	Esclave DP

Figure 7 – Modèle d'esclave DP (esclave DP modulaire)

Le maître DP (Classe 1) doit avoir seulement son Default AP. Le maître DP (Classe 1) doit communiquer avec le Default AP d'un esclave DP seulement.

Le maître DP (Classe 2) peut avoir plusieurs AP. Communiquant avec un esclave DP, le maître DP (Classe 2) peut être capable d'adresser tous les AP d'un esclave DP.

6.1.3.4.2 Élément de service application

Un élément de service application (ASE), tel que défini dans l'ISO/CEI 9545, est un jeu de fonctions d'application qui fournissent un moyen pour l'interfonctionnement des processus d'application dans un but spécifique. Les ASE fournissent un jeu de services pour acheminer les demandes et réponses à destination et en provenance de processus d'application et de leurs objets.

La couche d'application DP propose les ASE suivants (voir Figure 8):

ASE I/O Data
L'ASE I/O Data fournit un jeu de services pour acheminer des données E/S de manière cyclique. Ces données appartiennent toujours aux modules qui ont été configurés en termes de l'ASE Context. Un service de multidiffusion fournit la synchronisation des données E/S et rapporte l'état opérationnel du maître DP assigné. Facultativement, l'ASE I/O offre la possibilité de partages les données d'entrées d'un esclave DP avec d'autres esclaves DP.

ASE Process Data
L'ASE Process Data fournit un jeu de services pour acheminer des données de manière acyclique. L'application du maître DP demande chaque émission individuellement. L'ASE Process Data peut être relatif à tous les AP d'un esclave DP.

ASE Diagnosis
L'ASE Diagnosis fournit des services pour le maître DP pour rechercher des informations Diagnosis auprès d'un esclave DP qui est relatif au Default AP ou à ses modules qui font partie intégrante de la configuration.

ASE Alarm
L'ASE Alarm fournit un jeu de services pour acheminer les alarmes ou messages de statut émis par l'esclave DP. Le maître DP assigné acquitte les Alarm. Les messages de statut n'ont pas être acquittés par le maître DP.

ASE Management
L'ASE Management fournit un jeu de services pour gérer un maître DP (Classe 1) par un maître DP (Classe 2). Le maître DP (Classe 1) reçoit le jeu de paramètres de bus ou le jeu de paramètres d'esclave. Ensuite, les jeux reçus sont activés par le maître DP (Classe 2). Le maître DP (Classe 1) fournira les informations de diagnostic système si elles sont demandées.

ASE Context
L'ASE Context fournit un jeu de services pour

- paramétrer et configurer le Default AP lui-même et ses modules contenant des données E/S,
- transférer les commandes de commutation pour les structures redondantes et les commandes pour réinitialiser MS1 ou changer le mode de fonctionnement vers l'esclave DP,
- établir l'adresse de l'appareil d'un esclave DP,
- fournir la configuration réelle de l'esclave DP destinée à être lue par le maître DP (Classe 2),

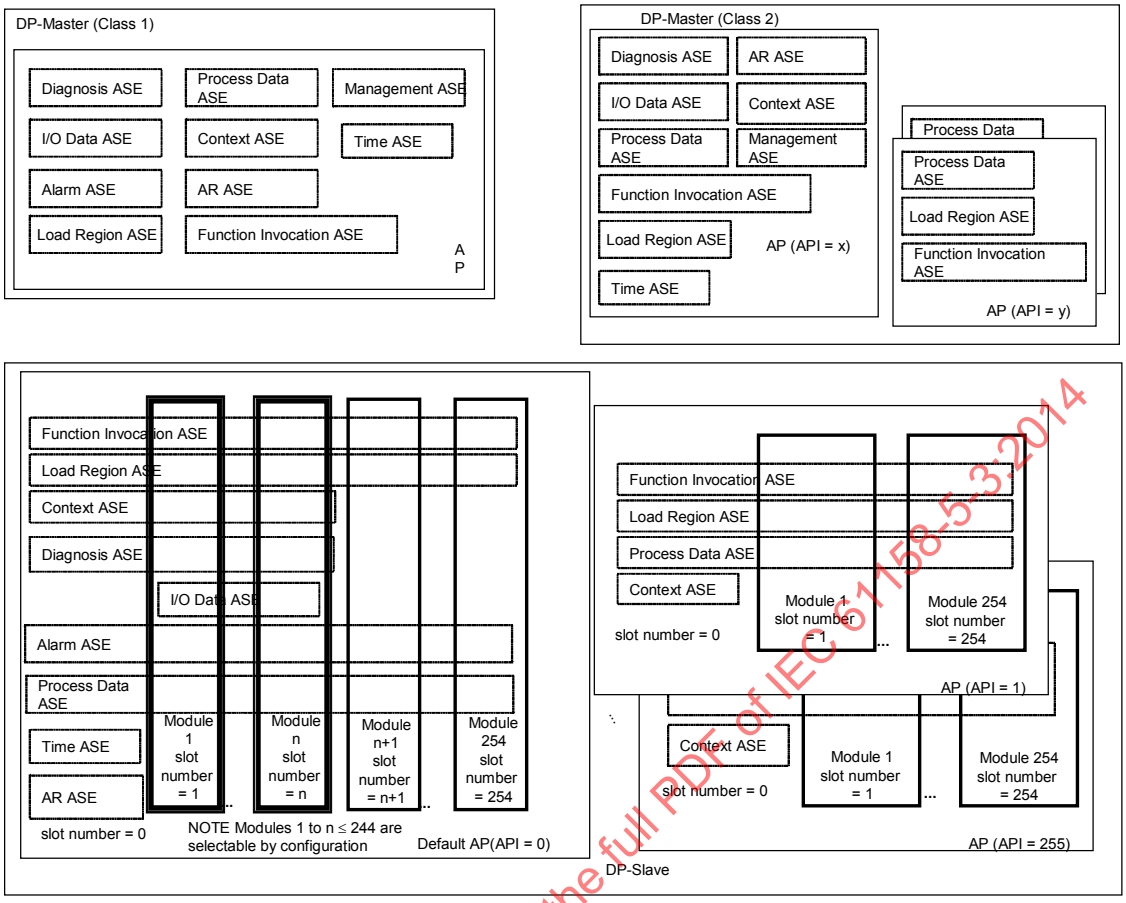
et pour établir ou libérer une association entre un AP individuel d'un maître DP (Classe 2) et un AP individuel de l'esclave DP pour la communication acyclique. Cette association appartient exclusivement à l'ASE Process Data.

ASE Load Region
L'ASE Load Region fournit un jeu de services pour acheminer des LR Data de manière acyclique. L'application du maître DP demande chaque émission individuellement. L'ASE LR Data peut être relatif à tous les AP d'un esclave DP.

ASE Function Invocation
L'ASE Function Invocation fournit un jeu de services pour invoquer des objets Function Invocation sans état et/ou orientés état. L'application du maître DP est tenue d'invoquer chaque fonction individuellement. L'ASE Function Invocation peut être relatif à tous les AP d'un esclave DP.

ASE Time
L'ASE Time fournit un jeu de données pour synchroniser les horloges de plusieurs appareils sur un segment de bus de terrain. L'application du maître DP (Classe 1) demande chaque synchronisation individuellement. L'ASE Time peut être relatif au Defaut AP d'un esclave DP.

ASE Application Relationship
L'ASE Application Relationship (ASE AR) fournit un modèle de description pour les types AR distincts. Cela inclut leurs caractéristiques de transfert ainsi que leurs états de communication courants.



Légende

Anglais	Français
Management ASE	ASE “Management”
AR ASE	ASE “AR”
Function Invocation ASE	ASE « Function Invocation »
Load Region ASE	ASE “région de charge”
Context ASE	ASE “Contexte”
Diagnosis ASE	ASE “Diagnostic”
I/O Data ASE	ASE “Données E/S”
Alarm ASE	ASE “Alarme”
Process Data ASE	ASE “Données de processus”
Time ASE	ASE “Temps”
Module 1 slot number = 1	Module 1 Numéro de baie =1
Module n slot number = n	Module n numéro de baie = n
Module n+1 slot number = n+1	Module n+1 numéro de baie = n+1
Module 254 slot number = 254	Module 254 numéro de baie = 254
NOTE Modules 1 to n ≤244 are selectable by configuration (n≤244)	Note Les Modules 1 à n≤244 sont sélectionnables par configuration
Default AP (API=0)	Processus d’application par défaut (API=0)

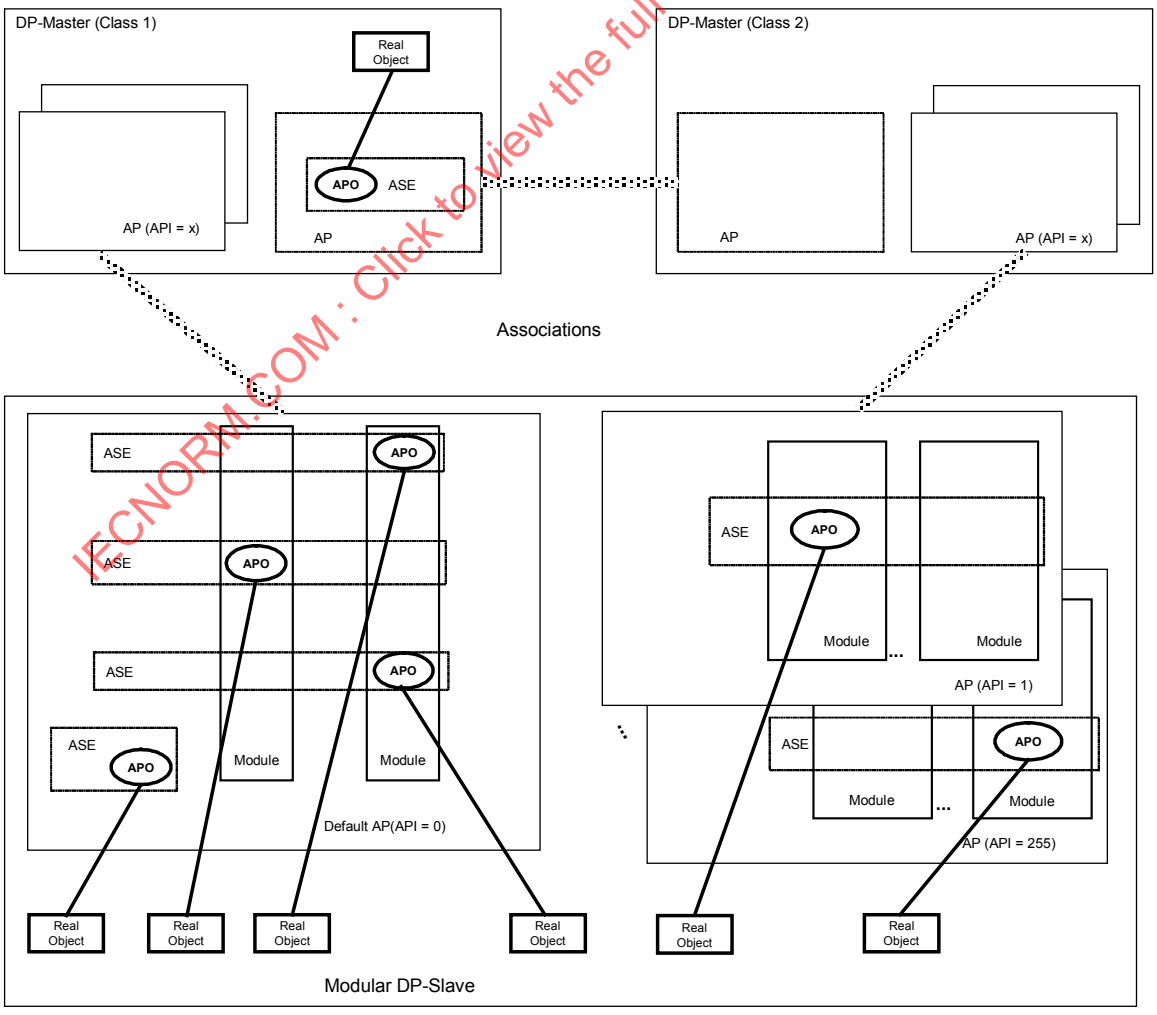
Anglais	Français
DP-Slave	Esclave DP
slot number = 0	numéro de baie = 0
Module 1 slot number = 1	Module 1 numéro de baie =1
AP (API=255)	AP (API=255)
DP-Slave	Esclave DP

Figure 8 – Éléments de service application (ASE)

6.1.3.4.3 Objets d'application

Un objet d'application (APO pour «application object») est une représentation réseau d'un aspect spécifique d'un AP. Chaque APO représente un jeu d'informations et de moyens de traitements d'un AP qui sont accessibles par le biais de services de la DP Application Layer. Les APO sont utilisés pour représenter ces moyens à d'autres AP dans un système DP de bus de terrain.

Afin de permettre à un AP de communiquer avec un AP d'un autre service, il est nécessaire que des APO soient disponibles. Les objets d'application (objets virtuels) représentent des objets de processus existants (objets réels), qu'un processus d'application a rendus visibles et accessibles à la communication (voir Figure 9).



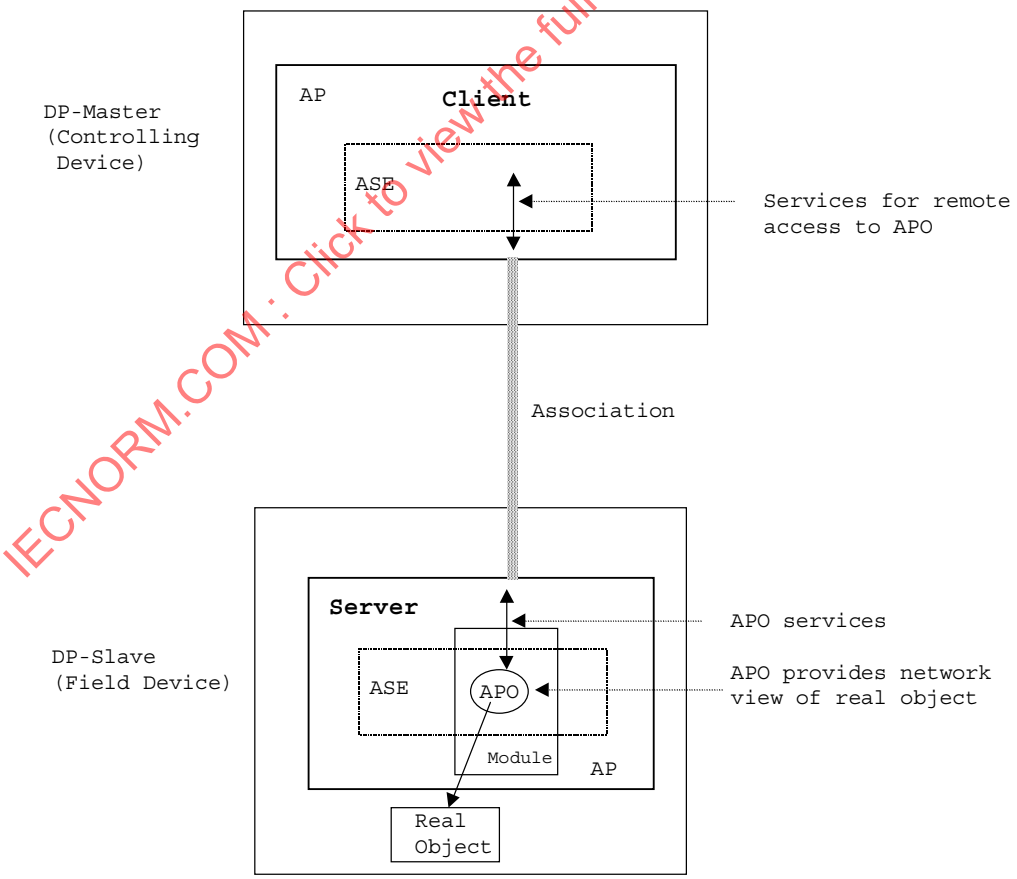
Légende

Anglais	Français
Modular DP-Slave	Esclave DP Modulaire
Real Object	Objet Réel
Default AP(API = 0)	Processus d'application par défaut (API=0)
Module	Module
ASE	ASE
APO	APO
AP (API = 255)	AP (API = 255)
AP (API = 1)	AP (API = 1)
DP-Master (Class 1)	DP-Master (Class 1) ou Maître DP (Classe 1)
DP-Master (Class 2)	DP-Master (Class 2) ou Maître DP (Classe 2)

Figure 9 – Processus d'application avec objets d'application (APO)

Dans la Figure 10, un AP distant agissant comme Client peut accéder à l'objet réel en envoyant des demandes par l'intermédiaire de l'APO qui représente l'objet réel. Les aspects locaux de l'AP assurent la conversion entre la vue réseau (l'APO) de l'objet réel et la vue d'AP interne de l'objet réel.

Au sein d'un AP, un APO est identifié par une baie et un indice. L'espace adresse qui est défini par la baie et l'indice peut être utilisé par plusieurs AP.



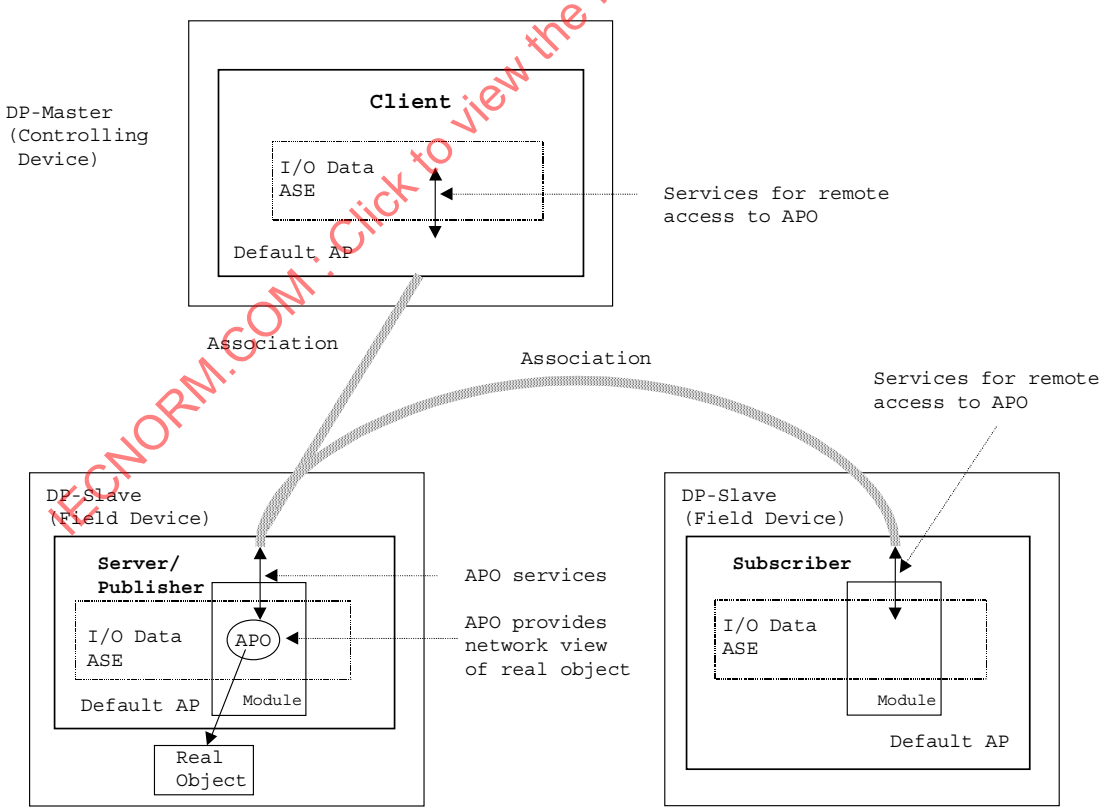
Légende

Anglais	Français
DP-Master (Controlling Device)	Maître DP (Appareil de commande)

Anglais	Français
AP	Processus d'application
Client	Client
ASE	ASE
Services for remote access to APO	Services pour accès distant aux objets de processus d'application
Association	Association
DP-Slave (Field Device)	Esclave DP (Appareil de terrain)
Server	Serveur
APO services	Services APO
APO	Objet de processus d'application
APO provides network view of real object	L'APO fournit une vue réseau de l'objet réel
Real Object	Objet réel
Module	Module

Figure 10 – Accès à un APO distant

La Figure 11 montre une association Client/Server (Client/Serveur) et une association Publisher/Subscriber (Éditeur/Abonné). Le Client peut accéder à l'objet réel en envoyant des demandes par l'intermédiaire de l'APO qui représente l'objet réel. L'Abonné peut s'abonner à tout ou partie de l'APO distant. L'association Éditeur/Abonné est toujours relative au Default AP et l'ASE I/O Data. Les aspects locaux de l'AP assurent la conversion entre la vue réseau (l'APO) de l'objet réel et la vue d'AP interne de l'objet réel.



Légende

Anglais	Français
DP-Master (Controlling Device)	Maître DP (Appareil de commande)

Anglais	Français
Client	Client
I/O Data	Données E/S
ASE	ASE
Services for remote access to APO	Services pour accès distant aux objets de processus d'application
Default AP	Processus d'application par défaut
Association	Association
DP-Slave (Field Device)	Esclave DP (Appareil de terrain)
Server/Publisher	Serveur/Éditeur
Subscriber	Abonné
APO provides network view of real object	L'APO fournit une vue réseau de l'objet réel
APO services	Services APO
APO	Objet de processus d'application
Real Object	Objet réel
Module	Module

Figure 11 – Accès à un APO distant pour association éditeur/abonné

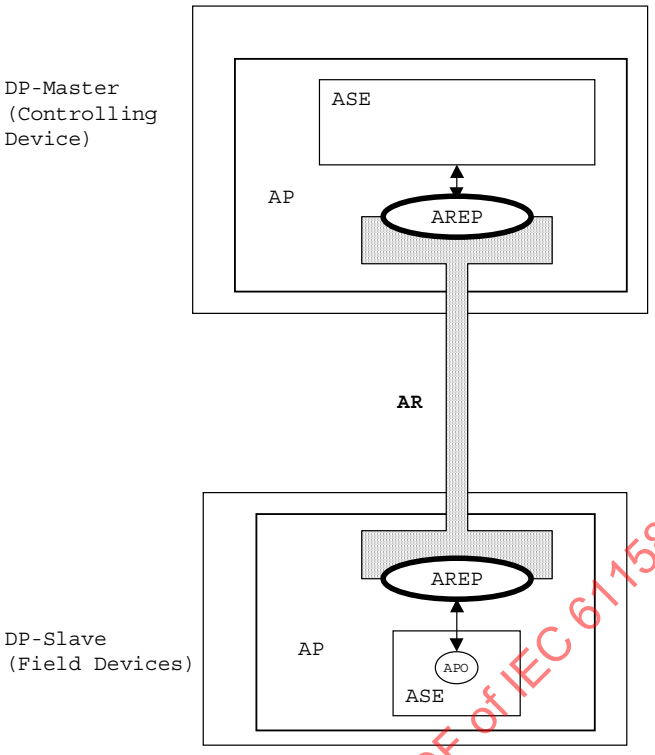
6.1.3.5 Relation entre applications

6.1.3.5.1 Généralités

Une Relation entre applications (AR) est une relation coopérative entre deux ou plusieurs AP dans le but d'échanger des informations et de coordonner leur fonctionnement conjoint (voir Figure 12). La relation est activée par l'échange d'unités de données de protocole application (APDU). Le DP de bus de terrain utilise différents types d'AR qui se distinguent par leur caractéristique d'acheminement.

6.1.3.5.2 Point d'extrémité de relation entre applications

Un AP a accès à la communication en utilisant des points d'extrémité de relation entre applications (AREP, voir l'ISO/CEI 7498-3). Un ou plusieurs AREP sont fixes et assignés de manière univoque à un AP. Ces points d'extrémité sont adressés par l'AP par l'intermédiaire d'un identificateur (AREP ID). Ces identificateurs sont spécifiques à un appareil et ne sont pas définis par la communication elle-même. Les AR sont définies comme un jeu d'AREP coopératifs. Entre deux AP, il peut exister une ou plusieurs AR, chacune ayant des AREP uniques. La Figure 12 montre un exemple d'une AR avec deux AREP.



Légende

Anglais	Français
DP-Master (Controlling Device)	Maître DP (Appareil de commande)
ASE	ASE
AP	Processus d'application
AREP	Point d'extrémité de relation entre applications
AR	Relation entre applications
DP-Slave (Field Device)	Esclave DP (Appareil de terrain)
APO	Objet de processus d'application

Figure 12 – Exemple d'une AR avec deux AREP

6.1.3.5.3 Vue d'ensemble de relations entre applications

La couche application DP propose les types d'AR suivants:

MS0: relation d'applications entre le processus d'application d'un maître DP (Classe 1) et tous les esclaves DP connexes et facultativement entre l'AP d'un ou plusieurs maîtres DP (Classe 2) et tous les esclaves DP connexes et facultativement entre l'AP d'un ou plusieurs esclaves DP avec tous les esclaves DP connexes pour les besoins suivants:

- échange cyclique des données E/S avec le maître DP (Classe 1),
- échange cyclique des données d'entrée entre esclaves DP (DXB),
- transfert de données acyclique pour la paramétrisation, la configuration et le diagnostic (maître DP (Classe 1)),
- transfert acyclique de commandes vers un ensemble des appareils de terrain (maître DP (Classe 1)),
- transfert cyclique de messages de synchronisation vers un ensemble des appareils de terrain (maître DP (Classe 1)),
- lecture acyclique des données E/S (maître DP (Classe 2)),
- lecture acyclique d'informations de configuration (maître DP (Classe 2)),
- lecture acyclique d'informations de diagnostic (maître DP (Classe 2)),
- écriture acyclique de paramètre rémanent (maître DP (Classe 2)).

MS1: relation orientée connexion entre les processus d'application d'un maître DP (Classe 1) et d'un esclave DP connexe pour les besoins suivants:

– lecture et écriture acycliques de variables,
– transfert acyclique d'alarmes,
– téléchargement vers l'amont et/ou l'aval de LR Data,
– invocation de fonctions sans état et/ou orientées état.

MS2: relation orientée connexion entre les processus d'application d'un maître DP (Classe 2) et ceux d'un esclave DP connexe pour les besoins suivants:

– lecture et écriture acycliques de variables,
– téléchargement vers l'amont et/ou l'aval de LR Data,
– invocation de fonctions sans état et/ou orientées état.

MS3: relation sans connexion entre les processus d'application d'un maître DP et ceux d'un ensemble d'esclaves DP connexes pour le besoin suivant:

– synchronisation du temps.

MM1: relation sans connexion entre les processus d'application d'un appareil de configuration (Maître de Classe 2) et ceux d'un appareil de commande connexe (Maître de Classe 1) pour les besoins suivants:

– téléchargement vers l'amont et vers l'aval pour des informations de configuration,
– télécharger vers l'amont d'informations de diagnostic,
– activation de la configuration précédemment transférée.

MM2: relation sans connexion entre les processus d'application d'un appareil de configuration (Maître de Classe 2) et ceux d'un ensemble des appareils de commande connexes (Maître de Classe 1) pour le besoin suivant:

– activation des configurations précédemment transférées.

6.1.4 Comportement dynamique de DP de bus de terrain

Avant de mettre en service un système DP, toutes les stations doivent avoir eu une adresse unique qui leur est assignée. Dans le cas des esclaves DP, cette adresse peut être établie en utilisant l'AR MS0 (voir 6.2.5.3.9).

Tous les esclaves DP auxquels une adresse individuelle n'a pas encore été assignée doivent démarrer avec l'adresse par défaut 126. Un seul appareil avec cette adresse doit être autorisé sur le réseau. Une adresse de maître DP ne doit pas être mise à l'adresse par

défaut. Un maître DP (Classe 2) peut accéder à un esclave DP par l'adresse par défaut et peut assigner une adresse spécifique à l'appareil en question.

Pour des raisons de sécurité, le maître DP (Classe 1) ne doit pas échanger de données E/S avec un esclave DP dont l'adresse est 126.

Pour échanger des données E/S, un maître DP (Classe 1) doit avoir un jeu de paramètres de maître (Master Parameter Set) valide qui peut être chargé et activé par le maître DP (Classe 2) par le biais de l'AR MM1 et de l'AR MM2. Le Master Parameter Set est constitué de plusieurs jeux de données et de paramètres de configuration, dédiés chacun à un esclave DP. Ce jeu de données doit contenir toutes les informations nécessaires (pour le système DP) pour décrire un esclave DP. En outre, le Master Parameter Set doit inclure les paramètres de bus ainsi que la table d'affectation d'adresses qui affecte à chaque donnée E/S distante prise séparément une adresse unique dans l'espace E/S de l'utilisateur du maître DP.

S'il existe un Master Parameter Set valide dans un maître DP (Classe 1), cet appareil commence à vérifier si les esclaves DP dédiés au maître DP en question sont présents ou non (cela est effectué par une lecture du diagnostic d'un esclave DP). Après la réponse appropriée, le maître DP (Classe 1) établira les paramètres et vérifiera la configuration de l'esclave DP. Un esclave DP acceptera une demande de vérification de configuration issue seulement du Maître qui établit les paramètres.

Après avoir correctement présenté ces deux fonctions, un maître DP (Classe 1) vérifie le statut de l'esclave DP en lisant les informations Diagnosis. Si la vérification de configuration a été couronnée de succès et la paramétrisation correcte, le maître DP (Classe 1) entrera dans le mode "échange de données" utilisateur. Un esclave DP accepte des demandes d'échange de données issues seulement d'un maître DP (Classe 1) qui a précédemment présenté les paramètres et la configuration. Si un événement diagnostic s'est produit dans le processus d'application d'un esclave DP, cet événement est signalé par l'envoi d'une réponse de haute priorité. Le maître DP (Classe 1) lira le diagnostic et informera l'utilisateur local.

Le maître DP (Classe 1) envoie des informations relatives à son propre mode de fonctionnement (CLEAR, OPERATE) à ses esclaves DP dédiés en vue de la synchronisation. Celle-ci se fera à chaque changement du mode de fonctionnement du maître DP (Classe 1) et, en plus, à intervalles réguliers (Data Control Time/2, c'est-à-dire Temps de commande de données/2).

C'est seulement dans le mode d'échange de données que les fonctionnalités étendues comme la lecture et l'écriture acycliques de variables, le transfert acyclique d'alarmes, le téléchargement amont et/ou aval de LR Data, l'invocation de fonctions sans état et/ou orientées états, la fonctionnalité DXB et le mode isochrone (Isochronous Mode) sont possibles, si elles sont prises en charge par le maître DP (Classe 1) et aussi par l'esclave DP. Les erreurs sur la communication acyclique (AR MS1) ont aussi une influence sur le transfert cyclique de données (AR MS0 AR) et réciproquement. Cela signifie que les deux relations de communication sont abandonnées. Chaque alarme est à acquitter par le maître DP (Classe 1) assigné.

Chaque maître DP (Classe 2) peut lire les informations de diagnostic, la configuration, les données d'entrée et de sortie (par le biais de l'AR MS0) de chaque esclave D bien que l'esclave DP soit sous le contrôle d'un maître DP (Classe 1). Un maître DP (Classe 2) peut prendre un esclave DP sous son contrôle. Dans ce cas, l'esclave DP arrêtera le mode "échange de données" avec le maître DP (Classe 1). Dans ce cas, le maître DP (Classe 1) commence à lire le diagnostic de façon cyclique. Si le maître DP (Classe 2) a fini la communication avec cet esclave DP, l'esclave DP mettra le champ "Master Add(ress)" à "invalid" (non valide). Cet événement déclenche le maître DP (Classe 1) pour qu'il ait de nouveau le contrôle sur cet esclave DP. Comme décrit ci-dessus, il enverra d'abord des paramètres et la configuration.

Si elle est prise en charge tant par le maître DP (Classe 2) que par l'esclave DP, la fonctionnalité étendue comme la lecture et l'écriture acycliques de variables, le téléchargement amont et/ou aval de LR Data et l'invocation de fonctions sans état et/ou orientées états est possibles (par le biais de l'AR MS2). L'AR MS2 est indépendante de l'AR MS0 et de l'AR MS1.

Le DP de bus de terrain définit un modèle de redondance avec les caractéristiques suivantes:

– Un seul modèle de redondance d'esclave pour toutes les structures de redondance (même pour les systèmes non redondants),
– Échelonnabilité: Redondance indépendante de maître, esclave, ligne,
– Ingénierie aisée: Aucun effort supplémentaire de l'utilisateur, aucun outil complexe nécessaire,
– Surveillance complète de tous les composants,
– Aucune influence sur la charge et la temporisation du bus,
– Fiabilité élevée,
– Courts temps de commutation,
– Aucune perte de données pendant la tolérance aux pannes.

Un esclave DP redondant peut être constitué de deux interfaces de communication avec des extensions de redondance spéciales et d'une voie de communication de redondance indépendante entre ces deux interfaces de communication. L'un des modules d'interface esclaves agit comme l'esclave primaire et l'autre comme l'esclave de secours.

Le modèle de redondance de DP de bus de terrain repose sur une commande de commutation, des commandes d'arrêter, de démarrer et de réinitialiser l'AR MS1, fournies par l'AR MS0 par le biais du service Check User Prm à l'application de l'esclave DP et un objet Diagnosis (Red Status) qui fournit des informations relatives à l'état de l'esclave DP redondant s'il est primaire ou de secours et si une commande fournie précédente (commutation, démarrage, arrêt réinitialisation de l'AR MS) a été traitée.

Au cours du traitement d'une commande de commutation ou d'une commande d'arrêt, de démarrage ou de réinitialisation, l'AR MS0 reste dans le mode d'échange de données. Seule l'AR MS1 est influencée et, donc, une prise de contrôle sans couture d'un esclave DP peut être accomplie.

La communication avec un esclave DP redondant est effectuée de la manière suivante:

AR MS0:

L'esclave primaire utilise l'AR MS0 (communication cyclique) pour envoyer et recevoir des données E/S. En outre, l'esclave primaire envoie toutes les informations de diagnostic relatives à lui-même et à l'esclave de secours.

L'esclave de secours peut aussi envoyer et recevoir des valeurs pour les données E/S par l'intermédiaire de l'AR MS0, mais ces valeurs ne sont pas pertinentes et doivent être ignorées par l'esclave de secours. L'esclave de secours peut émettre ses propres informations de diagnostic, mais elles doivent aussi faire partie du diagnostic du primaire.

AR MS1:

Seul l'esclave primaire recevra et émettra les services de l'AR MS1. Lorsque l'esclave de secours reçoit un service d'AR MS1, il doit répondre négativement.

AR MS2:

L'esclave primaire et l'esclave de secours peuvent tous deux recevoir des services acycliques de l'AR MS2. Les deux modules d'interface esclaves doivent traiter ces services pour permettre la communication avec l'appareil spécifique. Si des valeurs sont envoyées avec les services de l'AR MS2, ces données peuvent être envoyées vers un seul module d'interface esclave et doivent être échangées avec l'autre en utilisant la voie de communication redondante entre les deux interfaces de communication. Seules des données, qui sont tenues dans les deux modules d'interface esclaves, doivent être échangées pour les maintenir cohérentes.

Les caractéristiques pour la redondance pourraient aussi bien être utilisées pour des systèmes avec maîtres de secours et fournir des moyens pour un fonctionnement cyclique continu.

6.1.5 Valeurs de statut de DP de bus de terrain

Le Tableau 2 montre les Status Values (valeurs de statut) des primitives de service.

Tableau 2 – Valeurs de statut des primitives de service

Nom court	Signification
ABT_FE	APDU de demande non valide reçue
ABT_IA	Informations d'adresse supplémentaires non valides
ABT_IV	Service non valide pour l'utilisateur
ABT_OC	En attente de DL-DATA-REPLY.cnf
ABT_RE	APDU de réponse non valide reçue
ABT_SE	Erreur de séquence; service non autorisé dans cet état
ABT_STO	La valeur demandée de Send Timeout était trop petite.
ABT_TO	Temporisation de la connexion
AD	Accès refusé
DH	Acquittement possible pour les données envoyées, les données de réponse avec haute priorité disponible
DI	Données incomplètes
DS	Local-DL/PHY Entity n'est pas dans l'anneau à jeton logique ou déconnectée de la ligne.
EA	Zone trop grande (téléchargement amont/aval)
FE	Erreur de format dans une APDU de demande
GE	Erreur de groupe, groupe non valide à la demande de service (Commande globale)
IP	Paramètre non valide
IV	Paramètres non valides dans la demande
LE	Longueur de bloc de données trop grande (téléchargement amont/aval)
LR	Ressources locales indisponibles ou insuffisantes
LS	Service au DLSAP local ou DLSAP local non activé
NA	Ack négatif, aucune réaction de la station distante
NC	Master Parameter Set non compatible
NE	Zone inexistante
NI	Service non mis en œuvre
NO	Service dans cet état non possible
NR	Aucune donnée de réponse
RDH	Données de réponse de haute priorité et pas de ressources pour les données envoyées
RDL	Données de réponse de basse priorité et pas de ressources pour les données envoyées
RE	Erreur de format dans une APDU de réponse
RR	Ressources de l'entité de DL distante insuffisantes ou non disponibles
RS	Service ou adresse distante au DLSAP distant ou bien DLSAP distant non activé - le poste distant n'est pas un poste DP (DP-Station) - le poste distant n'est pas encore prêt pour ces services - le poste distant est associé à un autre demandeur - service facultatif non disponible
SC	Conflit de séquence
SV	Violation de séquence (seconde Base de temps active)
SE	Erreur de séquence
TO	Temporisation de fonction expirée
UE	Erreur d'interface DMPM distant / DL

6.2 ASE

6.2.1 ASE “Process data"

6.2.1.1 Vue d'ensemble

Dans l'environnement de la couche application DP, les processus d'application contiennent des données que des applications distantes sont capables de lire et écrire. L'ASE Process Data définit des attributs des objets Process Data d'usage général et fournit un jeu de services utilisés pour lire et écrire leurs valeurs. Un ASE Process Data est spécifique pour un AP et chaque AP peut contenir un ASE Process Data Les objets Process Data au sein d'un AP sont adressés par des Slot Number (numéros de baie) et des Index (indices). L'objet Process Data peut être lu et écrit, partiellement ou complètement.

Les maîtres DP (Classe 1) et le maître DP (Classe 2) sont capables de lire ou écrire la valeur de l'objet Process Data au sein du Default AP d'un esclave DP. Les objets Process Data à l'extérieur du Default AP d'un esclave DP peuvent seulement être accessibles par le maître DP (Classe 2).

Il est nécessaire que le maître DP (Classe 2) utilise l'ASE Context (voir 6.2.5.3.10) pour établir une association afin d'avoir accès aux objets Process Data d'un esclave DP. L'objet Process Data en combinaison avec l'ASE Context offre la possibilité de protection d'accès (voir 6.2.1.3).

Le maître DP (Classe 1) peut accéder aux valeurs de l'objet Process Data au sein du Default AP après que l'esclave DP est entré dans le mode d'échange de données. Cette association n'offre aucune protection d'accès pour l'objet Process Data accessible.

L'accès à des objets Process Data est réalisé conformément au modèle d'accès Client/Serveur. Le modèle Client/Serveur est caractérisé en ce que l'application Client envoie une demande de lecture ou d'écriture à une application Serveur qui répond en conséquence. L'activité du Serveur est stimulée par les Clients sur le réseau.

Le modèle formel de l'ASE Process Data est présenté par la spécification pour la classe Process Data, contenant une description de ses attributs ainsi que ses invocations, suivie d'une spécification particulière des services.

6.2.1.2 Spécification de la classe de données de processus (Process data)

6.2.1.2.1 Généralités

L'ASE Process Data définit deux types d'objets Process Data:

— les objets simples,

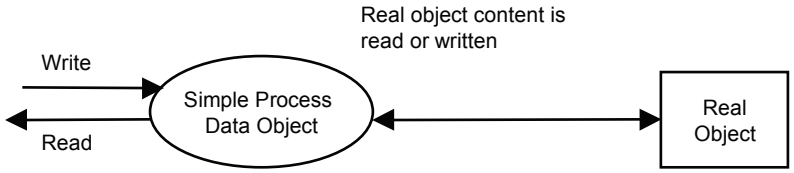
— les objets combinés,

Les objets simples ont une relation un à un aux objets réels. Les objets combinés ont une relation un à deux aux objets réels. Les objets combinés peuvent être lus ou écrits avec une interaction de services. Ce type d'objets peut être utilisé pour la communication qui exige une cohérence entre lecture et écriture ou une haute efficacité de protocole.

6.2.1.2.2 Spécification de la classe de données de processus simples

6.2.1.2.2.1 Modèle

Un objet Simple Process Data est toujours relié à un objet réel comme montré à la Figure 13.



Légende

Anglais	Français
Real Object	Objet réel
Simple Process Data Object	Objet de données de processus simple
Real object content is read or written	Le contenu de l'objet réel est lu ou écrit
Read	Lecture
Write	Écriture

Figure 13 – Relation d'un objet "données de processus simples" à l'objet réel

Un objet "Process Data" est décrit par le modèle suivant:

ASE DP: **ASE Process Data**
CLASS: **Simple Process Data**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attributs-clés:	Identifier
2.	(m)	Attribut:	Process Data Description
3.	(m)	Attribut:	Partial Access
4.	(m)	Attribut:	Access Rights MS1
5.	(m)	Attribut:	Access Rights MS2
6.	(c)	Contrainte:	Access Rights MS2 = Right to Read ou Right to Read/Write
6.1.	(m)	Attribut:	List of SCL Read
6.1.1.	(m)	Attribut:	SCL
7.	(c)	Contrainte:	Access Rights MS2 = Right to Write ou Right to Read/Write
7.1.	(m)	Attribut:	List of SCL Write
7.1.1.	(m)	Attribut:	SCL
8.	(o)	Attribut:	Process Data Name
9.	(o)	Attribut:	Local Detail

SERVICES:

1.	(o)	OpsService:	Read
2.	(o)	OpsService:	Write

6.2.1.2.2.2 Attributs

Identifier
Cet attribut-clé est un triplet constitué de l'API, du Slot Number et de l'Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Simple Process Data appartient. Cet Identifier ne doit pas être utilisé par un autre objet Action, un autre objet Function Invocation, un autre objet Load Region ou un autre objet Process Data.

Type d'attributUnsigned8, Unsigned8, Unsigned8

Valeurs admissibles du triplet: [0 à 254, 0 à 254, 0 à 254]

Process Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Partial Access
Cet attribut Boolean définit si un accès partiel à l'objet Simple Process Data est pris en charge ou non. Partial access égal à TRUE signifie que le contenu de l'objet Simple Process Data est lu ou écrit à partir du premier octet jusqu'à l'octet défini par la longueur donnée dans

le service. La longueur donnée dans le service doit être inférieure ou égale à la valeur de la longueur d'attribut dans le cas d'un accès en écriture. Pour un accès en lecture, l'objet tout entier est lu si la longueur donnée dans le service est supérieure à la longueur de l'objet. Partial access égal à FALSE signifie que le contenu de l'objet Simple Process Data peut seulement être entièrement lu ou écrit. Cela implique que la longueur donnée dans le service doit être égale ou (dans le cas de l'accès en lecture) supérieure à la longueur de l'objet

Type d'attribut: Boolean

Access Rights MS1
Cet attribut définit le type de droits d'accès définis pour une AR MS1 pour l'objet Simple Process Data.

Les valeurs admissibles sont montrées dans le Tableau 3.

Tableau 3 – Droits d'accès MS1

Valeur	Signification
0	Right to Write (Droit d'écrire)
1	Right to Read (Droit de lire)
2	Right to Read/Write (Droit de lire et d'écrire)
3	No Access (Aucun accès)

Access Rights MS2
Cet attribut définit le type de droits d'accès définis pour une AR MS2 pour l'objet Simple Process Data.

Les valeurs admissibles sont montrées dans le Tableau 4.

Tableau 4 – Droits d'accès MS2

Valeur	Signification
0	Right to Write (Droit d'écrire)
1	Right to Read (Droit de lire)
2	Right to Read/Write (Droit de lire et d'écrire)
3	No Access (Aucun accès)

List of SCL Read
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur d'octet pour la protection d'accès en lecture à cet objet Simple Process Data. Les règles pour accepter ou refuser un accès en lecture sont décrites en 6.2.1.3.

Type d'attribut: Unsigned8

List of SCL Write
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur d'octet pour la protection d'accès en écriture à cet objet Simple Process Data. Les règles pour accepter ou refuser un accès en écriture sont décrites en 6.2.1.3.

Type d'attribut: Unsigned8

Process Data Name
Cet attribut facultatif définit le nom de l'objet Process Data.

Type d'attribut: Visible String(32)

Local Detail
Cet attribut facultatif contient des informations locales complémentaires concernant l'objet Simple Process Data.

Type d'attribut: Octet String

6.2.1.2.2.3 Invocation de l'objet "simple process data"

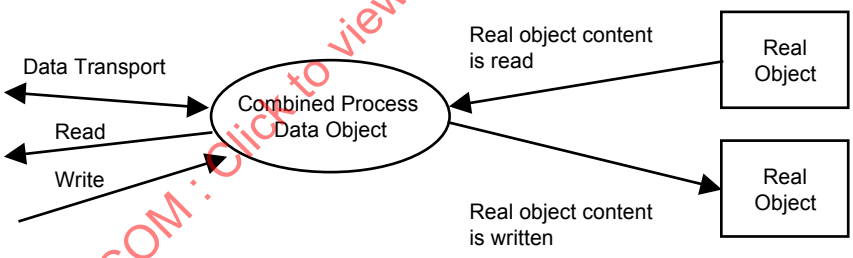
Pour l'invocation de l'objet Simple Process Data, les règles suivantes s'appliquent:

— Les objets Simple Process Data ne doivent pas dépasser la longueur totale de 240 octets. L'attribut Process Data Description est à positionner en conséquence.
— Au moins un service doit être autorisé pour l'accès à l'objet.
— Au moins une AR doit avoir accès à l'objet.
— Les droits d'accès sont à établir en conséquence au(x) service(s) autorisé(s).

6.2.1.2.3 Spécification de la classe de "données de processus combinées"

6.2.1.2.3.1 Modèle

Un objet Combined Process Data est toujours relié à deux objets réels comme montré à la Figure 14.



Légende

Anglais	Français
Real Object	Objet réel
Simple Process Data Object	Objet de données de processus simple
Real object content is read	Le contenu de l'objet réel est lu
Read	Lecture
Write	Écriture
Combined Process Data Object	Objet de données de processus combiné
Real object content is written.	Le contenu de l'objet réel est écrit.
Data Transport	Transport de données

Figure 14 – Relation d'un objet "données de processus combinées" aux objets réels

Un objet "Combined Process Data" est décrit par le modèle suivant:

ASE DP:			**ASE Process Data**
CLASS:			**Combined Process Data**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attributs-clés:	Identifier
2.	(m)	Attribut:	Read Process Data Description
3.	(m)	Attribut:	Read Partial Access
4.	(c)	Contrainte:	Access Rights MS2 = Right to Read ou Right to Read/Write
4.1.	(m)	Attribut:	List of SCL Read
4.1.1.	(m)	Attribut:	SCL
5.	(m)	Attribut:	Write Process Data Description
6.	(m)	Attribut:	Write Partial Access
7.	(c)	Contrainte:	Access Rights MS2 = Right to Write ou Right to Read/Write
7.1.	(m)	Attribut:	List of SCL Write
7.1.1.	(m)	Attribut:	SCL
8.	(m)	Attribut:	Access Rights MS1
9.	(m)	Attribut:	Access Rights MS2
10.	(o)	Attribut:	Process Data Name
11.	(o)	Attribut:	Local Detail
SERVICES:			
1.	(o)	OpsService:	Data Transport
2.	(o)	OpsService:	Read
3.	(o)	OpsService:	Write

6.2.1.2.3.2 Attributs

Identifier
Cet attribut-clé est un triplet constitué de l'API, du Slot Number et de l'Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Combined Process Data appartient. Cet Identifier ne doit pas être utilisé par un autre objet Action, un autre objet Function Invocation, un autre objet Load Region ou un autre objet Process Data.

Type d'attribut: Unsigned8, Unsigned8, Unsigned8

Valeurs admissibles du triplet: [0 à 254, 0 à 254, 0 à 254]

Read Process Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description, de l'ASE Process Data. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Read Partial Access
Cet attribut Boolean définit si un accès partiel en lecture à l'objet Combined Process Data est pris en charge ou non. Partial access égal à TRUE signifie que le contenu de l'objet Combined Process Data est lu ou écrit à partir du premier octet jusqu'à l'octet défini par la longueur donnée dans le service. L'objet tout entier est lu si la longueur donnée dans le service est supérieure à la longueur de l'objet. Partial access égal à FALSE signifie que le contenu de l'objet Combined Process Data peut seulement être entièrement lu. Cela implique que la longueur donnée dans le service doit être supérieure ou égale à la longueur de l'objet.

Type d'attribut: Boolean

List of SCL Read
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur d'octet pour la protection d'accès en lecture à cet objet Combined Process Data. Les règles pour accepter ou refuser un accès en lecture sont décrites en 6.2.1.3.

Type d'attribut: Unsigned8

Write Process Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Write Partial Access
Cet attribut Boolean définit si un accès partiel en écriture à l'objet Combined Process Data est pris en charge ou non. Partial access égal à TRUE signifie que le contenu de l'objet Combined Process Data est écrit à partir du premier octet jusqu'à l'octet défini par la longueur donnée dans le service. La longueur donnée dans le service doit être inférieure ou égale à la valeur de la longueur d'attribut. Partial access égal à FALSE signifie que le contenu de l'objet Combined Process Data peut seulement être entièrement écrit. Cela implique que la longueur donnée dans le service doit être égale à la longueur de l'objet.

Type d'attribut: Boolean

List of SCL Write
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur d'octet pour la protection d'accès en écriture à cet objet Combined Process Data. Les règles pour accepter ou refuser un accès en écriture sont décrites en 6.2.1.3.

Type d'attribut: Unsigned8

Access Rights MS1
Cet attribut définit le type de droits d'accès définis pour une AR MS1 pour l'objet Combined Process Data.

Les valeurs admissibles sont montrées dans le Tableau 5.

Tableau 5 – Droits d'accès MS1

Valeur	Signification
0	Right to Write (Droit d'écrire)
1	Right to Read (Droit de lire)
2	Right to Read/Write (Droit de lire et d'écrire)
3	No Access (Aucun accès)

Access Rights MS2
Cet attribut définit le type de droits d'accès définis pour une AR MS2 pour l'objet Combined Process Data.

Les valeurs admissibles sont montrées dans le Tableau 6.

Tableau 6 – Droits d'accès MS2

Valeur	Signification
0	Right to Write (Droit d'écrire)
1	Right to Read (Droit de lire)
2	Right to Read/Write (Droit de lire et d'écrire)
3	No Access (Aucun accès)

Process Data Name
Cet attribut facultatif définit le nom de l'objet Process Data.

Type d'attribut: Visible String(32)

Local Detail
L'attribut Local Detail contient des informations locales complémentaires concernant l'objet Simple Process Data.
Type d'attribut: Octet String

6.2.1.2.3.3 Invocation de l'objet "Combined process data"

Pour l'invocation de l'objet Combined Process Data, les règles suivantes s'appliquent:

— Les objets Combined Process Data ne doivent pas dépasser la longueur totale de 240 octets tant pour Read Process Data que pour Write Process Data. Les attributs Read Data Description et Write Data Description sont à mettre en conséquence.

— Au moins un service doit être autorisé pour l'accès à l'objet.

— Au moins une AR doit avoir accès à l'objet.

— Les droits d'accès sont à établir en conséquence au(x) service(s) facultatif(s) autorisé(s).

6.2.1.3 Protection d'accès sur les objets données de processus

L'ASE Process Data en combinaison avec l'ASE Context assure également la protection d'accès. Ce n'est pas 'une protection contre une intentionnelle utilisation impropre des moyens de communication d'un appareil de terrain, mais cela aide à protéger un système contre l'utilisation erronée accidentelle de Process Data.

La protection est assurée par deux mécanismes:

— le service Initiate établira un niveau de sécurité pour cette CR.

— l'objet Process Data aura une liste de valeurs SCL pour le service lecture et écriture. Un accès est accepté seulement si le SCL du service Initiate concorde (voir Tableau 7) avec un élément de la liste de SCL pour cet objet Process Data.

Cela permet à l'utilisateur de sélectionner un niveau pour les objets de lecture et d'écriture. Ainsi, un appareil de service peut utiliser une SCL qui permet l'accès à presque tous les objets. Un affichage textuel peut seulement avoir un accès en lecture à des objets spécifiques.

Tableau 7 – Règles de concordance de SCL

SCL dans Initiate.ind	SCL dans l'objet Process Data	Condition	Résultat
= 0	= 0	-	accès autorisé
≠ 0	= 0	-	accès autorisé
= 0	≠ 0	-	accès refusé
≠ 0	≠ 0	SCL dans Initiate.ind non contenu dans l'attribut List of SCL en l'objet Process Data	accès refusé
≠ 0	≠ 0	SCL dans Initiate.ind contenu dans List of SCL en l'objet Process Data	accès autorisé

6.2.1.4 Spécification de services de Process data

6.2.1.4.1 Read

Ce service confirmé peut être utilisé pour lire la valeur d'un objet Process Data. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2. Le Tableau 8 montre les paramètres du service.

Tableau 8 – Read

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Index	M	M(=)		
Length	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Length			M	M(=)
Data			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Process Data souhaité dans une baie spécifique (typiquement un module).

Index
Le paramètre Index est utilisé dans l'appareil de destination pour adresser l'objet Process Data souhaité.

Length
Le paramètre Length indique le nombre d'octets d'un objet Process Data à lire.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Data
Le paramètre Data contient la valeur de l'objet qui a été lue et est constitué du nombre d'octets indiqués dans le paramètre Length de la réponse. Ce paramètre doit être constitué des types de données définis à l'Article 5.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Decode
Ce paramètre sélectionne l'un des schémas d'erreur (Error scheme) suivants: DPV1, FMS, Profile_Specific.

Error Code 1
Lorsque Error Decode est égal à DPV1, le code d'erreur (Error Code) 1 prend l'une des valeurs suivantes:
read error (erreur de lecture), module failure (défaillance de module), version conflict

(conflit de versions), feature not supported (caractéristique non prise en charge), user specific (spécifique à un utilisateur), invalid index (indice non valide), invalid slot (baie non valide), type conflict (conflit de types), invalid area (zone non valide), state conflict (conflit d'états), access denied (accès refusé), invalid range (plage non valide), invalid parameter (paramètre non valide), invalid type (type non valide), read constrain conflict (conflit de contraintes en lecture), resource busy (ressource occupée), resource unavailable (ressource non disponible).

Error Code 2
Le paramètre Error Code 2 est spécifique à un utilisateur.

6.2.1.4.2 Write

Ce service confirmé peut être utilisé pour écrire la valeur d'un objet Process Data. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2. Le Tableau 9 montre les paramètres du service.

Tableau 9 – Write

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Index	M	M(=)		
Length	M	M(=)		
Data	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Length			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Process Data souhaité dans une baie spécifique (typiquement un module).

Index
Le paramètre Index est utilisé dans l'appareil de destination pour adresser l'objet Process Data souhaité.

Length
Le paramètre Length indique le nombre d'octets d'un objet Process Data à écrire.

Data
Le paramètre Data contient la valeur de l'objet Process Data à écrire et est constitué du nombre d'octets indiqués dans le paramètre Length de la demande. Ce paramètre doit être constitué des types de données définis à l'Article 5.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Decode
Ce paramètre sélectionne l'un des schémas d'erreur (Error scheme) suivants: DPV1, FMS, Profile_Specific.

Error Code 1
Lorsque Error Decode est égal à DPV1, le code d'erreur (Error Code) 1 prend l'une des valeurs suivantes:
write error (erreur d'écriture), module failure (défaillance de module), version conflict (conflit de versions), feature not supported (caractéristique non prise en charge), user specific (spécifique à un utilisateur), invalid index (indice non valide), write length error (erreur de longueur en écriture), invalid slot (baie non valide), type conflict (conflit de types), invalid area (zone non valide), state conflict (conflit d'états), access denied (accès refusé), invalid range (plage non valide), invalid parameter (paramètre non valide), invalid type (type non valide), write constrain conflict (conflit de contraintes en écriture), resource busy (ressource occupée), resource unavailable (ressource non disponible).

Error Code 2
Le paramètre Error Code 2 est spécifique à un utilisateur.

6.2.1.4.3 Data transport

Ce service est utilisé pour écrire et lire les valeurs d'un seul objet Combined Process Data. Ce service peut être utilisé seulement par le biais de l'AR MS2. Le Tableau 10 montre les paramètres du service.

Avec ce service, les données d'un objet Process Data sont transférées

— du maître DP vers l'esclave DP,

— de l'esclave DP vers le maître DP (Classe 2), ou

— du maître DP vers l'esclave DP et retour.

Tableau 10 – Data transport

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Index	M	M(=)		
Length	M	M(=)		
Data	C	C(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Length			M	M(=)
Data			C	C(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser la baie souhaitée (typiquement un module).

Index
Le paramètre Index est utilisé dans l'appareil de destination pour adresser les données souhaitées.

Length
Dans la demande, le paramètre Length indique le nombre d'octets du paramètre "Data" qui sont transférés du Client vers le Serveur. Dans la réponse, le paramètre Length indique le nombre d'octets du paramètre "Data" qui sont transférés du Serveur vers le Client.

Data
Le paramètre Data contient les valeurs transférées du Client vers le Serveur et est constitué du nombre d'octets indiqués dans le paramètre Length. Ce paramètre doit être constitué des types de données définis à l'Article 5.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Decode
Ce paramètre sélectionne l'un des schémas d'erreur (Error scheme) suivants: DPV1, FMS, Profile_Specific.

Error Code 1
Lorsque Error Decode est égal à DPV1, le code d'erreur (Error Code) 1 prend l'une des valeurs suivantes: read error (erreur de lecture), write error (erreur d'écriture), module failure (défaillance de module), version conflict (conflit de versions), feature not supported (caractéristique non prise en charge), user specific (spécifique à un utilisateur), invalid index (indice non valide), write length error (erreur de longueur en écriture), invalid slot (baie non valide), type conflict (conflit de types), invalid area (zone non valide), state conflict (conflit d'états), access denied (accès refusé), invalid range (plage non valide), invalid parameter (paramètre non valide), invalid type (type non valide), read constrain conflict (conflit de contraintes en lecture), write constrain conflict (conflit de contraintes en écriture), resource busy (ressource occupée), resource unavailable (ressource non disponible).

Error Code 2
Le paramètre Error Code 2 est spécifique à un utilisateur.

6.2.2 ASE I/O Data

6.2.2.1 Vue d'ensemble

Dans l'environnement couche d'application DP, le processus d'application par défaut d'un esclave DP peut contenir des objets I/O Data qui doivent être acheminés de manière cyclique sur le réseau. Par conséquent, l'ASE I/O Data définit les attributs des objets I/O Data et fournit un jeu de services utilisés pour récupérer et établir le tampon pour transporter leurs valeurs. Les objets I/O Data (données E/S) sont séparés en objet Input Data (Données d'entrée) et objet Output Data (Données de sortie).

Un objet Input Data ou un objet Output Data doit être partitionné en éléments Input Data ou éléments Output Data qui représentent les I/O Data des modules. Les Input Data distants souscris en combinaison avec les Output Data représentent les données de sortie réelles de l'esclave DP.

Des services supplémentaires sont fournis pour lire de façon acyclique les valeurs des objets I/O Data et pour indiquer les nouvelles valeurs pour l'objet Input Data et l'objet Output Data. De plus, des services sont fournis pour indiquer la présence de nouveaux objets Input Data souscrits et en récupérer le contenu.

Les objets I/O Data sont adressés de façon implicite par l'intermédiaire des services connexes. La granularité des données d'entrée ou de sortie dans un Serveur/Éditeur est fonction de l'objet MSO Configuration Elements correspondant. Un Client est toujours relatif à la totalité des Input Data et/ou Output Data d'un Serveur. Un Abonné peut être relatif à la totalité des Input Data ou à une partie des Input Data d'un Éditeur en fonction des valeurs d'attribut de l'objet DXB-Linktable.

L'ASE I/O Data utilise le modèle d'accès Client/Serveur et Éditeur/Abonné. Le modèle Client/Serveur est caractérisé en ce qu'une application Client achemine la valeur pour l'objet Output Data vers le tampon du Serveur. L'application Serveur récupère cette valeur par un service Get Output. La réception d'une nouvelle valeur est indiquée par le service New Output. L'application Serveur achemine la valeur de son objet Input Data par le service Set Input vers le tampon du Serveur dès que l'une des valeurs d'objet réel change. L'application Client récupère cette valeur par le service Get Input. L'émission du tampon Serveur vers le tampon Client et vice versa est effectuée de façon asynchrone par rapport aux services Get Output et Set Input.

Le modèle Éditeur/Abonné est caractérisé en ce que des esclaves DP agissent comme Éditeur qui édite les valeurs des objets Input Data et comme Abonnés qui s'abonnent à une ou plusieurs valeurs de l'objet Input Data d'un ou plusieurs Éditeurs (esclaves DP). L'application Éditeur achemine la valeur de son objet Input Data (objet Simple Input Data ou objet Extended Input Data) par le service Set Input vers le tampon Serveur dès que l'une des valeurs d'objets réels change. Le Client (maître DP (Classe 1)) amène l'Éditeur à éditer les

valeurs de l'objet Input Data object afin que ces valeurs soient disponibles du côté Client et en chaque Abonné présent sur le réseau. L'Abonné s'abonne à autant d'objets Input Data qu'en indiquent les valeurs d'attribut de DXB-Linktable ou DXB-Subscribertable. La réception de nouvelles valeurs d'Input Data souscrits est indiquée par le service New Publisher Data en l'Abonné. L'application Abonné récupère ces valeurs par le service Get Publisher Data.

Il est possible de mettre à jour des parties seulement des valeurs des objets I/O Data en fonction de la granularité donnée concernant la structure des objets. En outre, il est également possible qu'un esclave DP ait seulement soit un objet Input Data, soit un objet Output Data.

La plupart des esclaves DP prennent seulement en charge les structures Octet ou Word (mot) pour l'objet Input Data et l'objet Output Data. Ce type d'appareils utilise l'objet Simple Input and Output Data. Cependant, des appareils complexes utilisent des valeurs Floating Point (virgule flottante) ou des structures Doubleword (mots doubles), voire des combinaisons de plusieurs types de données. L'objet Extended Input/Output Data reflète les exigences de ces appareils évolués.

Les valeurs des objets I/O Data de la totalité ou d'un groupe des esclaves DP au sein d'un réseau donné peuvent être synchronisées par son maître DP assigné avec l'aide du service Global Control.

La synchronisation des valeurs des objets Input Data est réalisée la Freeze Command et la synchronisation des valeurs des objets Output Data est réalisée par la Sync Command. Ce mécanisme de synchronisation n'a aucun effet sur la réception des Input Data souscrits.

Avec chaque Freeze Command, la valeur courante de l'objet Input Data dans le tampon MS0 est gelée dans les esclaves DP adressés. Si l'application Serveur met maintenant à jour la valeur de l'objet Input Data par le service Set Input, cette valeur n'est pas émise vers le tampon jusqu'à ce que la prochaine Command se produise. Autrement dit, les esclaves DP adressés achemineront les valeurs "gelées" des objets Input Data. La synchronisation des objets Input Data est désactivée par une Freeze Command avec la valeur Unfreeze.

Avec la première Sync Command, la valeur courante de l'objet Ouput Data dans le tampon MS0 est gelée dans les esclaves DP adressés. Avec chaque Sync Command ultérieure, les valeurs précédemment émises des objets Output Data sont mises à jour au tampon d'AR MS0 pour un accès par le service Get Output. La synchronisation des objets Output Data est désactivée par une Sync Command avec la valeur Unsync. Une description particulière est donnée en 6.2.8.3.2.

Les valeurs des objets Input Data souscrits ne sont pas synchronisées aux valeurs de l'objet Output Data dan l'Abonné.

Le maître DP (Classe 1) envoie des informations concernant son propre mode de fonctionnement à ses esclaves DP assignés en vue de la synchronisation. Cela est réalisé à chaque changement du mode de fonctionnement du maître DP (Classe 1) ou a lieu dans un intervalle de temps qui peut être paramétré. Si l'état du maître DP est CLEAR, les valeurs de l'objet Output Data doivent être mises à une valeur correspondant à un état de sécurité.

Les valeurs des objets I/O Data pour la totalité ou un groupe des esclaves DP au sein d'un réseau donné peuvent être synchronisées par le maître DP (Classe 1) assigné par un mécanisme de synchronisation spécial (Isochronous Mode).

La synchronisation des valeurs des objets Input Data et la synchronisation des valeurs des objets Output Data sont accomplies par le message de synchronisation (SYNCH) et des valeurs de réglage fixes du temps relatives au message de synchronisation au sein des esclaves DP impliqués et du maître DP (Classe 1).

Les valeurs des objets Input Data souscrits dans l'Abonné prenant en charge l'Isochronous Mode peuvent également être synchronisées par l'Isochronous Mode.

Le maître DP (Classe 1) communique simultanément, par l'AR MS0, avec tous les objets Input Data et Output Data des esclaves DP assignés. De plus, l'application Client peut communiquer avec un objet Input Data, un objet Output Data, un Input Data Element ou un Output Data Element d'un seul esclave DP.

Le maître DP (Classe 2) communique, par l'AR MS0, avec les objets Input Data et Output Data d'un seul esclave DP.

Un esclave DP, configuré comme Abonné, communique avec l'/les objet(s) Input Data d'un ou plusieurs esclaves DP, configuré(s) comme Éditeur et au(x)quel(s) il est abonné par l'intermédiaire AR MS0.

Le mapping des Input Data souscrits avec la totalité ou des parties des Output Data Elements de l'esclave DP peut être effectuée de manière spécifique à une application ou peut être accomplie par l'application de l'esclave DP avec le DXB-Subscribertable conjointement aux données de configuration.

Le modèle formel de l'ASE "I/O Data" fait l'objet d'une présentation ci-après, suivie d'une description de ses services.

6.2.2.2 Spécification de la classe "I/O data"

6.2.2.2.1 Spécification de la classe "Simple input data"

6.2.2.2.1.1 Modèle

L'objet "Simple Input Data" est décrit par le modèle suivant:

ASE DP: **ASE I/O Data**
CLASS: **Simple Input Data**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of Input Data Elements
2.1.	(m)	Attribut:	Slot Number
2.2.	(m)	Attribut:	Simple Input Data Description
2.2.1.	(m)	Attribut:	Length
2.2.2.	(m)	Attribut:	Format
2.2.3.	(m)	Attribut:	Consistency
2.2.4.	(o)	Attribut:	Name

SERVICES:

1.	(m)	OpsService:	Set input
2.	(o)	OpsService:	Read Input
3.	(m)	OpsService:	Get Input
4.	(o)	OpsService:	New Input
5.	(m)	OpsService:	Get Publisher Data
6.	(o)	OpsService:	New Publisher Data
7.	(m)	OpsService:	Global Control
8.	(o)	OpsService:	SYNCH
9.	(o)	OpsService:	SYNCH Delayed
10.	(o)	OpsService:	DX Finished
11.	(o)	OpsService:	SYNCH Event

6.2.2.2.1.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Simple Input Data est adressé de façon implicite par les services.

List of Input Data Elements
Un élément Input Data Element se compose des éléments de liste suivants:

Slot Number
Cet attribut définit le module auquel appartient l'Input Data Element.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

Simple Input Data Description
Cet attribut est constitué des attributs suivants.

Length
L'attribut Length définit le nombre d'octets de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 128

Format
Cet attribut définit le format de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 11.

Tableau 11 – Format (description de données d'entrée simples)

Valeur	Signification
0	Octet
1	Word

Consistency
Cet attribut définit la cohérence de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 12.

Tableau 12 – Consistency (description de données d'entrée simples)

Valeur	Signification
0	Cohérence sur le Format défini de l'Input Data Element
1	Cohérence sur la totalité de l'Input Data Element

Name
Cet attribut définit le nom de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Type d'attribut: Visible String(32)

6.2.2.2.1.3 Invocation de l'objet "simple input data"

Pour l'invocation de l'objet Simple Input Data, les règles suivantes s'appliquent:

— Les objets Simple Input Data ne doivent pas dépasser la longueur totale de 244 octets. Les attributs List of Input Data Elements et Length sont à positionner en conséquence.

— Un seul objet Simple Input Data doit être invoqué dans un esclave DP.

— Un objet Simple Input Data doit seulement être invoqué si aucun objet Extended Input Data (voir 6.2.2.2.3) n'a été invoqué.

6.2.2.2.2 Spécification pour la classe "Simple output data"

6.2.2.2.2.1 Modèle

L'objet "Simple Output Data" est décrit par le modèle suivant:

ASE DP:			**ASE I/O Data**
CLASS:			**Simple Output Data**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of Output Data Elements
2.1.	(m)	Attribut:	Slot Number
2.2.	(m)	Attribut:	Simple Output Data Description
2.2.1.	(m)	Attribut:	Length
2.2.2.	(m)	Attribut:	Format
2.2.3.	(m)	Attribut:	Consistency
2.2.4.	(o)	Attribut:	Name
SERVICES:			
1.	(m)	OpsService:	Set Output
2.	(o)	OpsService:	Read Output
3.	(m)	OpsService:	Get Output
4.	(o)	OpsService:	New Output
5.	(m)	OpsService:	Global Control
6.	(o)	OpsService:	SYNCH
7.	(o)	OpsService:	SYNCH Delayed
8.	(o)	OpsService:	DX Finished
9.	(o)	OpsService:	SYNCH Event

6.2.2.2.2.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Simple Output Data est adressé de façon implicite par les services.

List of Output Data Elements
Un élément Output Data Element se compose des éléments de liste suivants:

Slot Number
Cet attribut définit le module auquel appartient l'Output Data Element.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

Simple Output Data Description
Cet attribut est constitué des attributs suivants.

Length
L'attribut Length définit le nombre d'octets de l'Output Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 128

Format
Cet attribut définit le format de l'Output Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 13.

Tableau 13 – Format (données de sortie simples)

Valeur	Signification
0	Octet
1	Word

Consistency
Cet attribut définit la cohérence de l'Output Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 14.

Tableau 14 – Consistency (données de sortie simples)

Valeur	Signification
0	Cohérence sur le Format défini de l'Output Data Element
1	Cohérence sur la totalité de l'Output Data Element

Name
Cet attribut facultatif définit le nom de l'Output Data Element.

Type d'attribut: Visible String(32)

6.2.2.2.2.3 Invocation de l'objet "output data"

Pour l'invocation de l'objet Simple Output Data, les règles suivantes s'appliquent:

— Les objets Simple Output Data ne doivent pas dépasser la longueur totale de 244 octets. Les attributs List of Output Data Elements et Length sont à positionner en conséquence.

— Un seul objet Simple Output Data doit être invoqué dans un esclave DP.

— Un objet Simple Output Data doit seulement être invoqué si aucun objet Extended Output Data (voir 6.2.2.2.4) n'a été invoqué.

6.2.2.2.3 Spécification de la classe "Extended input data"

6.2.2.2.3.1 Modèle

L'objet "Extended Input Data" est décrit par le modèle suivant:

ASE DP: **ASE I/O Data**
CLASS: **Extended Input Data**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of Input Data Elements
2.1.	(m)	Attribut:	Slot Number
2.2.	(m)	Attribut:	Selector of Data Type
2.2.1.	(s)	Attribut:	Extended Input Data Description
2.2.2.	(s)	Attribut:	Simple Input Data Description
2.2.2.1.	(m)	Attribut:	Length
2.2.2.2.	(m)	Attribut:	Format
2.2.2.3.	(m)	Attribut:	Consistency
2.3.	(o)	Attribut:	Name

SERVICES:

1.	(m)	OpsService:	Set input
2.	(o)	OpsService:	Read Input
3.	(m)	OpsService:	Get Input
4.	(o)	OpsService:	New Input
5.	(m)	OpsService:	Get Publisher Data
6.	(o)	OpsService:	New Publisher Data

7. (m) OpsService: Global Control
8. (o) OpsService: SYNCH
9. (o) OpsService: SYNCH Delayed
10. (o) OpsService: DX Finished
11. (o) OpsService: SYNCH Event

6.2.2.2.3.2 Attributs

Implicit

L'attribut "Implicit" indique que l'objet Extended Input Data est adressé de façon implicite par les services.

List of Input Data Elements
Un élément Input Data Element se compose des éléments de liste suivants:

Slot Number
Cet attribut définit le module auquel appartient l'Input Data Element.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

Selector of Data Type
Cet attribut définit si ce module est décrit conformément à l'Extended Input Data Description ou conformément à la Simple Input Data Description. S'il s'agit de l'Extended Input Data Description, la cohérence est sur la totalité de l'Input Data Element.

Extended Input Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Simple Input Data Description
Cet attribut est constitué des attributs suivants.

Length
L'attribut Length définit le nombre d'octets de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 128

Format
Cet attribut définit le format de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 15.

Tableau 15 – Format (données d'entrée étendues)

Valeur	Signification
0	Octet
1	Word

Consistency
Cet attribut définit la cohérence de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 16.

Tableau 16 – Consistency (données d'entrée étendues)

Valeur	Signification
0	Cohérence sur le Format défini de l'Input Data Element
1	Cohérence sur la totalité de l'Input Data Element

Name
Cet attribut facultatif définit le nom de l'Input Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Type d'attribut: Visible String(32)

6.2.2.2.3.3 Invocation de l'objet "extended input data"

Pour l'invocation de l'objet Extended Input Data, les règles suivantes s'appliquent:

— Les objets Extended Input Data ne doivent pas dépasser la longueur totale de 244 octets. Les attributs List of Input Data Elements, Input Data Description respectivement Format et Length sont à positionner en conséquence.

— Un seul objet Extended Input Data doit être invoqué dans un esclave DP.

— Un objet Extended Input Data doit seulement être invoqué si aucun objet Simple Input Data n'a été invoqué.

6.2.2.2.4 Spécification de la classe "Extended output data"

L'objet "Extended Output Data" est décrit par le modèle suivant:

ASE DP: **ASE I/O Data**
CLASS: **Extended Output Data**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of Output Data Elements
2.1.	(m)	Attribut:	Slot Number
2.2.	(m)	Attribut:	Selector of Data Type
2.2.1.	(s)	Attribut:	Extended Output Data Description
2.2.2.	(s)	Attribut:	Simple Output Data Description
2.2.2.1.	(m)	Attribut:	Length
2.2.2.2.	(m)	Attribut:	Format
2.2.2.3.	(m)	Attribut:	Consistency
2.3.	(o)	Attribut:	Name

SERVICES:

1.	(m)	OpsService:	Set Output
2.	(o)	OpsService:	Read Output
3.	(m)	OpsService:	Get Output
4.	(o)	OpsService:	New Output
5.	(m)	OpsService:	Global Control
6.	(o)	OpsService:	SYNCH
7.	(o)	OpsService:	SYNCH Delayed
8.	(o)	OpsService:	DX Finished
9.	(o)	OpsService:	SYNCH Event

6.2.2.2.4.1 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Extended Output Data est adressé de façon implicite par les services.

List of Output Data Elements
Un élément Output Data Element se compose des éléments de liste suivants:

Slot Number
Cet attribut définit le module auquel appartient l'Output Data Element.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

Selector of Data Type
Cet attribut définit si ce module est décrit conformément à l'Extended Output Data Description ou conformément à la Simple Output Data Description. S'il s'agit de l'Extended Output Data Description, la cohérence est sur la totalité de l'Output Data Element.

Extended Output Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Simple Output Data Description
Cet attribut est constitué des attributs suivants.

Length
L'attribut Length définit le nombre d'octets de l'Output Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 128

Format
Cet attribut définit le format de l'Output Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 17.

Tableau 17 – Format (données de sortie étendues)

Valeur	Signification
0	Octet
1	Word

Consistency
Cet attribut définit la cohérence de l'Output Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Les valeurs admissibles sont montrées dans le Tableau 18.

Tableau 18 – Consistency (données de sortie étendues)

Valeur	Signification
0	Cohérence sur le Format défini de l'Output Data Element
1	Cohérence sur la totalité de l'Output Data Element

Name
Cet attribut facultatif définit le nom de l'Output Data Element. La valeur de cet attribut est déterminée par les données de configuration correspondantes. La valeur de cet attribut est déterminée par les données de configuration correspondantes.

Type d'attribut: Visible String(32)

6.2.2.2.4.2 Invocation de l'objet "extended output data"

Pour l'invocation de l'objet Extended Output Data, les règles suivantes s'appliquent:

— Les objets Extended Output Data ne doivent pas dépasser la longueur totale de 244 octets. Les attributs List of Output Data Elements, Output Data Description respectivement Format et Length sont à positionner en conséquence.

— Un seul objet Extended Output Data doit être invoqué dans un esclave DP.

— Un objet Extended Output Data doit seulement être invoqué si aucun objet Simple Output Data n'a été invoqué.

6.2.2.3 Spécification de service I/O data

6.2.2.3.1 Set input

Ce service doit être utilisé par un esclave DP pour transmettre la valeur d'un objet Simple Input Data ou un Extended Input Data vers la couche application DP. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Par ce service, le processus d'application d'esclave DP met à jour le tampon local. Conformément aux caractéristiques de transport cyclique de tampon vers un autre tampon relatives à la relation entre applications utilisée, le comportement suivant est possible.

— Les demandes de Set Input arrivent plus vite que le contenu des tampons n'est transporté sur le réseau vers le Client et les Abonnés. Dans ce cas, ce n'est pas chaque valeur de l'objet Input Data fournie par le service Set Input qui est acheminée sur le réseau. Seule la dernière valeur est transportée, les autres sont écrasées localement en écriture.

— Les demandes de Set Input sont émises plus lentement que le contenu des tampons n'est transporté sur le réseau vers le Client et les Abonnés. Dans ce cas, chaque valeur de l'objet Input Data fournie par le service Set Input est acheminée plus d'une fois sur le réseau.

— Si les demandes de Set Input sont émises en synchronisme avec le transport du contenu des tampons, chaque valeur de l'objet Input Data fournie avec le service Set Input est acheminée sur le réseau.

Le Tableau 19 montre les paramètres du service.

Tableau 19 – Set Input

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Input Data	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Input Data
Ce paramètre contient la valeur de l'objet "Input Data".

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.2.3.2 Read input

Avec ce service, n'importe quel maître DP (Classe 2) peut lire la valeur de l'objet Input Data d'un esclave DP. Ce service délivrera un résultat positif seulement si l'esclave DP est déjà dans le mode d'échange de données avec son maître DP (Classe 1) assigné. Ce service doit seulement être utilisé conjointement à l'AR MS0. Le Tableau 20 montre les paramètres du service.

Tableau 20 – Read Input

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Input Data		M
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Input Data
Ce paramètre contient la valeur demandée de l'objet "Input Data".

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre donne des informations détaillées sur l'erreur survenue.

Valeurs admissibles: DS, NA, RS, UE, NR, RE

6.2.2.3.3 Get input

Ce service doit seulement être utilisé conjointement à l'AR MS0. Il est utilisé par un maître DP (Classe 1) pour lire:

— les valeurs de tous les objets Input Data des esclaves DP assignés si les paramètres CREP et Slot Number n'ont pas été utilisés dans la demande de service Get Input,

— la valeur de l'objet Input Data d'un seul esclave DP assigné si le paramètre Slot Number a été utilisé dans la demande de service Get Input,

— la valeur d'un Input Data Element d'un seul esclave DP assigné si les paramètres CREP et Slot Number ont été utilisés dans la demande de service Get Input.

Ce service peut être acquitté négativement si le maître DP (Classe 1) fonctionne dans le mode Isochronous Mode dans les cas suivants:

— si la demande de service est faite à l'extérieur de la fenêtre autorisée d'interactions du mode Enhanced Synchronized Isochronous Mode (SE),

— si la demande de service est faite entre le service Set Output avec le paramètre Final mis à TRUE et la prochaine indication SYNCH dans le cas du mode Buffered Synchronized Isochronous Mode (SE),

— si l'esclave DP adressé est désactivé (SC).

Le Tableau 21 montre les paramètres du service.

Tableau 21 – Get Input

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
CREP	U	
Slot Number	U	
Result(+)		S
AREP		M
CREP		C
Slot Number		C
Input Data		M
Result(-)		S
AREP		M
CREP		C
Slot Number		C
Status		C

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Si ce paramètre est présent, l'objet Input Data d'un seul esclave DP est adressé.

Slot Number
Si ce paramètre est présent, un Input Data Element d'un seul esclave DP est adressé. Si ce paramètre est utilisé, le paramètre CREP doit également être présent.

Valeurs admissibles: 1 à 244

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Input Data
Ce paramètre contient la valeur sélectionnée.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre donne des informations détaillées sur l'erreur survenue.

Valeurs admissibles: SC, SE

6.2.2.3.4 New input

Le service New Input signale à l'application du maître DP (Classe 1) que de nouvelles valeurs de tous les objets Input Data d'un esclave DP qui sont relatifs à l'AR MS0 sont présentes. Ce service doit seulement être utilisé conjointement à l'AR MS0. Le Tableau 22 montre les paramètres du service.

Tableau 22 – New Input

Nom de paramètre	Ind.
Argument	M
AREP	M
CREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à l'indication de service.

AREP
Ce paramètre est l'identificateur local de l'AR qui a transmis l'indication.

CREP
Ce paramètre et l'identificateur local d'un seul esclave DP.

6.2.2.3.5 Set output

Par ce service, le processus d'application Client du maître DP (Classe 1) demande l'acheminement:

– des valeurs de tous les objets Output Data des esclaves DP assignés si les paramètres CREP et Slot Number n'ont pas été utilisés dans la demande de service Set Output,
– de la valeur de l'objet Output Data d'un seul esclave DP assigné si le paramètre Slot Number a été utilisé dans la demande de service Set Output,
– la valeur d'un Output Data Element d'un seul esclave DP assigné si les paramètres CREP et Slot Number ont été utilisés dans la demande de service Set Output.

Ce service doit seulement être utilisé conjointement à l'AR MS0.

Conformément aux caractéristiques du transport cyclique d'un tampon à l'autre, le comportement suivant est possible:

– Les demandes de Set Output arrivent plus vite que le contenu des tampons n'est transporté sur le réseau. Dans ce cas, seules les valeurs fournies avec le plus récent service Set Output sont acheminées sur le réseau.

– Les demandes de Set Output sont émises plus lentement que le contenu des tampons n'est transporté sur le réseau. Dans ce cas, les valeurs fournies avec chaque service Set Output sont acheminées plus d'une fois sur le réseau.
– Si les demandes de Set Output sont émises en synchronisme avec le transport du contenu des tampons, les valeurs fournies avec chaque service Set Output sont acheminées une seule fois sur le réseau.

Ce service peut être acquitté négativement si le maître DP (Classe 1) fonctionne dans le mode Isochronous Mode dans les cas suivants:

– si la demande de service est faite à l'extérieur de la fenêtre autorisée d'interactions du mode Enhanced Synchronized Isochronous Mode (SE),
– si la demande de service est faite entre le service Set Output avec le paramètre Final mis à TRUE et la prochaine indication SYNCH dans le cas du mode Buffered Synchronized Isochronous Mode (SE),
– si l'esclave DP adressé est désactivé (SC).

Le Tableau 23 montre les paramètres du service.

Tableau 23 – Set Output

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
CREP	U	
Slot Number	U	
Output Data	M	
Final	M	
Result(+)		S
AREP		M
CREP		C
Slot Number		C
Result(-)		S
AREP		M
CREP		C
Slot Number		C
Status		C

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Si ce paramètre est présent, l'objet Output Data d'un seul esclave DP est adressé.

Slot Number
Si ce paramètre est présent, un Output Data Element d'un seul esclave DP est adressé. Si ce paramètre est utilisé, le paramètre CREP doit également être présent.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

Output Data
Ce paramètre contient la valeur sélectionnée.

Final
Ce paramètre indique à la couche application DP de changer le tampon intermédiaire de l'Input Data et de l'Output Data des esclaves DP. La couche application DP change ces tampons après l'indication DX Finished et avant la prochaine indication SYNCH du cycle de DP isochrone dans le cas du mode Buffered Synchronized Isochronous Mode.

Les valeurs admissibles sont montrées dans le Tableau 24.

Tableau 24 – Final

Valeur	Signification
TRUE	le changement du tampon intermédiaire des Input Data et des Output Data doit être effectué
FALSE	le changement du tampon intermédiaire des Input Data et des Output Data n'est pas demandé.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre donne des informations détaillées sur l'erreur survenue.

Valeurs admissibles: SC, SE

6.2.2.3.6 Read output

Avec ce service, n'importe quel maître DP (Classe 2) peut lire de façon cyclique la valeur de l'objet Output Data d'un esclave DP. Ce service délivrera un résultat positif seulement si l'esclave DP est déjà dans le mode d'échange de données avec son maître DP (Classe 1) assigné. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Si le maître DP (Classe 1) change le mode de fonctionnement en CLEAR, l'esclave DP doit mettre la valeur de l'Output Data à une valeur correspondant à un état de sécurité par exemple: valeur zéro, dernière valeur, ou valeur par défaut). Dans ce cas, la valeur reçue avec ce service peut ne pas correspondre à la valeur réelle de l'objet Output Data.

Le Tableau 25 montre les paramètres du service.

Tableau 25 – Read Output

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Output Data		M
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Output Data
Ce paramètre contient la valeur demandée de l'objet "Output Data".

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre donne des informations détaillées sur l'erreur survenue.

Valeurs admissibles: DS, NA, RS, UE, NR, RE

6.2.2.3.7 Get output

Ce service doit être utilisé par un esclave DP pour lire dans la couche application DP la valeur pour son objet Simple Ouput Data ou Extended Output Data. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 26 montre les paramètres du service.

Tableau 26 – Get Output

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Output Data		M
Clear Flag		M
New Flag		M
Result(-)		S
AREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Output Data
Ce paramètre contient la valeur de l'objet "Output Data".

Clear Flag
Ce paramètre indique si la valeur pour l'objet Output Data est valide ou effacée.

Les valeurs admissibles sont montrées dans le Tableau 27.

Tableau 27 – Clear Flag

Valeur	Signification
0	valeur pour l'objet Output Data valide (maître DP dans l'état OPERATE)
1	valeur pour l'objet Output Data effacée (maître DP dans l'état CLEAR)

New Flag
Ce paramètre indique si une nouvelle valeur pour l'objet Output Data est reçue ou non.

Les valeurs admissibles sont montrées dans le Tableau 28.

Tableau 28 – New Flag

Valeur	Signification
0	aucune nouvelle valeur reçue
1	nouvelle valeur reçue

Result(-)

Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.2.3.8 New output

Ce service indique à l'application de l'esclave DP qu'une nouvelle valeur pour son objet Simple Ouput Data ou Extended Output Data est présente. Le Tableau 29 montre les paramètres du service.

Tableau 29 – New Output

Nom de paramètre	Ind
Argument	M
AREP	M
Clear Flag	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à l'indication de service.

AREP
Ce paramètre est l'identificateur local de l'AR qui a transmis l'indication.

Clear Flag
Ce paramètre indique si la valeur pour l'objet Output Data est valide ou effacée.

Les valeurs admissibles sont montrées dans le Tableau 30.

Tableau 30 – Clear Flag

Valeur	Signification
0	valeur pour l'objet Output Data valide (maître DP dans l'état OPERATE)
1	valeur pour l'objet Output Data effacée (maître DP dans l'état CLEAR)

6.2.2.3.9 Global control

Ce service est utilisé par le maître DP (Classe 1) pour la synchronisation des valeurs des objets Input Data (Freeze Command) et la synchronisation des valeurs des objets Output Data (Sync Command) d'un groupe ou de la totalité des esclaves DP assignés. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Si le maître DP (Classe 1) fonctionne dans le mode Isochronous Mode avec la totalité ou un groupe de ses esclaves DP assignés, le groupe 8 du paramètre Group Select du service Global Control n'est plus une valeur de groupe valide pour ce paramètre.

De plus, ce service est utilisé par le maître DP (Classe 1) pour envoyer des informations relatives à son propre mode de fonctionnement (Clear Command) à tous les esclaves DP assignés.

Le Tableau 31 montre les paramètres du service.

Tableau 31 – Global Control

Nom de paramètre	Req	Ind	Cnf
Argument	M	M(=)	
AREP	M	M(=)	
Clear Command		M	
Sync Command	M	M(=)	
Freeze Command	M	M(=)	
Group Select	M	M(=)	
Result(+)			S
AREP			M
Result(-)			S
AREP			M
Status			M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Clear Command
Ce paramètre contient la Clear Command.

Les valeurs admissibles sont montrées dans le Tableau 32.

Tableau 32 – Clear Command

Valeur	Signification
0	Maître DP (Classe 1) dans l'état OPERATE: Les valeurs pour les objets Output Data sont valides.
1	Maître DP (Classe 1) dans l'état CLEAR: Les valeurs pour les objets Output Data doivent être mises à une valeur correspondant à un état de sécurité.

Sync Command
Ce paramètre commande la Sync Operation.

Les valeurs admissibles sont montrées dans le Tableau 33.

Tableau 33 – Sync Command

Valeur	Signification
0	aucune action
1	Sync
2	Unsync

Freeze Command
Ce paramètre commande la Freeze Operation.

Les valeurs admissibles sont montrées dans le Tableau 34.

Tableau 34 – Freeze Command

Valeur	Signification
0	aucune action
1	Freeze (geler)
2	Unfreeze (dégeler)

Group Select
Ce paramètre détermine quel(s) groupe(s) d'esclaves DP assignés doi(ven)t être adressé(s). La commande de contrôle prend effet si l'opération AND (ET Logique) entre l'attribut Group Identifier de l'objet ARL de l'esclave DP et le paramètre Select donne comme résultat une valeur différente de zéro. Si le paramètre Group Select est zéro, tous les esclaves DP assignés sont adressés.

Si le maître DP (Classe 1) fonctionne dans le mode Isochronous Mode, le groupe 8 n'est pas autorisé pour ce paramètre. La demande de service doit être confirmée négativement avec une valeur de statut appropriée (GE).

Result(+)
Ce paramètre de type sélection indique que la demande de service a été émise.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre donne des informations détaillées sur l'erreur survenue.

Valeurs admissibles: DS, NO, IV, GE

6.2.2.3.10 New publisher data

Ce service indique à l'application de l'esclave DP qu'une nouvelle valeur issue d'un objet Input Data souscrit est présent. Le Tableau 35 montre les paramètres du service.

Tableau 35 – New publisher data

Nom de paramètre	Ind
Argument	M
AREP	M
CREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à l'indication de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Ce paramètre est l'identificateur local de la CR (Éditeur) qui a transmis l'indication.

6.2.2.3.11 Get publisher data

Ce service doit être utilisé par l'application d'esclave DP (Abonné) pour lire la valeur issue d'un objet Input Data souscrit particulier dans la couche application DP. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 36 montre les paramètres du service.

Tableau 36 – Get publisher data

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
CREP	M	
Result(+)		S
AREP		M
CREP		M
Data		M
New Flag		M
Result(-)		S
AREP		M
CREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Ce paramètre est l'identificateur local de la CR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Data
Ce paramètre contient la valeur issue de l'objet "Output Data" souscrit.

New Flag
Ce paramètre indique si une nouvelle valeur issue de l'objet Input Data souscrit est reçue ou non.

Les valeurs admissibles sont montrées dans le Tableau 37.

Tableau 37 – New Flag

Valeur	Signification
0	aucune nouvelle valeur reçue
1	nouvelle valeur reçue

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.2.3.12 SYNCH

Le service SYNCH indique à l'application du maître DP (Classe 1) qu'un nouveau cycle DP Isochrone a été lancé.

Le Tableau 38 montre les paramètres du service.

Tableau 38 – SYNCH

Nom de paramètre	Ind
Argument	M
AREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à l'indication de service.

AREP
Ce paramètre est l'identificateur local de l'AR qui a transmis l'indication.

6.2.2.3.13 SYNCH delayed

Le service SYNCH Delayed doit être transmis à l'application du maître DP (Classe 1) seulement si le dernier message de synchronisation n'a pas été envoyé dans le temps autorisé $maxT_{SH}$. La valeur du temps $maxT_{SH}$ est contenue dans l'entrée AR MS0 connexe de l'ARL du maître DP (Classe 1). Le service donne également la différence temporelle mesurée selon les règles suivantes:

- le message de synchronisation a été envoyé antérieurement à $|T_{DP} - maxT_{SH}|$
- le message de synchronisation a été envoyé postérieurement à $|T_{DP} - maxT_{SH}|$

Le Tableau 39 montre les paramètres du service.

Tableau 39 – SYNCH Delayed

Nom de paramètre	Ind
Argument	M
AREP	M
Shift	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à l'indication de service.

AREP
Ce paramètre est l'identificateur local de l'AR qui a transmis l'indication.

Shift
Ce paramètre contient la valeur de la différence temporelle (décalage) mesurée en unités de t_{BIT}.

Type d'attribut: Unsigned32

Valeurs admissibles: 1 à $2^{32}-1$

6.2.2.3.14 DX finished

Le service DX Finished indique à l'application du maître DP (Classe 1) que la partie cyclique (DX) du cycle DP isochrone est parachevée.

Le Tableau 40 montre les paramètres du service.

Tableau 40 – DX Finished

Nom de paramètre	Ind
Argument	M
AREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à l'indication de service.

AREP
Ce paramètre est l'identificateur local de l'AR qui a transmis l'indication.

6.2.2.3.15 SYNCH event

Le service SYNCH Event indique à l'application de l'esclave DP si

– le maître DP (Classe 1) et la couche application DP de l'esclave DP ont commencé à fonctionner dans le mode Isochronous Mode,
– qu'un nouveau cycle DP isochrone a été lancé,
– le maître DP (Classe 1) et la couche application DP de l'esclave DP ont cessé de fonctionner dans le mode Isochronous Mode.

Le Tableau 41 montre les paramètres du service.

Tableau 41 – SYNCH Event

Nom de paramètre	Ind
Argument	M
AREP	M
Status	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à l'indication de service.

AREP
Ce paramètre est l'identificateur local de l'AR qui a transmis l'indication.

Status
Ce paramètre indique le statut du mode Isochronous Mode.

Les valeurs admissibles sont montrées dans le Tableau 42.

Tableau 42 – Status

Valeur	Signification
IsoM Start	démarrage du fonctionnement en Isochronous Mode
IsoM SYNCH	démarrage d'un nouveau cycle DP isochrone
IsoM Stop	arrêt du fonctionnement en Isochronous Mode

6.2.2.4 Comportement des objets I/O data

6.2.2.4.1 Comportement général de l'objet "output data"

Pour des raisons de performance, il convient que la valeur pour l'objet Output Data soit transférée dès la réception le plus rapidement possible à l'objet. La vitesse et les gigues de cette action constituent un paramètre de performance d'un esclave DP. Ce paramètre détermine l'exactitude de la synchronisation.

Pour des esclaves DP hautement synchronisés, il convient que cette action soit toujours exécutée au même instant après la Sync Command.

6.2.2.4.2 Caractéristiques d'un système DP avec fonctionnalité de mode isochrone

6.2.2.4.2.1 Comportement de base des données d'entrée et de sortie dans le mode isochrone

Un système DP fonctionnant dans le mode Isochronous Mode fournit les caractéristiques suivantes (spécifiées à des débits de données supérieurs à 1 000 kbit/s et à des cycles de DP isochrones entre 500 μs et 32 ms):

– tous les appareils DP prenant part à la fonctionnalité Isochronous Mode synchronisent leur système d'horloge interne avec le message de synchronisation SYNCH avec le cycle de DP isochrone;

– il est nécessaire que tous les appareils DP prenant part à la fonctionnalité Isochronous Mode soient sur le même réseau (la fonctionnalité Isochronous Mode sur plusieurs réseaux n'est pas possible; des répétiteurs sont autorisés);

– la gigue de SYNCH-Event.ind est inférieure à 1μs (les gigues au sein d'un esclave DP sont prises égales à 0);

– dans chaque cycle de DP, un service acyclique de pleine longueur sur l'AR MS1 peut facultativement être inclus (demande ou réponse) et être accompli s'il y a suffisamment de temps pour traiter complètement le service avant qu'il ne faille lancer le prochain cycle de DP isochrone;

– le fonctionnement d'un maître DP (Classe 2) sur le même réseau pour des besoins de configuration ou de diagnostic est possible (il convient d'inclure dans la durée de cycle le temps nécessaire pour exécuter les services en question).

Dans un système DP avec fonctionnalité Isochronous Mode activé, des esclaves DP avec la fonctionnalité de base peuvent être utilisés conjointement à des esclaves DP synchronisés. La fonctionnalité Isochronous Mode tolère et compense les erreurs de communication comme:

– les messages de synchronisation (SYNCH) défectueux; le message de synchronisation peut être perturbé au niveau du maître DP (Classe 1) au moment de la création ou de l'acheminement vers l'esclave DP ou au moment de la reconnaissance au niveau de l'esclave DP;

– l'Isochronous Mode est capable de compenser les gigues du message de synchronisation produites par le maître DP (Classe 1) et, s'ils sont présents, par des appareils de réseau comme les répétiteurs; Cette gigue doit être inférieure à 1μs (cela inclut les gigues au sein du maître DP (Classe 1));

– l'écart du cycle de DP isochrone réel par rapport à un cycle de DP isochrone idéal, non perturbé, ne doit dépasser 0,03 %;

– Le SYNCH-Event.ind peut être manqué en raison d'un télégramme perturbé au sein du DP isochrone; un maximum de trois messages de synchronisation envoyés consécutivement hors délai ou manquants est autorisé pour un maître DP (Classe 1) fonctionnant dans le mode Isochronous Mode;

– en raison de perturbations, les messages de synchronisation peuvent être détruits au moment de l'acheminement vers l'esclave DP; le mécanisme de récupération d'horloge

dans l'esclave DP fonctionnant dans le mode Isochronous Mode tolère un maximum de cinq messages de synchronisation consécutifs non reçus.

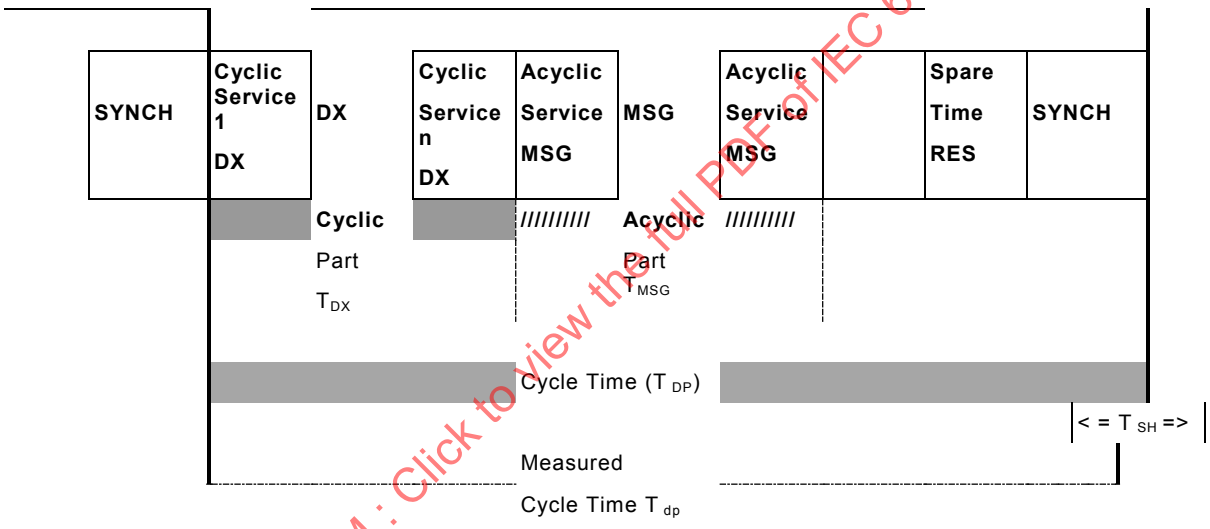
Si un système de DP doit fonctionner dans le mode Isochronous Mode, les définitions suivantes s'appliquent.

— Le maître DP (Classe 1) et les esclaves DP de correspondants doivent prendre en charge le mécanisme Fail Safe (sécurité intégrée).

— L'esclave DP doit fonctionner dans le mode Isochronous Mode seulement si les paramètres Fail Safe et IsoM Req fournis par la paramétrisation sont mis à TRUE. De plus, le nécessaire objet IsoM Parameter doit être fourni avec le service Check User Prm ou Check Ext User Prm à l'application de l'esclave DP pour positionner le paramètre Isochronous Mode de l'esclave DP.

6.2.2.4.2.2 Séquences de télégrammes d'un système DP fonctionnant dans le mode isochrone

La Figure 15 montre les séquences de systèmes DP fonctionnant dans le mode Isochronous Mode. Un cycle de DP dans le contexte isochrone est considéré comme allant du SYNCH (message de synchronisation) vers le prochain SYNCH.



Légende

Anglais	Français
SYNCH	Synchronisation
Cyclic Service 1	Service cyclique 1
DX	échange de données
Cyclic Part	Partie cyclique n
Acyclic Service 1	Service acyclique 1
Acyclic Part	Partie acyclique n
Spare time RES	Temps de réserve RES
Cycle time	Durée de cycle
Measured cycle time	Durée de cycle mesurée
MSG	Message

Figure 15 – Séquence d'un cycle de DP isochrone avec un seul maître DP (Classe 1)

Pour l'Isochronous Mode, les définitions suivantes doivent s'appliquer.

T_{DX} (Data eXchange Time, temps d'échange de données)
Le temps T_{DX} peut s'écouler pour gérer toute la communication MS0 d'un Maître (Classe 1) de DP de bus de terrain.

T_{DP} (Temps d'échange de Données et de Paramètres = durée de cycle DP isochrone)
Le temps T_{DP} peut s'écouler pour gérer toutes les parties du cycle DP isochrone. Le cycle DP isochrone est constitué des parties suivantes:

– SYNCH-Event,
– services cycliques (DX),
– services acycliques (MSG),
– temps de réserve (RES).

T_{MSG} (Temps de message)
Les temps T_{MSG} peuvent s'écouler pour gérer tous les services acycliques sur l'AR MS1 (T_{MSG1}) et l'AR MS2 (T_{MSG2}) de DP de bus de terrain ainsi que tous les autres services de DL avec Service_class = low. Ces services acycliques doivent être exécutés après les services cycliques. Pour assurer un cycle DP isochrone, cette partie doit être limitée.

T_{SH} (Shift Time, temps de décalage)
Ce temps doit être calculé avec $T_{measured\ cycle}$ — T_{DP}. Le décalage peut être dû aux raisons suivantes:

– envoi différé du message de synchronisation (SYNCH) au niveau du maître DP (Classe 1),
– calcul inexact du temps T_{DP} au niveau du maître DP (Classe 1).

La valeur maximale de ce temps ($maxT_{SH}$) est dépendante de la mise en œuvre et de la configuration. Si le temps T_{SH} dépasse une certaine limite ($maxT_{SH}$), l'application dans le maître DP (Classe 1) est informée par le service SYNCH Delayed indiquant le décalage mesuré.

Gigues
Les gigues peuvent être dues aux raisons suivantes:

(Cette définition détermine les parties requises pour une influence de PLL- Minimising Jitters (réduction au maximum des gigues), capturer des dérives et compenser les pertes de signaux. Elle doit être utilisée comme une base pour l'assurance qualité.)

– les gigues dues aux composants réseau tels que les répétiteurs ou les interfaces à fibres optiques,
– les gigues dépendant du débit d'émission (gigues < 1 t_{BIT} pour l'expéditeur, le destinataire et les composants réseau entre eux),
– temps de réaction dépendant de la mise en œuvre dans le poste d'envoi et de réception et les composants réseaux entre eux.

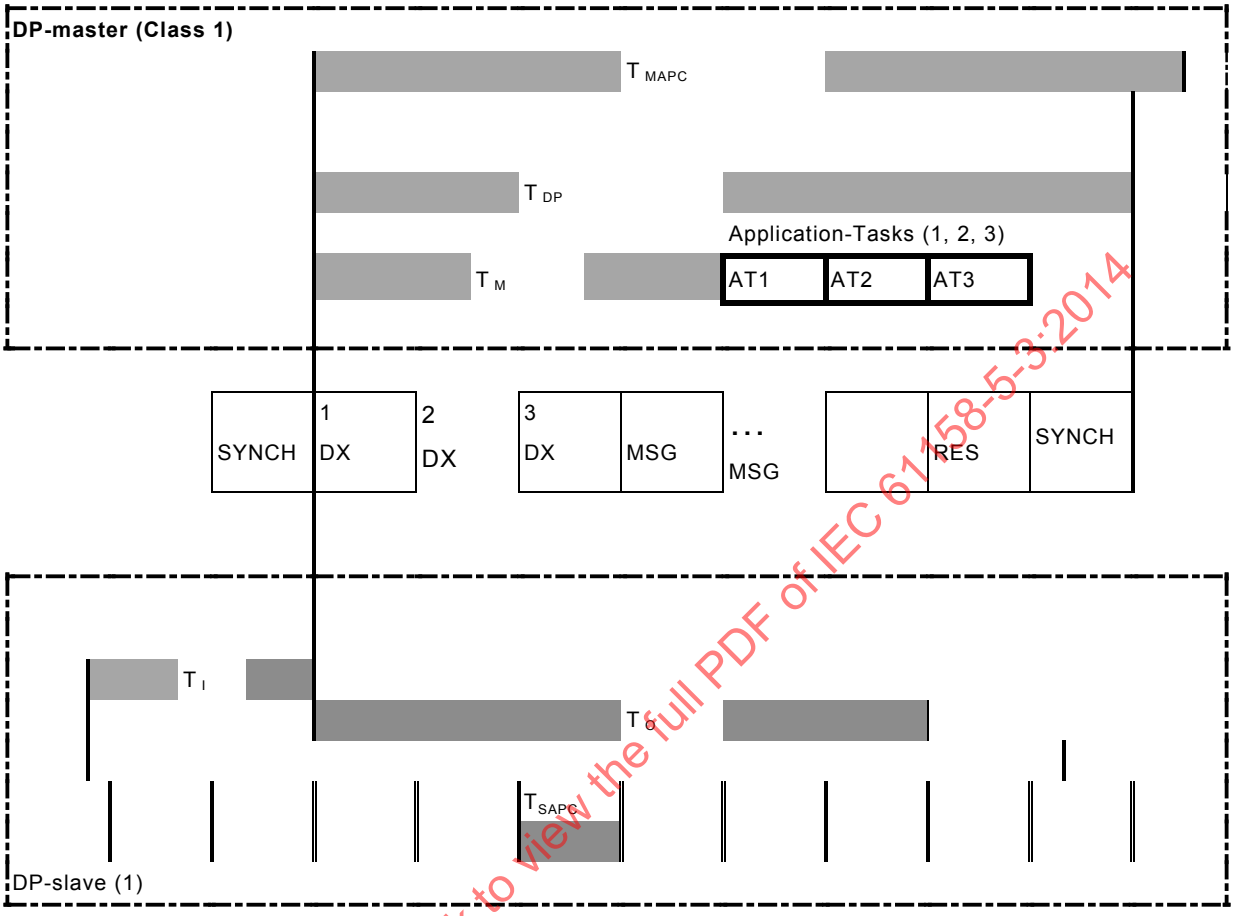
Les changements suivants de T_{DP} ne font pas partie de la définition des gigues

– écart maximal de T_{DP} dans les postes d'envoi et de réception (gigue < 0,03 %·T_{DP}),
– pertes de signalisation d'horloge,
– variations d'écart dues à la température et au vieillissement (1 ppm dans la plage d'une minute),
– SYNCH est envoyée trop tard.

RES (Temps de réserve)
Cette partie est définie comme le reste du temps entre la partie cyclique, la partie acyclique et la gestion de jeton et le prochain message de synchronisation. La durée du temps de réserve (RES) dépend des services envoyés dans le cycle de DP isochrone réel.

6.2.2.4.2.3 Relations temporelles d'un système DP fonctionnant en mode isochrone

La Figure 16 montre les relations temporelles des applications du maître DP (Classe 1) et de trois esclaves DP fonctionnant dans le mode Isochronous Mode.



Légende

Anglais	Français
DP-master (Class 1)	Maître DP (Classe 1)
Application-Tasks (1, 2, 3)	Tâches d'application (1,2,3) AT1 AT2 AT3
DP-slave (1)	Esclave DP(1)

Figure 16 – Relations temporelles supplémentaires dans un système de DP fonctionnant en mode isochrone

Pour le mode Isochronous Mode, les définitions de temps suivantes pour les relations entre applications doivent être appliquées.

T_{I_MIN} (Minimum T_I)
Ce paramètre temporel indique le temps minimal nécessaire pour récupérer et mettre à jour l'Input Data d'un esclave DP individuel.

T_I (Input Time, temps d'entrée)
Ce temps est utilisé pour déclencher l'acquisition et la mise à jour de l'objet Input Data d'un esclave DP fonctionnant en mode Isochronous Mode et peut être calculé par l'outil de configuration. Ce temps est relatif à la fin du cycle de DP isochrone.

T_{O_MIN} (Minimum T_O)
Ce paramètre temporel indique le temps minimal qui est nécessaire à la fin de la partie cyclique du cycle de DP isochrone (T_{DX}) pour récupérer et produire les Output Data donné en unités de T_{BASE_IO} d'un esclave DP individuel.

T_O (Output Time, temps de sortie)
Ce temps est utilisé pour déclencher le traitement de sortie de l'objet Output Data d'un esclave DP fonctionnant en mode Isochronous Mode et peut être calculé par l'outil de configuration. Ce temps est relatif au début du cycle de DP isochrone.

T_M (Master Time, temps de maître)
Ce temps est utilisé pour décaler le traitement des tâches d'application dans le maître DP (Classe 1) derrière la fin de la partie cyclique du cycle de DP isochrone (T_{DX}). Ce paramètre peut être fixé par l'application dans le maître DP (Classe 1) en fonction de $T_M > T_{DX}$. Si le temps T_M est mis à 0 (cas spécial), le maître DP (Classe 1) peut traiter les données d'entrée acheminées un cycle DP isochrone auparavant.

T_{MAPC}, T_{SAPC} (Master Application Cycle Time «Durée de cycle d'application de maître», Slave Application Cycle Time «Durée de cycle d'application d'esclave»)
Ces paramètres indiquent le temps dont les applications du maître DP (Classe 1) ou de l'esclave DP ont besoin pour traiter complètement une tâche d'application. Les valeurs de réglage de ces paramètres de durée de cycle peuvent être rendues spécifiques à un appareil. Il est permis d'avoir des valeurs de réglage différentes pour la durée de cycle d'application de maître (Master Application Cycle Time) et la durée de cycle d'application d'esclave (Slave Application Cycle Time) dans chaque esclave DP utilisant la base de temps commune T_{DP}. Les règles suivantes doivent s'appliquer:

— $T_{MAPC} \geq T_{DP} \geq T_{SAPC}$,

— $T_{MAPC} = n \cdot T_{DP}$ (*n* = 1 à 14; limité par le signe de vie de l'esclave).

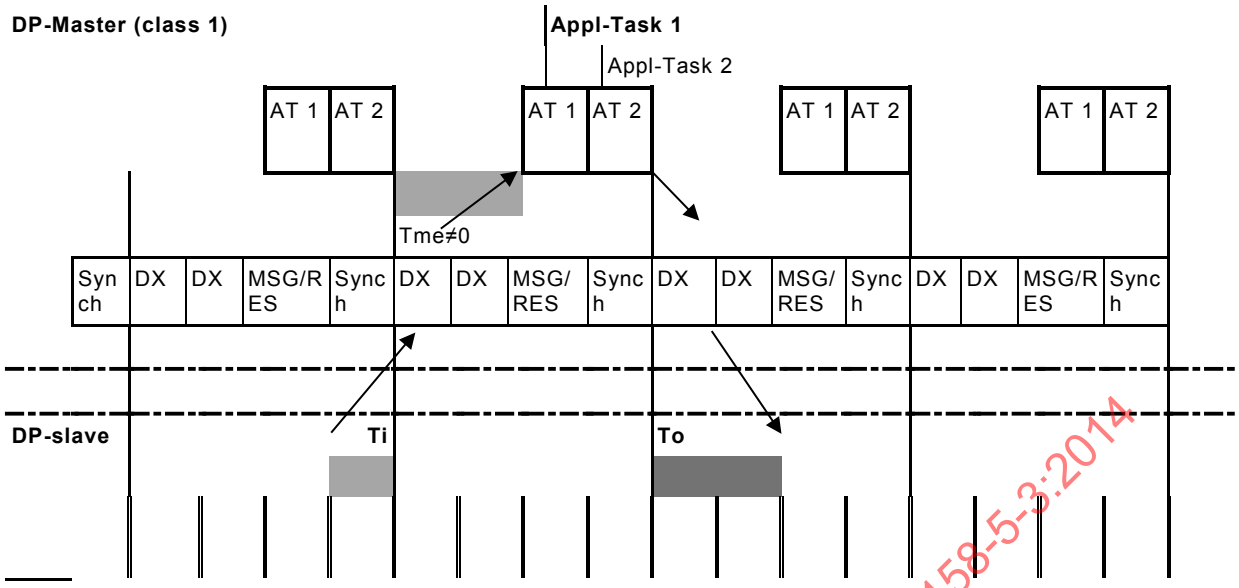
Les relations temporelles d'un système de DP, notamment celles de l'esclave DP, fonctionnant dans le mode Isochronous Mode sont seulement décrites pour le dit Esclave DP compact. Pour rendre le modèle également adapté à des esclaves DP modulaires, des aspects complémentaires sont à ajouter. Notamment, les relations temporelles doivent être étendues et un certain nombre de paramètres temporels supplémentaires peuvent être nécessaires.

Il est nécessaire de résoudre au moins les contraintes suivantes:

– au niveau d'un esclave DP modulaire, les paramètres temporels dépendent des modules branchés.
– chaque module peut également ajouter un décalage temporel (chaque type de module peut avoir un décalage temporel individuel).
– l'établissement de T_I et de T_O peut être restreint.
– l'appareil lui-même peut ajouter un décalage temporel supplémentaire en raison du temps de "transfert" des données E/S entre l'appareil d'interface et les modules.

6.2.2.4.2.4 Exemple de système DP avec cycle de DP isochrone optimisé

Dans l'exemple ci-après, les tâches d'application 1 et 2 dans le maître DP (Classe 1) peuvent être complètement traitées dans chaque cycle de DP isochrone.



Légende

Anglais	Français
DP-master (Class 1)	Maître DP (Classe 1)
Appl-Task	Tâche d'application
DP-slave	Esclave DP

Figure 17 – Système de DP avec cycle de DP isochrone optimisé

La Figure 17 montre un Système de DP où la séquence d'acquisition et d'acheminement des Input Data, le traitement de la tâche d'application et le transfert la production des Output Data sont optimisés. Cette optimisation réduit les cycles nécessaires pour accomplir une boucle de commande complète à moins de trois cycles de DP isochrones et se caractérise par les phases suivantes:

– Phase 1: acquisition et mise à jour de la valeur des Input Data dans l'esclave DP,

– Phase 2: acheminement de la valeur des Input Data vers le maître DP (Classe 1), traitement de la tâche d'application 2 avec la valeur réelle des Input Data et transfert de la nouvelle valeur pour les Output Data de la tâche d'application 2 vers le maître DP (Classe 1),

– Phase 3: acheminement de la valeur pour les Output Data vers l'esclave DP et production des nouvelles Output Data.

Ce modèle de synchronisation exige des capacités de traitement évoluées du maître DP (Classe 1) et exige une programmation optimisée dans le maître DP (Classe 1) et dans les esclaves DP.

Le temps mort dans ce modèle de modèle de synchronisation est donné par: $T_D = T_{DP} + T_I + T_O$.

Optimisation dans l'esclave DP (T_I)
Il convient que l'acquisition et la mise à jour de la valeur des Input Data de tous les esclaves DP impliqués soient faites à la fin de chaque cycle de DP isochrone le plus étroitement possible pour fournir la plupart des valeurs réelles des Input Data.

Optimisation dans l'esclave DP (T_O)
Il convient que la production de la valeur des Output Data de tous les esclaves DP impliqués soit faite à la fin de la partie cyclique (DX) de chaque cycle de DP isochrone le plus étroitement possible pour fournir les valeurs réelles pour les Output Data dès que possible.

Optimisation dans le maître DP (Classe 1) (T_M)
Le traitement des tâches d'application dans le maître DP (Classe 1) peut être décalé du paramètre T_M à la fin de la partie cyclique du cycle de DP isochrone (T_{DX}). À ce stade, les valeurs des Input Data de tous les esclaves DP impliqués ont été mises à jour dans le maître DP (Classe 1). Il est nécessaire de finir le traitement des tâches d'application dans le maître DP (Classe 1) dans le cycle de DP isochrone réel pour fournir de nouvelles valeurs pour les Output Data de tous les esclaves DP impliqués à acheminer au prochain cycle de DP isochrone.

6.2.2.4.3 Modèle d'application du maître DP (Classe 1) avec fonctionnalité de mode isochrone

6.2.2.4.3.1 Variantes du modèle

Deux modèles différents concernant l'interface entre la couche application DP et l'application du maître DP (Classe 1) lui-même sont fournis par les DP de bus de terrain pour le maître DP (Classe 1) prenant en charge le mode Isochronous Mode:

– Buffered Synchronized (tamponné synchronisé),

– Enhanced Synchronised (évolué synchronisé).

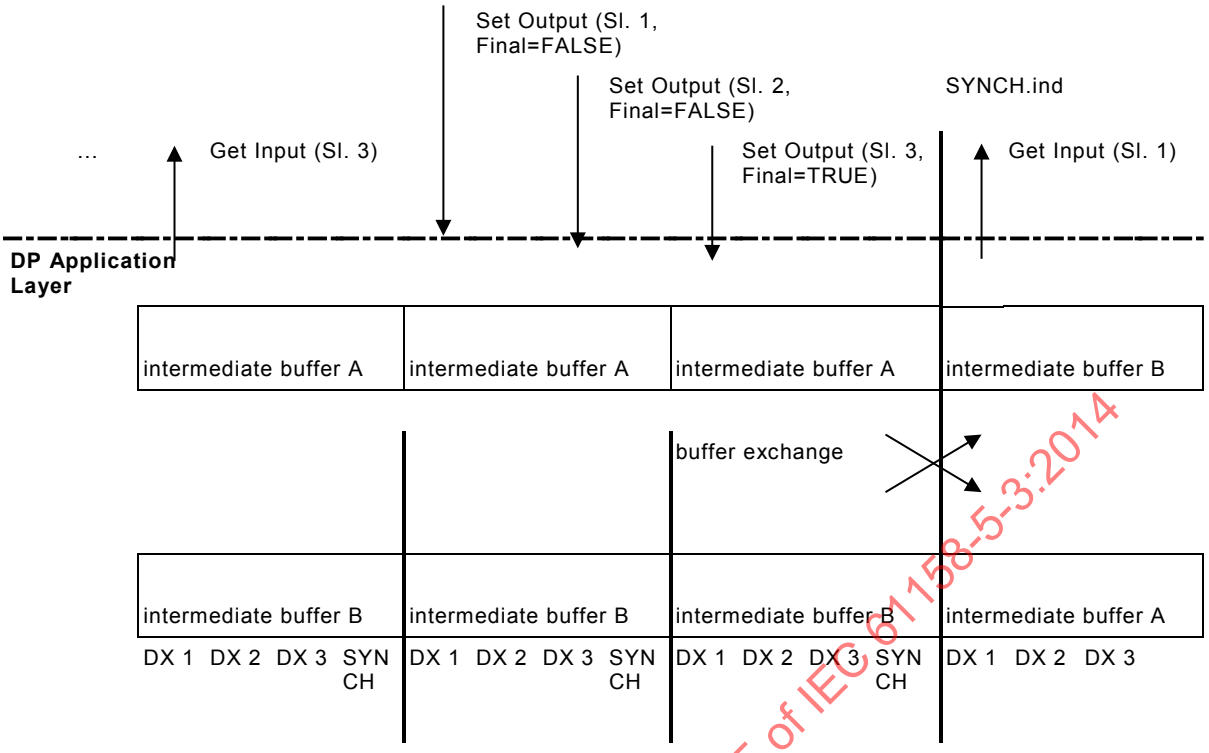
La couche application DP conjointement à la couche liaison de données fournit les nécessaires événements et messages pour synchroniser l'application du maître DP (Classe 1) avec la couche application DP pour les deux mécanismes de synchronisation.

6.2.2.4.3.2 Mode isochrone synchronisé tamponné

Le modèle Buffered Synchronized fournit un tampon intermédiaire pour stocker les Input Data issues des esclaves DP et les Output Data pour les esclaves DP au cours d'un ou plusieurs cycles de DP isochrones. Au cours de cette période, l'application peut accéder à ces tampons afin de lire et mettre à jour leur contenu, et ce, à tout moment. Ce mécanisme exige moins de performance de traitement des tâches d'application et moins de synchronisation entre l'application et la couche application DP.

En premier lieu, l'application doit lire la valeur des Input Data avec le service Get Input et doit mettre à jour la valeur des Output Data des esclaves DP avec le service Set Output. Pour informer la couche application DP de changer le tampon intermédiaire, il faut que l'application mette le paramètre Final du service Set Output à TRUE au moment de la mise à jour du dernier esclave DP. La couche application DP échange le tampon intermédiaire à la fin du cycle de DP isochrone courant. L'application doit attendre la prochaine indication SYNCH pour lancer le prochain cycle d'application. L'application est tenue d'assurer la séquence correcte d'interactions.

La Figure 18 illustre le Buffered Synchronized Isochronous Mode au niveau du maître DP (Classe 1).



Légende

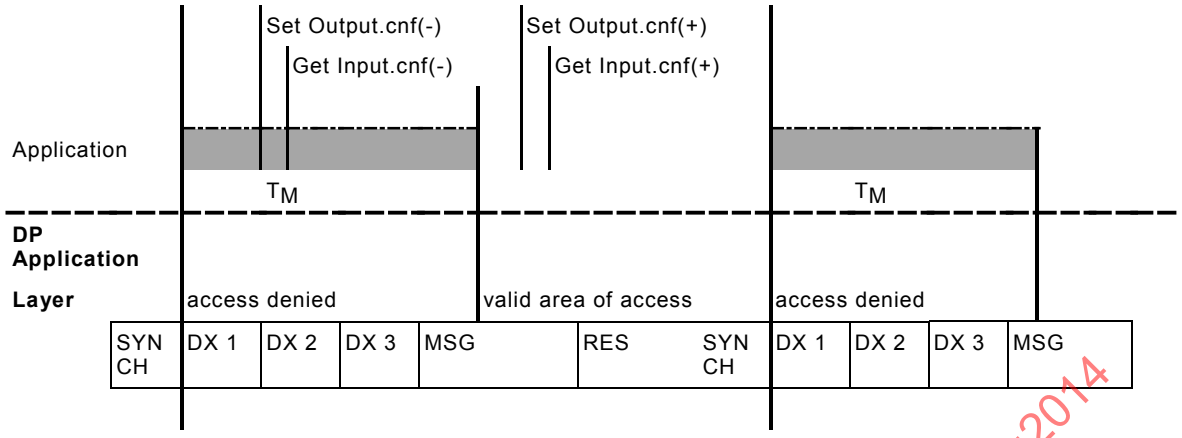
Anglais	Français
DP-Application Layer	Couche d'application DP
Intermediate buffer	Tampon intermédiaire
Buffer exchange	Échange de tampons

Figure 18 – Mode isochrone synchronisé tamponné au niveau du maître DP (Classe 1)

6.2.2.4.3.3 Mode isochrone synchronisé évolué

Le modèle Enhanced Synchronized est caractérisé par une séquence bien définie d'interactions entre l'application et la couche application DP du maître DP (Classe 1). La couche application DP donne les moyens de surveiller la séquence correcte d'interactions de l'application dans le cycle de DP isochrone et informe l'application en cas d'interactions erronées.

La Figure 19 illustre le mécanisme du mode Enhanced Synchronized Isochronous Mode.



Légende

Anglais	Français
DP-Application Layer	Couche application DP
Application	Application
access denied	accès refusé
valid area of access	zone d'accès valide

Figure 19 – Mode isochrone synchronisé évolué au niveau du maître DP (Classe 1)

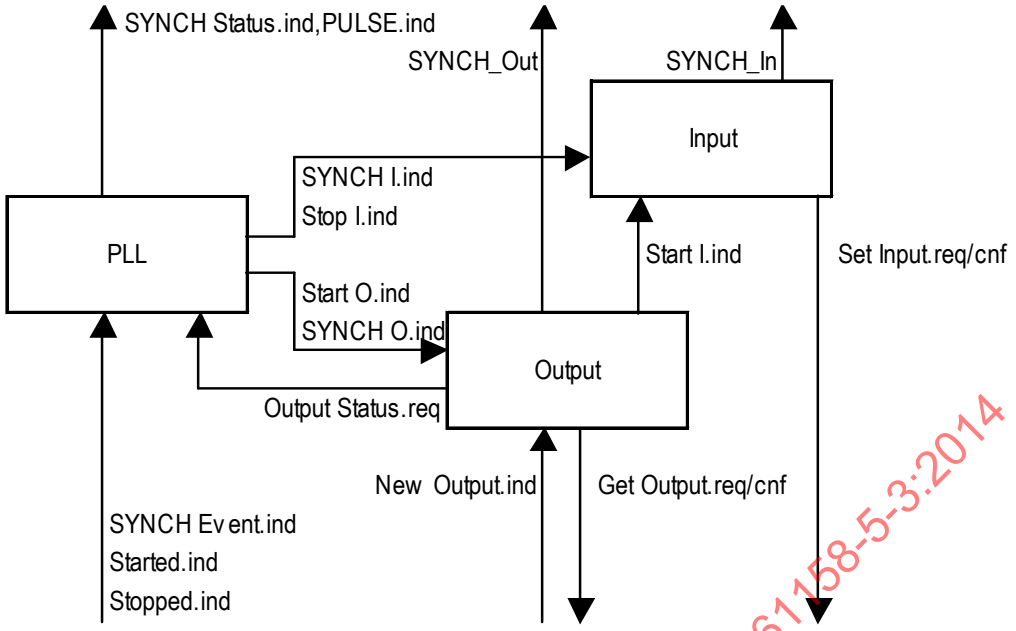
Ce mécanisme exige une performance de traitement évolué des tâches d'application et une très étroite synchronisation entre l'application et la couche application DP.

6.2.2.4.4 Description du diagramme d'états des données de sortie pour le mode isochrone

6.2.2.4.4.1 Généralités

Le comportement de l'objet Output Data est défini par le diagramme d'états des Output Data et de PLL dans le cas du mode Isochronous Mode.

Il existe plusieurs diagrammes d'états pour les objets Input et Output Data. La Figure 20 montre les interactions des diagrammes d'états de PLL, Input et Output.



Légende

Anglais	Français
PLL	Boucle à verrouillage de phase
Output	Sortie
Input	Entrée

Figure 20 – Interaction des diagrammes d'états Input, Output and PLL

Les diagrammes d'états des Output Data et de PLL représentent une partie de la fonctionnalité d'utilisateur et sont inclus pour assurer l'interopérabilité.

Les diagrammes d'états des Output Data et de PLL fournissent des événements à d'autres parties de la fonctionnalité d'utilisateur. Les diagrammes d'états des Output Data et de PLL ont des interfaces avec le diagramme d'états des Input Data dans le but de la coordination.

Les principales fonctionnalités du diagramme d'états des Output Data sont:

– Transférer les Output Data reçues vers l'application en temps utile.
– Vérifier la séquence correcte: SYNCH, Nouvelle sortie, lecture ou les Output Data à To-To_min.
– Vérifier le signe de vie des maîtres (Masters Sign of Life (MLS)).

Les principales fonctionnalités du diagramme d'états de PLL sont:

– Compensation des gigues du SYNCH-Event.ind dans l'esclave DP.
– Compensation de dérive.
– Production d'un plan d'impulsion (Pulse Scheme) pour l'application locale.

La primitive de Synchronisation (SYNCH) est acheminée par le service Global Control envoyé comme multidiffusion avec des réglages fixes des paramètres Group Select et Control Command Freeze et/ou Sync.

La synchronisation (SYNCH) indiquée à l'application de l'esclave DP peut sautiller, peut être perdue au moment de l'acheminement vers l'esclave DP ou peut être reçue trop tard parce que le maître DP (Classe 1) a envoyé le message de synchronisation en temps inopportun.

Une PLL (matériel ou logiciel intégré) doit être utilisée dans les esclaves DP pour réduire au maximum les gigues et les déphasages du message de synchronisation et pour compenser la perte de message de synchronisation. Elle peut aussi réduire la variation possible de la synchronisation à une valeur que peut tolérer l'esclave DP fonctionnant dans le mode Isochronous Mode. S la variation dépasse une certaine limite, l'application de l'esclave DP quitte le mode Isochronous Mode et passe à un état de sécurité.

La PLL dans l'esclave DP doit gérer les situations suivantes:

– Les gigues du message de synchronisation au niveau de l'esclave DP sont à lisser par la PLL. Si les gigues dépassent une certaine limite, la PLL reconnaîtra perte d'horloge.

– Les messages de synchronisation perdus en raison de perturbations de bus (la multidiffusion ne peut pas être tentée de nouveau) doivent être compensés.

– Dérive: Après le démarrage, le Maître envoie des messages de synchronisation. Un esclave DP est tenue de réajuster l'heure locale T_{DP} en T_{DP} de maîtres. À la première SYNCH, la fenêtre doit être mise à la première fenêtre, qui est au moins $T_{PLL\_W}*n1+T_{DP}*n2$ (n1>1, n2>0,000 3). Si les SYNCH suivantes s'inscrivent dans la fenêtre, le T_{DP} local est réajusté et la taille de la fenêtre doit être réduire jusqu'à ce que la fenêtre ait une taille de T_{PLL_W}. La PLL doit maintenant entrer dans le mode Operating normal. Les dérives supplémentaires dues à la variation de température ou au vieillissement sont réajustées dans le mode de fonctionnement normal. Il convient que les variations de dérive ou les gigues se traduisent par une faible variation de T_{DP}. Afin de réduire au maximum les variations de T_{DP}, les variations de dérive et les gigues doivent être limitées.

– Les déphasages dus au retard de ligne entre les différents esclaves DP peuvent être compensés, si la sortie de la PLL est décalée par rapport à l'horloge. Il faut que chaque esclave DP et le maître DP (Classe 1) aient un décalage temporel individuel pour être synchronisés à l'esclave DP qui a le retard de ligne le plus long.

6.2.2.4.4.2 Définitions des primitives

6.2.2.4.4.2.1 Primitives échangées entre l'AL et le diagramme d'états de PLL

Le Tableau 43 montre la primitive échangée entre l'utilisateur et le diagramme d'états de PLL.

Tableau 43 – Primitives émises par l'AL vers le diagramme d'états de la PLL

Nom de primitive	Source	Paramètres associés	Fonctions
SYNCH Event.ind	AL	AREP Status	
Started.ind	AL	AREP	
Stopped.ind	AL	AREP	

Les paramètres utilisés avec les primitives sont décrits dans la Spécification de service de l'objet Output Data (voir 6.2.2.3).

6.2.2.4.4.2.2 Primitives échangées entre l'utilisateur et le diagramme d'états de PLL

Le Tableau 44 montre la primitive échangée entre l'utilisateur et le diagramme d'états de PLL.

Tableau 44 – Primitives émises par l'utilisateur vers le diagramme d'états de la PLL

Nom de primitive	Source	Paramètres associés	Fonctions
SYNCH Status.ind	PLL	Status	Rapporte le changement de mode de fonctionnement, les avertissements et les erreurs.
PULSE.ind	PLL		Signal pour que le Processus d'application lance un nouveau cycle. Ce signal peut être subdivisé à partir de TDP (voir Figure 16, TSAPC).

Le paramètre Status utilisé avec la primitive SYNCH Status.ind rapporte les différents changements de mode d'opération, avertissements et erreurs Le Tableau 45 montre les valeurs admissibles de Status.

Tableau 45 – Valeurs admissibles de Status

Valeur	Source	Signification
PLL_STOP	PLL	La PLL est arrêtée, l'appareil est à resynchroniser.
PLL_SYNCH_ERR	PLL	La PLL détecte des erreurs de synchronisation et il faut la resynchroniser.
PLL_SYNCHED	PLL	La PLL est synchronisée à la SYNCH de maîtres
OUTPUT_SEQUENCE	Output	La sortie détecte des erreurs de séquence (Violation de la séquence: SYNCH > Nouvelle sortie > SYNCH_Out)
OUTPUT_RESETMLS	Output	La vérification du MLS de la sortie détecte des erreurs de MLS et resynchronise le MLS.
OUTPUT_RUNMLS	Output	La vérification du MLS de la sortie détecte le MLS en séquence.

6.2.2.4.4.2.3 Primitives échangées entre l'utilisateur et le diagramme d'états des entrées

Le Tableau 46 montre la primitive échangée entre l'utilisateur et le diagramme d'états des entrées.

Tableau 46 – Primitives émises par l'utilisateur vers le diagramme d'états des entrées

Nom de primitive	Source	Paramètres associés	Fonctions
SYNCH In	Input (Entrée)		Rassemble les valeurs des données utilisateur d'entrée.

6.2.2.4.4.2.4 Primitives échangées entre l'utilisateur et le diagramme d'états des sorties

Le Tableau 47 montre la primitive échangée entre l'utilisateur et le diagramme d'états des sorties.

Tableau 47 – Primitives émises par l'utilisateur vers le diagramme d'états des sorties

Nom de primitive	Source	Paramètres associés	Fonctions
SYNCH Out	Output (Sortie)		Transmettre le tampon des sorties aux données utilisateur de sortie.

6.2.2.4.4.2.5 Primitives échangées entre le diagramme d'états des sorties et le diagramme d'états de PLL

Le Tableau 48 et le Tableau 49 montrent la primitive échangée entre le diagramme d'états des sorties et le diagramme d'états de PLL.

Tableau 48 – Primitives émises par le diagramme d'états de PLL vers le diagramme d'états des sorties

Nom de primitive	Source	Paramètres associés	Fonctions
Start_O.ind	PLL		Démarrer le traitement des données de sortie
SYNCH_O.ind	PLL		Signal de déclenchement pour le traitement des données de sortie

Tableau 49 – Primitives émises par le diagramme d'états des sorties vers le diagramme d'états de PLL

Nom de primitive	Source	Paramètres associés	Fonctions
Output_Status.req	Output (Sortie)	Status	Rapporte le Status du traitement des Output Data

Le paramètre Status utilisé avec la primitive Output_Status.req rapporte les différents avertissements et erreurs.

6.2.2.4.4.2.6 Primitives échangées entre le diagramme d'états des entrées et le diagramme d'états de PLL

Le Tableau 50 montre la primitive échangée entre le diagramme d'états des entrées et le diagramme d'états de PLL.

Tableau 50 – Primitives émises par le diagramme d'états de PLL vers le diagramme d'états des entrées

Nom de primitive	Source	Paramètres associés	Fonctions
Stop_I.ind	PLL		Arrêter le traitement des Input Data
SYNCH_I.ind	PLL		Signal de déclenchement pour le traitement des données d'entrée

6.2.2.4.4.2.7 Primitives échangées entre le diagramme d'états des entrées et le diagramme d'états des sorties

Le Tableau 51 montre la primitive échangée entre le diagramme d'états des entrées et le diagramme d'états de sorties.

Tableau 51 – Primitives émises par le diagramme d'états des sorties vers le diagramme d'états des entrées

Nom de primitive	Source	Paramètres associés	Fonctions
Start I.ind	Output (Sortie)		Démarrer le traitement des données de sortie.

6.2.2.4.4.2.8 Primitives échangées entre l'AL et le diagramme d'états des sorties

Le Tableau 52 montre la primitive échangée entre le diagramme d'états des sorties et l'AL.

Tableau 52 – Primitives émises par le diagramme d'états des sorties vers l'AL

Nom de primitive	Source	Paramètres associés	Fonctions
Get Output.req	Output (Sortie)	AREP	

Le Tableau 53 montre la primitive échangée entre l'AL et le diagramme d'états des sorties.

Tableau 53 – Primitives émises par l'AL vers le diagramme d'états des sorties

Nom de primitive	Source	Paramètres associés	Fonctions
Get Output.cnf	AL	AREP Output Data Clear Flag New Flag	
New Output.ind	AL	AREP Clear Flag	

6.2.2.4.4.2.9 Primitives échangées entre le diagramme d'états des entrées et l'AL

Le Tableau 54 montre la primitive échangée entre le diagramme d'états des entrées et l'AL.

Tableau 54 – Primitives émises par le diagramme d'états des entrées vers l'AL

Nom de primitive	Source	Paramètres associés	Fonctions
Set Input.req	Input (Entrée)	AREP, Input Data	

Le Tableau 55 montre la primitive échangée entre le diagramme d'états des entrées et l'AL.

Tableau 55 – Primitives émises par l'AL vers le diagramme d'états des entrées

Nom de primitive	Source	Paramètres associés	Fonctions
Set Input.cnf	AL	AREP	

6.2.2.4.4.3 Définitions de valeurs locales de PLL

AREP

Cette variable locale contient l'AREP qui représente les relations entre applications.

W

Cette variable locale contient la valeur courante de la fenêtre. La fenêtre est utilisée pour détecter une primitive SYNCH Event.ind. valide.

Tdp_n

Cette variable locale contient la valeur courante de la durée de cycle réajustée. L'adaptation est nécessaire en raison des dérives d'horloges dans un maître DP (Classe 1).

count_error

Cette variable locale contient la valeur courante des erreurs pondérées en rapport avec la primitive SYNCH Event.ind au sein de la fenêtre.

RUN

Cette variable locale indique que le diagramme d'états des sorties a déjà été lancé.

cnt

Cette variable locale contient la valeur courante du compteur de segments.

seg

Cette variable locale contient la valeur courante du temps de segment (=Tdp_n/m).

TimerW

Ce temporisateur local contient la valeur courante soit du temps entre une primitive SYNCH Event.ind et le début de la prochaine fenêtre, soit le temps entre le début de la fenêtre et la primitive SYNCH Event.ind. S'il n'y a pas de SYNCH Event.ind dans la fenêtre, la fin de fenêtre remplace cet événement. La valeur du temporisateur contient le temps écoulé depuis le dernier démarrage.

TimerP

Ce temporisateur local contient la valeur courante du temps entre deux PULSE.ind. Il est utilisé pour subdiviser.

6.2.2.4.4.4 Définitions de constantes locales de PLL

E_limit
Cette constante locale limite le nombre d'erreurs de synchronisation acceptables au cours de l'intervalle de temps.

N_error
Cette constante locale est utilisée pour pondérer les erreurs de synchronisation et le succès de la synchronisation. Un facteur de 2 signifie que deux SYNCH Event.ind sont nécessaires pour compenser une erreur de synchronisation.

m
Cette constante locale est utilisée pour subdiviser Tdp_n.

6.2.2.4.4.5 Fonction locale

NOTE Certaines de ces fonctions sont aussi utilisées par le diagramme d'états des sorties et des entrées.

inc(x)
if (x=LIMIT) then x:=0 else x:=x+1

inc_LS(x)
if (x=LS_LIMIT) then x:=1 else x:=x+1

reduce_Window(W)
Réduit W tant que W>Tpllw. W ne doit pas être inférieur à Tpllw.

NOTE Cette fonction ne décrémente pas W comme une fonction linéaire, mais peut changer de manière discontinue (conservant W pendant un certain temps et le réduisant en divisant la valeur par 2).

adjust_Tdp(Tdp_n, error)
Adapter Tdp_n à une variation de Tdp. Pour prévenir les variations inutiles dues à des gigues, il convient qu'une erreur se traduise en une variation de Tdp_n inférieure à la valeur d'erreur.

6.2.2.4.4.6 Diagramme d'états de PLL

La Figure 21 montre le diagramme d'états du diagramme d'états de PLL.

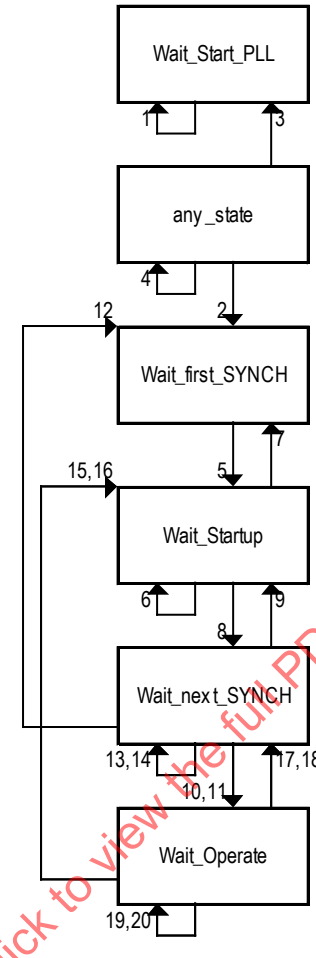


Figure 21 – Diagramme d'états de PLL

6.2.2.4.4.7 Table d'états de PLL

Le Tableau 56 montre les états d'un diagramme d'états de PLL.

Tableau 56 – Table d'états de PLL

#	État courant	Événement / Condition => Action	État suivant
1	Wait_Start_PLL	SYNCH Event.ind(AREP,Status) => ignore	Wait_Start_PLL
2	any_state	Started.ind(AREP) => W:= first_Window Tdp_n:= Tdp RUN:= 0 count_error:= 0	Wait_first_SYNCH
3	any_state	Stopped.ind(AREP) => Status:= PLL_STOP SYNCH Status.ind(Status) Stop_I.ind	Wait_Start_PLL
4	any_state	Output Status.req(Status) => SYNCH Status.ind(Status) if (Status = OUTPUT_RESETMLS) Stop_I.ind endif	any_state
5	Wait_first_SYNCH	SYNCH Event.ind(AREP,Status) => start TimerW(Tdp_n-W)	Wait_Startup
6	Wait_Startup	SYNCH Event.ind(AREP,Status) /count_error < e_limit => count_error:= count_error + 1	Wait_Startup
7	Wait_Startup	SYNCH Event.ind(AREP,Status) /count_error = e_limit => Status:= PLL_SYNCH_ERR W:= first_Window Tdp_n:= Tdp count_error:= 0 start TimerW(Tdp_n-W) SYNCH Status.ind(Status) Stop_I.ind	Wait_first_SYNCH
8	Wait_Startup	TimerW expired => start TimerW(2*W)	Wait_next_SYNCH
9	Wait_next_SYNCH	SYNCH Event.ind(AREP,Status) /W > Tpllw => Tdp_n:= adjust_Tdp(Tdp_n ,(TimerW-W)) W:=reduce_Window(W) start TimerW(Tdp_n-TimerW)	Wait_Startup
10	Wait_next_SYNCH	SYNCH Event.ind(AREP,Status) /W = Tpllw => Tdp_n:= adjust_Tdp(Tdp_n ,(TimerW-W)) count_error:= 0start TimerW(Tdp_n-TimerW)	Wait_Operate
11	Wait_next_SYNCH	TimerW expired /count_error < e_limit => count_error:= count_error + 1 start TimerW(Tdp-2*W)	Wait_Operate
12	Wait_next_SYNCH	TimerW expired /count_error = e_limit => Status:= PLL_SYNCH_ERR W:= first_Window Tdp_n:= Tdp RUN:= 0 count_error:= 0 start TimerW(Tdp_n-W) SYNCH Status.ind(Status) Stop_I.ind	Wait_first_SYNCH

#	État courant	Événement / Condition => Action	État suivant
13	Wait_next_SYNCH	TimerP expired /cnt < m => cnt:= cnt+1 Start TimerP(Seg) PULSE.ind	Wait_next_SYNCH
14	Wait_next_SYNCH	TimerP expired /cnt = m => PULSE.ind	Wait_next_SYNCH
15	Wait_Operate	SYNCH Event.ind(AREP,Status) /count_error < e_limit => count_error:= count_error + 1	Wait_Startup
16	Wait_Operate	SYNCH Event.ind(AREP,Status) /count_error = e_limit => Status:= PLL_SYNCH_ERR W:= first_Window Tdp_n:= Tdp RUN:= 0 count_error:= 0 start TimerW(Tdp_n-W) SYNCH Status.ind(Status) Stop_I.ind	Wait_Startup
17	Wait_Operate	TimerW expired /RUN = 1 => cnt:=1 Seg = (Tdp_n)/m Start TimerP(W+Delay) Start TimerW(2*W) SYNCH_I.ind SYNCH_O.ind	Wait_next_SYNCH
18	Wait_Operate	TimerW expired /RUN = 0 => Status:= PLL_SYNCHED cnt:=1 Seg = (Tdp_n)/m Start TimerP(W+Delay) RUN:= 1 Start TimerW(2*W) Start_O.ind SYNCH_I.ind SYNCH_O.ind SYNCH_Status.ind(Status)	Wait_next_SYNCH
19	Wait_Operate	TimerP expired /cnt < m => cnt:= cnt+1 Start TimerP(Seg) PULSE.ind	Wait_Operate
20	Wait_Operate	TimerP expired /cnt = m => PULSE.ind	Wait_Operate

6.2.2.4.4.8 Définitions de valeurs locales de la sortie

AREP

Cette variable locale contient l'AREP qui représente les relations entre applications.

wrong_output_update

Cette variable locale contient la valeur courante des erreurs pondérées relatives aux mises à jour des sorties.

MLS_start_counter

Cette variable locale contient la valeur courante du MLS correct au démarrage.

MLS_error_counter

Cette variable locale contient la valeur courante des erreurs de MLS.

TMAPC_counter

Cette variable locale contient la valeur courante du nombre de cycles depuis la dernière mise à jour de MLS.

old_MLS

Cette variable locale contient la valeur du dernier MLS.

MLS_State

Cette variable locale contient la valeur sous-état pour le calcul de MLS.

Les valeurs possibles sont:

- Wait_first_MLS Start-up of MLS
- Wait_next_MLS Wait for next following Start-up MLS
- Wait_MLS Wait for MLS during normal operation

O_Buffer

Cette variable locale contient les valeurs des données de sortie reçues. Elle contient une variable facultative MLS qui contient le signe de vie. Si cette variable est absente, l'état “Check_MLS” n'a aucune Action.

TimerTo

Ce temporisateur local contient la valeur courante du temps entre un SYNCH Event.ind et le transfert de l'Output_Data vers le Processus d'application.

Timer(To-Tomin)

Ce temporisateur local contient la valeur courante du temps entre un SYNCH Event.ind et le transfert de l'Output_Data vers le diagramme d'états des sorties.

6.2.2.4.4.9 Définitions de constantes locales

limit
Cette constante locale limite le nombre d'erreurs acceptables de mise à jour de sorties au cours de l'intervalle de temps.

n
Cette constante locale est utilisée pour pondérer les mises à jour de sorties erronées et celles qui ont réussi. Un facteur de 2 signifie que deux mises à jour réussies sont nécessaires pour compenser une mise à jour erronée.

MLS_factor
Cette constante locale est utilisée pour définir le nombre de cycles entre des incréments de MLS.

n_TMAPC
Cette constante locale est utilisée pour limiter les erreurs de MLS.

n_MLS
Cette constante locale est utilisée pour le démarrage pour avoir un nombre raisonnable de MLS corrects.

6.2.2.4.4.10 Fonction locale

inc(x)
if (x=LIMIT) then x:=0 else x:=x+1

6.2.2.4.4.11 Diagramme d'états des OUTPUT

La Figure 22 montre le diagramme d'états du diagramme d'états de l'OUTPUT.

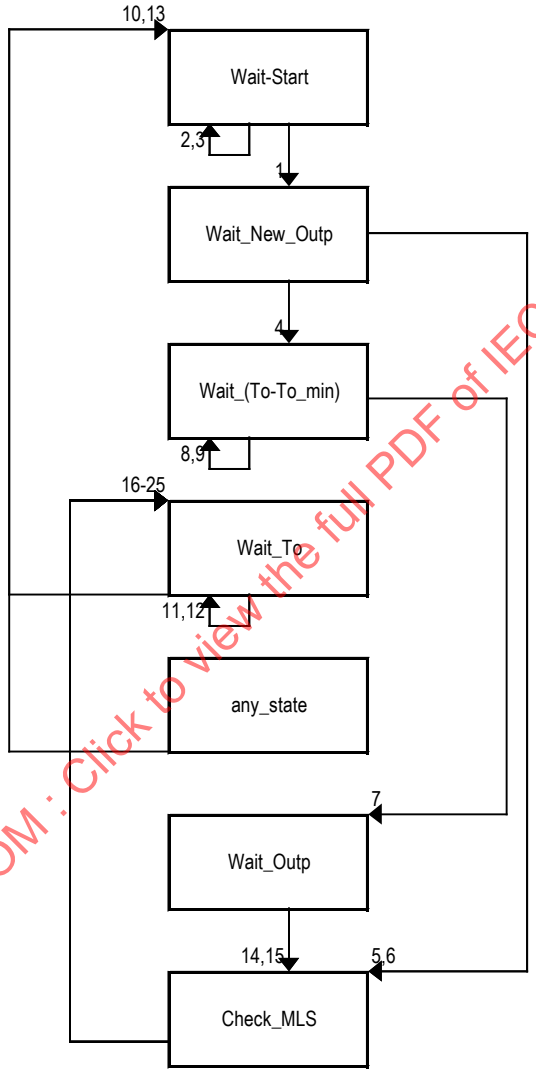


Figure 22 – Diagramme d'états de l'OUTPUT

6.2.2.4.4.12 Table d'états de l'OUTPUT

Le Tableau 57 montre les états d'un diagramme d'états de l'OUTPUT.

Tableau 57 – Table d'états de l'OUTPUT

#	État courant	Événement / Condition => Action	État suivant
1	Wait-Start	SYNCH_O.ind => Start Timer(To-To_min) Start TimerTo	Wait_New_Outp
2	Wait-Start	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Wait-Start
3	Wait-Start	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd = limit => Status:= OUTPUT_SEQUENCE Output_Status.req (Status)	Wait-Start
4	Wait_New_Outp	New_Output.ind (AREP, Clear_Flag) => if (wrong_output_upd>0) wrong_output_upd= wrong_output_upd-1	Wait_(To-To_min)
5	Wait_New_Outp	Timer(To-To_min) expired /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Check_MLS
6	Wait_New_Outp	Timer(To-To_min) expired /wrong_output_upd = limit => Status:= OUTPUT_SEQUENCE Output_Status.req (Status)	Check_MLS
7	Wait_(To-To_min)	Timer(To-To_min) expired => Get_Output.req(AREP)	Wait_Outp
8	Wait_(To-To_min)	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Wait_(To-To_min)
9	Wait_(To-To_min)	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd = limit => Status:= OUTPUT_SEQUENCE Output_Status.req (Status)	Wait_(To-To_min)
10	Wait_To	TimerTo expired => SYNCH_Out	Wait-Start
11	Wait_To	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd < limit => wrong_output_upd= wrong_output_upd+n	Wait_To
12	Wait_To	New_Output.ind (AREP, Clear_Flag) /wrong_output_upd = limit => Status:= OUTPUT_SEQUENCE Output_Status.req (Status)	Wait_To
13	any_state	Start_O.ind=>MLS_State:= Wait_first_MLSO_Buffer:=Nil(with O_Buffer.MLS:=0)old_MLS:=0MLS_error_counter:=0TMAPC_counter:=0MLS_start_counter:=0	Wait-Start
14	Wait_Outp	Get_Output.cnf (AREP, Output_Data, New_Flag, Clear_Flag) /New_Flag && !Clear_Flag => O_Buffer:= Output_Data	Check_MLS
15	Wait_Outp	Get_Output.cnf (AREP, Output_Data, New_Flag, Clear_Flag) /!(New_Flag && !Clear_Flag) =>	Check_MLS

#	État courant	Événement / Condition => Action	État suivant
16	Check_MLS	/MLS_State = Wait_first_MLS => MLS_State:= Wait_next_MLS Status:=OUTPUT_RESETMLS old_MLS:=O_Buffer.MLS Output_Status.req (Status)	Wait_To
17	Check_MLS	/MLS_State = Wait_next_MLS TMAPC_counter < MLS_factor => MLS_State:= Wait_next_MLS inc(TMAPC_counter)	Wait_To
18	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter < n_TMAPC => MLS_State:= Wait_next_MLS inc_LS(old_MLS) TMAPC_counter:=0 MLS_error_counter:= MLS_error_counter + n_error	Wait_To
19	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter >= n_TMAPC => MLS_State:= Wait_first_MLS Status:=OUTPUT_RESETMLS TMAPC_counter:=0 MLS_error_counter:=0 old_MLS:=0 MLS_start_counter:=0 Output_Status.req (Status)	Wait_To
20	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS = inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_start_counter < n_MLS => MLS_State:= Wait_next_MLS TMAPC_counter:=0 if (MLS_error_counter >0) MLS_error_counter:= MLS_error_counter –1 old_MLS:=MLS inc(MLS_start_counter)	Wait_To
21	Check_MLS	/MLS_State = Wait_next_MLS O_Buffer.MLS = inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_start_counter = n_MLS => MLS_State:= Wait_MLS Status:=OUTPUT_RUNMLS TMAPC_counter:=0 MLS_error_counter:=0 old_MLS:=O_Buffer.MLS MLS_start_counter:=0 Output_Status.req (Status) Start_I.ind	Wait_To
22	Check_MLS	/MLS_State = Wait_MLS TMAPC_counter < MLS_factor => MLS_State:= Wait_MLS inc(TMAPC_counter)	Wait_To

#	État courant	Événement / Condition => Action	État suivant
23	Check_MLS	/MLS_State = Wait_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter < n_TMAPC => MLS_State:= Wait_MLS inc_LS(old_MLS) TMAPC_counter:=0 MLS_error_counter:= MLS_error_counter + n_error	Wait_To
24	Check_MLS	/MLS_State = Wait_MLS O_Buffer.MLS <> inc_LS(old_MLS) & TMAPC_counter = MLS_factor & MLS_error_counter >= n_TMAPC => MLS_State:= Wait_first_MLS Status:=OUTPUT_RESETMLS TMAPC_counter:=0 MLS_error_counter:=0 old_MLS:=0 Output_Status.req (Status)	Wait_To
25	Check_MLS	/MLS_State = Wait_MLS O_Buffer.MLS = inc_LS(old_MLS) & TMAPC_counter = MLS_factor => MLS_State:= Wait_MLS TMAPC_counter:=0 if (MLS_error_counter >0) MLS_error_counter:= MLS_error_counter –1 old_MLS:=O_Buffer.MLS	Wait_To

6.2.2.4.5 Comportement de l'objet données d'entrée

Pour des raisons de performance, il convient que la valeur de l'objet Input Data soit transférée à la suite d'un changement le plus rapidement possible au tampon de MS0 en vue de l'émission. La vitesse et les gigues de cette action constituent un paramètre de performance d'un esclave DP. Ce paramètre détermine l'exactitude du gel.

Pour des esclaves DP hautement synchronisés, il convient que cette action soit toujours exécutée au même instant avant la Freeze Command. Cela exige qu'une Freeze Command soit envoyée de façon cyclique.

6.2.2.4.6 Description du diagramme d'états des données d'entrée pour le mode isochrone

6.2.2.4.6.1 Définitions de valeurs locales de l'entrée

AREP

Cette variable locale contient l'AREP qui représente les relations entre applications.

SLS

Cette variable locale contient la valeur courante du signe de vie d'esclave.

I_Buffer

Cette variable locale contient les valeurs de l'Input Data récupérés du Processus d'application. Elle contient une variable facultative SLS qui contient le signe de vie. Si SLS est absent, la mise à jour de SLS est une opération NULL.

TimerTix

Ce temporisateur local contient la valeur courante du temps entre un SYNCH Event.ind et le transfert de l'Input_Data vers l'AL.

Timer(TDP-Ti)

Ce temporisateur local contient la valeur courante du temps entre un SYNCH Event.ind et le transfert de l'Input_Data vers le diagramme d'états des entrées (Input).

6.2.2.4.6.2 Diagramme d'états de l'INPUT

La Figure 23 montre le diagramme d'états du diagramme d'états de l'INPUT.

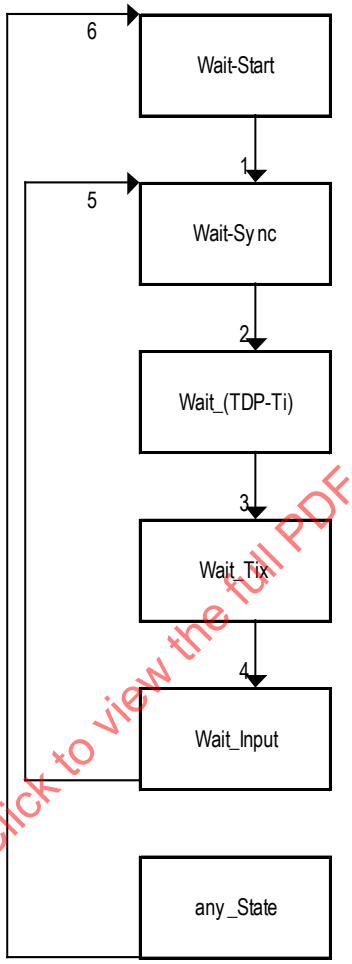


Figure 23 – Diagramme d'états de l'INPUT

6.2.2.4.6.3 Table d'états de l'INPUT

Le Tableau 58 montre les états d'un diagramme d'états de l'INPUT.

Tableau 58 – Table d'états de l'INPUT

#	État courant	Événement / Condition => Action	État suivant
1	Wait-Start	**Start_I.ind** =>	Wait-Sync
2	Wait-Sync	**SYNCH_I.ind** => Start Timer(TDP-Ti) Start TimerTix	Wait_(TDP-Ti)
3	Wait_(TDP-Ti)	**Timer(TDP-Ti) expired** => SYNCH_In	Wait_Tix
4	Wait_Tix	**TimerTix expired** => Input_Data:= I_Buffer Input_Data.SLS:= inc_LS(SLS) Set_Input.req (AREP,Input_Data)	Wait_Input
5	Wait_Input	**Set_Input.cnf(AREP)** =>	Wait-Sync
6	any_State	**Stop_I.ind** => SLS:= 0	Wait_Start

6.2.3 ASE Diagnosis

6.2.3.1 Vue d'ensemble

Dans l'environnement de la couche application DP, l'ASE Diagnosis d'un esclave DP contient des objets Diagnosis qui sont utilisés pour rapporter le statut du Processus d'application par défaut (Default Application Process) de l'esclave DP et des modules qui sont partie intégrante de la configuration. L'ASE Diagnosis définit les attributs des objets Diagnosis et fournit un ensemble de services locaux utilisés pour établir et récupérer leurs valeurs. Les objets Diagnosis suivants sont définis:

— Objet Device Related Diagnosis,

— Objet Status,

— Objet Module Status Data,

— Objet DXB-Link Status,

— Objet Identifier Related Diagnosis,

— Objet Channel Related Diagnosis.

L'application d'un esclave DP peut fournir la valeur d'un ou plusieurs objets Diagnosis à l'AR MS0. L'esclave DP doit fournir seulement les valeurs des objets Diagnosis dont les valeurs ont changé depuis leur dernière fourniture. L'esclave DP peut indiquer (avec le paramètre Ext Diag Overflow) que les valeurs de tous les objets Diagnosis avec une valeur modifiée ne pourraient pas être fournies avec une seule demande de service Set Slave Diag. Le maître DP (Classe 1) doit indiquer (avec le paramètre Ext Diag Overflow) dans la confirmation de Get Slave Diag que toutes les valeurs reçues des objets Diagnosis provenant de l'esclave DP ne pourraient pas être fournies à l'application.

L'esclave DP doit indiquer (avec le paramètre Ext Diag Flag) qu'un événement de diagnostic pertinent s'est produit et conduit à un changement d'au moins un objet Diagnosis (par exemple, il convient de le positionner si un élément de l'objet Identifier Related Diagnosis est mis à la valeur 1). Ce fanion entraîne une entrée dans l'attribut System Diagnosis de l'objet Master Diag du maître DP (Classe 1).

L'esclave DP peut indiquer la version réelle de l'esclave DP avec le paramètre Revision Number fourni par le service Set Slave Diag. Le Revision Number peut être utilisé par un maître DP (Classe 2) pour trouver le fichier GSD connexe de l'esclave DP.

Le modèle de Diagnosis utilise le modèle d'accès Client/Serveur. Le modèle Diagnosis repose sur une relation entre tampons. Ainsi, l'application Serveur met à jour (établit) les valeurs des objets Diagnosis dans son tampon local en vue de l'acheminement. L'application Client peut récupérer de son tampon local les valeurs des objets Diagnosis.

Le maître DP (Classe 1) communique simultanément, par l'AR MS0, avec tous les objets Diagnosis des esclaves DP assignés. De plus, l'application Client peut communiquer avec les objets Diagnosis d'un seul esclave DP.

Le maître DP (Classe 2) communique, par l'AR MS0, avec les objets Diagnosis d'un seul esclave DP.

En outre, des informations de diagnostic supplémentaires (attributs pour l'objet AR MS0) sont fournies par la couche application DP comme paramètres du service Get Slave Diag et du service Read Slave Diag.

La description formelle des divers objets Diagnosis est proposée ci-après, suivie d'une description de ses services.

6.2.3.2 Spécification de la classe Diagnosis

6.2.3.2.1 Spécification de la classe Device related diagnosis

6.2.3.2.1.1 Modèle

L'objet "Device Related Diagnosis" est décrit par le modèle suivant:

ASE DP: **ASE Diagnosis**
CLASS: **Device Related Diagnosis**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Attribut-clé: Implicit
2. (m) Attribut: Device Related Data Description

SERVICES:
1. (m) OpsService: Set Slave Diag
2. (m) OpsService: Get Slave Diag
3. (o) OpsService: Read Slave Diag
4. (o) OpsService: New Slave Diag

6.2.3.2.1.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Device Related Diagnosis est adressé de façon implicite par les services.

Device Related Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1. La longueur maximale de ce Device Related Data ne doit pas dépasser 62 octets.

6.2.3.2.1.3 Invocation de l'objet diagnostic relatif à un appareil

Pour l'invocation de l'objet Device Related Diagnosis, les règles suivantes doivent être prises en considération:

– Les objets Device Related Diagnosis doivent être utilisés seulement si l'attribut de l'appareil DPV1 = FALSE.
– Les objets Device Related Diagnosis ne doivent pas dépasser la longueur totale de 62 octets. L'attribut Device Related Data Description est à positionner en conséquence.

Plus d'un objet Device Related Diagnosis peuvent être invoqués dans un esclave DP.

6.2.3.2.2 Spécification de la classe Identifier related diagnosis

6.2.3.2.2.1 Modèle

Cette classe spécifie l'état des modules qui sont partie intégrante de la configuration. Chaque bit est relatif à un Module (Slot). Un élément mis à FALSE signifie qu'aucun diagnostic du modèle connexe n'existe ou qu'aucun module n'est placé. Un bit mis à TRUE indique une défaillance de module. Si une défaillance de module existe, les valeurs de l'objet Input Data et/ou de l'objet Output Data de ce module sont mises à la valeur par défaut par l'esclave DP.

L'objet "Identifier Related Diagnosis" est décrit par le modèle suivant:

ASE DP:			**ASE Diagnosis**
CLASS:			**Identifier Related Diagnosis**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of Identifier Status
2.1.	(m)	Attribut:	Identifier Status
SERVICES:			
1.	(m)	OpsService:	Set Slave Diag
2.	(m)	OpsService:	Get Slave Diag
3.	(o)	OpsService:	Read Slave Diag
4.	(o)	OpsService:	New Slave Diag

6.2.3.2.2.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Identifier Related Diagnosis est adressé de façon implicite par les services.

List of Identifier Status
Cet attribut est constitué des éléments de liste suivants:

Identifier Status
Les éléments de cet attribut de liste indiquent l'état de chaque module. Par exemple, si le premier module a une erreur, le premier élément de l'attribut List of Identifier Status a une valeur 1.

Les valeurs admissibles sont montrées dans le Tableau 59.

Tableau 59 – Identifier status

Valeur	Signification
0	Module en fonctionnement.
1	Module pas en fonctionnement.

6.2.3.2.2.3 Invocation de l'objet diagnostic relatif à un appareil

Pour l'invocation de l'objet Identifier Related Diagnosis, les règles suivantes doivent être prises en considération:

– Un seul objet Identifier Related Diagnosis doit être invoqué dans un esclave DP.

– En fonction du nombre maximal de modules, qui sont partie intégrante de la configuration, un objet Identifier Related Diagnosis ne doit pas dépasser la longueur totale de 32 octets.

6.2.3.2.3 Spécification de la classe Channel related diagnosis

6.2.3.2.3.1 Modèle

L'objet "Channel Related Diagnosis" est décrit par le modèle suivant:

ASE DP: **ASE Diagnosis**
CLASS: **Channel Related Diagnosis**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attributs-clés:	Identifier
2.	(m)	Attribut:	Channel Type
3.	(m)	Attribut:	IO Type
4.	(m)	Attribut:	Error Type

SERVICES:

1.	(m)	OpsService:	Set Slave Diag
2.	(m)	OpsService:	Get Slave Diag
3.	(o)	OpsService:	Read Slave Diag
4.	(o)	OpsService:	New Slave Diag

6.2.3.2.3.2 Attributs

Identifier
Cet attribut-clé est constitué de l'Identifier Number et du Channel Number pour indiquer la source de l'objet Channel Related Diagnosis.

Valeurs admissibles: 0 à 63, 0 à 63

Channel Type
Cet attribut indique la taille de voie à laquelle l'objet Channel Related Diagnosis se rapporte.

Les valeurs admissibles sont montrées dans le Tableau 60.

Tableau 60 – Channel type

Valeur	Signification
0	unspecific (non spécifique)
1	1 bit
2	2 bit
3	4 bit
4	octet
5	word (mot)
6	2 words (deux mots)

IO Type
Cet attribut indique l'IO Type de la voie à laquelle l'objet Channel Related Diagnosis se rapporte.

Les valeurs admissibles sont montrées dans le Tableau 61.

Tableau 61 – IO type

Valeur	Signification
1	Voie d'entrée
2	Voie de sortie
3	Voie bidirectionnelle

Error Type
Cet attribut indique le type d'erreur de l'objet Channel Related Diagnosis.

Les valeurs admissibles sont montrées dans la CEI 61158-6-3, 5.3.17.

6.2.3.2.3.3 Invocation de l'objet diagnostic relatif à une voie

Pour l'invocation de l'objet Channel Related Diagnosis, la règle suivante est à prendre en considération:

– Plusieurs objets Channel Related Diagnosis peuvent être invoqués dans un esclave DP.

6.2.3.2.4 Spécification de la classe Status

6.2.3.2.4.1 Modèle

Le modèle de statut permet le transfert de messages de statut de l'esclave DP vers le maître DP (Classe 1). Le modèle de statut est basé sur les baies.

Un objet Status est décrit par le modèle suivant:

ASE DP: **ASE Diagnosis**
CLASS: **Status**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Attributs-clés: Identifier
2. (m) Attribut: Status Specifier
3. (o) Attribut: Status Data Description

SERVICES:
1. (m) OpsService: Set Slave Diag
2. (m) OpsService: Get Slave Diag
3. (o) OpsService: Read Slave Diag
4. (o) OpsService: New Slave Diag

6.2.3.2.4.2 Attributs

Identifier
Cet attribut-clé est constitué de Slot Number et de Status Type pour indiquer le numéro de baie et le type de statut.

Valeurs admissibles pour Slot Number: 0 à 254

Les valeurs admissibles pour Status Type sont montrées dans le Tableau 62.

Tableau 62 – Status type

Valeur	Signification
1	Message de statut
32-126	Spécifique à un fabricant

Status Specifier
Cet attribut spécifie des informations de statut complémentaires.

Les valeurs admissibles sont montrées dans le Tableau 63.

Tableau 63 – Status specifier

Valeur	Signification
0	aucune différenciation supplémentaire
1	le statut apparaît.
2	le statut disparaît

Status Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

6.2.3.2.4.3 Invocation de l'objet statut

Pour l'invocation de l'objet Status, les règles suivantes s'appliquent:

– Les objets Status doivent être utilisés seulement si l'attribut de l'appareil DPV1 = TRUE
– Plusieurs objets Status peuvent être invoqués dans un esclave DP.
– Un objet Status ne doit pas dépasser la longueur totale de 59 octets. L'attribut Status Data Description est à positionner en conséquence.

6.2.3.2.5 Spécification de la classe Module status data

6.2.3.2.5.1 Modèle

Cette classe spécifie le statut des modules qui sont partie intégrante de la configuration.

L'objet "Module Status Data" est décrit par le modèle suivant:

ASE DP: **ASE Diagnosis**
CLASS: **Module Status Data**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	Status Specifier
3.	(m)	Attribut:	List of Module Status
3.1.	(m)	Attribut:	Module Status

SERVICES:

1.	(m)	OpsService:	Set Slave Diag
2.	(m)	OpsService:	Get Slave Diag
3.	(o)	OpsService:	Read Slave Diag
4.	(o)	OpsService:	New Slave Diag

6.2.3.2.5.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Module Status Data est adressé de façon implicite par les services.

Status Specifier
Cet attribut spécifie des informations de statut complémentaires.

Les valeurs admissibles sont montrées dans le Tableau 64.

Tableau 64 – Status specifier

Valeur	Signification
0	aucune différenciation supplémentaire
1	le statut apparaît.
2	le statut disparaît

List of Module Status
Cet attribut est constitué des éléments de liste suivants:

Module Status
Cet attribut spécifie des informations de statut complémentaires pour chaque module. Le premier élément de la liste contient le statut du premier module (Slot Number 1).

Les valeurs admissibles sont montrées dans le Tableau 65.

Tableau 65 – Module status

Valeur	Signification
0	données valides
1	données non valides - erreur de signal: les données du module correspondant sont non valides en raison d'une erreur (entrées non accessibles ou les sorties ne peuvent pas être établies, par exemple, en raison d'un court-circuit ou d'une surcharge)
2	données non valides - mauvais module: les données du module correspondant ne sont pas valides, en raison d'un mauvais module en place
3	données non valides - aucun module: les données du module correspondant ne sont pas valides, car aucun module n'est en place

6.2.3.2.5.3 Invocation de l'objet "module status data"

Pour l'invocation de l'objet Module Status Data, les règles suivantes s'appliquent:

– L'objet Module Status Data doit être utilisé seulement si l'attribut de l'appareil DPV1 = TRUE.
– Un seul objet Module Status Data doit être invoqué dans un esclave DP.
– Un objet Module Status Data ne doit pas dépasser la longueur totale de 59 octets.

6.2.3.2.6 Spécification de la classe DXB-link status

6.2.3.2.6.1 Modèle

Cette classe spécifie le statut des liaisons DXB (relations Éditeur/Abonné) contenues dans la DXB-Linktable au niveau de l'Abonné.

L'objet DXB-Link Status doit être utilisé conformément aux règles suivantes:

– L'objet DXB-Link Status doit être émis la première fois avec les informations Diagnosis si l'esclave DP a reçu une DXB-Linktable correcte et si l'esclave DP est entré dans l'état de transfert de données.
– l'objet DXB-Link Status doit être omis si la DXB-Linktable était marquée comme non valide (par exemple, réception de nouvelles données de paramétrisation sans une DXB-Linktable)

L'objet "DXB-Link Status" est décrit par le modèle suivant:

ASE DP: **ASE Diagnosis**
CLASS: **DXB-Link Status**
CLASS ID: non utilisé

PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	Status Specifier
3.	(m)	Attribut:	List of DXB-Link Status
3.1.	(m)	Attribut:	Publisher Address
3.2.	(m)	Attribut:	Link Status
3.3.	(m)	Attribut:	Link Error

SERVICES:

1.	(m)	OpsService:	Set Slave Diag
2.	(m)	OpsService:	Get Slave Diag
3.	(o)	OpsService:	Read Slave Diag
4.	(o)	OpsService:	New Slave Diag

6.2.3.2.6.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet DXB-Link Status est adressé de façon implicite par les services.

Status Specifier
Cet attribut spécifie des informations de statut complémentaires.

Les valeurs admissibles sont montrées dans le Tableau 66.

Tableau 66 – Status specifier

Valeur	Signification
0	aucune différenciation supplémentaire
1	le statut apparaît.
2	le statut disparaît.

List of DXB-Link Status
Cet attribut est constitué des éléments de liste suivants:

Publisher Address
Cet attribut indique l'adresse de DL de l'Éditeur correspondant.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 125

Link Status
Cet attribut indique les états de la DXB-Link connexe.

Les valeurs admissibles sont montrées dans le Tableau 67.

Tableau 67 – Link status

Valeur	Signification
TRUE	DXB-Link active; au moins un abonnement de l'Éditeur connexe a été accompli au cours de la dernière commande de chien garde (T_{WD})
FALSE	DXB-Link non active; aucun abonnement de l'Éditeur connexe n'a été accompli au cours de la dernière commande de chien garde (T_{WD}); l'Abonné peut indiquer des informations détaillées concernant la cause avec l'attribut Link Error

Link Error
Cet attribut indique des informations détaillées concernant la défaillance de la DXB-Link connexe.

Les valeurs admissibles sont montrées dans le Tableau 68.

Tableau 68 – Link error

Valeur	Signification
0	unspecific (non spécifique)
1	longueur erronée; la longueur de l'objet Input Data de l'Éditeur n'est pas alignée sur la valeur de l'attribut Input Data Length Publisher de l'entrée DXB-Link Entry connexe

6.2.3.2.6.3 Invocation de l'objet "DXB-link status"

Pour l'invocation de l'objet DXB-link status, les règles suivantes s'appliquent:

– L'objet DXB-Link status doit être utilisé seulement si l'attribut de l'appareil DPV1 est mis à TRUE.

– Un seul objet DXB-Link Status doit être invoqué dans un esclave DP.

– Un objet DXB-Link Status ne doit pas dépasser la longueur totale de 59 octets. L'attribut DXB-Link Status est à positionner en conséquence.

6.2.3.3 Spécification du service Diagnosis

6.2.3.3.1 Set slave diag

Ce service doit être utilisé par un esclave DP pour transmettre les valeurs des objets Diagnosis à la couche application DP. Les informations de diagnostic globales émises ne doivent pas dépasser la longueur totale de 238 octets. Par ce service, le processus d'application Serveur de l'esclave DP met à jour le tampon local. Conformément aux caractéristiques de l'AR MS0 d'un tampon à l'autre, le comportement suivant est possible:

– Les demandes de Set Slave Diag arrivent plus vite que le contenu des tampons n'est transporté sur le réseau. Dans ce cas, ce n'est pas chaque valeur des objets Diagnosis fournie par le service Set Slave Diag qui est acheminée sur le réseau. Seules les dernière valeurs sont transportées, les autres sont écrasées localement en écriture.

– Les demandes de Set Slave Diag émises plus lentement que le contenu des tampons sont transportées sur le réseau. Dans ce cas, les valeurs des objets Diagnosis fournies par le service Set Slave Diag sont acheminées plus d'une fois sur le réseau.

– Si les demandes de Set Slave Diag sont émises en synchronisme avec le transport du contenu des tampons, chaque valeur des objets Diagnosis fournie avec le service Set Slave Diag est acheminée sur le réseau.

Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 69 montre les paramètres du service.

Tableau 69 – Set Slave Diag

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Ext Diag Overflow	M	
Ext Diag Flag	M	
Ext Diag Data	C	
Device Related Diagnosis Data	U	
List of Identifier Status	U	
Identifier Status	M	
List of Channel Related Diagnosis Data	U	
Identifier Number	M	
Channel Number	M	
Channel Type	M	
IO Type	M	
Error Type	M	
List of Status Data	U	
Slot Number	M	
Status Type	M	
Status Specifier	M	
Status Data	M	
Module Status Data	U	
Status Specifier	M	
List of Module Status	M	
Module Status	M	
DXB-Link Status	U	
Status Specifier	M	
List of DXB-Link Status	M	
Publisher Address	M	
Link Status	M	
Link Error	M	
Revision Number	U	
Result		S
AREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Ext Diag Flag
Ce paramètre indique qu'un événement de diagnostic pertinent s'est produit et conduit à un changement d'au moins un objet Diagnosis (par exemple, il convient de le positionner si un élément de l'objet Identifier Related Diagnosis est mis à la valeur 1).

Les valeurs admissibles sont montrées dans le Tableau 70.

Tableau 70 – Ext Diag Flag

Valeur	Signification
TRUE	L'appareil est à l'état de diagnostic. Ainsi, il existe au moins un élément de diagnostic important (tel qu'un court-circuit). Il faut qu'une entrée soit présente dans les objets Diagnosis (objets Device Related Diagnosis, objet Identifier Related Diagnosis, objets Channel Related Diagnosis, objets Status, objet Module Status).
FALSE	L'appareil n'est PAS à l'état de diagnostic. Ainsi, aucune entrée de diagnostic importante n'existe. Si la valeur d'attribut est FALSE, un message de statut (par exemple, tension d'alimentation OK) peut être présent dans les données de diagnostic (Diagnosis Data) ou il n'existe pas de Diagnosis Data.

Ext Diag Overflow
Ce paramètre Boolean indique que les valeurs de tous objets Diagnosis avec une valeur modifiée ne pouvaient pas être fournies avec cette demande de service Set Slave Diag service.

En cas de condition de débordement, il est permis de tronquer seulement aux limites de l'objet Diagnosis.

Valeurs admissibles: TRUE, FALSE

Ext Diag Data
Ce paramètre est constitué des éléments suivants.

Device Related Diagnosis Data
Ce paramètre contient la valeur de l'objet Device Related Diagnosis (ce paramètre doit être utilisé seulement si DPV1 = FALSE).

List of Identifier Status
Ce paramètre est constitué des éléments de liste suivants:

Identifier Status
Ce paramètre contient la valeur de l'attribut Identifier Status de l'objet Identifier Related Diagnosis.

List of Channel Related Diagnosis Data
Ce paramètre est constitué des éléments de liste suivants:

Identifier Number
Ce paramètre contient la valeur de l'attribut Identifier Number d'un objet Channel Related Diagnosis.

Channel Number
Ce paramètre contient la valeur de l'attribut Channel Number d'un objet Channel Related Diagnosis.

Channel Type
Ce paramètre contient la valeur de l'attribut Channel Type d'un objet Channel Related Diagnosis.

IO Type
Ce paramètre contient la valeur de l'attribut IO Type d'un objet Channel Related Diagnosis.

Error Type
Ce paramètre contient la valeur d'un objet Channel Related Diagnosis.

List of Status Data
Ce paramètre est constitué des éléments de liste suivants:

Slot Number
Ce paramètre contient la valeur de l'attribut Slot Number d'un objet Status.

Status Type
Ce paramètre contient la valeur de l'attribut Status Type d'un objet Status.

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier d'un objet Status.

Status Data
Ce paramètre contient la valeur d'un objet Status.

Module Status Data
Ce paramètre est constitué des sous-paramètres suivants:

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier de l'objet Module Status Data.

List of Module Status
Ce paramètre est constitué des éléments de liste suivants:

Module Status
Ce paramètre contient la valeur de l'attribut Module Status de l'objet Module Status Data.

DXB-Link Status
Ce paramètre est constitué des sous-paramètres suivants:

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier de l'objet DXB-Link Status.

List of DXB-Link Status
Ce paramètre est constitué des éléments de liste suivants:

Publisher Address
Ce paramètre contient la valeur de l'attribut Publisher Address de l'objet DXB-Link Status.

Link Status
Ce paramètre contient la valeur de l'attribut Link Status de l'objet DXB-Link Status.

Link Error
Ce paramètre contient la valeur de l'attribut Link Error de l'objet DXB-Link Status.

Revision Number
Ce paramètre contient la valeur réelle de l'objet "Revision Number".

Result
Ce paramètre indique que la demande de service a réussi.

6.2.3.3.2 Get slave diag

Ce service peut être utilisé par un maître DP (Classe 1) pour lire les valeurs précédemment reçues des objets Diagnosis de l'un ou de tous les esclaves DP assignés qui sont relatifs à l'AR MS0. De plus, le service Get Slave Diag donne des informations de diagnostic fournies par la couche application DP de l'esclave DP et du maître DP (Classe 1). Ce service doit seulement être utilisé conjointement à l'AR MS0. Le Tableau 71 montre les paramètres du service.

Tableau 71 – Get Slave Diag

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
CREP	U	
Result(+)		S
AREP		M
CREP		C
List of Diag Data		M
Prm Req		M
Prm Fault		M
Cfg Fault		M
Not Supported		M
Station Not Ready		M
Stat Diag		M
WD On		M
Sync Mode		M
Freeze Mode		M
Master Add		M
Ident Number		M
Deactivated		M
Station Not Existent		M
Invalid Slave Response		M
Master Lock		M
Ext Diag Overflow		M
Ext Diag Flag		M
Ext Diag Data		C
Device Related Diagnosis Data		U
List of Identifier Status		U
Identifier Status		M
List of Channel Related Diagnosis Data		U
Identifier Number		M
Channel Number		M
Channel Type		M
IO Type		M
Error Type		M
List of Status Data		U
Slot Number		M
Status Type		M
Status Specifier		M
Status Data		M
Module Status Data		U
Status Specifier		M
List of Module Status		M

Nom de paramètre	Req	Cnf
Module Status		M
DXB-Link Status		U
Status Specifier		M
List of DXB-Link Status		M
Publisher Address		M
Link Status		M
Link Error		M
Red Diagnosis		U
Red Status		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Red Status2		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Prm Command Ack		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialied		M
TOH started		M
Red Status2		M
Role		M

Nom de paramètre	Req	Cnf
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialised		M
TOH started		M
Revision Number		U
Result(-)		S
AREP		M
CREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Si ce paramètre est présent, les objets Diagnosis d'un seul esclave DP sont lus.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

List of Diag Data
Ce paramètre est constitué des éléments de liste suivants:

Prm Req
Ce paramètre contient la valeur de l'attribut Prm Req de l'entrée ARL MS0 d'un esclave DP.

Prm Fault
Ce paramètre contient la valeur de l'attribut Prm Fault de l'entrée ARL MS0 d'un esclave DP.

Cfg Fault
Ce paramètre contient la valeur de l'attribut Cfg Fault de l'entrée ARL MS0 d'un esclave DP.

Not Supported
Ce paramètre contient la valeur de l'attribut Not Supported de l'entrée ARL MS0 d'un esclave DP.

Station Not Ready
Ce paramètre contient la valeur de l'attribut Station Not Ready de l'entrée ARL MS0 d'un esclave DP.

Stat Diag
Ce paramètre contient la valeur de l'attribut Stat Diag de l'entrée ARL MS0 d'un esclave DP.

WD On
Ce paramètre contient la valeur de l'attribut WD On de l'entrée ARL MS0 d'un esclave DP.

Sync Mode
Ce paramètre contient la valeur de l'attribut Sync Mode de l'entrée ARL MS0 d'un esclave DP.

Freeze Mode
Ce paramètre contient la valeur de l'attribut Freeze Mode de l'entrée ARL MS0 d'un esclave DP.

Master Add
Ce paramètre contient la valeur de l'attribut Master Add de l'entrée ARL MS0 d'un esclave DP.

Ident Number
Ce paramètre contient la valeur de l'attribut Ident Number de l'entrée ARL MS0 d'un esclave DP.

Deactivated
Ce paramètre contient la valeur de l'attribut Deactivated de l'entrée CRL connexe de l'esclave DP correspondant du maître DP (Classe 1).

Station Non Existent
Ce paramètre contient la valeur de l'attribut Station Non Existent de l'entrée CRL connexe de l'esclave DP correspondant du maître DP (Classe 1).

Invalid Slave Response
Ce paramètre contient la valeur de l'attribut Invalid Slave Response de l'entrée CRL connexe de l'esclave DP correspondant du maître DP (Classe 1).

Master Lock
Ce paramètre indique si, oui ou non, l'esclave DP est verrouillé par un maître DP. Si le paramètre Master Lock est TRUE, l'esclave DP est verrouillé par un autre maître DP. Si le paramètre Master Lock est False, l'esclave DP n'est pas verrouillé par un autre maître DP (Classe 1).

Ext Diag Overflow
Ce paramètre Boolean contient la valeur du paramètre Ext Diag Overflow du service Set Slave Diag si toutes les valeurs délivrées des objets Diagnosis ne sont pas contenues dans cette confirmation de service. Ce paramètre contient la valeur de l'attribut Ext Diag Overflow de l'entrée de CRL connexe de l'esclave DP correspondant du maître DP (Classe 1) si toutes les valeurs délivrées des objets Diagnosis ont contenues dans cette confirmation de service en raison de problèmes de stockage.

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

NOTE En cas de condition de débordement, il est permis de tronquer seulement aux limites de l'objet Diagnosis.

Ext Diag Flag
Ce paramètre contient la valeur du paramètre Ext Diag Flag du service Set Slave Diag d'un esclave DP.

Ext Diag Data
Ce paramètre est constitué des sous-paramètres suivants:

Device Related Diagnosis Data
Ce paramètre contient la valeur de l'objet Device Related Diagnosis d'un esclave DP.

List of Identifier Status
Ce paramètre est constitué des éléments de liste suivants:

Identifier Status
Ce paramètre contient la valeur de l'attribut Identifier Status de l'objet Identifier Related Diagnosis d'un esclave DP.

List of Channel Related Diagnosis Data
Ce paramètre est constitué des éléments de liste suivants:

Identifier Number
Ce paramètre contient la valeur de l'attribut Identifier Number d'un objet Channel Related Diagnosis d'un esclave DP.

Channel Number
Ce paramètre contient la valeur de l'attribut Channel Number d'un objet Channel Related Diagnosis d'un esclave DP.

Channel Type
Ce paramètre contient la valeur de l'attribut Channel Type d'un objet Channel Related Diagnosis d'un esclave DP.

IO Type
Ce paramètre contient la valeur de l'attribut IO Type d'un objet Channel Related Diagnosis d'un esclave DP.

Error Type
Ce paramètre contient la valeur de l'attribut Error Type d'un objet Channel Related Diagnosis d'un esclave DP.

List of Status Data
Ce paramètre est constitué des éléments de liste suivants:

Slot Number
Ce paramètre contient la valeur de l'attribut Slot Number d'un objet Status d'un esclave DP.

Status Type
Ce paramètre contient la valeur de l'attribut Status Type d'un objet Status d'un esclave DP.

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier d'un objet Status d'un esclave DP.

Status Data
Ce paramètre contient la valeur d'un objet Status d'un esclave DP.

Module Status Data
Ce paramètre est constitué des sous-paramètres suivants:

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier de l'objet Module Status Data d'un esclave DP.

List of Module Status
Ce paramètre est constitué des éléments de liste suivants:

Module Status
Ce paramètre contient la valeur de l'attribut Module Status de l'objet Module Status Data d'un esclave DP.

DXB-Link Status
Ce paramètre est constitué des sous-paramètres suivants:

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier de l'objet DXB-Link Status d'un esclave DP.

List of DXB-Link Status
Ce paramètre est constitué des éléments de liste suivants:

Publisher Address
Ce paramètre contient la valeur de l'attribut Publisher Address de l'objet DXB-Link Status d'un esclave DP.

Link Status
Ce paramètre contient la valeur de l'attribut Link Status de l'objet DXB-Link Status d'un esclave DP.

Link Error
Ce paramètre contient la valeur de l'attribut Link Error de l'objet DXB-Link Status d'un esclave DP.

Red Diagnosis
Ce paramètre est constitué des sous-paramètres suivants:

Red Status
Ce paramètre de type sélection indique que le paramètre Red Diagnosis contient des informations de diagnostic des interfaces de communications redondantes d'un Esclave DP redondant Ce paramètre est constitué des sous-paramètres suivants:

Seq Number
Ce paramètre contient le Sequence Number nécessaire pour l'identification univoque du Red Status reçu.

Valeurs admissibles: 0 à 31

Fonction
Ce paramètre est constitué des sous-paramètres suivants:

Primary Request
Ce paramètre Boolean mis à TRUE indique qu'une Primary Request (Demande primaire) a été exécutée à cette interface de communication. Ce paramètre mis à FALSE indique qu'aucune Primary Request (Demande primaire) n'a été exécutée à cette interface de communication.

Valeurs admissibles: TRUE, FALSE

MS1 Command
Ce paramètre indique qu'une MS1 Command comme Start MS1, Stop MS1, Reset MS1 ou aucune MS1 Command n'a été exécutée à cette interface de communication.

Valeurs admissibles: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
Ce paramètre Boolean mis à TRUE indique qu'une demande Check Properties a été exécutée à cette interface de communication. Ce paramètre mis à FALSE indique qu'aucune demande Check Properties n'a été exécutée à cette 'interface de communication.

Valeurs admissibles: TRUE, FALSE

Master State Clear
Ce paramètre Boolean mis à TRUE indique que le maître DP (Classe 1) connexe a mis l'interface de communication dans l'état Clear. Ce paramètre mis à FALSE indique que le maître DP (Classe 1) connexe n'a envoyé aucune commande Master State Clear.

Valeurs admissibles: TRUE, FALSE

Red Status1
Ce paramètre contient le statut de redondance de la première interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que cette interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que cette interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect (défaut de matériel) de cette interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état Fail Safe (sécurité intégrée).

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de cette interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Red Status2
Ce paramètre contient le statut de redondance de la seconde (redondante) interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que l'autre interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que l'autre interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect de l'autre interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état Fail Safe.

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de l'autre interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Prm Command Ack
Ce paramètre de type sélection indique que le paramètre Red Diagnosis contient l'acquittement d'une Prm Command reçue précédemment. Ce paramètre est constitué des sous-paramètres suivants:

Seq Number
Ce paramètre contient le Sequence Number nécessaire pour l'identification univoque du Prm Command Ack reçu.

Valeurs admissibles: 0 à 31

Fonction
Ce paramètre est constitué des sous-paramètres suivants:

Primary Request
Ce paramètre Boolean mis à TRUE indique qu'une Primary Request (Demande primaire) a été exécutée à cette interface de communication. Ce paramètre mis à FALSE indique qu'aucune Primary Request (Demande primaire) n'a été exécutée à cette interface de communication.

Valeurs admissibles: TRUE, FALSE

MS1 Command
Ce paramètre indique qu'une MS1 Command comme Start MS1, Stop MS1, Reset MS1 ou aucune MS1 Command n'a été exécutée à cette interface de communication.

Valeurs admissibles: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
Ce paramètre Boolean mis à TRUE indique qu'une demande Check Properties a été exécutée à cette interface de communication. Ce

paramètre mis à FALSE indique qu'aucune demande Check Properties n'a été exécutée à cette 'interface de communication.

Valeurs admissibles: TRUE, FALSE

Master State Clear
Ce paramètre Boolean mis à TRUE indique que le maître DP (Classe 1) connexe a mis l'interface de communication dans l'état Clear. Ce paramètre mis à FALSE indique que le maître DP (Classe 1) connexe n'a envoyé aucune commande Master State Clear.

Valeurs admissibles: TRUE, FALSE

Red Status1
Ce paramètre contient le statut de redondance de la première interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que cette interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que cette interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect (défaut de matériel) de cette interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état Fail Safe (sécurité intégrée).

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de cette interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Red Status2
Ce paramètre contient le statut de redondance de la seconde (redondante) interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que l'autre interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que l'autre interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect de l'autre interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état Fail Safe.

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de l'autre interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Revision Number
Ce paramètre contient la valeur réelle de l'objet "Revision Number".

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.3.3.3 Read slave diag

Ce service peut être utilisé par un maître DP (Classe 2) pour lire de façon acyclique les valeurs des objets Diagnosis d'un esclave DP par le biais d'une AR MS0. De plus, le service Read Slave Diag donne des informations de diagnostic fournies par la couche application DP de l'esclave DP et du maître DP (Classe 2).

Le Tableau 72 montre les paramètres du service.

Tableau 72 – Read Slave Diag

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Diag Data		M
Prm Req		M
Prm Fault		M
Cfg Fault		M
Not Supported		M
Station Not Ready		M
Stat Diag		M
WD On		M
Sync Mode		M
Freeze Mode		M
Master Add		M
Ident Number		M
Ext Diag Overflow		M
Ext Diag Flag		M
Ext Diag Data		C
Device Related Diagnosis Data		U
List of Identifier Status		U
Identifier Status		M
List of Channel Related Diagnosis Data		U
Identifier Number		M
Channel Number		M
Channel Type		M
IO Type		M
Error Type		M
List of Status Data		U
Slot Number		M
Status Type		M
Status Specifier		M
Status Data		M
Module Status Data		U
Status Specifier		M
List of Module Status		M
Module Status		M
DXB-Link Status		U
Status Specifier		M
List of DXB-Link Status		M
Publisher Address		M
Link Status		M

Nom de paramètre	Req	Cnf
Link Error		M
Red Diagnosis		U
Red Status		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Red Status2		M
Role		M
HW Defect		M
Data Exchange		M
Safe State		M
DL Initialised		M
TOH started		M
Prm Command Ack		S
Seq Number		M
Function		M
Primary Request		M
MS1 Command		M
Check Properties		M
Master State Clear		M
Red Status1		M
Role		M
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialied		M
TOH started		M
Red Status2		M
Role		M
HW Defect		M
Data Exchange		M
Master State Clear		M
DL Initialised		M
TOH started		M
Revision Number		U

Nom de paramètre	Req	Cnf
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Diag Data
Ce paramètre est constitué des éléments de liste suivants:

Prm Req
Ce paramètre contient la valeur de l'attribut Prm Req de l'entrée ARL MS0 de l'esclave DP sélectionné.

Prm Fault
Ce paramètre contient la valeur de l'attribut Prm Fault de l'entrée ARL MS0 de l'esclave DP sélectionné.

Cfg Fault
Ce paramètre contient la valeur de l'attribut Cfg Fault de l'entrée ARL MS0 de l'esclave DP sélectionné.

Not Supported
Ce paramètre contient la valeur de l'attribut Not Supported de l'entrée ARL MS0 de l'esclave DP sélectionné.

Station Not Ready
Ce paramètre contient la valeur de l'attribut Station Not Ready de l'entrée ARL MS0 de l'esclave DP sélectionné.

Stat Diag
Ce paramètre contient la valeur de l'attribut Stat Diag de l'entrée ARL MS0 de l'esclave DP sélectionné.

WD On
Ce paramètre contient la valeur de l'attribut WD On de l'entrée ARL MS0 de l'esclave DP sélectionné.

Sync Mode
Ce paramètre contient la valeur de l'attribut Sync Mode de l'entrée ARL MS0 de l'esclave DP sélectionné.

Freeze Mode
Ce paramètre contient la valeur de l'attribut Freeze Mode de l'entrée ARL MS0 de l'esclave DP sélectionné.

Master Add
Ce paramètre contient la valeur de l'attribut Master Add de l'entrée ARL MS0 de l'esclave DP sélectionné.

Ident Number
Ce paramètre contient la valeur de l'attribut Ident Number de l'entrée ARL MS0 de l'esclave DP sélectionné.

Ext Diag Overflow
Ce paramètre Boolean contient la valeur du paramètre Ext Diag Overflow du service Set Slave Diag si toutes les valeurs délivrées des objets Diagnosis ne sont pas contenues dans cette confirmation de service. Ce paramètre est mis à TRUE par l'AP du maître DP (Classe 2) si toutes les valeurs délivrées des objets Diagnosis ne sont pas contenues dans cette confirmation de services en raison de problèmes de stockage.

Attribute Type: Boolean

Valeurs admissibles: TRUE, FALSE

NOTE En cas de condition de débordement, il est permis de tronquer seulement aux limites de l'objet Diagnosis.

Ext Diag Flag
Ce paramètre contient la valeur du paramètre Ext Diag Flag du service Set Slave Diag de l'esclave DP sélectionné.

Ext Diag Data
Ce paramètre est constitué des sous-paramètres suivants:

Device Related Diagnosis Data
Ce paramètre contient la valeur de l'objet Device Related Diagnosis de l'esclave DP sélectionné.

List of Identifier Status
Ce paramètre est constitué des éléments de liste suivants:

Identifier Status
Ce paramètre contient la valeur de l'attribut Identifier Status de l'objet Identifier Related Diagnosis de l'esclave DP sélectionné.

List of Channel Related Diagnosis Data
Ce paramètre est constitué des éléments de liste suivants:

Identifier Number
Ce paramètre contient la valeur de l'attribut Identifier Number d'un objet Channel Related Diagnosis de l'esclave DP sélectionné.

Channel Number
Ce paramètre contient la valeur de l'attribut Channel Number d'un objet Channel Related Diagnosis de l'esclave DP sélectionné.

Channel Type
Ce paramètre contient la valeur de l'attribut Channel Type d'un objet Channel Related Diagnosis de l'esclave DP sélectionné.

IO Type
Ce paramètre contient la valeur de l'attribut IO Type d'un objet Channel Related Diagnosis de l'esclave DP sélectionné.

Error Type
Ce paramètre contient la valeur de l'attribut Error Type d'un objet Channel Related Diagnosis de l'esclave DP sélectionné.

List of Status Data
Ce paramètre est constitué des éléments de liste suivants:

Slot Number
Ce paramètre contient la valeur de l'attribut Slot Number d'un objet Status de l'esclave DP sélectionné.

Status Type
Ce paramètre contient la valeur de l'attribut Status Type d'un objet Status de l'esclave DP sélectionné.

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier d'un objet Status de l'esclave DP sélectionné.

Status Data
Ce paramètre contient la valeur d'un objet Status de l'esclave DP sélectionné.

Module Status Data
Ce paramètre est constitué des sous-paramètres suivants:

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier de l'objet Module Status Data de l'esclave DP sélectionné.

List of Module Status
Ce paramètre est constitué des éléments de liste suivants:

Module Status
Ce paramètre contient la valeur de l'attribut Module Status de l'objet Module Status Data de l'esclave DP sélectionné.

DXB-Link Status
Ce paramètre est constitué des sous-paramètres suivants:

Status Specifier
Ce paramètre contient la valeur de l'attribut Status Specifier de l'objet DXB-Link Status d'un esclave DP.

List of DXB-Link Status
Ce paramètre est constitué des éléments de liste suivants:

Publisher Address
Ce paramètre contient la valeur de l'attribut Publisher Address de l'objet DXB-Link Status d'un esclave DP.

Link Status
Ce paramètre contient la valeur de l'attribut Link Status de l'objet DXB-Link Status d'un esclave DP.

Link Error
Ce paramètre contient la valeur de l'attribut Link Error de l'objet DXB-Link Status d'un esclave DP.

Red Diagnosis
Ce paramètre est constitué des sous-paramètres suivants:

Red Status
Ce paramètre de type sélection indique que le paramètre Red Diagnosis contient des informations de diagnostic des interfaces de communications redondantes d'un Esclave DP redondant Ce paramètre est constitué des sous-paramètres suivants:

Seq Number
Ce paramètre contient le Sequence Number nécessaire pour l'identification univoque du Red Status reçu.

Valeurs admissibles: 0 à 31

Fonction
Ce paramètre est constitué des sous-paramètres suivants:

Primary Request
Ce paramètre Boolean mis à TRUE indique qu'une Primary Request (Demande primaire) a été exécutée à cette interface de communication. Ce paramètre mis à FALSE indique qu'aucune Primary Request (Demande primaire) n'a été exécutée à cette interface de communication.

Valeurs admissibles: TRUE, FALSE

MS1 Command
Ce paramètre indique qu'une MS1 Command comme Start MS1, Stop MS1, Reset MS1 ou aucune MS1 Command n'a été exécutée à cette interface de communication.

Valeurs admissibles: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
Ce paramètre Boolean mis à TRUE indique qu'une demande Check Properties a été exécutée à cette interface de communication. Ce paramètre mis à FALSE indique qu'aucune demande Check Properties n'a été exécutée à cette 'interface de communication.

Valeurs admissibles: TRUE, FALSE

Master State Clear
Ce paramètre Boolean mis à TRUE indique que le maître DP (Classe 1) connexe a mis l'interface de communication dans l'état Clear. Ce paramètre mis à FALSE indique que le maître DP (Classe 1) connexe n'a envoyé aucune commande Master State Clear.

Valeurs admissibles: TRUE, FALSE

Red Status1
Ce paramètre contient le statut de redondance de la première interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que cette interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que cette interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect (défaut de matériel) de cette interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état Fail Safe (sécurité intégrée).

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de cette interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Red Status2
Ce paramètre contient le statut de redondance de la seconde (redondante) interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que l'autre interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que l'autre interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect de l'autre interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état Fail Safe.

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de l'autre interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Prm Command Ack
Ce paramètre de type sélection indique que le paramètre Red Diagnosis contient l'acquittement d'une Prm Command reçue précédemment. Ce paramètre est constitué des sous-paramètres suivants:

Seq Number
Ce paramètre contient le Sequence Number nécessaire pour l'identification univoque du Prm Command Ack reçu.

Valeurs admissibles: 0 à 31

Fonction
Ce paramètre est constitué des sous-paramètres suivants:

Primary Request
Ce paramètre Boolean mis à TRUE indique qu'une Primary Request (Demande primaire) a été exécutée à cette interface de communication. Ce paramètre mis à FALSE indique qu'aucune Primary Request (Demande primaire) n'a été exécutée à cette interface de communication.

Valeurs admissibles: TRUE, FALSE

MS1 Command
Ce paramètre indique qu'une MS1 Command comme Start MS1, Stop MS1, Reset MS1 ou aucune MS1 Command n'a été exécutée à cette interface de communication.

Valeurs admissibles: Start MS1, Stop MS1, Reset MS1, no Action

Check Properties
Ce paramètre Boolean mis à TRUE indique qu'une demande Check Properties a été exécutée à cette interface de communication. Ce paramètre mis à FALSE indique qu'aucune demande Check Properties n'a été exécutée à cette 'interface de communication.

Valeurs admissibles: TRUE, FALSE

Master State Clear
Ce paramètre Boolean mis à TRUE indique que le maître DP (Classe 1) connexe a mis l'interface de communication dans l'état Clear. Ce paramètre mis à FALSE indique que le maître DP (Classe 1) connexe n'a envoyé aucune commande Master State Clear.

Valeurs admissibles: TRUE, FALSE

Red Status1
Ce paramètre contient le statut de redondance de la première interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que cette interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que cette interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect (défaut de matériel) de cette interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que cette interface de communication est dans l'état Fail Safe (sécurité intégrée).

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de cette interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Red Status2
Ce paramètre contient le statut de redondance de la seconde (redondante) interface de communication d'un esclave DP redondant. Ce paramètre est constitué des sous-paramètres suivants:

Role
Ce paramètre mis à Primary indique que l'autre interface de communication agit comme l'interface primaire. Ce paramètre mis à Backup indique que l'autre interface de communication agit comme l'interface de secours (Backup). Ce paramètre mis à Power On indique que l'affectation des rôles entre deux interfaces de communication n'est pas encore terminée (par exemple à la mise sous tension de l'esclave DP).

Valeurs admissibles: Primary, Backup, Power On

HW Defect
Ce paramètre Boolean mis à TRUE indique qu'un HW Defect de l'autre interface de communication a été détecté.

Valeurs admissibles: TRUE, FALSE

Data Exchange
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état d'échange de données.

Valeurs admissibles: TRUE, FALSE

Safe State
Ce paramètre Boolean mis à TRUE indique que l'autre interface de communication est dans l'état Fail Safe.

Valeurs admissibles: TRUE, FALSE

DL Initialised
Ce paramètre Boolean mis à TRUE indique que le paramètre DL de l'autre interface de communication a été chargé.

Valeurs admissibles: TRUE, FALSE

TOH started
Ce paramètre Boolean mis à TRUE indique que le temporisateur Output Hold Timer de cette interface de communication a été lancé.

Valeurs admissibles: TRUE, FALSE

Revision Number
Ce paramètre contient la valeur du Revision Number de l'esclave DP sélectionné.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre contient la cause de la défaillance.

Valeurs admissibles: RE, DS, UE, RS, NA, NR, RR

6.2.3.3.4 New slave diag

Ce service peut être utilisé par un maître DP (Classe 1) pour indiquer à l'application que de nouvelles valeurs reçues des objets Diagnosis d'un esclave DP assigné qui est relatif à l'AR MS0 sont présentes. De plus, ce service peut être utilisé pour indiquer que de nouvelles informations de diagnostic fournies par la couche application DP de l'esclave DP ou du maître DP (Classe 1) sont présentes.

Le Tableau 73 montre les paramètres du service.

Tableau 73 – New Slave Diag

Nom de paramètre	Ind
Argument	M
AREP	M
CREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR connexe.

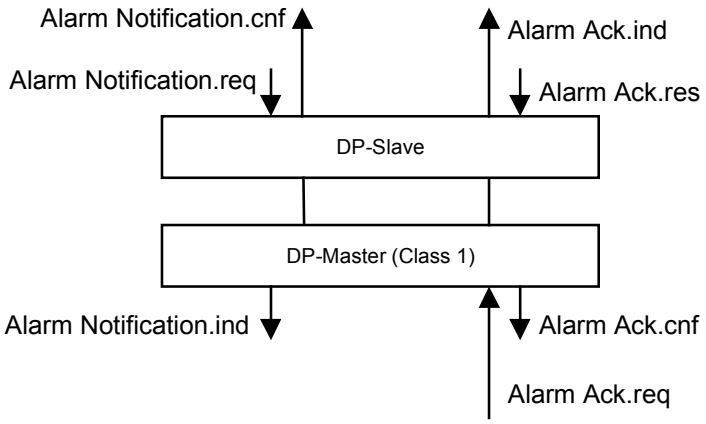
CREP
Ce paramètre est l'identificateur local de la CR connexe (Esclave DP).

6.2.4 ASE Alarm

6.2.4.1 Vue d'ensemble

Le modèle d'alarme permet le transfert d'une alarme de l'esclave DP vers le maître DP (Classe 1) assigné et l'acquittement explicite de l'alarme par le maître DP (Classe 1) par le biais de l'AR MS1.

La Figure 24 montre les interactions d'alarme entre l'esclave DP (Alarm Notification «notification d'alarme») et le maître DP (Alarm Ack «acquittement d'alarme»).



Légende

Anglais	Français
DP-master (Class 1)	Maître DP (Classe 1)
DP-slave	Esclave DP

Figure 24 – Traitement d'une alarme dans le système DP

Pour l'émission d'une alarme, les conditions suivantes sont à remplir:

– il faut que l'esclave DP soit dans le mode d'échange de données,
– il faut que l'AR MS1 soit ouverte,
– il faut que le type d'alarme correspondant soit activé,
– la limite des alarmes actives n'est pas dépassée.

En plus de l'émission d'alarmes, un diagnostic relatif à une voie (Channel Related Diagnosis) et/ou un diagnostic relatif à un identificateur (Identifier Related Diagnosis) peu(ven)t être émis et identifier la source exacte de l'alarme.

Deux variantes du traitement des alarmes (voir 6.2.10.3.1) sont définies:

— une seule alarme d'un type d'alarme spécifique peut être activée à la fois (Type Mode),
— plusieurs alarmes (2 à 32) indépendantes du type peuvent être actives à la fois (Sequence Mode).

Le maître DP (Classe 1) et l'esclave DP doivent mettre en œuvre le Type Mode et peuvent en plus mettre en œuvre le Sequence Mode si les alarmes sont prises en charge. Les alarmes sont placées en file d'attente. L'alarme reste active jusqu'à ce que le maître DP (Classe 1) l'acquitte explicitement.

L'acquittement d'alarme est accompli soit par l'intermédiaire de la CR, qui achemine les services Read et Write service, soit facultativement par l'intermédiaire d'une CR distincte sur l'AR MS1. Dans le second cas de figure, les alarmes ne sont pas différées par les services de lecture et d'écriture.

Les types d'alarmes suivants sont définis. Ils peuvent être étendus pour être spécifiques à un fabricant:

Diagnosis Alarm
Une Diagnosis Alarm (Alarme de diagnostic) signale un événement au sein d'un module (par exemple: surchauffe, court-circuit, etc.).

Process Alarm
Une Process Alarm (Alarme de processus) signale la survenue d'un événement dans le processus connecté (par exemple: valeur limite supérieure dépassée).

Pull Alarm
Une baie signale le retrait d'un module.

Plug Alarm
Une baie signale l'insertion d'un module.

Status Alarm
Une Status Alarm (Alarme de statut) signale un changement d'état d'un module, par exemple, run (en marche), stop (à l'arrêt) ou ready (prêt).

Update Alarm
Une Update Alarm (Alarme de mise à jour) signale le changement d'un paramètre dans un module, par exemple, par une opération locale ou un accès distant.

6.2.4.2 Spécification de la classe Alarm

6.2.4.2.1 Modèle

Un objet Alarm est décrit par le modèle suivant:

ASE DP:			**ASE Alarm**
CLASS:			**ALARM DATA**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attributs-clés:	Identifier
2.	(m)	Attribut:	Add Ack
3.	(m)	Attribut:	Alarm Specifier
4.	(o)	Attribut:	Alarm Data Description
SERVICES:			
1.	(m)	OpsService:	Alarm Notification
2.	(m)	OpsService:	Alarm Ack

6.2.4.2.2 Attributs

Identifier
Cet attribut-clé est constitué de Slot Number et d'Alarm Type pour indiquer le numéro de baie et le type d'alarme.

Valeurs admissibles pour Slot Number: 0 à 254

Les valeurs admissibles pour Alarm Type sont montrées dans le Tableau 74.

Tableau 74 – Alarm type

Valeur	Signification
1	Diagnosis Alarm (Alarme de diagnostic)
2	Process Alarm (Alarme de processus)
3	Pull Alarm (Alarme de retrait)
4	Plug Alarm (Alarme d'insertion)
5	Status Alarm (Alarme de statut)
6	Update Alarm (Alarme de mise à jour)
32-126	Manufacturer Specific (Spécifique à un fabricant)

Add Ack
Cet attribut indique au maître DP (Classe 1) que cette alarme exige un acquittement supplémentaire par l'utilisateur. Cela peut être accompli, par exemple, par un service d'écriture (write). L'alarme reste active après l'Alarm Ack (acquittement d'alarme).

Les valeurs admissibles sont montrées dans le Tableau 75.

Tableau 75 – Add Ack

Valeur	Signification
TRUE	acquittement supplémentaire requis
FALSE	aucun acquittement supplémentaire requis

Alarm Specifier
Cet attribut spécifie des informations d'alarme complémentaires.

Les valeurs admissibles sont montrées dans le Tableau 76.

Tableau 76 – Alarm specifier

Valeur	Signification
0	aucune différenciation supplémentaire
1	Une alarme survient et le module connexe est perturbé.
2	Une alarme disparaît et le module connexe n'a plus d'autres erreurs.
3	Une alarme disparaît et le module connexe est encore perturbé.

Alarm Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

6.2.4.2.3 Invocation de l'objet de données d'alarme

Pour l'invocation de l'objet Alarm Data, les règles suivantes s'appliquent:

— Les objets Alarm Data doivent être utilisés seulement si l'attribut de l'appareil DPV1 = TRUE,

— Plusieurs objets Alarm Data peuvent être invoqués dans un esclave DP,

— Pour chaque module, un seul objet Alarm Data d'un type doit être invoqué,

— Les objets Alarm Data ne doivent pas dépasser la longueur totale de 59 octets. L'attribut Alarm Data Description est à positionner en conséquence.

6.2.4.3 Spécification des services Alarm

6.2.4.3.1 Alarm notification

Ce service est utilisé pour transférer une notification d'alarme de l'esclave DP vers le maître DP (Classe 1). Ce service doit seulement être utilisé conjointement à l'AR MS1. Le Tableau 77 montre les paramètres du service.

Tableau 77 – Alarm notification

Nom de paramètre	Req	Ind	Cnf
Argument	M	M(=)	
AREP	M	M(=)	
Slot Number	M	M(=)	
Alarm Type	M	M(=)	
Seq Nr	M	M(=)	
Add Ack	M	M(=)	
Alarm Specifier	M	M(=)	
Alarm Data	U	C(=)	
Result(+)			S
AREP			M
Slot Number			M
Alarm Type			M
Seq Nr			M
Result(-)			S
AREP			M
Slot Number			M
Alarm Type			M
Seq Nr			M
Status			M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Ce paramètre contient la valeur de l'attribut Slot Number de l'objet Alarm Data.

Alarm Type
Ce paramètre contient la valeur de l'attribut Alarm Type de l'objet Alarm Data.

Seq Nr
Ce paramètre est utilisé comme un identificateur unique pour la transaction d'une notification d'alarme par la couche application DP.

Valeurs admissibles: 0 à 31

Add Ack
Ce paramètre contient la valeur de l'attribut Add Ack de l'objet Alarm Data.

Alarm Specifier
Ce paramètre contient la valeur de l'attribut Alarm Specifier de l'objet Alarm Data.

Alarm Data
Ce paramètre contient la valeur de l'objet "Alarm Data".

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre contient la cause de la défaillance.

Valeurs admissibles: MSAL1S_Stopped, Not_Enabled, Limit_Expired, Sequence_Nr_Pending, No_Start

6.2.4.3.2 Alarm ack

Ce service est utilisé par le maître DP (Classe 1) pour acquitter la réception d'une notification d'alarme qui a été précédemment reçue en provenance de l'esclave DP. Ce service doit seulement être utilisé conjointement à l'AR MS1.

Le Tableau 78 montre les paramètres du service.

Tableau 78 – Alarm Ack

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Alarm Type	M	M(=)		
Seq Nr	M	M(=)		
Result(+)			S	S
AREP			M	M(=)
Slot Number			M	M(=)
Alarm Type			M	M(=)
Seq Nr			M	M(=)
Result(-)			S	S
AREP			M	M(=)
Slot Number			M	M(=)
Alarm Type			M	M(=)
Seq Nr			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Ce paramètre contient la valeur de l'attribut Slot Number de l'objet Alarm Data reçu précédemment.

Alarm Type
Ce paramètre contient la valeur de l'attribut Alarm Type de l'objet Alarm Data reçu précédemment

Seq Nr
Ce paramètre contient le numéro de séquence de la notification d'alarme reçue précédemment.

Valeurs admissibles: 0 à 31

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.5 ASE Context

6.2.5.1 Vue d'ensemble

L'ASE Context fournit un jeu de services pour paramétrer et configurer le Default AP d'un esclave DP lui-même et ses module contenant des données E/S.

Le processus d'application de plusieurs esclaves DP a besoin d'un paramètre (par exemple, la plage d'un signal analogique) provenant du maître DP (Classe 1) assigné pour fournir les données issues de l'objet Input Data et/ou de l'objet Output Data conformément aux exigences de l'AP du maître DP (Classe 1). Ces esclaves DP fournissent dans l'ASE Context un objet MS0 Parameter ou, pour plus de flexibilité, plusieurs objets MS0 Structured Parameter. Les valeurs pour les attributs de ces objets sont fournies par le maître DP (Classe 1) et assignées après validation de la part de l'AP de l'esclave DP.

L'ASE Context fournit des objets et les services connexes pour la paramétrisation en toute flexibilité d'un esclave DP. Les objets suivants sont définis:

— Objet MS0 User Parameter (objet de paramétrisation de base),
— Objet MS0 Structured User Parameter,
— Objet DXB-Linktable,
— Objet DXB-Subscribertable,
— Objet IsoM Parameter,
— Objet Time AR Parameter.

La fonctionnalité d'une application d'esclave DP implique les nécessaires objets de paramétrisation, à acheminer, à la phase de démarrage du système DP, du maître DP (Classe 1) vers les esclaves DP assignés par le biais de l'AR MS0. L'application de l'esclave DP est tenue de vérifier les données de paramétrisation envoyées par le maître DP (Classe 1). Si la vérification échoue, l'esclave DP peut répondre par une entrée appropriée dans les informations de diagnostic.

L'objet MS0 Configuration Elements décrit la structure de module, la plage des données d'entrée et de sortie et contient des informations relatives à la cohérence des données. L'ASE Context définit un ensemble de services utilisés pour établir et récupérer leurs valeurs. L'application de l'esclave DP comparera les données de configuration envoyées par le maître DP (Classe 1) à la configuration locale de l'esclave DP. Si elles concordent, l'AR MS0 avec cet esclave DP est établie. La configuration réelle de l'esclave DP peut être lue par un maître DP (Classe 2) avec le service connexe.

L'objet DXB-Subscribertable conjointement à l'objet MS0 Configuration Elements donne les informations dont l'Abonné a besoin pour faire correspondre les données de sortie du maître DP (Classe 1) et/ou les données d'entrée souscrites d'un ou plusieurs Éditeurs aux zones de données des Output Data Elements de l'esclave DP.

L'utilisateur doit vérifier le contenu de l'objet DXB-Subscribertable fourni avec le service Check User Prm ou le service Check Ext User Prm et les informations de configuration fournies avec le service Check Cfg. En cas de résultat positif de ces vérifications, l'utilisateur doit extraire les informations nécessaires de l'objet DXB-Subscribertable dont la couche application DP a besoin pour l'abonnement. L'utilisateur doit charger cette information avec le service Load CRL DXB Link Entries.

Pour ce mapping, les règles suivantes doivent s'appliquer:

Les zones de données d'un Output Data Element peuvent être assignées:

— complètement au maître DP (Classe 1) (par défaut), ou

— partiellement au maître DP (Classe 1) et partiellement à un ou plusieurs Éditeurs, ou

— complètement à un ou plusieurs Éditeurs.

Le niveau le plus bas des informations de mapping contenues dans l'objet DXB-Subscribertable est un octet, aucun mapping de bit n'est pris en charge.

Il est permis que, pour une ou plusieurs zones de données d'un Output Data Element, aucune valeur ne soit fournie à partir du maître DP (Classe 1) ou à partir d'un Éditeur. Pour ces zones de donnée, les valeurs par défaut (zéro) doivent être utilisées.

Une zone de données d'un Output Data Element doit toujours être fournie soit par le maître DP (Classe 1), soit par un Éditeur. Il est interdit de fournir des valeurs seulement pour une partie d'une zone de données d'un Output Data Element.

Des variations des zones de données (longueur des Output Data) qui sont fournies par le maître DP (Classe 1) peuvent être réalisées par des variations des attributs Simple Output Data Description ou Extended Output Data Description pour cet Output Data Element.

Pour les Output Data Elements d'un esclave DP où aucune donnée de sortie n'est exigée du maître DP (Classe 1), l'attribut Length de la Simple ou Extended Output Data Description (aucune Output Data) doit être mis à zéro. Pour ces modules, aucune Output Data n'est transférée du maître DP (Classe 1).

Il convient de placer au début de l'Output Data Element les zones de données d'un Output Data Element qui sont habituellement affectées au maître DP (Classe 1). Dans ce cas, aucune information de mapping n'est nécessaire dans l'objet DXB-Subscribertable (mapping par défaut).

Les vérifications suivantes peuvent être accomplies pour chaque module au niveau de l'Abonné en fonction des informations fournies avec le service Check Cfg et les informations contenues dans l'objet DXB-Subscribertable.

Cas 1:

La longueur indiqué dans l'attribut Simple ou Extended Output Data Description d'un module est mise à zéro (aucune donnée de sortie issue du maître DP (Classe 1)) et l'objet DXB-Subscribertable contient au moins une entrée pour ce module, alors les règles suivantes doivent être appliquées:

— La somme de la longueur donnée dans l'attribut Length Data Area des entrées dans l'objet DXB-Subscribertable pour le Simple or Extended Output Data Element doit être la

longueur du Simple ou Extended Output Data Element, s'il existe une entrée pour chaque zone de données de l'Output Data Element.

— Le chevauchement des valeurs des attributs Offset Data Area et Length Data Area des entrées dans l'objet DXB-Subscribertable pour cet Output Data Element est interdit.

— L'objet DXB-Subscribertable ne doit contenir aucune DXB Subscriber Entry du maître DP (Classe 1) (Master Data) pour cet Output Data Element.

— Les zones de données non couvertes de l'Output Data Element doivent être alimentées de valeurs par défaut.

Cas 2:

La longueur indiquée dans l'attribut Simple ou Extended Output Data Description d'un module est différente de zéro et inférieure à la longueur complète de l'Output Data Element (des parties de l'Output Data pour cet Output Data Element sont fournies par le maître DP (Classe 1)) et l'objet DXB-Subscribertable contient au moins une entrée pour ce module, alors les règles suivantes doivent être appliquées:

— La somme des valeurs des attributs Length Data Area of the DXB Subscriber Entries dans l'objet DXB-Subscribertable du maître DP (Classe 1) (Master Data, c'est-à-dire «Données de maître») doit être égale à la longueur donnée dans l'attribut Simple ou Extended Output Data Description de cet Output Data Element.

— Si l'objet DXB-Subscribertable ne contient aucune entrée pour le maître DP (Classe 1) (Master Data, c'est-à-dire «Données de maître»), les valeurs pour cet Output Data Element issues du maître DP (Classe 1) doivent être mises en correspondance (mappées) avec les premières zones de données de cet Output Data Element (mapping par défaut).

— Le mapping des données souscrites de l'Éditeur doit être effectuée en fonction des entrées dans l'objet DXB-Subscribertable.

— Le chevauchement des valeurs des attributs Offset Data Area et Length Data Area des entrées dans l'objet DXB-Subscribertable pour cet Output Data Element est interdit tout comme l'est celui avec les zones de données fournies par le maître DP (Classe 1) par le biais du mapping par défaut).

— Les zones de données non couvertes de l'Output Data Element doivent être alimentées de valeurs par défaut.

Cas 3:

La longueur indiquée dans l'attribut Simple or Extended Output Data Description d'un module est égale à la longueur de l'Output Data Element (toutes les données de sortie pour cet Output Data Element sont fournies par le maître DP (Classe 1), alors la règle suivante doit être appliquée:

— L'objet DXB-Subscribertable ne doit contenir aucune entrée pour cet Output Data Element.

L'ASE Context fournit également le moyen de changer l'adresse d'un esclave DP par le biais de l'AR MS0. Cela peut être effectué par un maître DP (Classe 2).

La longueur d'un Input Data Element d'un module ne doit pas être influencée par les possibles variations des affectations des zones de données de l'Output Data Element.

L'ASE Context est utilisé pour établir ou libérer une association entre un AP individuel d'un maître DP (Classe 2) et un AP individuel de l'esclave DP pour la communication acyclique. Cette association appartient exclusivement à l'ASE Process Data.

La description formelle de l'ASE Context est présentée ci-après, suivie d'une description de ses services.

6.2.5.2 Spécification de la classe Context

6.2.5.2.1 Spécification de la classe MS0 user parameter

6.2.5.2.1.1 Modèle

Cette classe spécifie l'objet pour les données de paramètres qui peuvent être connexes à l'esclave DP lui-même et/ou au module de l'esclave DP.

L'objet MS0 User Parameter est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**MS0 User Parameter**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(o)	Attribut:	User Prm Data
2.1.	(o)	Attribut:	Device Related User Parameter
2.1.1	(m)	Attribut:	Data Description
2.2.	(o)	Attribut:	List of Module Related User Parameter
2.2.1	(m)	Attribut:	Data Description
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(m)	OpsService:	Check User Prm Result

6.2.5.2.1.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet User Parameter est adressé de façon implicite par les services.

User Prm Data
Cet attribut est un agrégat du Device Related User Parameter et/ou du List of Module Related User Parameter et se compose des éléments suivants.

Device Related User Parameter
Cet attribut se compose des éléments suivants:

Data Description
Cet attribut contient soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

List of Module Related User Parameter
Cet attribut se compose des éléments de liste suivants (chaque élément correspond à un module).

Data Description
Cet attribut contient soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

6.2.5.2.1.3 Invocation de l'objet MS0 user parameter

Pour l'invocation de l'objet MS0 User Parameter, les règles suivantes s'appliquent:

— L'objet MS0 User Parameter ne doit pas dépasser la longueur totale de 237 octets (seulement 234 si l'attribut de l'appareil DPV1 = TRUE). Les attributs Data Description et List of Module Related User Parameter sont à positionner en conséquence.

— L'objet MS0 User Parameter doit exister seulement si l'attribut de l'appareil DPV1 est mis à TRUE et le paramètre Prm Structure fourni avec le service Check User Prm est mis à FALSE ou si l'attribut de l'appareil DPV1 est mis à FALSE.

— Un seul objet MS0 User Parameter peut être invoqué dans un esclave DP.

6.2.5.2.2 Spécification de la classe MS0 structured user parameter

6.2.5.2.2.1 Modèle

Contrairement à l'objet MS0 User Parameter, cette classe définit des objets distincts pour l'appareil lui-même et pour chacun de ses modules. En outre, ces objets permettent l'identification et l'affectation uniques.

Il convient que cette structuration soit utilisée par des appareils évolués avec une forte variation des paramètres qui peuvent ne pas être présents dans chaque application. Cet objet est obligatoire pour les esclaves DP qui prennent en charge le mode Isochronous Mode et pour l'acheminement de l'objet DXB-Linktable.

L'objet MS0 Structured User Parameter est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**MS0 Structured User Parameter**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Identifier
2.	(m)	Attribut:	Structured User Prm Data
2.1.	(s)	Attribut:	Device Related User Parameter
2.1.1	(m)	Attribut:	Data Description
2.2.	(s)	Attribut:	Module Related User Parameter
2.2.1	(m)	Attribut:	Data Description
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.2.2 Attributs

Identifier
Cet attribut-clé est constitué de Slot Number et de Prm Data Type pour indiquer le numéro de la baie et le type de données de paramètres que contient l'objet MS0 Structured User Parameter spécifié. Si l'attribut Slot Number continent la valeur 0, les Structured Prm Data qui suivent sont Device Related User Parameter, autrement elles sont Module Related User Parameter

Type d'attribut de Slot Number: Unsigned8

Valeurs admissibles pour Slot Number: 0 à 244

Type d'attribut de Prm Data Type:

Les valeurs admissibles pour Prm Data Type sont montrées dans le Tableau 79.

Tableau 79 – Prm data type

Valeur	Signification
129	Parameter Data (Données de paramètres)
32–128	Manufacturer Specific (Spécifique à un fabricant)

Structured User Prm Data
Cet attribut contient soit l'attribut DeviceRelated User Parameter, soit l'attribut Module Related User Parameter indiqué par l'attribut Identifier et se compose des éléments suivants.

Device Related User Parameter
Cet attribut se compose des éléments suivants:

Data Description
Cet attribut contient soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Module Related User Parameter
Cet attribut se compose des éléments suivants.

Data Description
Cet attribut contient soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

6.2.5.2.2.3 Invocation de l'objet MS0 structured user parameter

Pour l'invocation de l'objet MS0 Structured User Parameter, les règles suivantes s'appliquent:

— Les objets MS0 Structured User Parameter doivent exister seulement si l'attribut de l'appareil DPV1 est mis à TRUE.

— Les objets MS0 Structured User Parameter peuvent être fournis avec le service Check User Prm si le paramètre Prm Structure fourni avec le service Check User Prm est mis à TRUE.

— La description de tous les objets MS0 Structured User Parameter ne doit pas dépasser la longueur totale de 230 octets si les valeurs sont fournies avec le service Check User Prm. L'attribut Data Description est à positionner en conséquence.

— La description de tous les objets MS0 Structured User Parameter ne doit pas dépasser la longueur totale de 240 octets si les valeurs sont fournies avec le service Check Ext User Prm. L'attribut Data Description est à positionner en conséquence.

— Plusieurs objets MS0 Structured User Parameter peuvent être invoqués dans un esclave DP.

— Un seul objet MS0 Structured User Parameter du même type doit être invoqué dans chaque baie.

6.2.5.2.3 Spécification de la classe DXB-linktable

6.2.5.2.3.1 Modèle

Cette classe spécifie les entrées de l'objet DXB-Linktable. Chaque entrée spécifie un Éditeur auprès duquel l'Abonné doit s'abonner.

L'objet DXB-Linktable est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**DXB-Linktable**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of DXB Link Entries
2.1.	(m)	Attribut:	Publisher Address
2.2.	(m)	Attribut:	Input Data Length Publisher
2.3.	(m)	Attribut:	Source Offset Publisher
2.4.	(m)	Attribut:	Length
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.3.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet DXB-Linktable est adressé de façon implicite par les services.

List of DXB Link Entries
Cet attribut est constitué des éléments de liste suivants:

Publisher Address
Cet attribut indique l'adresse de DL d'un esclave DP (Éditeur) auprès duquel l'Abonné doit s'abonner. Les règles suivantes doivent s'appliquer pour cet attribut:

Une valeur entre 0 et 125 représente une adresse d'Éditeur valide et, donc, une DXB-Link active.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 125

Input Data Length Publisher
Cet attribut définit le nombre d'octets de l'objet Simple Extended Input Data ou de l'objet Extended Input Data de l'Éditeur connexe.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

Source Offset Publisher
Cet attribut définit le début (décalage) de l'abonnement au sein de l'objet Simple Input Data ou Extended Input Data de l'Éditeur connexe donné par l'attribut Publisher Address.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 243

Length
Cet attribut définit le nombre d'octets auxquels l'Abonné doit s'abonner auprès de l'Éditeur connexe commençant au décalage donné dans l'attribut Source Offset Publisher.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

6.2.5.2.3.3 Invocation de l'objet "DXB-linktable"

Pour l'invocation de l'objet DXB-Linktable, les règles suivantes s'appliquent:

— L'objet DXB-Linktable doit exister seulement si l'attribut de l'appareil DPV1 de l'esclave DP est mis à TRUE.

— Une seule entrée pour chaque Éditeur doit être contenue dans l'objet DXB-Linktable.

— Les valeurs pour l'objet DXB-Linktable peuvent être fournies avec le service Check User Prm si le paramètre Prm Structure fourni avec le service Check User Prm est mis à TRUE.

— La description de l'objet DXB-Linktable ne doit pas dépasser la longueur totale de 230 octets si les valeurs de l'objet DXB-Linktable sont fournies avec le service Check User Prm. L'attribut List of DXB Link Entries est à positionner en conséquence.

— La description de l'objet DXB-Linktable ne doit pas dépasser la longueur totale de 240 octets si les valeurs de l'objet DXB-Linktable sont fournies avec le service Check Ext User Prm. L'attribut List of DXB Link Entries est à positionner en conséquence.

— Un seul objet DXB-Linktable doit être invoqué dans un esclave DP.

— L'objet DXB-Linktable doit seulement être invoqué dans un esclave DP si aucun objet DXB-Subscribertable n'est invoqué et réciproquement.

6.2.5.2.4 Spécification de la classe DXB-subscribertable

6.2.5.2.4.1 Modèle

Cette classe spécifie les entrées de l'objet DXB-Subscribertable. Chaque entrée spécifie la source, un Éditeur ou le maître DP (Classe 1), et les informations de mapping connexes nécessaires pour transférer les données correspondantes vers les zones de données d'un Output Data Element du module connexe.

L'objet DXB-Subscribertable est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**DXB-Subscribertable**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of DXB Subscriber Entries
2.1.	(s)	Attribut:	Publisher Address
2.1.1.	(m)	Attribut:	Input Data Length Publisher
2.1.2.	(m)	Attribut:	Source Offset Publisher
2.2.	(s)	Attribut:	Master Data
2.2.1.	(m)	Attribut:	Source Offset Master
2.3.	(m)	Attribut:	Dest Slot Number
2.4.	(m)	Attribut:	Offset Data Area
2.5.	(m)	Attribut:	Length Data Area
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.4.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet DXB-Subscribertable est adressé de façon implicite par les services.

List of DXB Subscriber Entries
Cet attribut est constitué des éléments de liste suivants:

Publisher Address
Cet attribut de sélection indique l'adresse de DL d'un esclave DP (Éditeur) auprès duquel l'Abonné doit s'abonner. Les règles suivantes doivent s'appliquer pour cet attribut:

Une valeur entre 0 et 125 représente une adresse d'Éditeur valide et, donc, une DXB-Link active.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 125

Input Data Length Publisher
Cet attribut définit le nombre d'octets de l'objet Simple Extended Input Data ou de l'objet Extended Input Data de l'Éditeur connexe.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

Source Offset Publisher
Cet attribut définit le début (décalage) de l'abonnement au sein de l'objet Simple Input Data ou Extended Input Data de l'Éditeur connexe donné par l'attribut Publisher Address.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 243

Master Data
Cet attribut de sélection indique que, pour cette entrée, les données doivent être dérivées du maître DP (Classe 1) connexe.

Type d'attribut: Unsigned8

Valeurs admissibles: 128

Source Offset Master
Cet attribut définit le début (décalage) au sein de l'Output Data du maître DP (Classe 1).

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 243

Dest Slot Number
Cet attribut indique le Slot Number du module auquel les valeurs souscrites de l'objet Input Data de l'Éditeur adressé ou les valeurs de l'Output Data adressé du maître DP (Classe 1) connexe doivent être mises en correspondance (mappées).

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 244

Offset Data Area
Cet attribut définit le début (décalage) au sein de l'Output Data du module (zone de données) auquel les valeurs souscrites de l'objet Input Data de l'Éditeur adressé ou les valeurs de l'Output Data adressé du maître DP (Classe 1) connexe doivent être mises en correspondance (mappées).

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 244

Length Data Area
Cet attribut définit le nombre d'octets issus de l'Éditeur connexe auxquels l'Abonné doit s'abonner, en commençant au décalage donné dans l'attribut Source Offset Publisher dans le cas d'un Éditeur. Si les données doivent être dérivées du maître DP (Classe 1), le décalage de début est donné dans l'attribut Source Offset Master.

Les valeurs souscrites de l'objet Input Data de l'Éditeur adressé ou les valeurs de l'Output Data adressé du maître DP (Classe 1) connexe doivent être mises en correspondance (mappées) aux zones de données du module données dans les attributs Dest Slot Number et Offset Data Area de la DXB Subscriber Entry connexe.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 244

6.2.5.2.4.3 Invocation de l'objet "DXB-subscribertable"

Pour l'invocation de l'objet DXB-Subscribertable, les règles suivantes s'appliquent:

— L'objet DXB-subscribertable doit exister seulement si l'attribut de l'appareil DPV1 de l'esclave DP est mis à TRUE.

— Les valeurs pour l'objet DXB-subscribertable peuvent être fournies avec le service Check User Prm si le paramètre Prm Structure fourni avec le service Check User Prm est mis à TRUE.

— La description de l'objet DXB-Subscribertable ne doit pas dépasser la longueur totale de 230 octets si les valeurs de l'objet DXB-Subscribertable sont fournies avec le service Check User Prm. L'attribut List of DXB Link Entries est à positionner en conséquence.

— La description de l'objet DXB-Subscribertable ne doit pas dépasser la longueur totale de 240 octets si les valeurs de l'objet DXB-Subscribertable sont fournies avec le service Check Ext User Prm. L'attribut List of DXB Link Entries est à positionner en conséquence.

— Un seul objet DXB-Subscribertable doit être invoqué dans un esclave DP.

— L'objet DXB-Subscribertable doit seulement être invoqué dans un esclave DP si aucun objet DXB-Linktable n'est invoqué et réciproquement.

6.2.5.2.5 Spécification de la classe IsoM parameter

6.2.5.2.5.1 Modèle

Cette classe spécifie les paramètres nécessaires pour qu'un esclave DP fonctionne dans le mode Isochronous Mode.

L'objet IsoM Parameter est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**IsoM Parameter**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	T_{BASE_DP}
3.	(m)	Attribut:	T_{DP}
4.	(m)	Attribut:	T_{MAPC}
5.	(m)	Attribut:	T_{BASE_IO}
6.	(m)	Attribut:	T_I
7.	(m)	Attribut:	T_O
8.	(m)	Attribut:	T_{DX}
9.	(m)	Attribut:	T_{PLL_W}
10.	(m)	Attribut:	T_{PLL_D}
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.5.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet IsoM Parameter est adressé de façon implicite par les services.

T_{BASE_DP}
Cet attribut indique la base de temps (en unités de 1/12 μs) pour le temps T_{DP}.

Valeurs admissibles: 375, 750, 1 500, 3 000, 6 000, 12 000 (la valeur 1 500 est obligatoire pour le mode Isochronous Mode standard)

T_{DP}
Cet attribut indique le temps T_{DP} (en unités de T_{BASE_DP}. Ce temps inclut toutes les parties du cycle de DP isochrone:

— message de synchronisation (SYNCH),

— services cycliques (DX),

— services acycliques (MSG),

— jeton (TOK),

— temps de réserve (RES).

Type d'attribut: Unsigned16

Valeurs admissibles: 1 à 2^{16} - 1

T_{MAPC}
Cet attribut indique le Master Application Cycle Time (en unités de T_{DP}) dont l'application du maître DP (Classe 1) a besoin pour traiter complètement une tâche d'application. Du point de vue isochrone, il y a exactement une tâche utilisateur pour commander la partie isochrone de l'application.

Valeurs admissibles: 1 à 14

T_{BASE_IO}
Cet attribut indique la base de temps (en unités de 1/12 µs) pour les temps T_I et T_O de l'esclave DP.

Valeurs admissibles: 375, 750, 1 500, 3 000, 6 000, 12 000 (la valeur 1 500 est obligatoire pour le mode Isochronous Mode standard)

T_I
Cet attribut indique le temps T_I (en unités de T_{BASE_IO}) pour l'esclave DP.

Valeurs admissibles: 0 (cas spécial), 1 à 2^{16} - 1

T_O
Cet attribut indique le temps T_O (en unités de T_{BASE_IO}) pour l'esclave DP.

Valeurs admissibles: 0 (cas spécial), 1 à 2^{16} - 1

T_{DX}
Cet attribut indique le temps T_{DX} (en unités de 1/12 µs) pour l'esclave DP.

Valeurs admissibles: 0 (cas spécial), 1 à 2^{32} - 1

T_{PLL_W}
Cet attribut indique le temps T_{PLL_W} (en unités de 1/12 µs) pour la PLL de l'esclave DP.

Valeurs admissibles: 1 à 2^{16} - 1 (valeur par défaut 12)

T_{PLL_D}
Cet attribut indique le temps T_{PLL_D} (en unités de 1/12 µs) pour la PLL de l'esclave DP.

Type d'attribut: Unsigned16

Valeurs admissibles: 0 à 2^{16} - 1

6.2.5.2.5.3 Invocation de l'objet IsoM parameter

Pour l'invocation de l'objet IsoM Parameter, les règles suivantes s'appliquent:

– L'objet IsoM Parameter doit exister seulement si l'attribut de l'appareil DPV1 de l'esclave DP correspondant est mis à TRUE.

– Les valeurs de l'objet IsoM Parameter peuvent être fournies avec le service Check User Prm si le paramètre Prm Structure fourni avec le service Check User Prm est mis à TRUE.

– Les valeurs de l'objet IsoM Parameter peuvent être fournies avec le service Check Ext User Prm.

Un seul objet IsoM Parameter doit être invoqué dans un esclave DP.

6.2.5.2.6 Spécification de la classe Time AR parameter

6.2.5.2.6.1 Modèle

Cette classe spécifie les paramètres nécessaires pour qu'un esclave DP fonctionne comme un récepteur de temps (Time Receiver).

L'objet Time AR Parameter est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**Time AR Parameter**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	Clock Sync Interval
3.	(o)	Attribut:	CS Delay Time
SERVICES:			
1.	(m)	OpsService:	Check User Prm
2.	(o)	OpsService:	Check Ext User Prm
3.	(m)	OpsService:	Check User Prm Result
4.	(o)	OpsService:	Check Ext User Prm Result

6.2.5.2.6.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Time AR Parameter est adressé de façon implicite par les services.

Clock Sync Interval
Cet attribut contient la valeur pour l'attribut "intervalle de synchronisation d'horloge" (Clock Sync Interval) de l'ARL de l'esclave DP fonctionnant comme récepteur de temps.

Valeurs admissibles: 1 à 2^{16} - 1; Valeur par défaut: 1 000

Base de temps: 10

CS Delay Time
Cet attribut contient la valeur pour l'attribut CS Delay Time de l'ARL de l'esclave DP fonctionnant comme récepteur de temps.

Valeur par défaut: 0

Type: Network Time Difference

6.2.5.2.6.3 Invocation de l'objet Time AR parameter

Pour l'invocation de l'objet Time AR Parameter, les règles suivantes sont à respecter:

– L'objet Time AR Parameter doit exister seulement si l'attribut de l'appareil DPV1 de l'esclave DP correspondant est mis à TRUE.

– Les valeurs de l'objet Time AR Parameter peuvent être fournies avec le service Check User Prm si le paramètre Prm Structure fourni avec le service Check User Prm est mis à TRUE.

– Les valeurs de l'objet Time AR Parameter peuvent être fournies avec le service Check Ext User Prm.

Un seul objet Time AR Parameter doit être invoqué dans un esclave DP.

6.2.5.2.7 Spécification de la classe MS0 configuration elements

6.2.5.2.7.1 Modèle

L'objet MS0 Configuration Elements est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**MS0 Configuration Elements**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	List of Cfg Data
2.1.	(o)	Attribut:	Input Data Reference
2.2.	(o)	Attribut:	Output Data Reference
2.3.	(o)	Attribut:	Manufacturer Specific Data
2.4.	(o)	Attribut:	Module Name
SERVICES:			
1.	(m)	OpsService:	Set Cfg
2.	(m)	OpsService:	Get Cfg
3.	(m)	OpsService:	Check Cfg
4.	(m)	OpsService:	Check Cfg Result

6.2.5.2.7.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet MS0 Configuration Elements est adressé de façon implicite par les services.

List of Cfg Data
Cet attribut est un agrégat des données de configuration d'un esclave DP qui est décrit par l'attribut Cfg Identifier ou l'attribut Special Cfg Identifier. Il doit contenir au moins en entrée Cfg Data. Une entrée Cfg Data est constituée des attributs suivants.

Input Data Reference
Référence à un Input Data Element d'un objet Simple Input Data ou d'un objet Extended Input Data.

Output Data Reference
Référence à un Output Data Element d'un objet Simple Output Data ou d'un objet Extended Output Data.

Manufacturer Specific Data
Cet attribut contient un ajout spécifique à un fabricant aux attributs Input Data Reference et/ou Output Data Reference. Le contenu est spécifique à un appareil ou spécifique à un module.

Type d'attribut: Octet string

Module Name
Cet attribut définit un nom d'un élément de liste.

Type d'attribut: Visible String(32)

6.2.5.2.7.3 Invocation de l'objet MS0 configuration elements

Pour l'invocation de l'objet MS0 Configuration Elements, les règles suivantes s'appliquent:

– Un seul objet MS0 Configuration Elements doit être invoqué dans un esclave DP.
– La description de l'objet MS0 Configurations Elements ne doit pas dépasser 244 octets. L'attribut List of Cfg Data est à positionner en conséquence.

6.2.5.2.8 Spécification de la classe Remanent parameter

6.2.5.2.8.1 Modèle

Un objet Remanent Parameter est décrit par le modèle suivant:

ASE DP:	**ASE Context**
CLASS:	**Remanent Parameter**
CLASS ID:	non utilisé

PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	No Add Change
3.	(m)	Attribut:	New Slave Add
4.	(o)	Attribut:	Rem Para
4.1	(m)	Attribut:	Data Description
SERVICES:			
1.	(m)	OpsService:	Set Slave Add

6.2.5.2.8.2 Attributs

Implicit
L'attribut "Implicit" indique qu'un objet Remanent Parameter est adressé de façon implicite par le service.

No Add Change
Cet attribut spécifie s'il est permis, oui ou non, de changer de nouveau l'adresse de l'esclave DP. Après la réinitialisation initiale, l'adresse de DL de l'esclave DP est mise à l'adresse par défaut (126) et l'attribut No Add Change est mis à FALSE. Pour faire passer cet attribut de TRUE à FALSE, une réinitialisation initiale de l'esclave DP est nécessaire.

Valeurs admissibles: FALSE, TRUE

New Slave Add
Cet attribut contient l'adresse de l'esclave DP qui est à établir.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 125

Rem Para
Cet attribut contient les paramètres à stocker de façon permanente.

Data Description
Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

6.2.5.2.8.3 Invocation de l'objet Rem Para

Pour l'invocation de l'objet Remanent Parameter, les règles suivantes s'appliquent:

– Un seul objet Remanent Parameter doit être invoqué dans un esclave DP.

– L'objet Remanent Parameter ne doit pas dépasser 242 octets. L'attribut Data Description est à positionner en conséquence.

6.2.5.2.9 Spécification de la classe MS2 user parameter

6.2.5.2.9.1 Modèle

L'objet MS2 User Parameter est décrit par le modèle suivant:

ASE DP:			**ASE Context**
CLASS:			**MS2 User Parameter**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	Features Supported
2.1.	(m)	Attribut:	Supported Feature
3.	(m)	Attribut:	Profile Features Supported
3.1.	(m)	Attribut:	Supported Profile Feature
4.	(m)	Attribut:	Profile Ident Number
SERVICES:			

1. (m) OpsService: Initiate
2. (m) OpsService: Abort

6.2.5.2.9.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet MS2 User Parameter est adressé de façon implicite par les services.

Features Supported
Cet attribut est constitué des éléments de liste suivants:

Supported Feature
Cet attribut reflète les caractéristiques réelles sur l'AR MS2 d'un esclave DP. Le maître DP (Classe 2) et l'esclave DP s'informent l'un l'autre de la fonctionnalité de service prise en charge. L'esclave DP et le maître DP (Classe 2) ont la possibilité de réajuster leur fonctionnalité conformément aux exigences. L'application utilisateur du maître DP doit décider si, oui ou non, la fonctionnalité rapportée est suffisante.

Les valeurs admissibles sont montrées dans le Tableau 80.

Tableau 80 – Supported feature

Valeur	Signification
1	Service Read et Write pris en charge
2 à 16	réservé pour d'autres caractéristiques

Profile Features Supported
Cet attribut est constitué des éléments de liste suivants:

Supported Profile Feature
Cet attribut reflète les caractéristiques de profil réelles sur l'AR MS2 d'un esclave DP. Le maître DP (Classe 2) et l'esclave DP s'informent l'un l'autre des caractéristiques de profil prises en charge. Concernant la définition utilisée pour "profil", le profil est identifié par le Profile Ident Number. L'application utilisateur du maître DP doit décider si, oui ou non, la fonctionnalité rapportée est suffisante.

Les valeurs admissibles sont montrées dans le Tableau 81.

Tableau 81 – Supported profile feature

Valeur	Signification
1 à 16	caractéristique à définir par les profils

Profile Ident Number
L'attribut Profile Ident Number contient l'identificateur unique d'une définition de profil. Il est nécessaire que tous les appareils utilisant la même définition de profil emploient le même Profile Ident Number. Le Profile Ident Number est pris dans le groupe des Ident Number pour les profils spécifiques à un vendeur ou autorisés. La valeur 0 indique qu'aucun profil n'est pris en charge.

Type d'attribut: Unsigned16

6.2.5.2.9.3 Invocation de l'objet MS2 user parameter

Pour l'invocation de l'objet MS2 user parameter, les règles suivantes sont à respecter:

Un seul objet MS2 user parameter doit être invoqué dans un esclave DP.

6.2.5.2.10 Spécification de la classe Red Context

6.2.5.2.10.1 Modèle

Cette classe spécifie le statut des interfaces de communication redondantes d'un esclave DP et l'état d'une commande de commutation fournie antérieurement.

Le modèle suivant décrit l'objet Red Context:

ASE DP: **ASE Context**
CLASS: **Red Context**
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:
1. (m) Attribut-clé: Communication Interface Identifier
2. (m) Attribut: Role
SERVICES:
1. (m) OpsService: Check User Prm
2. (m) OpsService: Check User Prm Result

6.2.5.2.10.2 Attributs

Communication Interface Identifier

Cet attribut identifie de façon univoque l'interface de communication.

Role

Cet attribut définit le rôle réel de l'interface de communication avec les valeurs selon le Tableau 82.

Valeurs admissibles: Primary, Backup, Power On

Tableau 82 – Role

Valeur	Signification
Primary	À l'interface Primary (Primaire), il n'existe aucune contrainte entre l'interface de communication et les ASE.
Backup	Dans le cas de Backup (Secours), les éléments suivants entre l'interface de communication et les ASE sont actifs: ASE Diagnosis (tous les services) ASE Context (Check User Prm, Check Ext User Prm, Check Cfg) ASE Process Data (tous les services, seulement par l'AR MS2) ASE Load Region (tous les services, seulement par l'AR MS2) ASE Function Invocation (tous les services, seulement par l'AR MS2)
Power On	Dans le cas de Power On (Mise sous tension), les éléments suivants entre l'interface de communication et les ASE sont actifs: ASE Diagnosis (tous les services) ASE Context (Check User Prm, Check Ext User Prm)

6.2.5.2.10.3 Invocation de l'objet Red Context

Pour l'invocation de l'objet Red Context, les règles suivantes sont à respecter:

— L'objet Red Context doit être utilisé seulement si l'attribut de l'appareil DPV1 est mis à TRUE.

— Pour chaque interface de communication, un seul objet Red Context doit être invoqué.

6.2.5.3 Spécification de services de Context

6.2.5.3.1 Check user Prm

La couche application DP de l'esclave DP notifie à l'application le moment où une paramétrisation valide est reçue du maître DP parce que ce service demande une validation de User Prm Data ou de List of Structured Prm Data. Une indication de Check User Prm supplémentaire est seulement émise après que le service Check User Prm est complété avec un service Check User Prm Result.

Le service Check User Prm doit être acquitté négativement (le paramètre Prm OK du service Check User Prm Result doit être mis à FALSE) par l'application de l'esclave DP si les conditions suivantes sont remplies:

– le paramètre Prm Structure est mis à FALSE et les données de paramétrisation (User Prm Data) ne sont ni permises ni acceptables par l'application de l'esclave DP,

– le paramètre Prm Structure est mis à TRUE et au moins un objet MS0 Structured User Parameter, l'objet DXB-Linktable, l'objet DXB-Subscribertable, ou l'objet IsoM Parameter n'existe pas,

– le paramètre Prm Structure est mis à TRUE et au moins une valeur pour un objet MS0 Structured User Parameter ou pour l'objet DXB-Linktable ou pour l'objet DXB-Subscribertable ou pour l'objet IsoM Parameter n'est ni autorisée ni acceptable par l'application de l'esclave DP.

Le Tableau 83 montre les paramètres du service.

Tableau 83 – Check user Prm

Nom de paramètre	Ind
Argument	M
AREP	M
Prm Structure	M
User Prm Data	C
Device Prm Data	U
Module Prm Data	U
List of Structured Prm Data	C
Structured User Prm Data	U
Device Related User Parameter	U
Device Prm Data	M
Module Related User Parameter	U
Slot Number	M
Module Prm Data	M
DXB-Linktable	U
List of DXB Link Entries	M
Publisher Address	M
Input Data Length Publisher	M
Source Offset Publisher	M
Length	M
DXB-Subscribertable	U
List of DXB Subscriber Entries	M
Publisher Address	C
Input Data Length Publisher	M
Source Offset Publisher	M
Master Data	C
Source Offset Master	M
Dest Slot Number	M
Offset Data Area	M
Length Data Area	M
IsoM Parameter	U
$T_{BASE\ DP}$	M
T_{DP}	M
T_{MAPC}	M
$T_{BASE\ IO}$	M
T_I	M

Nom de paramètre	Ind
T_O	M
T_{DX}	M
$T_{PLL\ W}$	M
$T_{PLL\ D}$	M
Prm Command	U
Seq Number	M
Primary Request	M
MS1 Command	M
Master State Clear	M
Check Properties	M
Primary Request used	M
MS1 Command used	M
Address Change	M
Address Offset	M
Output Hold Time	M
Time AR Parameter	U
Clock Sync Interval	M
CS Delay Time	U

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR.

Prm Structure
Ce paramètre indique quels types d'objets de paramétrisation sont fournis par le service Check User Prm. Ce paramètre doit seulement être évalué si l'attribut de l'appareil DPV1 est mis à TRUE.

Les valeurs admissibles sont montrées dans le Tableau 84.

Tableau 84 – Prm structure

Valeur	Signification
TRUE	Les paramètres qui suivent sont relatifs aux objets suivants: MS0 Structured User Parameter DXB-Linktable IsoM Parameter
FALSE	Les paramètres qui suivent sont seulement relatifs à l'objet MS0 User Parameter.

User Prm Data
Ce paramètre conditionnel est seulement présent si le paramètre Prm Structure contient la valeur FALSE. Il se compose de l'un et/ou l'autre des sous-paramètres suivants.

Device Prm Data
Ce paramètre contient la valeur de l'attribut Device Related User Parameter de l'objet MS0 User Parameter.

Module Prm Data
Ce paramètre contient la valeur de l'attribut Module Related User Parameter de l'objet MS0 User Parameter.

List of Structured Prm Data
Ce paramètre conditionnel est seulement présent si le paramètre Prm Structure contient la valeur TRUE. Ce paramètre est constitué d'un ou plusieurs des éléments de liste suivants.

Structured User Prm Data
Ce paramètre se compose de l'un et/ou l'autre des sous-paramètres suivants.

Device Related User Parameter
Ce paramètre indique la présence de Device Related User Parameter.

Device Prm Data
Ce paramètre contient la valeur de l'attribut Device Related User Parameter de l'objet MS0 Structured User Parameter.

Module Related User Parameter
Ce paramètre indique la présence de Module Related User Parameter.

Slot Number
Ce paramètre contient la valeur de l'attribut Slot Number de l'objet MS0 Structured User Parameter.

Module Prm Data
Ce paramètre contient la valeur de l'attribut Module Related User Parameter de l'objet MS0 Structured User Parameter.

DXB-Linktable
Ce paramètre indique la présence de DXB-Linktable.

List of DXB Link Entries
Ce paramètre est constitué des éléments de liste suivants:

Publisher Address
Ce paramètre contient la valeur de l'attribut Publisher Address d'une entrée DXB-Linktable de l'objet DXB-Linktable.

Input Data Length Publisher
Ce paramètre contient la valeur de l'attribut Input Data Length Publisher d'une entrée DXB-Linktable de l'objet DXB-Linktable.

Source Offset Publisher
Ce paramètre contient la valeur de l'attribut Source Offset Publisher d'une entrée DXB-Linktable de l'objet DXB-Linktable.

Length
Ce paramètre contient la valeur de l'attribut Length d'une entrée DXB-Linktable de l'objet DXB-Linktable.

DXB-Subscribertable
Ce paramètre indique la présence de DXB-Subscribertable.

List of DXB Subscriber Entries
Ce paramètre est constitué des éléments de liste suivants:

Publisher Address
Ce paramètre contient la valeur de l'attribut Publisher Address d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Input Data Length Publisher
Ce paramètre contient la valeur de l'attribut Input Data Length Publisher d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Source Offset Publisher
Ce paramètre contient la valeur de l'attribut Source Offset Publisher d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Master Data
Ce paramètre contient la valeur de l'attribut Master Data d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Source Offset Master
Ce paramètre contient la valeur de l'attribut Source Offset Master d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Dest Slot Number
Ce paramètre contient la valeur de l'attribut Dest Slot Number d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Offset Data Area
Ce paramètre contient la valeur de l'attribut Offset Data Area d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Length Data Area
Ce paramètre contient la valeur de l'attribut Length Data Area d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

IsoM Parameter
Ce paramètre indique la présence de IsoM Parameter.

TBASE_DP
Ce paramètre contient la valeur de l'attribut T_{BASE_DP} de l'objet IsoM Parameter.

T_{DP}
Ce paramètre contient la valeur de l'attribut T_{DP} de l'objet IsoM Parameter.

TMAPC
Ce paramètre contient la valeur de l'attribut T_{MAPC} de l'objet IsoM Parameter.

TBASE_IO
Ce paramètre contient la valeur de l'attribut T_{BASE_IO} de l'objet IsoM Parameter.

T_I
Ce paramètre contient la valeur de l'attribut T_I de l'objet IsoM Parameter.

T_O
Ce paramètre contient la valeur de l'attribut T_O de l'objet IsoM Parameter.

T_{DX}
Ce paramètre contient la valeur de l'attribut T_{DX} de l'objet IsoM Parameter.

TPLL_W
Ce paramètre contient la valeur de l'attribut T_{PLL_W} de l'objet IsoM Parameter.

TPLL_D
Ce paramètre contient la valeur de l'attribut T_{PLL_D} de l'objet IsoM Parameter.

Prm Command
Ce paramètre indique la présence de Prm Command.

Seq Number
Ce paramètre contient le Sequence Number nécessaire pour l'identification univoque du Prm Command reçu.

Valeurs admissibles: 0 à 31

Primary Request
Si ce paramètre est mis à TRUE, l'interface de communication sélectionnée doit fonctionner comme une primaire. Toutes les fonctions MS1 sont actives et sont traitées comme si la redondance n'était pas activée. Il faut que toutes les informations de diagnostic, y compris celles relatives à une voie, ou tous les autres messages de statut soient également présent(e)s.

Ce paramètre mis à FALSE signifie que le rôle de l'interface de communication (primary, backup) ne doit pas être modifié.

Valeurs admissibles: TRUE, FALSE

MS1 Command
Ce paramètre définit la commande souhaitée pour l'AR MS1 de l'interface de communication sélectionnée.

Les valeurs admissibles sont montrées dans le Tableau 85.

Tableau 85 – MS1 Command

Valeur	Signification
no Action	Aucune commande pour l'AR MS1 n'est présente.
Stop MS1	Ce paramètre spécifie le moment où l'AR MS1 de l'esclave DP doit être arrêtée par la couche application DP. Il n'y a plus de services d'AR MS1 possibles.
Start MS1	Ce paramètre spécifie le moment où l'AR MS1 de l'esclave DP doit être (re)démarrée par la couche application DP. Après le (re)démarrage, tous les services d'AR MS1 sont possibles si le rôle de l'interface de communication est primaire.
Reset MS1	Signifie que l'AR MS1doit être arrêtée et ensuite démarrée.

Master State Clear
Ce paramètre indique l'état réel de l'AR MS0 du maître DP (Classe 1). Ce paramètre mis à TRUE signifie que l'AR MS0 AR du maître DP (Classe 1) est dans le mode de fonctionnement CLEAR. Ce paramètre mis à FALSE signifie que l'AR MS0 AR du maître DP (Classe 1) est dans le mode de fonctionnement OPERATE.

Valeurs admissibles: TRUE, FALSE

Check Properties
Ce paramètre indique si, oui ou non, l'esclave DP doit vérifier les propriétés fournies par le service Check User Prm. Le paramètre mis à TRUE indique que les propriétés doivent être vérifiées (par exemple: au démarrage de l'esclave DP). Ce paramètre mis à FALSE indique que la vérification des propriétés doit être omise (par exemple: dans le mode "échange de données" de l'esclave DP).

Valeurs admissibles: TRUE, FALSE

Primary Request used
Ce paramètre indique si, oui ou non, l'interface de communication d'un esclave DP redondant prend en charge des demandes primaires. L'esclave DP doit répondre négativement (avec le paramètre Prm OK du service Check User Prm mis à FALSE) si ces demandes ne sont pas prises en charge. Dans l'autre cas, le Check Properties doit être accepté.

Valeurs admissibles: TRUE, FALSE

MS1 Command used
Ce paramètre indique si, oui ou non, les commandes MS1 Start MS1, Stop MS1 et Reset MS1 sont prises en charge par l'interface de communication d'un esclave DP redondant. L'esclave DP doit répondre négativement (avec le paramètre Prm OK du service Check User Prm mis à FALSE) si ces demandes ne sont pas prises en charge. Dans l'autre cas, le Check Properties doit être accepté.

Valeurs admissibles: TRUE, FALSE

Address Change
Ce paramètre indique si, oui ou non, des changements d'adresse sont demandés avec la demande primaire. TRUE indique que des changements d'adresse sont demandés et FALSE indique qu'aucun changement d'adresse n'est demandé. L'esclave DP doit répondre négativement (avec le paramètre

Prm OK du service Check User Prm mis à FALSE) si ces demandes ne sont pas prises en charge. Dans l'autre cas, le Check Properties doit être accepté.

Valeurs admissibles: TRUE, FALSE

Address Offset
Ce paramètre indique si, oui ou non, le décalage d'adresse de 64 est utilisé au démarrage de l'esclave DP. L'esclave DP doit répondre négativement (avec le paramètre Prm OK du service Check User Prm mis à FALSE) si le décalage d'adresse n'est pas pris en charge. Dans l'autre cas, le Check Properties doit être accepté.

Valeurs admissibles: TRUE, FALSE

Output Hold Time
Ce paramètre indique le temps où l'esclave DP redondant ne doit pas changer ses valeurs de sortie lorsqu'il accomplit une commutation redondante. Lorsque l'interface de communication de secours (backup) devient la primaire (et vice versa), l'utilisateur peut se servir des valeurs de sortie précédemment émises à partir de l'ancienne interface de communication primaire ou l'utilisateur peut attendre le premier échange de données après la commutation pour écrire de nouvelles valeurs dans ses sorties. Une possible commutation de backup vers primary démarrera toujours l'Output_Hold_Time.

Timebase: 10 ms.

Valeurs admissibles: 0 à 2^{16} - 1

Time AR Parameter
Ce paramètre indique la présence de Time AR Parameter.

Clock Sync Interval
Ce paramètre contient la valeur pour l'attribut Clock Sync Interval de l'objet ARL DP-slave.

CS Delay Time
Ce paramètre contient la valeur pour l'attribut CS Delay Time de l'objet ARL DP-slave.

6.2.5.3.2 Check user Prm result

Ce service est utilisé dans un esclave DP comme une réponse au service Check User Prm précédemment émis. Le Default AP d'un esclave DP utilise ce service pour fournir le résultat de la validation de User Prm Data ou de l'attribut List of Structured Prm Data. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 86 montre les paramètres du service.

Tableau 86 – Check user Prm result

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Prm OK	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR souhaitée.

Prm OK
Le paramètre Prm OK égal à TRUE spécifie que les valeurs de User Prm Data ou de l'attribut List of Structured Prm Data acheminées par le précédent service Check User Prm sont acceptables par l'application de l'esclave DP. Si elles ne sont pas acceptables pour l'application de l'esclave DP, ce paramètre doit avoir la valeur FALSE. Dans ce cas, les attributs Prm Req et Prm Fault de la MS0 ARL Entry connexe sont mis à TRUE par la couche application DP.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué. L'esclave DP doit ignorer la paramétrisation des maîtres DP.

Status
Ce paramètre indique la cause de l'échec.

Les valeurs admissibles sont montrées dans le Tableau 87.

Tableau 87 – Valeurs de Status

Valeur	Signification
Reset	Réinitialisation encore en cours
Not Pending	Aucune validation en attente
New Prm	Réception de nouvelles données User Prm Data ou de nouvelle List of Structured Prm Data pendant la validation
Inv Master Add	Un maître DP différent a paramétré l'esclave DP ou bien le maître DP a libéré l'esclave DP (la validation des précédentes User Prm Data ou List of Structured Prm Data a été inutile).

6.2.5.3.3 Check Ext user Prm

Ce service doit seulement être utilisé si l'AR MS0 propose une CR supplémentaire pour l'acheminement des données de paramétrisation et l'attribut de l'appareil DPV 1 est mis à TRUE.

La couche application DP de l'esclave DP notifie à l'application si des données de paramétrisation supplémentaires sont reçues du maître DP (Classe 1). Ce service demande une validation de l'attribut List of Structured Prm Data. Une indication de Check Ext User Prm supplémentaire est seulement émise après que le service Check Ext User Prm est complété avec un service Check Ext User Prm Result.

Le service Check Ext User Prm doit être acquitté négativement (le paramètre Ext Prm OK du service Check Ext User Prm Result doit être mis à FALSE) si au moins un objet de paramétrisation est inconnu, non autorisé ou inacceptable par l'application de l'esclave DP.

Le Tableau 88 montre les paramètres du service.

Tableau 88 – Check Ext user Prm

Nom de paramètre	Ind
Argument	M
AREP	M
List of Structured Prm Data	M
Structured User Prm Data	U
Device Related User Parameter	U
Device Prm Data	M
Module Related User Parameter	U
Slot Number	M
Module Prm Data	M
DXB-Linktable	U
List of DXB Link Entries	M
Publisher Address	M
Input Data Length Publisher	M
Source Offset Publisher	M
Length	M
DXB-Subscribertable	U
List of DXB Subscriber Entries	M
Publisher Address	C
Input Data Length Publisher	M
Source Offset Publisher	M
Master Data	C
Source Offset Master	M
Dest Slot Number	M
Offset Data Area	M
Length Data Area	M
IsoM Parameter	U
T_{BASE_DP}	M
T_{DP}	M
T_{MAPC}	M
T_{BASE_IO}	M
T_I	M
T_O	M
T_{DX}	M
T_{PLL_W}	M
T_{PLL_D}	M
Time AR Parameter	U
Clock Sync Interval	M
CS Delay Time	U

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR.

List of Structured Prm Data
Ce paramètre est constitué d'un ou plusieurs des éléments de liste suivants.

Structured User Prm Data
Ce paramètre se compose de l'un et/ou l'autre des sous-paramètres suivants.

Device Related User Parameter
Ce paramètre indique la présence de Device Related User Parameter.

Device Prm Data
Ce paramètre contient la valeur de l'attribut Device Related User Parameter de l'objet MS0 Structured User Parameter.

Module Related User Parameter
Ce paramètre indique la présence de Module Related User Parameter.

Slot Number
Ce paramètre contient la valeur de l'attribut Slot Number de l'objet MS0 Structured User Parameter.

Module Prm Data
Ce paramètre contient la valeur de l'attribut Module Related User Parameter de l'objet MS0 Structured User Parameter.

DXB-Linktable
Ce paramètre indique la présence de DXB-Linktable.

List of DXB Link Entries
Ce paramètre est constitué des éléments de liste suivants:

Publisher Address
Ce paramètre contient la valeur de l'attribut Publisher Address d'une entrée DXB-Linktable de l'objet DXB-Linktable.

Input Data Length Publisher
Ce paramètre contient la valeur de l'attribut Input Data Length Publisher d'une entrée DXB-Linktable de l'objet DXB-Linktable.

Source Offset Publisher
Ce paramètre contient la valeur de l'attribut Source Offset Publisher d'une entrée DXB-Linktable de l'objet DXB-Linktable.

Length
Ce paramètre contient la valeur de l'attribut Length d'une entrée DXB-Linktable de l'objet DXB-Linktable.

DXB-Subscribertable
Ce paramètre indique la présence de DXB-Subscribertable.

List of DXB Subscriber Entries
Ce paramètre est constitué des éléments de liste suivants:

Publisher Address
Ce paramètre contient la valeur de l'attribut Publisher Address d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Input Data Length Publisher
Ce paramètre contient la valeur de l'attribut Input Data Length Publisher d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Source Offset Publisher
Ce paramètre contient la valeur de l'attribut Source Offset Publisher d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Master Data
Ce paramètre contient la valeur de l'attribut Master Data d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Source Offset Master
Ce paramètre contient la valeur de l'attribut Source Offset Master d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Dest Slot Number
Ce paramètre contient la valeur de l'attribut Dest Slot Number d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Offset Data Area
Ce paramètre contient la valeur de l'attribut Offset Data Area d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

Length Data Area
Ce paramètre contient la valeur de l'attribut Length Data Area d'une entrée DXB-Subscribertable de l'objet DXB-Subscribertable.

IsoM Parameter
Ce paramètre indique la présence de IsoM Parameter.

TBASE_DP
Ce paramètre contient la valeur de l'attribut T_{BASE_DP} de l'objet IsoM Parameter.

T_{DP}
Ce paramètre contient la valeur de l'attribut T_{DP} de l'objet IsoM Parameter.

TMAPC
Ce paramètre contient la valeur de l'attribut T_{MAPC} de l'objet IsoM Parameter.

TBASE_IO
Ce paramètre contient la valeur de l'attribut T_{BASE_IO} de l'objet IsoM Parameter.

T_I
Ce paramètre contient la valeur de l'attribut T_I de l'objet IsoM Parameter.

T_O
Ce paramètre contient la valeur de l'attribut T_O de l'objet IsoM Parameter.

T_{DX}
Ce paramètre contient la valeur de l'attribut T_{DX} de l'objet IsoM Parameter.

TPLL_W
Ce paramètre contient la valeur de l'attribut T_{PLL_W} de l'objet IsoM Parameter.

TPLL_D
Ce paramètre contient la valeur de l'attribut T_{PLL_D} de l'objet IsoM Parameter.

Time AR Parameter
Ce paramètre indique la présence de Time AR Parameter.

Clock Sync Interval
Ce paramètre contient la valeur pour l'attribut Clock Sync Interval de l'objet ARL DP-slave.

CS Delay Time
Ce paramètre contient la valeur pour l'attribut CS Delay Time de l'objet ARL DP-slave.

6.2.5.3.4 Check Ext user Prm result

Ce service est utilisé dans un esclave DP comme une réponse au service Check Ext User Prm précédemment émis. Le Default AP d'un esclave DP utilise ce service pour fournir le résultat de la validation de l'attribut List of Structured Prm Data. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 89 montre les paramètres du service.

Tableau 89 – Check Ext user Prm result

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Ext Prm OK	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR souhaitée.

Ext Prm OK
Le paramètre Ext Prm OK égal à TRUE spécifie que l'attribut List of Structured Prm Data acheminé par le précédent service Check User Prm est acceptable par l'application de l'esclave DP. Si l'attribut List of Structured Prm Data n'est pas acceptable pour l'application de l'esclave DP, ce paramètre doit avoir la valeur FALSE. Dans ce cas, les attributs Prm Req et Cfg Fault de la MS0 ARL Entry connexe sont mis à TRUE par la couche application DP.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué. L'esclave DP doit ignorer la paramétrisation des maîtres DP.

Status
Ce paramètre indique la cause de l'échec.

Les valeurs admissibles sont montrées dans le Tableau 90.

Tableau 90 – Valeurs de Status

Valeur	Signification
Reset	Réinitialisation encore en cours
Not Pending	Aucune validation en attente
New Prm	Réception d'une nouvelle List of Structured Prm Data pendant la validation
Inv Master Add	Un maître DP différent a paramétré l'esclave DP ou bien le maître DP a libéré l'esclave DP (la validation de la précédente List of Structured Prm Data a été inutile).

6.2.5.3.5 Check Cfg

La couche application DP de l'esclave DP notifie à l'application le moment où des Cfg Data (données de configuration) sont reçues du maître DP parce que ce service demande une validation des Cfg Data par l'application. Une indication de Check Cfg supplémentaire est seulement émise après que le service Check Cfg est complété avec un service Check Cfg Result. L'esclave DP compare ses propres données de configuration aux Cfg Data reçues en provenance du maître DP.

La vérification doit couvrir les détails suivants:

— la structure de module,

— la plage de l'Input and/or Output Data Element,

— le format,

— la cohérence.

La cohérence affecte également l'esclave DP et le maître DP. La vérification de la cohérence dans l'élément de liste de Cfg Data doit engendrer seulement dans les cas suivants un défaut de configuration:

— L'esclave DP a besoin de la cohérence pour un Input et/ou Output Data Element et le maître DP indique l'absence de cohérence.

— L'esclave DP ne peut pas assurer la cohérence pour un Input et/ou Output Data Element et le DP-master exige la cohérence pour son Input et/ou Output Data Element.

Les esclaves DP peuvent accepter différents jeux de Cfg Data. Dans ce cas, le Check Cfg doit être couronné de succès si un jeu de données de configuration concorde. Si le paramètre de service Check Cfg Mode est mis à TRUE; la vérification des Cfg Data peut être effectuée d'une manière différente spécifique à l'utilisateur. Par exemple, un module temporairement indisponible peut être accepté même si les Cfg Data contiennent le Configuration Identifier respectif.

Le Tableau 91 montre les paramètres du service.

Tableau 91 – Check Cfg

Nom de paramètre	Ind
Argument	—
AREP	M
Check Cfg Mode	C
Cfg Data	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR.

Check Cfg Mode
Ce paramètre indique comment l'application de l'esclave DP doit vérifier les Cfg Data. Ce paramètre doit seulement être évalué si l'attribut de l'appareil DPV1 est mis à TRUE.

Cfg Data
Ce paramètre contient les Cfg Data à vérifier par l'application de l'esclave DP.

6.2.5.3.6 Check Cfg result

Ce service est utilisé dans un esclave DP comme une réponse au service Check Cfg précédemment émis. Le Default AP de l'esclave DP utilise ce service pour fournir le résultat de la validation des Cfg Data. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 92 montre les paramètres du service.

Tableau 92 – Check Cfg result

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Cfg OK	M	
Input Data Len	M	
Output Data Len	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR souhaitée.

Cfg OK
Le paramètre Cfg OK égal à TRUE spécifie que les Cfg Data acheminées par le précédent service Check Cfg sont acceptables par l'application de l'esclave DP. L'application de l'esclave DP peut mettre à jour les Real Cfg Data (données de configuration réelles) de l'AR MS0 avec le service Set Cfg. Si les Cfg Data ne sont pas acceptables pour l'application de l'esclave DP, ce paramètre doit avoir la valeur FALSE. Dans ce cas, les attributs Prm Req et Cfg fault sont mis à TRUE par l'couche application DP.

Input Data Len
Ce paramètre contient la longueur réelle de "Input Data".

Output Data Len
Ce paramètre contient la longueur réelle de "Output Data".

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre indique la cause de l'échec.

Les valeurs admissibles sont montrées dans le Tableau 93.

Tableau 93 – Valeurs de Status

Valeur	Signification
Reset	Réinitialisation encore en cours
Not Pending	Aucune validation de Cfg Data en attente
New Cfg	Réception de nouvelles Cfg Data pendant la validation
Inv Master Add	Un maître DP différent a paramétré l'esclave DP ou bien le maître DP a libéré l'esclave DP (la validation des Cfg Data précédentes a été inutile).

6.2.5.3.7 Set Cfg

Ce service est utilisé par l'application de l'esclave DP pour transmettre des données de configuration à la couche application DP. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 94 montre les paramètres du service.

Tableau 94 – Set Cfg

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Cfg Data	M	
Result(+)		S
AREP		M
Result(-)		S
AREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR souhaitée.

Cfg Data
Ce paramètre contient les Cfg Data.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.5.3.8 Get Cfg

Ce service permet au maître DP (Classe 2) de lire les données de configuration de l'esclave DP par le biais de l'AR MS0.

Le Tableau 95 montre les paramètres du service.

Tableau 95 – Get Cfg

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Cfg Data		M
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Cfg Data
Ce paramètre contient les Cfg Data.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Le paramètre Status indique le type de défaillance du service.

Valeurs admissibles: DS, NA, RS, UE, NR, RE

6.2.5.3.9 Set slave add

Ce service permet au maître DP (Classe 2) de changer l'adresse d'un esclave DP et de transmettre des données esclaves rémanentes. Le service Set Slave Add est couronné de succès seulement si l'Ident Number émis correspond à l'Ident Number de l'esclave DP et l'attribut No Add Change (voir 6.2.10.3.1) de l'esclave DP a la valeur FALSE. Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 96 montre les paramètres du service.

Tableau 96 – Set Slave Add

Nom de paramètre	Req	Ind	Cnf
Argument	M	M(=)	
AREP	M	M(=)	
New Slave Add	M	M(=)	
Ident Number	M	M(=)	
No Add Chg	M	M(=)	
Rem Slave Data	U	C(=)	
Result(+)			S
AREP			M
Result(-)			S
AREP			M
Status			M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR souhaitée.

New Slave Add
Ce paramètre contient la valeur de l'attribut New Slave Add de l'objet Remanent Parameter de l'esclave DP à positionner.

Ident Number
Ce paramètre contient l'Ident Number à vérifier avec l'Ident Number de l'esclave DP.

No Add Chg
Ce paramètre contient la valeur de l'attribut No Add Change de l'objet Remanent Parameter. L'attribut No Add Change prendra la valeur du paramètre de service en cas de succès; autrement, No Add Change reste inchangé.

Rem Slave Data
Ce paramètre contient la valeur de l'attribut Rem Para de l'objet Remanent Parameter.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi. Cette confirmation ne reflète pas si le service a réussi ou non, c'est-à-dire si les nouvelles valeurs ont été acceptées par l'esclave DP ou non. Par conséquent, le maître DP doit vérifier l'exécution correcte du service en utilisant le service Read Slave Diag avec la nouvelle adresse de l'esclave DP.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Le paramètre Status indique le type de défaillance du service.

Valeurs admissibles: DS, NA, RS, RR, UE, RE

6.2.5.3.10 Initiate

Ce service doit être utilisé par un maître DP (Classe 2) pour établir une connexion à l'esclave DP souhaité. Ce service doit seulement être utilisé conjointement à l'AR MS2.

Le Tableau 97 montre les paramètres du service.

Tableau 97 – Initiate

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Send Timeout	M			
Features Supported	M	M(=)		
Profile Features Supported	M	M(=)		
Profile Ident Number	M	M(=)		
Add Addr Param	M	M(=)		
S-Type	M	M(=)		
S-Len	M	M(=)		
D-Type	M	M(=)		
D-Len	M	M(=)		
S-Addr	M	M(=)		
S-API	M	M(=)		
S-SCL	M	M(=)		
S-Network Address	C	C(=)		
S-MAC Address	C	C(=)		
D-Addr	M	M(=)		
D-API	M	M(=)		
D-SCL	M	M(=)		
D-Network Address	C	C(=)		
D-MAC Address	C	C(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Max Len Data Unit			M	M(=)
Features Supported			M	M(=)
Profile Features Supported			M	M(=)
Profile Ident Number			M	M(=)
Add Addr Param			M	M(=)
S-Type			M	M(=)
S-Len			M	M(=)
D-Type			M	M(=)
D-Len			M	M(=)
S-Addr			M	M(=)
S-API			M	M(=)
S-SCL			M	M(=)
S-Network Address			C	C(=)
S-MAC Address			C	C(=)

Nom de paramètre	Req	Ind	Rsp	Cnf
D-Addr			M	M(=)
D-API			M	M(=)
D-SCL			M	M(=)
D-Network Address			C	C(=)
D-MAC Address			C	C(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Decode			M	M(=)
Error Code 1			M	M(=)
Error Code 2			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Max Len Data Unit
Ce paramètre contient la valeur de l'attribut Max Len Data Unit de l'objet ARL DP-slave d'un esclave DP.

Send Timeout
Ce paramètre définit le temps de commande pour surveiller l'AR MS2 demandée par le maître DP (Classe 2). Le maître DP et l'esclave DP s'informent l'un l'autre du paramètre Send Timeout pris en charge. L'esclave DP réajuste son paramètre Send Timeout. Le maître DP (Classe 2) décide si la valeur communiquée de Send Timeout est suffisante ou non.

Features Supported
Le maître DP et l'esclave DP s'informent l'un l'autre de la fonctionnalité de service prise en charge. L'esclave DP a la possibilité de réajuster sa fonctionnalité selon les exigences du Maître. L'application utilisateur du maître DP doit décider si, oui ou non, la fonctionnalité rapportée est suffisante.

Profile Features Supported
Le maître DP et l'esclave DP s'informent l'un l'autre de la fonctionnalité de service prise en charge concernant la définition de profil utilisée. Le profil est identifié par le Profile Ident Number. La signification de bits définis est spécifique au profil ou au vendeur.

Profile Ident Number
Ce paramètre identifie de façon univoque une définition de profil. Tous les appareils utilisant la même définition de profil sont tenus d'utiliser le même Profile Ident Number. Le Profile Ident Number est pris dans le groupe des Ident Number pour les profils spécifiques à un vendeur ou autorisés. La valeur 0 indique qu'aucun profil n'est pris en charge. Si le profil demandé est pris en charge par l'esclave DP, le Profile Ident Number est miroité dans la réponse. Si le profil demandé n'est pas pris en charge par l'esclave DP, l'esclave DP répond négativement ou avec un Profile Ident Number pris en charge par l'esclave DP.

Add Addr Param
Add Addr Param consiste en deux parties, à savoir informations d'adresse complémentaires de la source et informations d'adresse complémentaires de la destination. Les informations d'adresse complémentaires contiennent le paramètre API plus une adresse Réseau/MAC facultative.

D-Addr et S-Addr (API, SCL, Network Address, MAC Address) sont relatives au sens des télégrammes, ce qui signifie que la source devient destination et vice versa dans la réponse.

Add Addr Param contient les sous-paramètres suivants:

S-Type
Ce sous-paramètre indique la présence (S-Type = 1) de l'adresse Réseau/MAC facultative dans l'Add Addr Param de la source.

S-Len
Ce sous-paramètre indique la longueur du sous-paramètre S-Addr.

D-Type
Ce sous-paramètre indique la présence (D-Type = 1) de l'adresse Réseau/MAC facultative dans l'Add Addr Param de la destination.

D-Len
Ce sous-paramètre indique la longueur du sous-paramètre D-Addr.

S-Addr
Ce sous-paramètre contient les informations d'adresse complémentaires de la source:

S-API
Ce sous-paramètre identifie l'AP de la source.

S-SCL
Ce sous-paramètre identifie le niveau d'accès de la source.

S-Network Address
Si S-Type = 1, ce sous-paramètre identifie l'adresse réseau de la source conformément aux adresses réseau de l'ISO/OSI.

S-MAC Address
Si S-Type = 1, ce sous-paramètre identifie l'adresse MAC de la source.

D-Addr
Ce sous-paramètre contient les informations d'adresse complémentaires de la destination:

D-API
Ce sous-paramètre identifie l'AP de la destination.

D-SCL
Ce sous-paramètre identifie le niveau d'accès de la destination.

D-Network Address
Si D-Type = 1, ce sous-paramètre identifie l'adresse réseau de la destination conformément aux adresses réseau de l'ISO/OSI.

D-MAC Address
Si D-Type = 1, ce sous-paramètre identifie l'adresse MAC de la destination.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Decode
Ce paramètre sélectionne l'un des schémas d'erreur (Error scheme) suivants: DPV1, FMS, Profile_Specific.

Error Code 1
Lorsque Error Decode est égal à DPV1, le code d'erreur (Error Code) 1 prend l'une des valeurs suivantes:
version conflict (conflit de versions), feature not supported (caractéristique non prise en charge), user specific (spécifique à un utilisateur), state conflict (conflit d'états), invalid range (plage non valide), invalid parameter (paramètre non valide), invalid type (type non valide), resource busy (ressource occupée), resource unavailable (ressource non disponible).

Error Code 2
Le paramètre Error Code 2 est spécifique à un utilisateur.

6.2.5.3.11 Abort

Le maître DP et l'esclave DP utilisent ce service pour abandonner une AR. Avec Abort.req, l'utilisateur réinitialise le FSPM. Dans des conditions de défaut, le FSPM informe l'utilisateur avec un Abort.ind et se réinitialise. Ce service doit seulement être utilisé conjointement à l'AR MS2.

Le Tableau 98 montre les paramètres du service.

Tableau 98 – Abort

Nom de paramètre	Req	Ind
Argument	M	M
AREP	M	M(=)
Locally Generated		C
Subnet	C	C(=)
Instance	C	C(=)
Reason Code	C	C(=)
Additional Detail		C

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est utilisé comme identificateur local de l'AR souhaitée.

Locally Generated
Ce paramètre existe seulement dans l'indication dans le cas d'une AR MS2. Il indique que le service Abort a été produit par le partenaire distant (Locally Generated = FALSE) ou dans le poste local (Locally Generated = TRUE).

Subnet
Ce paramètre est utilisé dans le cas d'une AR MS2 pour coder les informations relatives à l'emplacement de la source du service Abort.

Instance
Ce paramètre est utilisé dans le cas d'une AR MS2 pour coder les informations relatives à l'instance de protocole qui a détecté le problème ayant conduit à l'Abort.

Les valeurs admissibles sont montrées dans le Tableau 99.

Tableau 99 – Instance

Valeur	Signification
DL	Data-link layer (Couche liaison de données)
MS2	AR MS2
User	Utilisateur

Reason Code
Le Reason Code indique la cause de l'Abort dans le cas d'une AR MS2.

Valeurs admissibles: UE, RR, RS, NR, DH, LR, RDL, RDH, DS, NA, ABT_SE, ABT_FE, ABT_TO, ABT_RE, ABT_IV, ABT_STO, ABT_IA, ABT_OC.

Additional Detail
Ce paramètre contient les informations de détails dans le cas d'un abandon qui est causé par un Initiate inapproprié (par exemple: valeur de Send Timeout trop petite) dans le cas d'une AR MS2. Ce paramètre est présent seulement si le paramètre Reason Code a la valeur ABT_STO et le paramètre Instance est égal à MS2. Dans ce cas, le paramètre Additional Detail contient la valeur du paramètre Send Timeout reçu de l'esclave DP.

6.2.5.3.12 MS0 init DP-slave

Ce service est utilisé pour initialiser l'AR MS0 de l'esclave DP.

Le Tableau 100 montre les paramètres du service.

Tableau 100 – MS0 init DP-slave

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result		M
AREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result
Ce paramètre indique que la demande de service a réussi

6.2.5.3.13 MS1 init DP-slave

Ce service est utilisé pour initialiser l'AR MS1 de l'esclave DP.

Le Tableau 101 montre les paramètres du service.

Tableau 101 – MS1 init DP-slave

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result		M
AREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result
Ce paramètre indique que la demande de service a réussi

6.2.5.3.14 MS2 init DP-slave

Ce service est utilisé pour initialiser toutes les AR MS2 de l'esclave DP.

Le Tableau 102 montre les paramètres du service.

Tableau 102 – MS2 init DP-slave

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result		M
AREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result
Ce paramètre indique que la demande de service a réussi

6.2.5.3.15 DP-slave started

Avec ce service, les caractéristiques d'alarme prises en charge sur cette AR MS1 sont indiquées. L'indication se produit après que l'esclave DP est entré dans le mode "échange de données" et le transfert de messages d'alarme et de statut est désormais possible.

Le Tableau 103 montre les paramètres du service.

Tableau 103 – DP-slave started

Nom de paramètre	Ind
Argument	M
AREP	M
Actual Enabled Alarms	M
Alarm Sequence	M
Alarm Limit	C

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Actual Enabled Alarms
Ce paramètre indique quels types d'alarmes sont pris en charge par le maître DP (Classe 1).

Valeurs admissibles: Update Alarm enabled / disabled, Manufacturer Specific Alarm enabled / disabled, Diagnosis Alarm enabled / disabled, Process Alarm enabled / disabled, Pull Plug Alarm enabled / disabled.

Alarm Sequence
Ce paramètre Boolean indique si une seule alarme d'un Alarm Type spécifique peut être active à la fois (Alarm Sequence=FALSE) ou plusieurs alarmes (2 à 32) de type identique ou différent peuvent être actives à la fois (Alarm Sequence=TRUE).

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

Alarm Limit
Ce paramètre indique le nombre maximal d'alarmes parallèles.

Les valeurs admissibles sont montrées dans le Tableau 104.

Tableau 104 – Alarm limit

Valeur	Signification
0	1 alarme de chaque type
1	2 alarmes au total
2	4 alarmes au total
3	8 alarmes au total
4	12 alarmes au total
5	16 alarmes au total
6	24 alarmes au total
7	32 alarmes au total

6.2.5.3.16 DP-slave stopped

Dans le cas d'une AR MS0, ce service de la couche application indique à l'utilisateur que l'esclave DP a quitté le mode "échange de données" et que l'AR MS1 est fermée. En conséquence, le transfert d'alarmes ainsi que le traitement des services Write et Read ne sont plus possibles.

Dans le cas d'une AR MS1, ce service indique à l'utilisateur que la relation de communication acyclique au maître DP (Classe 1) a été arrêtée. Cela signifie qu'il n'y a plus d'échange cyclique de données, l'AR MS0 et l'AR MS1 d'un esclave DP sont mises à l'état initial. Le Tableau 105 montre les paramètres du service.

Tableau 105 – DP-slave stopped

Nom de paramètre	Ind
Argument	M
AREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

6.2.5.3.17 Reset DP-slave

Par ce service, l'utilisateur d'un esclave DP réinitialise la couche application.

Le Tableau 106 montre les paramètres du service.

Tableau 106 – Reset DP-slave

Nom de paramètre	Req	Cnf
Argument		

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service. Néanmoins, il n'y a pas de paramètres spécifiques dans ce service.

6.2.5.3.18 DP-slave fault

Par ce service, la couche application indique à l'utilisateur d'un esclave DP que toutes les AR ont été réinitialisées à cause d'erreurs fatales.

Le Tableau 107 montre les paramètres du service.

Tableau 107 – DP-slave fault

Nom de paramètre	Ind
Argument	

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service. Néanmoins, il n'y a pas de paramètres spécifiques dans ce service.

6.2.5.3.19 Application ready DP-slave

Par ce service, l'utilisateur d'un esclave DP indique à la couche application que l'application est prête Ce service doit seulement être utilisé conjointement à l'AR MS0.

Le Tableau 108 montre les paramètres du service.

Tableau 108 – Application ready DP-slave

Nom de paramètre	Req
Argument	M
AREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

6.2.5.3.20 Start subscriber

Ce service est utilisé par l'application de l'esclave DP pour lancer l'abonnement d'une certaine DXB-Link.

Le Tableau 109 montre les paramètres du service.

Tableau 109 – Start subscriber

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
CREP	M	
Result(+)		S
AREP		M
CREP		M
Result(-)		S
AREP		M
CREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Ce paramètre est l'identificateur local de la CR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.5.3.21 Stop subscriber

Ce service est utilisé par l'application de l'esclave DP pour arrêter l'abonnement d'une certaine DXB-Link.

Le Tableau 110 montre les paramètres du service.

Tableau 110 – Stop subscriber

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
CREP	M	
Result(+)		S
AREP		M
CREP		M
Result(-)		S
AREP		M
CREP		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Ce paramètre est l'identificateur local de la CR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.5.3.22 Publisher active

Ce service indique à l'application de l'Abonné (esclave DP) l'état de l'Éditeur d'une certaine DXB-Link. Ce service est émis par la couche application DP seulement si l'état de l'Éditeur a changé depuis la dernière période de chien de garde.

Le Tableau 111 montre les paramètres du service.

Tableau 111 – Publisher active

Nom de paramètre	Ind
Argument	M
AREP	M
CREP	M
Status	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

CREP
Ce paramètre est l'identificateur local de la CR souhaitée (Éditeur).

Status
Ce paramètre indique l'état de l'Éditeur connexe.

Les valeurs admissibles sont montrées dans le Tableau 112.

Tableau 112 – Status

Valeur	Signification
TRUE	l'Éditeur connexe est actif: pendant la dernière période de chien de garde, l'objet Input Data souscrit de l'Éditeur connexe a été reçu au moins une fois.
FALSE	l'Éditeur connexe n'est pas actif; pendant la dernière période de chien de garde, l'objet Input Data souscrit de l'Éditeur connexe n'a pas été reçu en raison des problèmes suivants: la valeur de l'attribut Input Data Length Publisher de l'Éditeur connexe de l'entrée de CRL connexe n'était pas alignée sur la longueur réelle de l'objet Input Data édité par l'Éditeur connexe l'Éditeur connexe n'était pas dans le mode "échange de données" défaillance de l'Éditeur connexe

6.2.5.3.23 Init DP-master Cl1

Par ce service, l'utilisateur initialise la couche application d'un maître DP (Classe 1). Les paramètres sont décrits dans la présente spécification, parties 3 et 4.

Le Tableau 113 montre les paramètres du service.

Tableau 113 – Init DP-master CI1

Nom de paramètre	Req	Cnf
Argument	M	
Bus Para	M	
TS	M	
Data rate	M	
TSL	M	
$minT_{SDR}$	M	
$maxT_{SDR}$	M	
TQUI	M	
TSET	M	
TTR	M	
G	M	
HSA	M	
max retry limit	M	
Result		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

Bus Para
Ce paramètre est constitué des sous-paramètres suivants:

TS
Ce paramètre contient l'adresse de DL de ce maître DP (Classe 1).

Data rate
Ce paramètre contient le débit de données de ce maître DP (Classe 1).

TSL
Ce paramètre contient la durée d'intervalle de temps pour ce maître DP (Classe 1).

$minT_{SDR}$
Ce paramètre contient le plus petit temps de retard de poste pour ce maître DP (Classe 1).

$maxT_{SDR}$
Ce paramètre contient le plus grand temps de retard de poste pour ce maître DP (Classe 1).

TQUI
Ce paramètre contient le temps d'extinction d'émetteur pour ce maître DP (Classe 1).

TSET
Ce paramètre contient le temps d'établissement pour ce maître DP (Classe 1).

TTR
Ce paramètre contient le temps de rotation cible pour ce maître DP (Classe 1).

G
Ce paramètre contient le facteur d'actualisation de liste GAP pour ce maître DP (Classe 1).

HSA
Ce paramètre contient l'adresse (DL) de poste la plus haute pour ce maître DP (Classe 1).

max retry limit
Ce paramètre contient le nombre maximal de répétitions de tentative pour ce maître DP (Classe 1).

Result
Ce paramètre indique que la demande de service a réussi.

6.2.5.3.24 DP-master CI1 started

Dans le cas d'une AR MS0, ce service de la couche application indique à l'utilisateur d'un maître DP (Classe 1) que l'esclave DP assigné est dans le mode "échange de données".

Dans le cas d'une AR MS1, ce service de la couche application indique que le traitement des demandes d'alarme pour cet esclave DP est possible.

Le Tableau 114 montre les paramètres du service.

Tableau 114 – DP-master CI1 started

Nom de paramètre	Ind
Argument	M
AREP	M
Alarm Limit	C

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée (Adresse d'esclave DP).

Alarm Limit
Dans le cas d'une AR MS1, ce paramètre indique le nombre maximal d'alarmes parallèles.

Les valeurs admissibles sont montrées dans le Tableau 115.

Tableau 115 – Alarm limit

Valeur	Signification
0	1 alarme de chaque type
1	2 alarmes au total
2	4 alarmes au total
3	8 alarmes au total
4	12 alarmes au total
5	16 alarmes au total
6	24 alarmes au total
7	32 alarmes au total

6.2.5.3.25 DP-master CI1 stopped

Ce service de la couche application indique à l'utilisateur d'un maître DP (Classe 1) que l'AR connexe a été arrêtée. Dans le cas d'une AR MS0, l'AR MS1 a été également arrêtée. Un nouveau démarrage est nécessaire pour redémarrer l'AR MS0 et l'AR MS1.

Le Tableau 116 montre les paramètres du service.

Tableau 116 – DP-master CI1 stopped

Nom de paramètre	Ind
Argument	M
AREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

6.2.5.3.26 Reset DP-master CI1

Par ce service, l'utilisateur d'un maître DP (Classe 1) réinitialise la couche application.

Le Tableau 117 montre les paramètres du service.

Tableau 117 – Reset DP-master CI1

Nom de paramètre	Req	Cnf
Argument	M	
Result		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service. Néanmoins, il n'y a pas de paramètres spécifiques dans ce service.

Result
Ce paramètre indique que la demande de service a réussi

6.2.5.3.27 DP-master CI1 fault

Par ce service, la couche application indique à l'utilisateur d'un maître DP (Classe 1) qu'une erreur fatale s'est produite et que toutes les AR ont été arrêtées. Une communication ultérieure est possible seulement après un Reset.req de l'utilisateur.

Le Tableau 118 montre les paramètres du service.

Tableau 118 – DP-master CI1 fault

Nom de paramètre	Ind
Argument	—

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service. Néanmoins, il n'y a pas de paramètres spécifiques dans ce service.

6.2.5.3.28 DP-master CI1 reject

Par ce service, la couche application rejette un précédent Read/ Write.req envoyé de l'utilisateur d'un maître DP (Classe 1) vers l'AR MS1, si l'AR MS1 a été forcée de s'arrêter.

Le Tableau 119 montre les paramètres du service.

Tableau 119 – DP-master CI1 reject

Nom de paramètre	Ind
Argument	M
AREP	M
Reason	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée (Address of DP-slave).

Reason
Le paramètre Status indique le succès ou l'échec du service.

Valeurs admissibles: REJ ABORT,REJ LE,REJ PS,REJ SE,REJ IV

6.2.5.3.29 Set mode DP-master CI1

Par ce service, l'utilisateur d'un maître DP (Classe 1) peut établir le mode de fonctionnement de l'AR MS0. De plus, l'utilisateur récupère des informations relatives à l'état réel du système de bus en cas d'erreurs.

Le Tableau 120 montre les paramètres du service.

Tableau 120 – Set mode DP-master CI1

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
USIF State	M	
Result(+)		M
AREP		M
Bus Accessible		M
Result(-)		
AREP		M
Bus Accessible		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

USIF State
La couche application fournit quatre modes de fonctionnement pour l'utilisateur:

OFFLINE
La communication vers tous les participants DP (maître DP et esclave DP) est arrêtée. La couche MAC de la DL locale est dans l'état hors ligne. La couche application attend un signal pour démarrer.

STOP
En changeant le mode de fonctionnement de OFFLINE à STOP, le jeu de paramètres de bus est chargé dans la DL par la couche application. La DL était activée. Les services de répondeur pour l'AR MM1 et l'AR MM2 peuvent être traités. Les AR MS0 et MS1 ne sont pas activées (aucune interaction aux esclaves DP).

CLEAR
Dans ce mode de fonctionnement, le maître DP (Classe 1) tente de configurer les esclaves DP qui lui sont assignés et démarre le mode "échange de données". Les données d'entrée des esclaves DP sont lues et transmises à l'utilisateur. Les données de sortie utilisateur sont ignorées. Les esclaves DP assignés reçoivent une valeur 0 ou aucune donnée pour leurs objets Output Data. (en fonction de l'attribut Fail Safe).

OPERATE
Le maître DP (Classe 1) est dans le mode "échange de données" avec les esclaves DP assignés. Les valeurs d'entrée provenant des esclaves DP sont transférées à l'utilisateur et les valeurs de sortie provenant de l'utilisateur sont transmises aux esclaves DP. Si la couche application quitte le mode OPERATE, toutes les sorties des esclaves DP changent dans l'état de sécurité au moyen du service Global Control (Control Command = Clear Data, Group Select = 0).

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Bus Accessible
Ce paramètre Boolean indique si le maître DP est capable d'accéder au bus (TRUE) ou non (FALSE).

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

6.2.5.3.30 DP-master CI1 mode changed

Par ce service, la couche application indique à l'utilisateur d'un maître DP (Classe 1):

— le changement de son mode de fonctionnement de OPERATE à CLEAR si une erreur s'est produite et l'attribut Error Action Flag est mis,

— le changement de son mode de fonctionnement de STOP à OFFLINE si une erreur s'est produite pendant l'établissement des paramètres de bus.

Le Tableau 121 montre les paramètres du service.

Tableau 121 – DP-master CI1 mode changed

Nom de paramètre	Ind
Argument	M
AREP	M
USIF State	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

USIF State
Valeurs admissibles: OFFLINE, CLEAR (voir 6.2.5.3.29)

6.2.5.3.31 Load bus Par DP-master CI1

Par ce service, l'utilisateur d'un maître DP (Classe 1) peut charger un nouveau jeu de paramètres de bus. Ce service n'est pas autorisé dans le mode de fonctionnement OFFLINE. Dans le mode STOP, les changements apportés à tous les paramètres de bus sont possibles. Dans les modes de fonctionnement CLEAR et OPERATE de la couche application, il est interdit de charger un jeu de paramètres de bus avec des changements apportés aux paramètres DL critiques variables Data_rate et TS.

Les paramètres de bus modifiés ne sont pas stockés. Donc, tous les paramètres de ce service sont perdus après un changement de STOP à OFFLINE du mode de fonctionnement de la couche application. Les changements à stocker peuvent seulement être effectués au sein du jeu de paramètres de bus (Bus Para) du service Init.

Le Tableau 122 montre les paramètres du service.

Tableau 122 – Load bus Par DP-master CI1

Nom de paramètre	Req	Cnf
Argument	M	
Bus Para	M	
Result(+)		S
Status		M
Result(-)		S
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Bus Para
Ce paramètre contient le jeu de paramètres de bus qui doit être chargé. La structure est définie dans le service Master CI1 Init.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Le paramètre Status indique la raison du succès ou de l'échec du service.

Valeurs admissibles: NO, IV

6.2.5.3.32 Mark DP-master CI1

Par ce service, l'utilisateur d'un maître DP (Classe 1) peut fonctionner en synchronisme avec l'AR MS0. Après que l'utilisateur a envoyé ce service vers la couche application, l'utilisateur recevra un acquittement de la part de la couche application après que tous les esclaves DP activés auront été sondés au moins une fois.

Le Tableau 123 montre les paramètres du service.

Tableau 123 – Mark DP-master CI1

Nom de paramètre	Req	Cnf
Argument	M	
AREP	M	
Result(+)		S
AREP		M
Dia		M
Result(-)		S
AREP		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Dia
Ce paramètre indique (si la valeur est égale à TRUE) si le maître DP n'a pas exécuté une séquence d'échange de données vers au moins un esclave DP activé.

Valeurs admissibles: TRUE, FALSE

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Le paramètre Status indique le succès ou l'échec du service. La valeur NO est signalée si la couche application est dans le mode de fonctionnement OFFLINE ou STOP ou une précédente Mark.req est encore active.

Valeurs admissibles: NO

6.2.5.3.33 Abort DP-master Cl1

Ce service permet de libérer une AR MS0, MS1 ou MM1 d'un maître DP (Classe 1). L'indication peut être émise par l'AR MS1 si l'esclave DP répond de manière incorrecte. Dans ce cas, les CR de MS0 vers ces esclaves DP sont libérées aussi.

Le Tableau 124 montre les paramètres du service.

Tableau 124 – Abort DP-master Cl1

Nom de paramètre	Req	Cnf
Argument	M	M
AREP	M	M

Argument

L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

6.2.5.3.34 Read value DP-master Cl1

L'utilisateur d'un maître DP (Classe 1) lit n'importe lesquelles des variables de DL avec ce service.

Le Tableau 125 montre les paramètres du service.

Tableau 125 – Read value DP-master Cl1

Nom de paramètre	Req	Cnf
Argument	M	
Variable	M	
Result		M
Value		M
Status		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

Variable
Le paramètre Variable sélection la Variable de DL dont la valeur courante est à lire (voir la CEI 61158-3-3 pour une liste de toutes les variables de DL possibles).

Result
Ce paramètre indique que la demande de service a réussi

Value
Ce paramètre contient la valeur réelle de la Variable de DL lue.

Status
Le paramètre Status indique le succès ou l'échec du service.

Valeurs admissibles: OK, NO, IV

6.2.5.3.35 Delete SC DP-master CI1

Par ce service, les compteurs statistiques de DL peuvent être effacés (mis à zéro) par l'utilisateur d'un maître DP (Classe 1).

Le Tableau 126 montre les paramètres du service.

Tableau 126 – Delete SC DP-master CI1

Nom de paramètre	Req	Cnf
Argument	M	
Address	M	
Result		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

Address
Ce paramètre spécifie l'adresse de DL des compteurs statistiques à effacer.

Type d'attribut Unsigned8

Valeurs admissibles: 0 à 125

Result
Ce paramètre indique que la demande de service a réussi.

6.2.5.3.36 DP-master CI1 event

Par ce service, les événements du service DLM-Event sont indiqués à l'utilisateur d'un maître DP (Classe 1) comme un Event (événement).

Le Tableau 127 montre les paramètres du service.

Tableau 127 – DP-master CI1 event

Nom de paramètre	Ind
Argument	M
Event	M
Add Info	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

Event
Ce paramètre peut prendre les valeurs qui sont prévues pour le service DLM-Event.

Add Info
Le paramètre indique les informations ajoutées de la DL.

6.2.5.3.37 Init DP-master CI2

Par ce service, l'utilisateur initialise le paramètre de DL d'un maître DP (Classe 2).

Le Tableau 128 montre les paramètres du service.

Tableau 128 – Init DP-master CI2

Nom de paramètre	Req	Cnf
Argument	M	
Bus Para	M	
TS	M	
Data rate	M	
TSL	M	
$minT_{SDR}$	M	
$maxT_{SDR}$	M	
TQUI	M	
TSET	M	
TTR	M	
G	M	
HSA	M	
max retry limit	M	
Result		M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

Bus Para
Ce paramètre est constitué des sous-paramètres suivants:

TS
Ce paramètre contient l'adresse de DL de ce maître DP (Classe 2).

Data rate
Ce paramètre contient le débit de données de ce maître DP (Classe 2).

TSL
Ce paramètre contient la durée d'intervalle de temps pour ce maître DP (Classe 2).

$minT_{SDR}$
Ce paramètre contient le plus petit temps de retard de poste pour ce maître DP (Classe 2).

$maxT_{SDR}$
Ce paramètre contient le plus grand temps de retard de poste pour ce maître DP (Classe 2).

TQUI
Ce paramètre contient le temps d'extinction d'émetteur pour ce maître DP (Classe 2).

TSET
Ce paramètre contient le temps d'établissement pour ce maître DP (Classe 2).

TTR
Ce paramètre contient le temps de rotation cible pour ce maître DP (Classe 2).

G
Ce paramètre contient le facteur d'actualisation de liste GAP pour ce maître DP (Classe 2).

HSA
Ce paramètre contient l'adresse (DL) de poste la plus haute pour ce maître DP (Classe 2).

max retry limit
Ce paramètre contient le nombre maximal de répétitions de tentative pour ce maître DP (Classe 2).

Result

Ce paramètre indique que la demande de service a réussi.

6.2.5.3.38 Reset DP-master CI2

Par ce service, l'utilisateur d'un maître DP (Classe 2) réinitialise la couche application.

Le Tableau 129 montre les paramètres du service.

Tableau 129 – Reset DP-master CI2

Nom de paramètre	Req	Cnf
Argument		

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service. Néanmoins, il n'y a pas de paramètres spécifiques dans ce service.

6.2.5.3.39 DP-master CI2 fault

Par ce service, la couche application indique une erreur DL fatale à l'utilisateur d'un maître DP (Classe 2).

Le Tableau 130 montre les paramètres du service.

Tableau 130 – DP-master CI2 fault

Nom de paramètre	Ind
Argument	

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service. Néanmoins, il n'y a pas de paramètres spécifiques dans ce service.

6.2.5.3.40 DP-master CI2 reject

Par ce service, la couche application rejette un précédent service Read, Write, Data Transport issu de l'utilisateur d'un maître DP (Classe 2).

Le Tableau 131 montre les paramètres du service.

Tableau 131 – DP-master CI2 reject

Nom de paramètre	Ind
Argument	M
AREP	M
Reason	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée (Adresse d'esclave DP).

Reason
Le paramètre Status indique le succès ou l'échec du service.

Valeurs admissibles: REJ ABORT,REJ LE,REJ PS,REJ SE,REJ IV

6.2.5.3.41 DP-master CI2 closed

Par ce service, la couche application informe l'utilisateur d'un maître DP (Classe 2) que l'AR MS2 a été fermée.

Le Tableau 132 montre les paramètres du service.

Tableau 132 – DP-master CI2 closed

Nom de paramètre	Ind
Argument	M
AREP	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

6.2.5.3.42 DP-master CI2 event

Par ce service, les événements de la DLM sont indiqués à l'utilisateur d'un maître DP (Classe 2) comme un Event (événement).

Le Tableau 133 montre les paramètres du service.

Tableau 133 – DP-master CI2 event

Nom de paramètre	Ind
Argument	M
Event	M
Add Info	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

Event
Ce paramètre peut prendre les valeurs qui sont prévues pour le service DLM-Event.

Add Info
Le paramètre indique les informations ajoutées de la DL.

6.2.6 ASE Management

6.2.6.1 Vue d'ensemble

L'ASE Management est utilisé pour acheminer des paramètres entre un maître DP (Classe 2) et un maître DP (Classe 1). De plus, le maître DP (Classe 2) peut lire des informations de diagnostic du maître DP (Classe 1) et de ses esclaves DP assignés. L'accès du maître DP (Classe 2) au maître DP (Classe 1) est accompli conformément au modèle d'accès Client/Serveur.

6.2.6.2 Spécification de la classe Management

6.2.6.2.1 Spécification de la classe Master diag

6.2.6.2.1.1 Modèle

Un objet Master Diag est décrit par le modèle suivant:

ASE DP:			**Management ASE**
CLASS:			**Master Diag**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	Master Status
2.1.	(m)	Attribut:	USIF State
2.2.	(m)	Attribut:	Ident Number
2.3.	(m)	Attribut:	Hardware Release DP
2.4.	(m)	Attribut:	Firmware Release DP
2.5.	(m)	Attribut:	Hardware Release User
2.6.	(m)	Attribut:	Firmware Release User
3.	(m)	Attribut:	System Diagnosis
4.	(m)	Attribut:	Data Transfer List
5.	(o)	Attribut:	List of DP-slave Diag
5.1	(m)	Attribut:	DP-slave Address
5.2	(m)	Attribut:	DP-slave Diag
SERVICES:			
1.	(o)	OpsService:	Get Master Diag

6.2.6.2.1.2 Attributs

Implicit
L'attribut "Implicit" indique que l'objet Master Diag est adressé de façon implicite par le service.

Master Status
Ce paramètre est constitué des éléments suivants.

USIF State
Cet attribut décrit l'état d'un maître DP (Classe 1).

Les valeurs admissibles sont montrées dans le Tableau 134.

Tableau 134 – USIF state

Valeur	Signification
0	OFFLINE
1	STOP Dans cet état, le maître DP (Classe 1) n'exécute aucun service au niveau des AR MS0 et MS1. Les AR MM1 et MM2 peuvent être établies.
2	CLEAR Le maître DP actionne l'AR MS0. Les données de sortie sont dans l'état de sécurité. Les services connexes à l'AR MS1 peuvent être exécutés. Les AR MM1 et MM2 peuvent être établies.
3	OPERATE Le maître DP actionne l'AR MS0. Les données de sortie sont valides. Les services connexes à l'AR MS1 peuvent être exécutés. Les AR MM1 et MM2 peuvent être établies.

Ident Number
Cet attribut reflète l'Ident Number du maître DP (Classe 1).

Type d'attribut: Unsigned16

Hardware Release DP
Cet attribut spécifie la version matérielle de la mise en œuvre DP du maître DP (Classe 1).

Type d'attribut: Unsigned8

Firmware Release DP
Cet attribut spécifie la version firmware/logicielle de la mise en œuvre DP du maître DP (Classe 1).

Type d'attribut: Unsigned8

Hardware Release User
Cet attribut spécifie la version matérielle de la mise en œuvre utilisateur du maître DP (Classe 1).

Type d'attribut: Unsigned8

Firmware Release User
Cet attribut spécifie la version firmware/logicielle de la mise en œuvre utilisateur du maître DP (Classe 1).

Type d'attribut: Unsigned8

System Diagnosis
Le diagnostic système reflète l'état de tous les esclaves DP assignés au maître DP (Classe 1). Cet état lui-même est une information d'un bit pour chaque esclave DP qui est mise à la valeur 1 si son attribut Ext Diag Flag des informations de diagnostic est mis ou s'il ne répond pas ou s'il répond incorrectement et, de ce fait, l'AR MS0 est à rétablir. L'état d'un esclave DP est réinitialisé à la valeur 0 après que l'AR MS0 est rétablie avec succès ou si l'esclave DP est désactivé par l'AP du maître DP (Classe 1) ou si l'USIF State du maître DP (Classe 1) est STOP.

Type d'attribut: List of 0,1

Data Transfer List
L'attribut Data Transfer List reflète le statut du mode "échange de données" de tous les esclaves DP assignés au maître DP (Classe 1).

Ce statut lui-même est une information d'un bit pour chaque esclave DP qui est mise à la valeur 1 si l'échange de données avec l'esclave DP assigné a réussi ou si entre deux demandes d'échange de données il n'y a qu'une seule demande de diagnostic. Tous les autres cas réinitialisent l'information bit de statut de l'esclave DP à la valeur 0.

Type d'attribut: List of 0,1

List of DP-slave Diag
Cet attribut est constitué des éléments de liste suivants:

DP-slave Address
Cet attribut spécifie l'adresse de DL de l'esclave DP.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 125

DP-slave Diag
Cet attribut reflète les informations de diagnostic de l'esclave DP, y compris les informations ajoutées par les couches d'application DP pour chaque esclave DP.

6.2.6.2.1.3 Invocation de l'objet master diag

Pour l'invocation de l'objet Master Diag, les règles suivantes s'appliquent:

Un seul objet Master Diag doit être invoqué dans un maître DP (Classe 1).

6.2.6.2.2 Spécification de la classe Master parameter

6.2.6.2.2.1 Spécification de vue d'ensemble

L'objet Master Parameter est constitué d'un élément Bus Para et d'un ou plusieurs éléments Slave Para. Les valeurs pour l'objet Master Parameter peuvent être chargées par l'intermédiaire de l'AR MM1.

Le Download (téléchargement vers l'aval) des valeurs d'attribut pour l'objet Master Parameter peut être effectué en une seule séquence ou bien les valeurs d'attribut pour l'élément Bus Para et celles pour les éléments Slave Para peuvent être chargées séparément.

Pour le téléchargement vers l'aval des valeurs d'attribut pour l'objet Master Parameter, il faut que la couche application soit dans le mode de fonctionnement STOP. Le mode de fonctionnement de la couche application du maître DP (Classe 1) doit être changé avec le service Act Param par le biais de l'AR MM1 conjointement avec le service local Set Mode DP-master CI1.

Si les valeurs d'attribut pour l'objet Master Parameter sont chargées en une seule séquence, il est nécessaire que les valeurs soient d'abord stockées dans un tampon intermédiaire. Après réception du service Act Param ou Act Para Brct, l'utilisateur doit transférer les valeurs du tampon intermédiaire aux attributs de l'objet Master Parameter. Ensuite, l'utilisateur doit transférer ces valeurs en conséquence avec les services Load ARL, Load CRL (le paramètre Update mis à FALSE) et Load Bus Par DP-master CI1. Le service Act Param doit être confirmé conformément aux règles suivantes:

— positivement, si tous les services de chargement sont confirmés positivement,

— négativement, si un ou plusieurs services de chargement sont confirmés négativement.

Si les valeurs d'attribut pour les éléments Bus Para sont chargées, les valeurs doivent d'abord être stockées dans un tampon intermédiaire. Après réception du service Act Param ou Act Para Brct, l'utilisateur doit transférer, avec le service Load Bus Par DP-master CI1, les valeurs du tampon intermédiaire vers la couche application DP. Le service Act Param doit être confirmé positivement, si le service de chargement demandé précédemment a été confirmé positivement et après que les valeurs ont été transférées aux attributs de l'objet Master Parameter.

Le service Act Param doit être confirmé négativement, si le service de chargement demandé précédemment a été confirmé négativement. Dans ce cas, les valeurs émises précédemment pour les éléments Bus Para doivent être omises.

Si les valeurs d'attribut pour un élément Slave Para sont chargées, il est nécessaire que les valeurs soient d'abord stockées dans un tampon intermédiaire. Après réception du service Act Param, l'utilisateur est tenu de transférer les valeurs du tampon intermédiaire aux attributs de l'objet Master Parameter.

Ensuite, l'utilisateur doit transférer ces valeurs en conséquence avec les services ARL Slave Update DP-master CI1 et le Load CRL (le paramètre Update mis à TRUE). Le service Act Param doit être confirmé conformément aux règles suivantes:

— positivement, si tous les services de chargement sont confirmés positivement,

— négativement, si un ou plusieurs services de chargement sont confirmés négativement.

Des précautions doivent être prises par l'utilisateur, car la mise à jour d'un élément Slave Para peut interférer avec un Bus Para déjà chargé et peut modifier le comportement du système.

6.2.6.2.2.2 Modèle

Un objet Master Parameter est décrit par le modèle suivant:

ASE DP:			**ASE Management**
CLASS:			**Master Parameter**
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Implicit
2.	(m)	Attribut:	Bus Para
2.1.	(m)	Attribut:	Bus Para Len
2.2.	(m)	Attribut:	TS
2.3.	(m)	Attribut:	Data rate
2.4.	(m)	Attribut:	T_{SL}
2.5.	(m)	Attribut:	min T_{SDR}
2.6.	(m)	Attribut:	max T_{SDR}
2.7.	(m)	Attribut:	T_{QUI}
2.8.	(m)	Attribut:	T_{SET}
2.9.	(m)	Attribut:	T_{TR}
2.10.	(m)	Attribut:	G
2.11.	(m)	Attribut:	HSA
2.12.	(m)	Attribut:	max retry limit
2.13.	(m)	Attribut:	Error Action Flag
2.14.	(o)	Attribut:	Isochronous Mode
2.15.	(o)	Attribut:	IsoM Freeze
2.16.	(o)	Attribut:	IsoM Sync
2.17.	(m)	Attribut:	Min Slave Interval
2.18.	(m)	Attribut:	Poll Timeout
2.19.	(m)	Attribut:	Data Control Time
2.20.	(o)	Attribut:	Alarm Max
2.21.	(m)	Attribut:	Max User Global Control
2.22.	(m)	Attribut:	Master User Data Len
2.23.	(m)	Attribut:	Master Class2 Name
2.24.	(m)	Attribut:	Master User Data

2.25.	(o)	Attribut:	T_{CT}
2.26.	(o)	Attribut:	maxT_{SH}
3.	(o)	Attribut:	List of Slave Para
3.1.	(m)	Attribut:	Slave DL-address
3.2.	(m)	Attribut:	Slave Para Len
3.3.	(o)	Attribut:	Extra Alarm SAP
3.4.	(o)	Attribut:	DPV1 Data Types
3.5.	(o)	Attribut:	DPV1 Supported
3.6.	(m)	Attribut:	Fail Safe
3.7.	(m)	Attribut:	Active
3.8.	(m)	Attribut:	Slave Type
3.9.	(m)	Attribut:	New Prm
3.10.	(m)	Attribut:	Max Diag Data Len
3.11.	(o)	Attribut:	Max Channel Data Length
3.12.	(m)	Attribut:	Diag Upd Delay
3.13.	(o)	Attribut:	Alarm Mode
3.14.	(m)	Attribut:	NA to Abort
3.15.	(m)	Attribut:	Ignore AClr
3.16.	(o)	Attribut:	Publisher Flag
3.17.	(o)	Attribut:	MS1 Timeout
3.18.	(o)	Attribut:	Prm Command supported
3.19.	(m)	Attribut:	Prm Data Len
3.20.	(m)	Attribut:	Prm Data
3.21.	(m)	Attribut:	Cfg Data Len
3.22.	(m)	Attribut:	Cfg Data
3.23.	(m)	Attribut:	Add Tab Len
3.24.	(m)	Attribut:	Add Tab
3.25.	(m)	Attribut:	Slave User Data Len
3.26.	(o)	Attribut:	Slave User Data
3.27.	(o)	Attribut:	Ext User Prm Data Len
3.28.	(o)	Attribut:	Ext User Prm Data
3.29.	(m)	Attribut:	List of assigned CREPs for the MS0 AR
3.29.1.	(m)	Attribut:	CREP
3.30.	(m)	Attribut:	Assigned AREP for the MS1 AR
3.31.	(m)	Attribut:	Assigned AREP for the MS3 AR
4.	(m)	Attribut:	Assigned AREP for the MS0 AR

SERVICES:

1.	(o)	OpsService:	Start Seq
2.	(o)	OpsService:	Download
3.	(o)	OpsService:	Upload
4.	(o)	OpsService:	End Seq
5.	(o)	OpsService:	Act Para Brct
6.	(o)	OpsService:	Act Param

6.2.6.2.2.3 Attributs

Implicit
Cet attribut indique que l'objet Master Parameter est adressé de façon implicite par les services.

Bus Para
Cet attribut combiné est constitué des éléments suivants.

Bus Para Len
Cet attribut contient la longueur de Bus Para, y compris le paramètre de longueur "length".

Type d'attribut: Unsigned16

Valeurs admissibles: 66 à $2^{16} - 1$

TS
Cet attribut contient la propre adresse de DL du maître DP.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 125

Data rate
Cet attribut contient le numéro de code du débit de données sélectionné.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 135.

Tableau 135 – Data rate

Valeur	Signification
0	9,6 kbit/s
1	19,2 kbit/s
2	93,75 kbit/s
3	187,5 kbit/s
4	500 kbit/s
6	1 500 kbit/s
7	3 000 kbit/s
8	6 000 kbit/s
9	12 000 kbit/s
10	31,25 kbit/s
11	45,45 kbit/s

T_{SL}, min T_{SDR}, max T_{SDR}, T_{QUI}, T_{SET}, T_{TR}, G, HSA, max retry limit
Ces attributs sont décrits dans la CEI 61158-3-3 et dans la CEI 61158-4-3.

Error Action Flag
Cet attribut reflète la réaction du maître DP (Classe 1) en cas de transferts de données défaillants. Cet attribut est évalué dans les modes de fonctionnement OPERATE et CLEAR du maître DP (Classe 1). Si le transfert de données vers au moins un esclave DP activé était impossible pendant une certaine durée (Data Control Time) et si l'attribut Error Action Flag est mis, le mode de fonctionnement changera de OPERATE à CLEAR. Si l'attribut Error Action Flag est effacé, le maître DP (Classe 1) restera dans le mode de fonctionnement OPERATE malgré des transferts erronés de données.

Si le mode de fonctionnement du maître DP (Classe 1) doit être changé de CLEAR à OPERATE avec l'attribut Error Action Flag mis à TRUE, les transferts de données vers tous les esclaves DP activés sont à accomplir avec succès pendant une certaine durée (Data Control Time)

Les valeurs admissibles sont montrées dans le Tableau 136.

Tableau 136 – USIF state

Valeur	Signification
0	pas de changement du mode de fonctionnement en cas d'erreur
1	changement du mode de fonctionnement en cas d'erreur

Isochronous Mode
Cet attribut indique si le maître DP (Classe 1) doit fonctionner dans le mode isochrone (Isochronous Mode). Deux modèles différents concernant l'interface entre la couche application DP et l'application du maître DP (Classe 1) lui-même sont fournis par les DP de bus de terrain pour le maître DP (Classe 1), voir 6.2.2.4.3.2)

Les valeurs admissibles sont montrées dans le Tableau 137.

Tableau 137 – Isochronous mode

Valeur	Signification
Non synchronisé	Le maître DP (Classe 1) ne doit pas fonctionner dans le mode Isochronous Mode
Buffered Synchronized (Tamponné synchronisé)	Le maître DP (Classe 1) ne doit pas fonctionner dans le mode Buffer Synchronized Isochronous Mode
Enhanced Synchronized (Renforcé synchronisé)	Le maître DP (Classe 1) ne doit pas fonctionner dans le mode Enhanced Synchronized Isochronous Mode

IsoM Freeze
Cet attribut Boolean indique si le message de synchronisation (SYNCH) doit être envoyé avec le paramètre Control Command Freeze mis à TRUE.

Valeurs admissibles: TRUE, FALSE

IsoM Sync
Cet attribut Boolean indique si le message de synchronisation (SYNCH) doit être envoyé avec le paramètre Control Command Sync mis à TRUE.

Valeurs admissibles: TRUE, FALSE

Min Slave Interval
Cet attribut spécifie la plus petite durée admissible entre deux cycles consécutifs de sondage d'esclave DP. Il assure que la séquence de demandes de services cycliques issue du maître DP (Classe 1) peut être traitée par l'esclave DP. Cette durée est satisfaite par le maître DP (Classe 1) pour chaque service cyclique Maître-esclave, avec l'exception du service Global_Control. Pour le service Global_Control, l'utilisateur a la responsabilité de la conformité au Min Slave Interval.

Type d'attribut: Unsigned16

Valeurs admissibles: 1 à 2^{16} - 1

Base de temps: 100 µs

Poll Timeout
Dans le cas de l'AR MM1, cet attribut spécifie la durée maximale nécessaire pour récupérer la réponse.

Type d'attribut: Unsigned16

Valeurs admissibles: 1 à 2^{16} - 1

Base de temps: 1

Data Control Time
Cet attribut spécifie la durée maximale pour un échange de données avec chaque esclave DP activé. Ce temps est utilisé pour envoyer le service Global Control de façon cyclique.

Type d'attribut: Unsigned16

Valeurs admissibles: 1 à 2^{16} - 1

Base de temps: 10

Alarm Max
Cette variable contient le nombre maximal d'alarmes par esclave DP, qui peuvent être traitées par le maître DP. Il décrit une capacité du Maître. Le Maître doit être capable de traiter un nombre minimum de sept alarmes, car il existe sept classes d'alarmes différentes.

Type d'attribut: Unsigned8

Valeurs admissibles: 7 à 32

Max User Global Control
Cet attribut définit le nombre maximal de demandes de Global Control qui peuvent être lancées simultanément par l'Utilisateur. Ce paramètre décrit l'aptitude du maître DP (Classe 1). Une valeur commode dans la pratique pour Max User Global Control est 16. Pour chacun des 8 groupes d'esclaves, une commande Sync Command et une commande Freeze Command peuvent être lancées simultanément.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 255

Master User Data Len
Cet attribut contient la longueur de Master_User_Data, y compris le paramètre de longueur "length".

Type d'attribut: Unsigned16

Valeurs admissibles: 34 à 2^{16} - 1

Master Class2 Name
Cet attribut indique le nom du maître DP (Classe 2) qui a fourni les valeurs pour les attributs de cet objet Master Parameter.

Type d'attribut: Visible String(32)

Master User Data
Cet attribut contient des données spécifiques à un fabricant qui sont nécessaires pour le jeu de paramètres de bus.

Type d'attribut: Octet String

T_{CT}
Cet attribut spécifie le temps (en unités de t_{BIT}) qui est nécessaire pour traiter un cycle Isochronous DP complet et correspond au temps T_{DP}.

Valeurs admissibles: 1 à 2^{24} - 1

maxT_{SH}
Cet attribut spécifie le décalage temporel maximal admissible (en unités de t_{BIT}) du temps T_{DP} mesuré par le maître DP (Classe 1). Si la durée de cycle Isochronous DP mesuré dépasse cette limite, l'application du maître DP (Classe 1) est informée par le service SYNCH Delayed indiquant le décalage survenu.

Valeurs admissibles: 1 à 2^{8} - 1

Slave Para
Cet attribut combiné est constitué des éléments suivants.

Slave DL-address
Cet attribut reflète l'adresse de DL de l'esclave DP. Cette adresse est utilisée comme un identificateur (Area Code) pour adresser le jeu de paramètres d'esclave correspondant, par exemple dans le but d'un téléchargement aval ou amont.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 125

Slave Para Len
Cet attribut contient la longueur de Slave Para, y compris le paramètre de longueur "length". Un jeu de paramètres d'esclave peut être supprimé en mettant le Slave Para Len à zéro.

Type d'attribut: Unsigned16

Valeurs admissibles: 0 à 2^{16} - 1

Extra Alarm DLSAP
Cet attribut Boolean sélectionne la voie pour le service Alarm Ack.

– Si l'attribut est mis à FALSE, le maître DP (Classe 1) acquitte les alarmes par le biais de DLSAP 51.
– Si l'attribut est mis à TRUE, le maître DP (Classe 1) acquitte les alarmes par le biais de DLSAP 50.

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

DPV1 Data Types
Cet attribut Boolean reflète si, oui ou non, l'esclave DP utilise des types de données.

– Si l'attribut est mis à FALSE, l'esclave DP utilise seulement des types de base (Octet, Word, Array) pour l'attribut Data Description.
– Si l'attribut est mis à TRUE, l'esclave DP utilise des types de données pour l'attribut Data Description.

Type d'attribut: Boolean

DPV1 Supported
Cet attribut Boolean reflète si, oui ou non, l'esclave DP utilise la fonctionnalité étendue.

– Si l'attribut est mis à FALSE, l'esclave DP n'utilise aucune fonctionnalité étendue.
– Si l'attribut est mis à TRUE, l'esclave DP utilise la fonctionnalité étendue.

Type d'attribut: Boolean

Fail Safe
Cet attribut Boolean reflète si, oui ou non, l'esclave DP prend en charge la fonctionnalité Fail Safe (sécurité intégrée). Si l'attribut est mis à FALSE, l'esclave DP ne prend pas en charge la fonctionnalité Fail Safe. Si l'attribut est mis à TRUE, l'esclave DP prend en charge la fonctionnalité Fail Safe.

Type d'attribut: Boolean

Active
Cet attribut Boolean reflète si, oui ou non, l'esclave DP est à activer. Si l'attribut Active a la valeur FALSE, le maître DP (Classe 1) arrête le traitement d'esclave DP à cet esclave DP assigné. Dans cet état, il est permis de changer le jeu de paramètres d'esclave, par exemple par un maître DP (Classe 2) avec le service Download. Si l'attribut Active-Flag a la valeur TRUE, le traitement de l'esclave DP est effectué par le maître DP (Classe 1) à cet esclave DP assigné. Le changement du jeu de paramètres d'esclave dans cet état est limité (voir aussi la description de l'attribut New Prm).

Type d'attribut: Boolean

Slave Type
Cet attribut contient une dénotation de types spécifiques à un fabricant pour l'esclave DP (voir le Tableau 138).

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 138.

Tableau 138 – Slave type

Valeur	Signification
0	esclave DP
1-15	réservé
16-255	spécifique à un fabricant

New Prm
Cet attribut Boolean reflète si, oui ou non, les données de paramétrisation sont à transférer vers l'esclave DP dans le mode "échange de données". Si cet attribut est mis à FALSE, les données de sortie sont transférées vers l'esclave DP dans le mode "échange de données". Si cet attribut est mis à TRUE, pour un cycle de données, les données de paramétrisation sont transférées en lieu et place des données de sortie. Il est nécessaire de garantir la cohérence des données de paramétrisation par des mesures locales. Après l'émission réussie de ces données de paramétrisation, le maître DP (Classe 1) met cet attribut à FALSE et le mode "échange de données" vers l'esclave DP est poursuivi.

Type d'attribut: Boolean

Max Diag Data Len
Cet attribut contient la longueur maximale de Diag Data pour l'esclave DP.

Type d'attribut: Unsigned8

Valeurs admissibles: 6 à 244

Max Channel Data Length
Cet attribut définit la longueur maximale d'une APDU MS1 (paramètre de longueur et en-tête de 4 octets) pour l'esclave DP correspondant.

Type d'attribut: Unsigned8

Valeurs admissibles: 4 à 244

Diag Upd Delay
L'attribut reflète le nombre de demandes de service Diag accomplies dans l'état DIAG2 du maître DP (Classe 1) alors que Prm Req est mis dans Diag Data de l'esclave DP (pour les esclaves DP à performance réduite).

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 15 (extensible jusqu'à 255)

Alarm Mode
Cet attribut spécifie le nombre maximal d'alarmes actives possibles.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 139.

Tableau 139 – Alarm mode

Valeur	Signification
0	1 alarme de chaque type
1	2 alarmes au total
2	4 alarmes au total
3	8 alarmes au total
4	12 alarmes au total
5	16 alarmes au total
6	24 alarmes au total
7	32 alarmes au total

NA To Abort
Cet attribut Boolean reflète le comportement du maître DP (Classe 1) si l'esclave DP assigné ne répond pas en l'AR MS0 dans l'état DIAG2 et le mode "échange de données".

– Si cet attribut est mis à FALSE, le maître DP continue en cas d'absence de réponse (Status = NA) de l'esclave DP assigné.
– Si cet attribut est mis à TRUE, le maître DP abandonne l'AR MS0 en cas d'absence de réponse (Status = NA) de l'esclave DP assigné ("abandonner" signifie envoyer une commande de déverrouillage et rétablir l'AR MS0).

Type d'attribut: Boolean

Ignore ACIr
Cet attribut Boolean en combinaison avec l'attribut Error Action Flag permet une réaction spécifique à un esclave DP du maître DP (Classe 1) en cas d'échec de transferts de données.

– Si l'attribut est mis à FALSE, le maître DP (Classe 1) traite la fonction d'autoeffacement telle que définie dans l'attribut Error Action Flag.
– Si l'attribut est mis à TRUE, le maître DP (Classe 1) ignore la fonction d'autoeffacement telle que définie dans l'attribut Error Action Flag.

Type d'attribut: Boolean

Publisher Flag
Cet attribut Boolean indique si l'esclave DP correspondant doit fonctionner comme Éditeur. Si l'attribut est mis à FALSE, l'esclave DP correspondant ne doit pas fonctionner comme Éditeur. Si l'attribut est mis à TRUE, l'esclave DP correspondant doit être adressé comme Éditeur par le maître DP (Classe 1).

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

MS1 Timeout
Cet attribut contient la valeur de Timeout de la communication acyclique sur l'AR MS1 AR et spécifie la durée maximale pendant laquelle un esclave DP peut fournir une réponse. Le temps contenu dans cet attribut est le temps maximal de l'esclave DP assigné.

Type d'attribut: Unsigned16

Valeurs admissibles: 1 à 2^{16} - 1

Base de temps: 10

Prm Command supported
Cet attribut Boolean reflète si, oui ou non, l'esclave DP utilise PrmCmd dans la fonctionnalité étendue.

– Si l'attribut est mis à FALSE, l'esclave DP ne prend en charge aucun PrmCmd.
– Si l'attribut est mis à TRUE, l'esclave DP prend en charge PrmCmd.

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

Prm Data Len
Cet attribut contient la longueur de Prm Data, y compris le paramètre de longueur "length".

Type d'attribut: Unsigned8

Valeurs admissibles: 9 à 246

Prm Data
Cet attribut reflète les données de paramétrisation envoyées vers l'esclave DP correspondant par le maître DP (Classe 1). La structure de cet attribut est décrite en 6.2.10.4.2.2.

Cfg Data Len
Cet attribut contient la longueur de Cfg Data, y compris le paramètre de longueur "length".

Type d'attribut: Unsigned8

Valeurs admissibles: 3 à 246

Cfg Data
Cet attribut reflète les informations de configuration envoyées à l'esclave DP. La structure de cet attribut est décrite en 6.2.10.4.2.2.

Add Tab Len
Cet attribut contient la longueur de Add_Tab, y compris le paramètre de longueur "length".

Type d'attribut: Unsigned16

Valeurs admissibles: 2 à 2^{16} - 31

Add Tab
Cet attribut contient la table d'affectation d'adresses de l'esclave DP. Dans le cas d'un automate programmable (Programmable Logic Controller), la liste contient les adresses d'automate programmable des adresses décentralisées.

Slave User Data Len
Cet attribut contient la longueur de Slave User Data, y compris le paramètre de longueur "length".

Type d'attribut: Unsigned16

Valeurs admissibles: 2 à 2^{16} - 31

Slave User Data
Cet attribut contient des données spécifiques au fabricant qui caractérisent l'esclave DP pour le maître DP (Classe 1).

Type d'attribut: Octet String

Ext User Prm Data Len
Cet attribut contient la longueur de Ext Prm Data, y compris le paramètre de longueur "length".

Type d'attribut: Unsigned8

Valeurs admissibles: 2 à 246

Ext User Prm Data
Cet attribut reflète les données de paramétrisation utilisateur étendues envoyées vers l'esclave DP correspondant par le maître DP (Classe 1). Le maître DP (Classe 1) enverra ces données entre les données de paramètre et les informations de configuration.

List of assigned CREPs for the MS0 AR
Cet attribut contient les CREP de l'AR MS0 assignée pour cet esclave DP et se compose des éléments de liste suivants:

CREP
Cet attribut contient un CREP de l'AR MS0.

Assigned AREP for the MS1 AR
Cet attribut contient l'AREP de l'AR MS1 assignée pour cet esclave DP.

Assigned AREP for the MS3 AR
Cet attribut contient l'AREP de l'AR MS3 assignée pour cet esclave DP.

Assigned AREP for the MS0 AR
Cet attribut contient l'AREP de l'AR MS0 assignée pour tous les esclaves DP.

6.2.6.2.2.4 Invocation de l'objet Master Parameter

Pour l'invocation de l'objet Master Parameter, les règles suivantes s'appliquent:

— Un seul objet Master Parameter peut être invoqué dans un maître DP (Classe 1).

— Jusqu'à 125 éléments Slave Para avec une valeur unique pour l'attribut Slave DL-address peuvent être contenus dans un objet Master Parameter d'un maître DP (Classe 1).

6.2.6.3 Spécification de services de Management

6.2.6.3.1 Get master diag

Avec ce service, un maître DP (Classe 2) peut lire un seul élément (Master Status, System Diagnosis, Data Transfer List, DP-slave Diag d'un seul esclave DP) dans l'objet Master Diag d'un maître DP (Classe 1). Ce service doit seulement être utilisé conjointement à l'AR MM1.

Le Tableau 140 montre les paramètres du service.

Tableau 140 – Get Master Diag

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
MDiag Identifier	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Diagnosis Data			M	M(=)
DP-slave Diag			S	S(=)
System Diagnosis			S	S(=)
Data Transfer List			S	S(=)
Master Status			S	S(=)
Result(-)			S	S(=)
AREP			M	M(=)
Status			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR souhaitée.

MDiag Identifier
Ce paramètre spécifie l'élément de l'objet Master Diag destiné à être lu.

Les valeurs admissibles sont montrées dans le Tableau 141.

Tableau 141 – MDiag identifier

Valeur	Signification
0-125	DP-slave Diag de l'esclave DP avec l'adresse de DL correspondante
126	System Diagnosis
127	Master Status
128	Data Transfer List

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Diagnosis Data
Ce paramètre contient l'un des éléments suivants de l'objet Master Diag tel que demandé.

DP-slave Diag
Ce paramètre contient le Diag Data de l'esclave DP adressé.

System Diagnosis
Ce paramètre contient le diagnostic système (System Diagnosis) de l'objet Master Diag (voir 6.2.6.2.1).

Master Status
Ce paramètre contient le Master Status (statut de maître) de l'objet Master Diag (voir 6.2.6.2.1).

Data Transfer List
Ce paramètre contient la Data Transfer List (Liste de transferts de données) de l'objet Master Diag (voir 6.2.6.2.1).

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre fournit des informations détaillées relatives à la défaillance.

Valeurs admissibles: DS, NA, RS, RR, UE, RE, TO, FE, NE, IP, AD, EA, LE

6.2.6.3.2 Start seq

Avec le service Start Seq, le début d'une séquence Upload ou Download est signalé par le maître DP (Classe 2) dans le maître DP (Classe 1). Dans ce cas, le maître DP (Classe 1) peut établir une protection d'accès pour la zone adressée jusqu'à ce que le service End Seq signale la fin de la séquence. Les données adressées peuvent être émises dans des blocs avec les services Upload et Download. Ce service doit seulement être utilisé conjointement à l'AR MM1.

Le Tableau 142 montre les paramètres du service.

Tableau 142 – Start Seq

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M(=)		
Timeout	M	M(=)		
Result(+)			S	S(=)
AREP			M	M
Max Len Data Unit			C	C
Result(-)			S	S(=)
AREP			M	M
Status			M	M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR souhaitée.

Area Code
Le paramètre Area Code identifie l'élément de l'objet Master Parameter qui doit être chargé ou lu. Si le paramètre Area Code est mis à la valeur de 255, la protection d'accès locale pour un Area Code déterminé n'est pas garantie pour les services qui suivent. La protection d'accès locale assure qu'au cours de la séquence de services, l'utilisateur du maître DP (Classe 1) ne peut pas accéder aux données partiellement chargées.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 143.

Tableau 143 – Area code (start seq)

Valeur	Signification
0 à 125	Slave Para de l'esclave DP avec l'adresse de DL correspondante
126	reserved (réservé)
127	Bus Para
128	reserved (réservé)
129	statistic counters (compteurs statistiques)
130 à 135	reserved (réservé)
136 à 139	pour le transfert du Master Parameter Set (jeu de paramètres de maître)
140 à 254	reserved (réservé)
255	Start Seq: Aucune protection d'accès locale

Timeout
Ce paramètre définit le temps de commande entre deux services Upload/Download successifs. Si ce temps expire, la protection d'accès est désactivée. Les données transférées précédemment sont non valides.

Type d'attribut: Unsigned16

Base de temps: 1

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Max Len Data Unit
Le paramètre Max Len Data Unit définit la longueur possible maximale du paramètre Data dans les APDU Upload/Download qui suivent.

Type d'attribut: Unsigned8

Plage: 1 à 240

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre fournit des informations détaillées relatives à la défaillance.

Valeurs admissibles: DS, NA, RS, RR, UE, TO, FE, RE, NE, AD, IP, NI, SE, SC, EA, LE

6.2.6.3.3 Download

Ce service est utilisé pour transférer les valeurs pour les attributs de l'objet Master Parameter, et ce, du maître DP (Classe 2) vers un maître DP (Classe 1). Le début de la séquence Download peut être signalé avec le service Start Seq. Le jeu réel de données est émis dans des blocs avec le service Download dans la zone adressée. Après que le transfert est parachevé, la séquence doit être terminée avec le service End Seq, si la séquence Download avait été lancée par le service Start Seq. Ce service doit seulement être utilisé conjointement à une AR MM1.

Le Tableau 144 montre les paramètres du service.

Tableau 144 – Download

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M		
Add Offset	M	M		
Data	M	M		
Result(+)			S	S(=)
AREP			M	M
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR souhaitée.

Area Code
Le paramètre Area Code identifie l'élément de l'objet Master Parameter qui doit être chargé.

Type d'attribut: Unsigned8

Valeurs admissibles: voir 6.2.6.3.2

Add Offset
Ce paramètre contient le décalage par rapport au début de la zone qui est spécifiée par le paramètre Area Code.

Type d'attribut: Unsigned16

Data
Ce paramètre contient les données à transférer.

Type d'attribut: Octet String

Longueur: 1 à 240

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre fournit des informations détaillées relatives à la défaillance.

Valeurs admissibles: DS, NA, RS, RR, UE, TO, FE, RE, NE, AD, EA, LE, SC, NI, NC

6.2.6.3.4 Upload

Ce service est utilisé pour transférer les valeurs des attributs de l'objet Master Parameter, et ce, du maître DP (Classe 1) vers un maître DP (Classe 2). La séquence Upload peut être exécutée de la même manière que la séquence Download. Il est possible de transférer des zones de données prises séparément sans les services Start Seq et End Seq. Dans ce cas, la cohérence des données transférées n'est pas garantie. Ce service doit seulement être utilisé conjointement à une AR MM1.

Le Tableau 145 montre les paramètres du service.

Tableau 145 – Upload

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M		
Add Offset	M	M		
Data Len	M	M		
Result(+)			S	S(=)
AREP			M	M
Data			C	C(=)
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR souhaitée.

Area Code
Le paramètre Area Code identifie l'élément de l'objet Master Parameter qui doit être lu.

Type d'attribut: Unsigned8

Valeurs admissibles: voir 6.2.6.3.2

Add Offset
Ce paramètre identifie le décalage par rapport au début de la zone qui est spécifiée par le paramètre Area Code.

Type d'attribut: Unsigned16

Data Len
Le paramètre définit la longueur des données demandées.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 240

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Data
Ce paramètre contient les données demandées.

Type d'attribut: Octet String

Length: 1 à 240

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre fournit des informations détaillées relatives à la défaillance.

Valeurs admissibles: DS, NA, RS, RR, UE, TO, FE, RE, NE, EA, LE, NI, SC, AD

6.2.6.3.5 End seq

Ce service est utilisé pour repérer la fin d'une séquence de transferts cohérents de blocs de données de l'objet Master Parameter. Ce service doit seulement être utilisé conjointement à une AR MM1.

Le Tableau 146 montre les paramètres du service.

Tableau 146 – End Seq

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Result(+)			S	S(=)
AREP			M	M
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR souhaitée.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre fournit des informations détaillées relatives à la défaillance.

Valeurs admissibles: DS, NA, RS, RR, UE, TO, FE, RE, NI, SE, NE, AD, EA, LE, NC

6.2.6.3.6 Act Para Brct

Ce service est utilisé pour activer les valeurs précédemment transférées pour l'objet Master Parameter par le maître DP (Classe 2). Ce service peut être accompli simultanément à un

(simple) ou plusieurs (multidiffusion) maître(s) DP (Classe 1). Dans les modes de fonctionnement CLEAR et OPERATE du maître DP (Classe 1), l'activation de l'élément Bus Para avec des paramètres de DL 'data rate' (débit de données) ou "TS" modifiés est rejetée. Ce service est seulement confirmé localement, car un changement possible du débit de données doit prendre effet simultanément en tous les maîtres DP (Classe 1) adressés. Ce service doit seulement être utilisé conjointement à une AR MM2.

Le Tableau 147 montre les paramètres du service.

Tableau 147 – Act Para Brct

Nom de paramètre	Req	Ind	Cnf
Argument	M	M	
AREP	M	M	
Area Code	M	M	
Result(+)			S
AREP			M
Result(-)			S
AREP			M
Status			M

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR souhaitée.

Area Code
Le paramètre Area Code identifie l'élément de l'objet Master Parameter qui doit être activé.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 148.

Tableau 148 – Area code (Act Para Brct)

Valeur	Signification
0 à 126	non autorisées
127	Bus Para
128 à 129	reserved (réservées)
130 à 135	reserved (réservées)
136 à 139	pour l'activation de l'objet Master Parameter
140 à 254	reserved (réservée)
255	non autorisée

En présence de plusieurs maîtres DP (Classe 2), il doit être garanti que les anciens et nouveaux jeux de paramètres de bus sont compatibles.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre fournit des informations détaillées relatives à la défaillance.

Valeurs admissibles: DS

6.2.6.3.7 Act param

Ce service est utilisé par le maître DP (Classe 2) pour les besoins suivants:

— activer ou désactiver un esclave DP paramétré,
— changer le mode de fonctionnement de la couche application DP,
— accepter et activer les valeurs pour les éléments Bus Para de l'objet Master Parameter.

Si le jeu de paramètres de bus chargé contient un débit de données modifié ou une adresse de poste modifiée pour le maître DP (Classe 1), le nouveau jeu de paramètres de bus doit être activé avec le service Act Para Brct. Ce service doit seulement être utilisé conjointement à une AR MM1.

Le Tableau 149 montre les paramètres du service.

Tableau 149 – Act param

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M		
AREP	M	M		
Area Code	M	M(=)		
Activate	M	M(=)		
Result(+)			S	S(=)
AREP			M	M
Result(-)			S	S(=)
AREP			M	M
Status			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local de l'AR souhaitée.

Area Code
Le paramètre Area Code identifie l'élément de l'objet Master Parameter qui doit être activé/désactivé, à l'exception du mode de fonctionnement.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 150.

Tableau 150 – Area code (Act param)

Valeur	Signification
0 à 125	Slave Para de l'esclave DP avec l'adresse de DL correspondante Le (Fanion/Bit) "Active" dans le jeu de paramètres d'esclave du maître DP (Classe 1) est mis comme spécifié dans le paramètre Activate.
126	Réservée
127	Bus Para
128	Mode de fonctionnement (pour la description, voir 6.2.5.3.29)
129	Réservée
130 à 135	reserved (réservées)
136 à 139	activation du jeu de paramètres de maître (Master Parameter Set)
140 à 255	reserved (réservées)

Activate
La signification et les valeurs admissibles de ce paramètre dépendent du paramètre Area Code.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 151.

Tableau 151 – Activate

Area Code	Valeurs admissibles pour le paramètre Activate	
0 à 125: (Slave Para)	0	(= Deactivate)
	128	(= Activate)
127: (Bus Para)	255	(= Activate)
128: (mode de fonctionnement)	64	(= STOP)
	128	(= CLEAR)
	192	(= OPERATE)

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Status
Ce paramètre fournit des informations détaillées relatives à la défaillance.

Valeurs admissibles: DS, NA, RS, RR, UE, TO, FE, RE, NE, AD, IP, SC, NI, DI, EA, LE

6.2.7 ASE Load Region

6.2.7.1 Vue d'ensemble

Un objet Load Region représente une zone de mémoire non structurée dont le contenu peut être téléchargé vers l'amont (Pull) ou téléchargé vers l'aval (Push). Dans ce contexte, "Non structuré" signifie que la zone de mémoire est représentée seulement comme étant une séquence ordonnée d'octets. Aucune autre structure n'est apparente.

Un objet Load Region peut représenter une zone de mémoire volatile anonyme (telle que celle mise en œuvre par une mémoire d'ordinateur dynamique) ou un objet nommé de mémoire non volatile (tel qu'un fichier). Le contenu d'un objet Load Region est appelé "load

image" (image de charge). Les images de charge peuvent contenir des programmes ou des données. Le transfert des LR Data à destination ou en provenance d'un objet Load Region est accompli en utilisant les services Load Region.

Les objets Load Region doivent être nettement séparés des objets Process Data et des objets Function Invocation. Aucun chevauchement n'est permis.

Pour maintenir l'intégrité, un seul processus de téléchargement vers l'aval pour un objet Load Region est permis à la fois. L'attribut Load Region State de l'objet Load Region est utilisé pour indiquer si un objet Load Region est vide, en train d'être téléchargé vers l'aval ou chargé (pour une liste complète des états, voir la définition de l'attribut). L'objet Load Region peut être effacé en utilisant le service Initiate Load.

L'attribut Load Region State indique également l'état d'avancement d'un téléchargement vers l'amont. Plusieurs téléchargements vers l'amont d'un objet Load Region par le biais d'AR différentes sont autorisés. Cet attribut est défini pour séparer l'état du contenu de l'objet Load Region du fonctionnement consistant à avoir le contenu téléchargé vers l'amont.

L'ASE Load Region fournit des services qui permettre de lancer le téléchargement vers l'amont ou le téléchargement vers l'aval (Pull ou Push) d'objets Load Region. Ils identifient l'objet Load Region et de tirer (Pull) ou pousser (Push) comme paramètres de la demande de service "Initiate load) (déclenchement de chargement).

Les LR Data peuvent être transférées en segments en utilisant le service Pull Segment ou le service Push Segment.

Les segments de LR Data sont téléchargés vers l'amont par le Client en émettant des demandes de service Pull Segment pour eux. L'AP qui contient les LR Data répond en retournant le segment demandé, en indiquant avec un paramètre le moment où le segment final des LR Data a été retourné.

Les segments de LR Data sont téléchargés vers l'aval par le Client ayant l'AP contenant l'image des LR Data en envoyant des segments individuels vers le Serveur et en attendant une réponse qui indique si, oui ou non, le segment de LR Data a été reçu. Dans ce cas, le Client indique le moment où le dernier segment de LR Data a été transféré. Après que le dernier segment de LR Data a été reçu, l'AP qui a lancé le transfert met fin au processus de chargement en émettant une demande de Terminate Load à l'AP distant. La demande de Terminate Load peut également être utilisée par un AP si le processus de chargement ne peut pas être achevé avec succès.

Pour prévenir les impasses au niveau du Serveur, une temporisation doit être émise pour commander les activités d'une AR concernant les procédures de téléchargement amont ou aval.

6.2.7.2 Spécification de la classe Load region

6.2.7.2.1 Modèle

L'objet Load Region est modélisé comme un objet Server (Serveur). L'objet Load Region spécifie les attributs et les services utilisés par les AP Client pour télécharger vers l'amont/aval les LR Data. La spécification de l'AP Client ne relève pas du domaine d'application de la présente spécification.

Un objet Load Region est décrit par le modèle suivant:

ASE DP:	**Load Region ASE**
CLASS:	Load Region
CLASS ID:	non utilisé
PARENT CLASS:	TOP

ATTRIBUTES:

1.	(m)	Attribut-clé:	Identifier
2	(m)	Attribut:	LR Size
3.	(m)	Attribut:	Access Rights MS1
4.	(m)	Attribut:	Access Rights MS2
5.	(c)	Contrainte:	Access Rights MS2 = Right to Pull ou Right to Pull/Push
5.1.	(m)	Attribut:	List of SCL Pull
5.1.1.	(m)	Attribut:	SCL
6.	(c)	Contrainte:	Access Rights MS2 = Right to Push ou Right to Pull/Push
6.1.	(m)	Attribut:	List of SCL Push
6.1.1.	(m)	Attribut:	SCL
7.	(m)	Attribut:	Load Region State
8.	(m)	Attribut:	Actual LR Size
9.	(o)	Attribut:	Identifier of Related Function Invocation
10.	(m)	Attribut:	Max Segment Length
11.	(m)	Attribut:	Max Response Delay
12.	(m)	Attribut:	Uploadcount
13.	(o)	Attribut:	Load Region Additional Information
14.	(o)	Attribut:	Load Region Name
15.	(o)	Attribut:	Local Detail

SERVICES:

1.	(m)	OpsService:	Initiate Load
2.	(o)	OpsService:	Push Segment
3.	(o)	OpsService:	Pull Segment
4.	(m)	OpsService:	Terminate Load

6.2.7.2.2 Attributs

Identifier
Cet attribut-clé est un triplet constitué de l'API, du Slot Number et du LR Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Load Region appartient. Cet Identifier ne doit pas être utilisé par un autre objet Action, un autre objet Load Region, un autre objet Function Invocation ou un autre objet Process Data.

Type d'attribut: Unsigned8, Unsigned8, Unsigned16

Valeurs admissibles du triplet: [0 à 254, 0 à 254, 0 à 65 199 (65 200 à 65 535 réservés)]

LR Size
Cet attribut spécifie la longueur de l'objet Load Region en octets.

Type d'attribut: Unsigned32

Valeurs admissibles: 1 à $2^{32} - 1$

Access Rights MS1
Cet attribut définit le type de droits d'accès définis pour une AR MS1 pour l'objet Load Region.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 152.

Tableau 152 – Access rights MS1

Valeur	Signification
0	Right to Push (Droit de pousser)
1	Right to Pull (Droit de tirer)
2	Right to Pull/Push (Droit de tirer/pousser)
3	No Access (Aucun accès)

Access Rights MS2
Cet attribut définit le type de droits d'accès définis pour une AR MS2 pour l'objet Load Region.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 153.

Tableau 153 – Access rights MS2

Valeur	Signification
0	Right to Push (Droit de pousser)
1	Right to Pull (Droit de tirer)
2	Right to Pull/Push (Droit de tirer/pousser)
3	No Access (Aucun accès)

List of SCL Pull
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur pour la protection d'accès Pull pour cet objet Load Region. Les règles d'acceptation ou de rejet d'un accès Pull sont décrites en 6.2.1.3 conformément à l'accès en lecture.

Type d'attribut: Unsigned8

List of SCL Push
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur pour la protection d'accès Push pour cet objet Load Region. Les règles d'acceptation ou de rejet d'un accès Push sont décrites en 6.2.1.3 conformément à la protection d'accès en écriture.

Type d'attribut: Unsigned8

Load Region State
Cet attribut spécifie l'état de l'objet Load Region.

Type d'attribut: Unsigned8

Les valeurs admissibles de cet attribut sont montrées dans le Tableau 154.

Tableau 154 – Load region state

Valeur	Signification
0	DOWNLOADABLE
1	DOWNLOADING
2	DOWNLOAD SUCCESS
3	LOADED
4	IN-USE

Contraintes de l'attribut Load Region State:

– Un objet Load Region dont le contenu est stocké dans une mémoire non effaçable ne peut pas prendre en charge des transferts en téléchargement vers l'aval. Ses valeurs d'état valides sont LOADED et IN USE.

– Un objet Load Region dont le contenu est stocké dans une mémoire effaçable prend en charge des transferts en téléchargement vers l'aval. Ses valeurs d'état valides sont DOWNLOADABLE, DOWNLOADING, DOWNLOADING SUCCESS, LOADED et IN USE,

Actual LR Size
Cet attribut spécifie la longueur utilisée de l'objet Load Region en octets. Un objet Load Region dont le contenu est effacé doit avoir une valeur égale à 0 pour cet attribut.

Type d'attribut: Unsigned32

Valeurs admissibles: 0 à 2^{32} - 1

Identifier of Related Function Invocation
Cet attribut facultatif est un triplet constitué de l'API, du Slot Number et du FI Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Function Invocation connexe appartient.

Type d'attribut: Unsigned8, Unsigned8, Unsigned16

Valeurs admissibles du triplet: Valeurs admissibles:[0 à 254, 0 à 254, 0 à 65 199 (65 200 à 65 535 réservés)]

Max Segment Length
Cet attribut spécifie la longueur maximale en octets de chaque Push Segment ou Pull Segment qui doit être utilisé pour le téléchargement (vers l'amont ou vers l'aval) de cet objet Load Region.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 234

Max Response Delay
Cet attribut spécifie la durée maximale (en unités de 10 ms) dont la réponse d'une demande de service Initiate Load, Pull Segment, Push Segment ou Terminate Load peut être différée par le Serveur en raison du traitement de la demande. La valeur de cet attribut doit être inférieure à la valeur de l'attribut MS1 Timeout de l'objet ARL DP-master (Class 1).

Type d'attribut: Unsigned16

Valeurs admissibles: 1 à 2^{16} - 1

Uploadcount
Cet attribut indique le nombre de Clients qui téléchargent actuellement vers l'amont le contenu de l'objet Load Region. Il faut que la valeur de cet attribut soit égale à zéro pour télécharger vers l'aval l'objet Load Region. Cet attribut est incrémenté chaque fois qu'un Client télécharge vers l'amont l'objet Load Region en parallèle. Cet attribut est décrémenté chaque fois qu'un Client finit le téléchargement vers l'amont de l'objet Load Region.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 255

Load Region Additional Information
Cet attribut facultatif contient des informations complémentaires relatives à l'objet Load Region; par exemple: les formats permis pour les données.

Type d'attribut: Octet String

Load Region Name
Cet attribut facultatif définit le nom de l'objet Load Region.

Type d'attribut: Visible String(32)

Local Detail
Cet attribut facultatif contient des informations locales complémentaires concernant l'objet Load Region.

Type d'attribut: Octet String

6.2.7.2.3 Invocation de l'objet load region

Pour l'invocation de l'objet Load Region, les règles suivantes s'appliquent.

— Les objets Load Region ne doivent pas dépasser la longueur totale de $2^{32} - 1$ octets. L'attribut LR Size est à positionner en conséquence.

— Au moins le service Push Segment ou le service Pull Segment ou les deux doivent être autorisés pour l'accès de l'objet Load Region.

— Au moins une AR doit avoir accès à l'objet Load Region.

— Un seul objet Function Invocation doit être connexe à un objet Load Region.

— Les droits d'accès doivent être établis en conséquence au(x) service(s) autorisé(s).

6.2.7.3 Spécification de services de Load region

6.2.7.3.1 Initiate load

Ce service est utilisé par un AP pour lancer le téléchargement (vers l'aval ou vers l'amont) d'un Load Region. Un paramètre du service indique si l'image de chargement est transférée dans l'objet Load Region en utilisant le service Pull Segment ou de l'objet Load Region en utilisant le service Push Segment. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2. Le Tableau 155 montre les paramètres du service.

Tableau 155 – Initiate load

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Load Type	M	M(=)		
Load Image Size	C	C(=)		
Intersegment Request Timeout	M	M(=)		
Additional Information	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Actual LR Size			M	M(=)
Max Response Delay			M	M(=)
Max Segment Length			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Load Region souhaité dans une baie spécifique (typiquement un module).

LR Index
Ce paramètre spécifie l'indice de l'objet Load Region à télécharger vers l'aval ou à télécharger vers l'amont.

Load Type
Ce paramètre spécifie si l'objet Load Region est à télécharger vers l'aval ou à télécharger vers l'amont.

Valeurs admissibles: Push, Pull

Load Image Size
Ce paramètre conditionnel indique le nombre d'octets à télécharger vers l'aval dans l'objet Load Region si le paramètre Load Type est égal à Push. Si le paramètre Load Type est égal à Pull, le paramètre Load Image Size est absent. Ce paramètre doit avoir une valeur égale à zéro si le contenu de l'objet Load Region doit être effacé.

Type d'attribut: Unsigned32

Valeurs admissibles: 0 à 2^{32} - 1

Intersegment Request Timeout
Ce paramètre spécifie la période de temporisation côté Serveur entre les demandes de service Pull Segment ou Push Segment, en unités de 100 ms. Les valeurs par défaut sont montrées dans le Tableau 156.

Type d'attribut: Unsigned16

Valeurs admissibles: 1 à 2^{16} - 1

Tableau 156 – Valeurs par défaut pour le paramètre Intersegment Request Timeout

Débits en bauds (kbit/s)	≤187,5	500	1 500	3 000	6 000	12 000
Intersegment Request Timeout (s)	40	20	10	10	10	10

Additional Information
Ce paramètre facultatif spécifie des informations complémentaires pour la procédure de chargement ultérieure (par exemple: le format souhaité pour les données).

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Actual LR Size
Ce paramètre contient la valeur

– de l'attribut Actual LR Size de l'objet Load Region si le paramètre Load Type a eu la valeur Pull dans la primitive "request" précédente,

– du paramètre Load Image Size si le paramètre Load Type a eu la valeur Push dans la primitive "request" précédente.

Type d'attribut: Unsigned32

Valeurs admissibles: 0 à 2^{32} - 1

Max Response Delay
Ce paramètre contient la valeur de l'attribut Max Response Delay de l'objet Load Region. L'AP Client doit surveiller les séquences de demandes et de réponses de service Initiate Load, Push Segment, Pull Segment ou Terminate Load avec ce paramètre temporel.

Max Segment Length
Ce paramètre contient la valeur de l'attribut Max Segment Length de l'objet Load Region.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: state conflict (conflit d'états), load image size error (erreur de taille d'image de chargement), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide).

6.2.7.3.2 Push segment

Ce service facultatif est utilisé pour transférer des LR Data vers l'objet Load Region après que le processus de téléchargement vers l'aval a été lancé. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 157 montre les paramètres du service.

Tableau 157 – Push segment

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Segment Length	M	M(=)		
Segment Number	M	M(=)		
More Follows	M	M(=)		
Data	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Load Region souhaité dans une baie spécifique (typiquement un module).

LR Index
Ce paramètre spécifie l'indice de l'objet Load Region à télécharger vers l'aval.

Segment Length
Ce paramètre indique le nombre d'octets de l'objet Max Segment Length à télécharger vers l'aval.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 234

Segment Number
Ce paramètre indique le numéro du segment qui est transféré. Les numéros de segment sont dans l'ordre croissant, sans trous, en commençant par la valeur 1.

Type d'attribut: Unsigned32

Valeurs admissibles: 1 à 2^{32} - 1

More Follows
Ce paramètre indique si, oui ou non, des segments supplémentaires restent à émettre.

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

Data
Le paramètre Data contient le segment de l'objet Load Image à télécharger vers l'aval et est constitué du nombre d'octets indiqués dans le paramètre Segment Length de la demande. Il est nécessaire que ce paramètre soit constitué du type de données Octet String.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: state conflict (conflit d'états), length error (erreur de longueur), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide), feature not supported (caractéristique non prise en charge).

6.2.7.3.3 Pull segment

Ce service facultatif est utilisé pour transférer des LR Data à partir de l'objet Load Region après que le processus de téléchargement vers l'amont a été lancé. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 158 montre les paramètres du service.

Tableau 158 – Pull segment

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Segment Length	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Segment Length			M	M(=)
Segment Number			M	M(=)
More Follows			M	M(=)
Data			C	C(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Load Region souhaité dans une baie spécifique (typiquement un module).

LR Index
Ce paramètre spécifie l'indice de l'objet Load Region à télécharger vers l'amont.

Segment Length
Ce paramètre indique le nombre d'octets de l'objet Load Region qui doit être téléchargé par l'amont avec cette demande.

Type d'attribut: Unsigned8

Valeurs admissibles: 1 à 234

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Segment Length
La valeur de ce paramètre est égale au paramètre équivalent de la demande si les octets restants à télécharger vers l'amont constituent une longueur supérieure ou égale à la longueur de segment demandée. Autrement, ce paramètre contient le nombre d'octets restants.

Type d'attribut: Unsigned8

Valeurs admissibles: 0 à 234

Segment Number
Ce paramètre indique le numéro du segment qui est transféré. Les numéros de segment sont dans l'ordre croissant, sans trous, en commençant par la valeur 1.

Type d'attribut: Unsigned32

Valeurs admissibles: 1 à 2^{32} - 1

More Follows
Ce paramètre Boolean indique si, oui ou non, des segments supplémentaires restent à émettre.

Type d'attribut: Boolean

Valeurs admissibles: TRUE, FALSE

Data
Ce paramètre conditionnel est présent seulement si la valeur du paramètre Segment Length de la réponse est différente de zéro et contient le segment de Load Image à télécharger vers l'amont et se compose du nombre d'octets indiqué dans le paramètre Segment Length de la réponse. Il faut que ce paramètre soit constitué du type de données Octet String.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeur admissible: state conflict (conflit d'états), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide), feature not supported (caractéristique non prise en charge).

6.2.7.3.4 Terminate load

Ce service est utilisé pour mettre fin au processus de téléchargement vers l'amont ou vers l'aval. Il doit être utilisé à la suite d'un achèvement réussi ou dans le but d'abandonner un téléchargement vers l'amont ou un téléchargement vers l'aval en cours.

Si le processus de téléchargement vers l'amont ou vers l'aval est en cours d'abandon avec le service Terminate Load, alors

— dans le cas d'un téléchargement vers l'aval, le contenu de l'objet Load Region doit être supprimé, l'attribut Actual LR Size doit être mis à zéro et l'attribut Load Region State doit être mis à DOWNLOADABLE);
— dans le cas d'un téléchargement vers l'amont, la valeur de l'attribut Uploadcount de l'objet Load Region doit être décrémentée.

Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 159 montre les paramètres du service.

Tableau 159 – Terminate load

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
LR Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Load Region souhaité dans une baie spécifique (typiquement un module).

LR Index
Ce paramètre spécifie l'indice de l'objet Load Region dont le processus de téléchargement vers l'amont ou de téléchargement vers l'aval doit être arrêté.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeur admissible: state conflict (conflit d'états), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide).

6.2.7.4 Comportement de l'objet Load Region

6.2.7.4.1 Description du diagramme d'états de Load region

6.2.7.4.1.1 Vue d'ensemble

Le comportement d'un objet Load Region est défini par le diagramme d'états des Load Region. Chaque objet Load Region est représenté par une propre instance de ce diagramme d'états.

Le diagramme d'états des Load Region représente une partie de la fonctionnalité d'utilisateur et est inclus pour assurer l'interopérabilité. Il est supposé que les autres parties de la fonctionnalité d'utilisateur accomplissent les vérifications suivantes:

— validation de l'existence et de l'adressage de l'objet Load Region,

— validation des droits d'accès,

— validation des restrictions temporaires du système relatives à l'utilisation de l'objet Load Region

et génèrent les Error Code appropriés, s'il y a lieu.

Deux types sont définis pour les diagrammes d'états des Load Region. L'un représente le comportement des objets Load Region dont le contenu est stocké dans une mémoire effaçable et, donc, peut 'être téléchargé vers l'aval. L'autre représente le comportement des objets Load Region dont le contenu est stocké dans une mémoire non effaçable et, donc, ne peut pas être téléchargé vers l'aval.

L'objet Load Region peut être utilisé par des objets Function Invocation. Par conséquent, les diagrammes d'états des Load Region fournissent une interface aux diagrammes d'états des Function Invocation pour des besoins de coordination.

Les principales fonctionnalités du diagramme d'états des Load Region sont:

— contrôler la séquence valide de services exécutés sur l'objet Load Region,

— assurer qu'un téléchargement vers l'aval est exécuté seulement si aucun téléchargement vers l'amont n'est en cours,

— assurer qu'un téléchargement vers l'amont ou un téléchargement vers l'aval est exécuté seulement si l'objet Load Region n'est pas utilisé par un objet Function Invocation,

— assurer qu'un téléchargement vers l'amont est exécuté seulement si aucun téléchargement vers l'aval n'est en cours,

— surveiller l'activité Client.

6.2.7.4.1.2 Définitions des primitives

6.2.7.4.1.2.1 Primitives échangées entre l'utilisateur et le diagramme d'états des régions de charge

Le Tableau 160 et le Tableau 161 montrent les primitives échangées entre l'utilisateur et le diagramme d'états des Load Region.

Tableau 160 – Primitives émises par l'utilisateur vers le diagramme d'états des Load Region

Nom de primitive	Source	Paramètres associés	Fonctions
Abort.ind	User("Utilisateur")	AREP Locally Generated Subnet Instance Reason Code Additional Detail	—
DP-slave Stopped.ind	User("Utilisateur")	AREP	—
Initiate Load.ind	User("Utilisateur")	AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request, Timeout	—
Pull Segment.ind	User("Utilisateur")	AREP, Slot Number, LR Index, Segment Length	—
Push Segment.ind	User("Utilisateur")	AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data	—
Terminate Load.ind	User("Utilisateur")	AREP, Slot Number, LR Index	—

Tableau 161 – Primitives émises par le diagramme d'états des Load Region vers l'utilisateur

Nom de primitive	Source	Paramètres associés	Fonctions
Initiate Load.rsp(+)	Load Region ("région de charge")	AREP, Actual LR Size, Max Response Delay, Max Segment Length	—
Initiate Load.rsp(-)	Load Region	AREP, Error Code	—
Pull Segment.rsp(+)	Load Region	AREP, Segment Length, Segment Number, More Follows, Data	—
Pull Segment.rsp(-)	Load Region	AREP, Error Code	—
Push Segment.rsp(+)	Load Region	AREP	—
Push Segment.rsp(-)	Load Region	AREP, Error Code	—
Terminate Load.rsp(+)	Load Region	AREP	—
Terminate Load.rsp(-)	Load Region	AREP, Error Code	—

Les paramètres utilisés avec les primitives sont décrits dans la Spécification de service de l'objet Load Region (voir 6.2.7.3).

6.2.7.4.1.2.2 Primitives échangées entre les diagrammes d'états des invocations de fonction et des régions de charge

Le Tableau 162 et le Tableau 163 montrent les primitives échangées entre les diagrammes d'états des Function Invocation et des Load Region.

Tableau 162 – Primitives émises par le diagramme d'états des Function Invocation vers le diagramme d'états des Load Region

Nom de primitive	Source	Paramètres associés	Fonctions
DeleteInUse.ind	Function Invocation ("Invocation de fonction")	FI Identifier, LR Identifier	L'objet Function Invocation indique avec la primitive que l'objet Load Region n'est plus à utiliser.
SetInUse.ind	Function Invocation	FI Identifier, LR Identifier	L'objet Function Invocation indique avec la primitive qu'il souhaite utiliser l'objet Load Region.

Tableau 163 – Primitives émises par le diagramme d'états des Load Region vers le diagramme d'états des Function Invocation

Nom de primitive	Source	Paramètres associés	Fonctions
DeleteInUse.rsp (+)	Load Region	FI Identifier, LR Identifier	L'objet Load Region répond avec la primitive à une DeleteInUse.ind reçue précédemment.
SetInUse.rsp (+/-)	Load Region	FI Identifier, LR Identifier	L'objet Load Region répond avec la primitive à une SetInUse.ind reçue précédemment.

6.2.7.4.1.3 Définitions d'états

Le Tableau 164 montre les définitions des états du diagramme d'états des Load Region.

Tableau 164 – Définitions des états de Load Region

État	Description
DOWNLOADABLE	Cet état indique l'objet Load Region est vide, mais est capable d'être téléchargé vers l'aval.
DOWNLOADING	Cet état indique que la séquence de téléchargement vers l'aval a été lancée.
DOWNLOAD SUCCESS	Cet état transitoire indique que la séquence de téléchargement vers l'aval a réussi, mais n'a pas encore été arrêtée.
LOADED	Cet état indique que la séquence de téléchargement vers l'aval s'est terminée avec succès.
IN-USE	Cet état indique que l'objet Load Region est chargé et est actuellement utilisé par un objet Function Invocation.

6.2.7.4.1.4 Définitions de variables locales

AREP List

L'AREP List consiste en une liste unidimensionnelle contenant un ou plusieurs AREP qui représentent les relations entre applications qui exécutent actuellement la séquence de téléchargement vers l'aval ou vers l'amont.

Segment Number List

La variable Segment Number List consiste en une liste bidimensionnelle contenant un ou plusieurs AREP, également contenu(s) dans la variable AREP List, et du paramètre Segment Number réel de la séquence exécutée sur cette relation.

Actual Length List

La variable Actual Length List consiste en une liste bidimensionnelle contenant un ou plusieurs AREP, également contenu(s) dans la variable AREP List, et l'Actual Length de la quantité de données déjà échangées dans la séquence exécutée sur cette relation.

FI Identifier

Cette variable locale contient l'Identifier (API, Slot Number, FI Index) de l'objet Function Invocation qui utilise actuellement cet objet Load Region.

6.2.7.4.1.5 Définitions de temporisateurs

Intersegment Request Timer

Avec le temporisateur Intersegment Request Timer, le diagramme d'états des Load Region surveille l'activité Client au cours d'une séquence de téléchargement vers l'amont ou vers l'aval.

6.2.7.4.1.6 Définitions de fonctions

Les fonctions suivantes contenues dans le Tableau 165 sont utilisées par les diagrammes d'états des Load Region qui décrivent un objet Load Region stocké tant dans une mémoire effaçable que dans une mémoire non effaçable.

Tableau 165 – Tableau des fonctions de Load Region

Nom de fonction	Description
IsServiceAccepted(Load Type, Load Image Size, Additional Information)	Cette fonction vérifie en appelant des sous-fonctions si le service est acceptable et concorde aux conditions locales des objets. A) Si les résultats des sous-fonctions appelées sont comme suit: IsLRSizeOK(Load Image Size) = TRUE IsPushAndUpload() = FALSE IsAddInfoOK(Additional Information) = TRUE elle retourne TRUE. B) Autrement, elle retourne FALSE.
IsPushAndUpload()	Cette fonction vérifie si, oui ou non, une séquence Upload (téléchargement vers l'amont) est en cours. A) Si l'attribut Uploadcount est supérieur à zéro, elle retourne TRUE. B) Autrement, elle retourne FALSE.
IsLRSizeOK(Load Image Size)	Cette fonction vérifie si, oui ou non, le paramètre Load Image Size concorde avec l'attribut LR Size. A) Si Load Image Size est inférieur ou égal à R Size, elle retourne TRUE. B) Autrement, elle retourne FALSE.
IsAddInfoOK(Additional Information)	Cette fonction vérifie si, oui ou non, le paramètre Additional Information concorde avec l'attribut Load Region Additional Information. A) Si Additional Information est compatible avec l'attribut Load Region Additional Information, elle retourne TRUE. B) Autrement, elle retourne FALSE.
StoreSegmentNumber(AREP, Segment Number)	Cette fonction incrémente le Segment Number avec un et le stocke comme étant relatif à l'AREP.
GetSegmentNumber(AREP)	Cette fonction retourne le Segment Number relatif à l'AREP.
CheckSegmentNumber(AREP, Segment Number)	Cette fonction compare le Segment Number avec le Segment Number relatif à l'AREP qui a été mémorisé. A) Si le Segment Number est égal au Segment Number relatif à l'AREP qui a été mémorisé, elle retourne TRUE. B) Autrement, elle retourne FALSE.
StoreActualLength(AREP, Actual Length)	Cette fonction mémorise l'Actual Length comme étant relative à l'AREP.
GetActualLength(AREP)	Cette fonction retourne l'Actual Length relative à l'AREP.

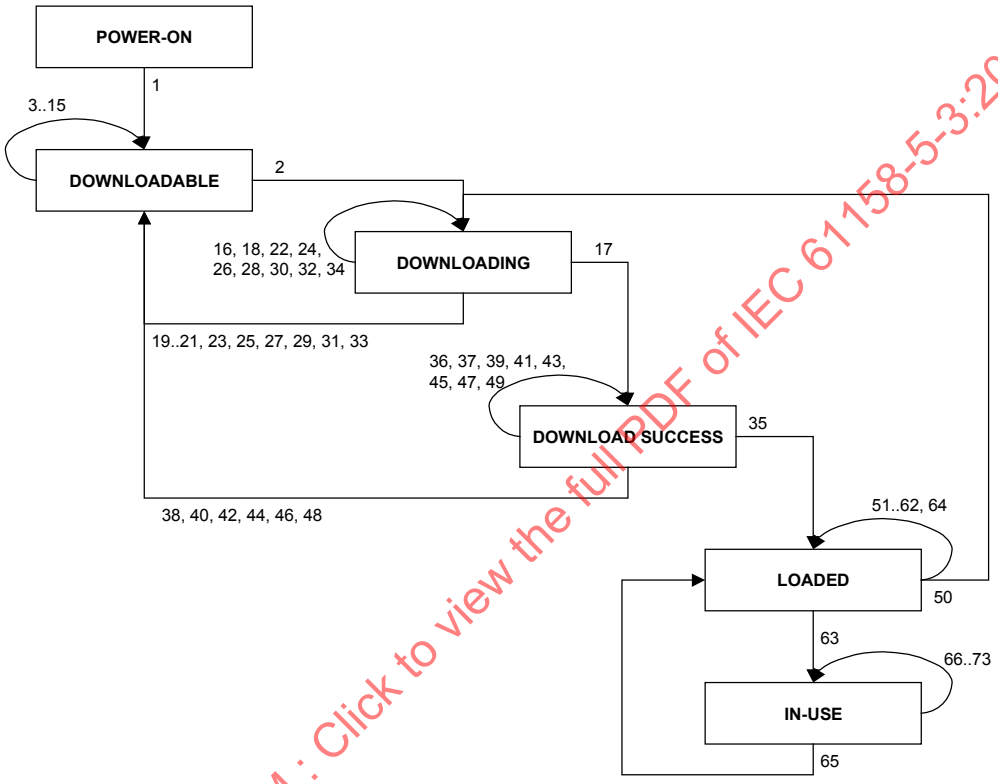
Nom de fonction	Description
GetActualSegLength(AREP, Segment Length)	Cette fonction lit l'Actual Length relative à l'AREP mémorisée en appelant la fonction GetActualLength et calcule le Segment Length réel conformément à l'équation suivante: A) Si la différence (Actual LR Size - Actual Length) est supérieure ou égale à Segment Length et inférieure ou égale à Max Segment Length, elle retourne Segment Length. B) Si la différence (Actual LR Size - Actual Length) est supérieure ou égale à Segment Length et supérieure à Max Segment Length, elle retourne Max Segment Length. C) Si la différence (Actual LR Size - Actual Length) est inférieure à Segment Length et inférieure ou égale à Max Segment Length, elle retourne (Actual LR Size - Actual Length). D) Si la différence (Actual LR Size - Actual Length) est inférieure à Segment Length et supérieure à Max Segment Length, elle retourne Max Segment Length. En outre, cette fonction calcule la nouvelle Actual Length conformément à l'équation suivante: Actual Length = Actual Length + valeur retournée. La nouvelle Actual Length est mémorisée comme étant relative à l'AREP en appelant la fonction StoreActualLength.
CheckActualSegLength(AREP, Segment Length)	Cette fonction lit l'Actual Length relative à l'AREP mémorisée en appelant la fonction GetActualLength et vérifie Segment Length conformément aux conditions suivantes: A) Si la valeur de Segment Length est supérieure à Max Segment Length, elle retourne FALSE. B) Si la valeur de Segment Length est supérieure à (Actual LR Size -Actual Length), elle retourne FALSE. C) Autrement, elle retourne TRUE.
IsLastSegment(AREP)	Cette fonction lit l'Actual Length relative à l'AREP mémorisée en appelant la fonction GetActualLength et calcule s'il s'agit, oui ou non, du dernier segment à transférer conformément aux conditions suivantes: A) Si la valeur de Actual Length est égale à Actual LR Size, elle retourne FALSE. B) Autrement, elle retourne TRUE.
StoreAREP(AREP)	Cette fonction stocke l'AREP dans l'AREP List.
CheckAREP(AREP)	Cette fonction vérifie si, oui ou non, l'AREP est contenu dans l'AREP List conformément aux conditions suivantes: A) Si l'AREP est contenu, elle retourne TRUE. B) Autrement, elle retourne FALSE.
DeleteAREP(AREP)	Cette fonction supprime l'AREP dans l'AREP List.
StartIntersegReqTimer(AREP, T1)	Cette fonction démarre ou redémarre le temporisateur Intersegment Request Timer relatif à l'AREP.
StopIntersegReqTimer(AREP)	Cette fonction arrête le temporisateur Intersegment Request Timer relatif à l'AREP.
CheckMoreFollows(AREP; More Follows)	Cette fonction lit l'Actual Length relative à l'AREP mémorisée en appelant la fonction GetActualLength et vérifie s'il s'agit, oui ou non, du dernier segment à recevoir conformément aux conditions suivantes: A) Si la valeur de Actual Length est égale à Actual LR Size, elle retourne TRUE. B) Autrement, elle retourne FALSE.
ClearContent	Cette fonction supprime le contenu de l'objet Load Region.
StoreFIIdentifier(FI Identifier)	Cette fonction stocke le FI Identifier.
CheckFIIdentifier(FI Identifier)	Cette fonction vérifie le FI Identifier conformément aux conditions suivantes: A) Si FI Identifier est égal au FI Identifier stocké, elle retourne TRUE. B) Autrement, elle retourne FALSE.
DeleteFIIdentifier(FI Identifier)	Cette fonction supprime le FI Identifier stocké.
StoreData(Data)	Cette fonction stocke le paramètre Data dans le contenu de l'objet Load Region.

Nom de fonction	Description
GetData()	Cette fonction récupère les Data du contenu de l'objet Load Region.

6.2.7.4.2 Diagrammes d'états des Load region

6.2.7.4.2.1 Diagramme d'états des Load region pour mémoire effaçable

La Figure 25 montre le diagramme d'états d'un objet Load Region pour mémoire effaçable.



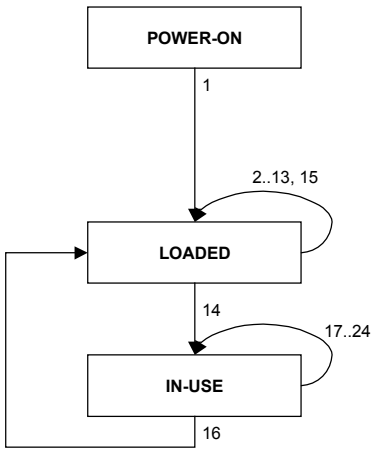
Légende

Anglais	Français
POWER-ON	POWER-ON (Mise sous tension)
DOWNLOADABLE	DOWNLOADABLE (Téléchargeable vers l'aval)
DOWNLOADING	DOWNLOADING (Téléchargement vers l'aval)
DOWNLOAD SUCCESS	DOWNLOAD SUCCESS (Succès du téléchargement)
LOADED	LOADED (Chargé)
IN-USE	IN-USE) En utilisation

Figure 25 – Diagramme d'états des Load Region pour mémoire effaçable

6.2.7.4.2.2 Diagramme d'états des Load region pour mémoire non effaçable

La Figure 26 montre le diagramme d'états d'un objet Load Region pour mémoire non effaçable.



Légende

Anglais	Français
POWER-ON	POWER-ON (Mise sous tension)
LOADED	LOADED (Chargé)
IN-USE	IN-USE (En utilisation)

Figure 26 – Diagramme d'états des Load Region pour mémoire non effaçable

6.2.7.4.3 Tables d'états des Load region

6.2.7.4.3.1 Table d'états des Load region pour mémoire effaçable

Le Tableau 166 montre la table d'états d'un objet Load Region pour mémoire effaçable.

Tableau 166 – Table d'états des Load Region pour mémoire effaçable

#	État courant	Événement /Condition =>Action	État suivant
1	POWER-ON	=> Uploadcount:=0	DOWNLOADABLE
2	DOWNLOADABLE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) && (IsServiceAccepted(Load Type, Load Image Size, Additional Information)=TRUE) => StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) T1:=Intersegment Request Timeout Actual LR Size:=Load Image Size Load Region State:=DOWNLOADING StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	DOWNLOADING
3	DOWNLOADABLE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) && (IsServiceAccepted(Load Type, Additional Information)=TRUE) => T1:=Intersegment Request Timeout Uploadcount:=Uploadcount+1 StoreAREP(AREP) StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	DOWNLOADABLE

4	DOWNLOADABLE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information)
		/(Load Type=Push) &&
		(IsServiceAccepted(Load Type, Load Image Size, Additional Information)=FALSE)
		=>
		if (IsPushAndUpload()=TRUE)
		Error Code:=state conflict
		elseif(IsLRSizeOK(Load Image Size)=FALSE)
		Error Code:=load image size error
		Endif
		Initiate Load.rsp(-) (AREP, Error Code)
			DOWNLOADABLE

#	État courant	Événement /Condition =>Action	État suivant
5	DOWNLOADABLE	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
6	DOWNLOADABLE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Segment Length:=0 Segment Number:=1 More Follows:=FALSE StartIntersegReqTimer(AREP, T1) Pull Segment.rsp(+) (AREP, Segment Length, Segment Number, More Follows)	DOWNLOADABLE
7	DOWNLOADABLE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
8	DOWNLOADABLE	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE
9	DOWNLOADABLE	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	DOWNLOADABLE
10	DOWNLOADABLE	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) Uploadcount:=Uploadcount-1 IntersegReqTimerExpired.req(AREP)	DOWNLOADABLE
11	DOWNLOADABLE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	DOWNLOADABLE

#	État courant	Événement /Condition =>Action	État suivant
12	DOWNLOADABLE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	DOWNLOADABLE
13	DOWNLOADABLE	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	DOWNLOADABLE
14	DOWNLOADABLE	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	DOWNLOADABLE
15	DOWNLOADABLE	SetInUse.ind (FI Identifier, LR Identifier) => SetInUse.rsp(-) (FI Identifier, LR Identifier)	DOWNLOADABLE
16	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=TRUE) && (CheckMoreFollows(AREP; More Follows)=TRUE)&& (More Follows=TRUE) => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) StartIntersegReqTimer(AREP, T1) GetActualSegLength(AREP, Segment Length) StoreData(Data) Push Segment.rsp(+) (AREP)	DOWNLOADING
17	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=TRUE) && (CheckMoreFollows(AREP; More Follows)=TRUE)&& (More Follows=FALSE) => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) StartIntersegReqTimer(AREP, T1) GetActualSegLength(AREP, Segment Length) StoreData(Data) Push Segment.rsp(+) (AREP)	DOWNLOAD SUCCESS

#	État courant	Événement /Condition =>Action	État suivant
18	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /CheckAREP(AREP)=FALSE => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADING
19	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=FALSE) => Error Code:=invalid parameter DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
20	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=FALSE) => Error Code:=length error DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
21	DOWNLOADING	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /(CheckAREP(AREP)=TRUE) && (CheckSegmentNumber(AREP, Segment Number)=TRUE) && (CheckActualSegLength(AREP, Segment Length)=TRUE) && (CheckMoreFollows(AREP; More Follows)=FALSE) => Error Code:=invalid parameter DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE

#	État courant	Événement /Condition =>Action	État suivant
22	DOWNLOADING	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADING
23	DOWNLOADING	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=TRUE) && (Load Type=Push) => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE
24	DOWNLOADING	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=FALSE) && (Load Type=Push) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADING
25	DOWNLOADING	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
26	DOWNLOADING	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADING

#	État courant	Événement /Condition =>Action	État suivant
27	DOWNLOADING	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE
28	DOWNLOADING	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOADING
29	DOWNLOADING	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE
30	DOWNLOADING	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	DOWNLOADING
31	DOWNLOADING	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE
32	DOWNLOADING	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	DOWNLOADING
33	DOWNLOADING	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) ClearContent Actual LR Size:=0 IntersegReqTimerExpired.req(AREP)	DOWNLOADABLE

#	État courant	Événement /Condition =>Action	État suivant
34	DOWNLOADING	SetInUse.ind (FI Identifier, LR Identifier) => SetInUse.rsp(-) (FI Identifier, LR Identifier)	DOWNLOADING
35	DOWNLOAD SUCCESS	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	LOADED
36	DOWNLOAD SUCCESS	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
37	DOWNLOAD SUCCESS	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
38	DOWNLOAD SUCCESS	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=TRUE) && (Load Type=Push) => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOADABLE
39	DOWNLOAD SUCCESS	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(CheckAREP(AREP)=FALSE) && (Load Type=Push) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS

#	État courant	Événement /Condition =>Action	État suivant
40	DOWNLOAD SUCCESS	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
41	DOWNLOAD SUCCESS	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
42	DOWNLOAD SUCCESS	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /CheckAREP(AREP)=TRUE => Error Code:=state conflict DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0 Push Segment.rsp(-) (AREP, Error Code)	DOWNLOADABLE
43	DOWNLOAD SUCCESS	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) /CheckAREP(AREP)=FALSE => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	DOWNLOAD SUCCESS
44	DOWNLOAD SUCCESS	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE
45	DOWNLOAD SUCCESS	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	DOWNLOAD SUCCESS

#	État courant	Événement /Condition =>Action	État suivant
46	DOWNLOAD SUCCESS	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) StopIntersegReqTimer(AREP) ClearContent Actual LR Size:=0	DOWNLOADABLE
47	DOWNLOAD SUCCESS	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	DOWNLOAD SUCCESS
48	DOWNLOAD SUCCESS	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) ClearContent Actual LR Size:=0 IntersegReqTimerExpired.req(AREP)	DOWNLOADABLE
49	DOWNLOAD SUCCESS	SetInUse.ind (FI Identifier, LR Identifier) => SetInUse.rsp(-) (FI Identifier, LR Identifier)	DOWNLOAD SUCCESS
50	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) && (IsServiceAccepted(Load Type, Load Image Size, Additional Information)=TRUE) => StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) T1:=Intersegment Request Timeout Actual LR Size:=Load Image Size Load Region State:=DOWNLOADING StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	DOWNLOADING

#	État courant	Événement /Condition =>Action	État suivant
51	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) && (IsServiceAccepted(Load Type, Additional Information)=TRUE) => T1:=Intersegment Request Timeout Uploadcount:=Uploadcount+1 StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	LOADED

52	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information)
		/(Load Type=Push) &&
		(IsServiceAccepted(Load Type, Load Image Size, Additional Information)=FALSE)
		=>
		if (IsPushAndUpload()=TRUE)
		Error Code:=state conflict
		Elseif(IsLRSizeOK(Load Image Size)=FALSE)
		Error Code:=load image size error
		Endif
		Initiate Load.rsp(-) (AREP, Error Code)
			LOADED

53	LOADED	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	LOADED
54	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) Segment Length:=GetActualSegLength(AREP, Segment Length) More Follows:=IsLastSegment(AREP) StartIntersegReqTimer(AREP, T1) Data:=GetData() Pull Segment.rsp(+) (AREP, Segment Length, Segment Number, More Follows, Data)	LOADED

#	État courant	Événement /Condition =>Action	État suivant
55	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	LOADED
56	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	LOADED
57	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	LOADED
58	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED
59	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	LOADED
60	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED
61	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	LOADED
62	LOADED	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) Uploadcount:=Uploadcount-1 IntersegReqTimerExpired.req(AREP)	LOADED

#	État courant	Événement /Condition =>Action	État suivant
63	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount=0 => StoreFIIdentifier(FI Identifier) SetInUse.rsp(+) (FI Identifier, LR Identifier)	IN-USE
64	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount>0 => SetInUse.rsp(-) (FI Identifier, LR Identifier)	LOADED
65	IN-USE	DeleteInUse.ind (FI Identifier, LR Identifier) => DeleteFIIdentifier(FI Identifier) DeleteInUse.rsp(+)(FI Identifier, LR Identifier)	LOADED
66	IN-USE	SetInUse.ind (FI Identifier, LR Identifier) /CheckFIIdentifier(FI Identifier)=FALSE => SetInUse.rsp(-) (FI Identifier, LR Identifier)	IN-USE
67	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	IN-USE
68	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	IN-USE
69	IN-USE	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	IN-USE
70	IN-USE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	IN-USE
71	IN-USE	Terminate Load.ind (AREP, Slot Number, LR Index) => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	IN-USE
72	IN-USE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) =>	IN-USE

#	État courant	Événement /Condition =>Action	État suivant
73	IN-USE	DP-slave Stopped.ind (AREP) =>	IN-USE

6.2.7.4.4 Table d'états des Load region pour mémoire non effaçable

Le Tableau 167 montre la table d'états d'un objet Load Region pour mémoire non effaçable.

Tableau 167 – Table d'états des Load Region pour mémoire non effaçable

#	État courant	Événement /Condition =>Action	État suivant
1	POWER-ON	=> Uploadcount:=0	LOADED
2	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) => Error Code:= accès refusé Initiate Load.rsp(-) (AREP, Error Code)	LOADED
3	LOADED	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) && (IsServiceAccepted(Load Type, Additional Information)=TRUE) => T1:=Intersegment Request Timeout Uploadcount:=Uploadcount+1 StoreAREP(AREP) Segment Number:=0 StoreSegmentNumber(AREP, Segment Number) Actual Length:=0 StoreActualLength(AREP, Actual Length) StartIntersegReqTimer(AREP, T1) Initiate Load.rsp(+) (AREP, Actual LR Size, Max Response Delay, Max Segment Length)	LOADED
4	LOADED	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	LOADED

#	État courant	Événement /Condition =>Action	État suivant
5	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=TRUE => Segment Number:=GetSegmentNumber(AREP) StoreSegmentNumber(AREP, Segment Number) Segment Length:=GetActualSegLength(AREP, Segment Length) More Follows:=IsLastSegment(AREP) StartIntersegReqTimer(AREP, T1) Data:=GetData() Pull Segment.rsp(+) (AREP, Segment Length, Segment Number, More Follows, Data)	LOADED
6	LOADED	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	LOADED
7	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP) Terminate Load.rsp(+) (AREP)	LOADED
8	LOADED	Terminate Load.ind (AREP, Slot Number, LR Index) /(CheckAREP(AREP)=FALSE => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	LOADED
9	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED
10	LOADED	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) /(CheckAREP(AREP)=FALSE =>	LOADED
11	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=TRUE => DeleteAREP(AREP) Uploadcount:=Uploadcount-1 StopIntersegReqTimer(AREP)	LOADED

#	État courant	Événement /Condition =>Action	État suivant
12	LOADED	DP-slave Stopped.ind (AREP) /(CheckAREP(AREP)=FALSE =>	LOADED
13	LOADED	IntersegReqTimer Expired => AREP:=GetTimerAREP() DeleteAREP(AREP) Uploadcount:=Uploadcount-1 IntersegReqTimerExpired.req(AREP)	LOADED
14	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount=0 => StoreFIIdentifier(FI Identifier) SetInUse.rsp(+) (FI Identifier, LR Identifier)	IN-USE
15	LOADED	SetInUse.ind (FI Identifier, LR Identifier) /Uploadcount>0 => SetInUse.rsp(-) (FI Identifier, LR Identifier)	LOADED
16	IN-USE	DeleteInUse.ind (FI Identifier, LR Identifier) => DeleteFIIdentifier(FI Identifier) DeleteInUse.rsp(+)(FI Identifier, LR Identifier)	LOADED
17	IN-USE	SetInUse.ind (FI Identifier, LR Identifier) /CheckFIIdentifier(FI Identifier)=FALSE => SetInUse.rsp(-) (FI Identifier, LR Identifier)	IN-USE
18	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Load Image Size, Intersegment Request Timeout, Additional Information) /(Load Type=Push) => Error Code:= accès refusé Initiate Load.rsp(-) (AREP, Error Code)	IN-USE
19	IN-USE	Initiate Load.ind (AREP, Slot Number, LR Index, Load Type, Intersegment Request Timeout, Additional Information) /(Load Type=Pull) => Error Code:=state conflict Initiate Load.rsp(-) (AREP, Error Code)	IN-USE
20	IN-USE	Push Segment.ind (AREP, Slot Number, LR Index, Segment Length, Segment Number, More Follows, Data) => Error Code:=state conflict Push Segment.rsp(-) (AREP, Error Code)	IN-USE

#	État courant	Événement /Condition =>Action	État suivant
21	IN-USE	Pull Segment.ind (AREP, Slot Number, LR Index, Segment Length) => Error Code:=state conflict Pull Segment.rsp(-) (AREP, Error Code)	IN-USE
22	IN-USE	Terminate Load.ind (AREP, Slot Number, LR Index) => Error Code:=state conflict Terminate Load.rsp(-) (AREP, Error Code)	IN-USE
23	IN-USE	Abort.ind (AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail) =>	IN-USE
24	IN-USE	DP-slave Stopped.ind (AREP) =>	IN-USE

6.2.8 ASE Function invocation

6.2.8.1 Vue d'ensemble

L'ASE Function Invocation définit deux objets, à savoir l'objet Action sans état et l'objet Function Invocation orienté état.

Les invocations de fonction sans état, appelées "actions", se déroulent jusqu'à leur achèvement lorsqu'elles sont invoquées et ne peuvent pas être interrompues. En outre, certaines actions atomiques retournent une valeur en réponse à leur invocation, alors que d'autres ne le font pas. Celles qui le font peuvent être utilisées pour modéliser des *fonctions* logicielles, et celles qui ne le font pas peuvent être utilisées pour modéliser des *procédures* logicielles.

D'autre part, les invocations de fonction orientées état peuvent être commandées pendant leur exécution. Des services sont définis pour les arrêter, les reprendre ou les réinitialiser une fois qu'elles ont été invoquées. Elles ne retournent pas de valeur en réponse à leur lancement. S'il est nécessaire d'arrêter une invocation de fonction une fois qu'elle a démarré, elle doit être définie comme étant orientée état.

Les invocations de fonction orientées état représentent la vue réseau d'une invocation spécifique d'une fonction utilisateur. Si la fonction utilisateur peut être invoquée plus d'une fois simultanément, des objets Function Invocation distincts sont nécessaires pour représenter chaque invocation.

Les invocations de fonction orientées état peuvent être utilisées pour modéliser des processus logiciels ou des opérations utilisateur qui peuvent être lancé(e)s et commandé(e)s.

NOTE Un exemple d'invocation de fonction orientée état qui peut être utilisée pour modéliser une opération utilisateur est l'opération "playback" (restitution) d'un enregistreur de vidéos. Une fois qu'elle a été lancée, l'opération de restitution peut être arrêtée (suspendue) et reprise plus tard.

Pour prendre en charge le concept de modélisation de processus logiciel, la définition d'invocations de fonction inclut des attributs facultatifs qui peuvent être utilisés pour les relier à des objets Load Region.

Les objets Function Invocation doivent être nettement séparés des objets Process Data et des objets Load Region. Aucun chevauchement n'est permis.

6.2.8.2 Spécifications de la classe modèle d'invocation de fonction

6.2.8.2.1 Définition de la classe des Function invocation

6.2.8.2.1.1 Modèle formel pour les invocations de fonction

Un objet Function Invocation est décrit par le modèle suivant:

ASE DP:			**ASE Function Invocation**
CLASS:			Function Invocation
CLASS ID:			non utilisé
PARENT CLASS:			TOP
ATTRIBUTES:			
1.	(m)	Attribut-clé:	Identifier
3.	(m)	Attribut:	Access Rights MS1
4.	(m)	Attribut:	Access Rights MS2
4.1.	(m)	Attribut:	List of SCL
4.1.1.	(m)	Attribut:	SCL
7.	(m)	Attribut:	Function Invocation State
8.	(o)	Attribut:	Load Region Object Identifier
8.1	(c)	Attribut:	Load Region Object In Use
9.	(o)	Attribut:	Execution Argument Data Description
10.	(o)	Attribut:	Function Invocation Name
11.	(o)	Attribut:	Local Detail
SERVICES:			
1.	(m)	OpsService:	Start
2.	(o)	OpsService:	Stop
3.	(o)	OpsService:	Resume
4.	(m)	OpsService:	Reset
5.	(m)	OpsService:	Get FI State

6.2.8.2.1.2 Attributs

Identifier
Cet attribut-clé est un triplet constitué d'un API, d'un Slot Number et d'un FI Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Function Invocation appartient. Cet Identifier ne doit pas être utilisé par un autre objet Action, un autre objet Function Invocation, un autre objet Load Region ou un autre objet Process Data.

Type d'attribut: Unsigned8, Unsigned8, Unsigned16

Valeurs admissibles du triplet: [0 à 254, 0 à 254, 0 à 65 199 (65 200 à 65 535 réservés)]

Access Rights MS1
Cet attribut définit le type de droits d'accès définis pour une AR MS1 pour l'objet Function Invocation.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 168.

Tableau 168 – Access rights MS1

Valeur	Signification
2	Right to Start, Stop, Resume, Reset (Droit de lancer, d'arrêter, de reprendre, de réinitialiser)
3	No Access (Aucun accès)

Access Rights MS2
Cet attribut définit le type de droits d'accès définis pour une AR MS2 pour l'objet Function Invocation.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 169.

Tableau 169 – Access rights MS2

Valeur	Signification
2	Right to Start, Stop, Resume, Reset (Droit de lancer, d'arrêter, de reprendre, de réinitialiser)
3	No Access (Aucun accès)

List of SCL
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur pour la protection d'accès à cet objet Function Invocation. Les règles pour accepter ou refuser un accès sont décrites en 6.2.1.3.

Type d'attribut: Unsigned8

Function Invocation State
Cet attribut spécifie l'état de l'objet Function Invocation.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 170.

Tableau 170 – Function Invocation state

Valeur	Signification
0	IDLE
1	STARTING
2	RUNNING
3	STOPPING
4	STOPPED
5	RESUMING
6	UNRUNNABLE
7	RESETTING
8	WAIT-SET-IN-USE
9	WAIT-SET-IN-USE-2
10	WAIT-DEL-IN-USE
11	WAIT-DEL-IN-USE-2
12	EXEC-ERR
13	EXEC-TERM

Load Region Object Identifier
Cet attribut facultatif clé est un triplet constitué de l'API, du Slot Number et du LR Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Load Region connexe appartient.

Type d'attribut: Unsigned8, Unsigned8, Unsigned16

Valeurs admissibles du triplet: [0 à 254, 0 à 254, 0 à 65 199 (65 200 à 65 535 réservés)]

Load Region Object In Use
Cet attribut Boolean indique si, oui ou non, l'objet Load Region connexe est réellement utilisé par cet objet Function Invocation.

Type d'attribut: Boolean

Les valeurs admissibles sont montrées dans le Tableau 171.

Tableau 171 – Load Region object in use

Valeur	Signification
FALSE	Objet Load Region pas en cours d'utilisation
TRUE	Objet Load Region en cours d'utilisation

Execution Argument Data Description
Cet attribut facultatif contient l'attribut Data Description pour le paramètre Execution Argument du service Start. Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Function Invocation Name
Cet attribut facultatif définit le nom de l'objet Function Invocation.

Type d'attribut: Visible String(32)

Local Detail
Cet attribut facultatif contient des informations locales complémentaires concernant l'objet Function Invocation.

Type d'attribut: Octet String

6.2.8.2.1.3 Invocation de l'objet function invocation

Pour l'invocation de l'objet Function Invocation, les règles suivantes s'appliquent:

— Au moins une AR doit avoir accès à l'objet Function Invocation.

— Un seul objet Load Region doit être connexe à un objet Function Invocation.

— Les droits d'accès doivent être établis en conséquence au(x) service(s) autorisé(s).

— Les services facultatifs doivent être sélectionnés conformément au mode d'exécution (cyclique, non cyclique, ...) de l'invocation de fonction.

6.2.8.2.2 Définition de la classe Action

6.2.8.2.2.1 Modèle formel des Action

Un objet Action est décrit par le modèle suivant:

ASE DP: **ASE Function Invocation**
CLASS: Action
CLASS ID: non utilisé
PARENT CLASS: TOP
ATTRIBUTES:

1.	(m)	Attribut-clé:	Identifier
3.	(m)	Attribut:	Access Rights MS1
4.	(m)	Attribut:	Access Rights MS2
4.1.	(m)	Attribut:	List of SCL
4.1.1.	(m)	Attribut:	SCL
8.	(o)	Attribut:	Load Region Object Identifier
8.1	(c)	Attribut:	Load Region Object In Use
9.	(o)	Attribut:	Execution Argument Data Description
10.	(o)	Attribut:	Result Argument Data Description
11.	(o)	Attribut:	Action Name
12.	(o)	Attribut:	Local Detail

SERVICES:

1.	(m)	OpsService:	Call

6.2.8.2.2.2 Attributs

Identifier
Cet attribut-clé est un triplet constitué de l'API, du Slot Number et du FI Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Action appartient. Cet Identifier ne doit pas être utilisé par un autre objet Action, un autre objet Function Invocation, un autre objet Load Region ou un autre objet Process Data.

Type d'attribut: Unsigned8, Unsigned8, Unsigned16

Valeurs admissibles du triplet: [0 à 254, 0 à 254, 0 à 65 199 (65 200 à 65 535 réservés)]

Access Rights MS1
Cet attribut définit le type de droits d'accès définis pour une AR MS1 pour l'objet Action.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 172.

Tableau 172 – Access rights MS1

Valeur	Signification
2	Right to Call (Droit d'appeler)
3	No Access (Aucun accès)

Access Rights MS2
Cet attribut définit le type de droits d'accès définis pour une AR MS2 pour l'objet Action.

Type d'attribut: Unsigned8

Les valeurs admissibles sont montrées dans le Tableau 173.

Tableau 173 – Access rights MS2

Valeur	Signification
2	Right to Call (Droit d'appeler)
3	No Access (Aucun accès)

List of SCL
Cet attribut est constitué des éléments de liste suivants:

SCL
L'attribut SCL contient une valeur pour la protection d'accès à cet objet Action. Les règles pour accepter ou refuser un accès sont décrites en 6.2.1.3.

Type d'attribut: Unsigned8

Load Region Object Identifier
Cet attribut facultatif clé est un triplet constitué de l'API, du Slot Number et du LR Index pour définir à quels Application Process Identifier, module et Index au sein du module spécifié l'objet Load Region connexe appartient.

Type d'attribut: Unsigned8, Unsigned8, Unsigned16

Valeurs admissibles du triplet: [0 à 254, 0 à 254, 0 à 65 199 (65 200 à 65 535 réservés)]

Load Region Object In Use
Cet attribut Boolean indique si, oui ou non, l'objet Load Region connexe est réellement utilisé par cet objet Action.

Type d'attribut: Boolean

Les valeurs admissibles sont montrées dans le Tableau 174.

Tableau 174 – Load Region object in use

Valeur	Signification
FALSE	Objet Load Region pas en cours d'utilisation
TRUE	Objet Load Region en cours d'utilisation

Execution Argument Data Description
Cet attribut facultatif contient l'attribut Data Description pour le paramètre Execution Argument de la demande de service Call. Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Result Argument Data Description
Cet attribut facultatif contient l'attribut Data Description pour le paramètre Execution Argument de la réponse de service Call. Cet attribut peut contenir soit une Simple Data Description, soit une Array Data Description, soit une Record Data Description. Pour la description de données, le modèle utilisé est celui donné dans la CEI 61158-5-10, 5.1.

Action Name
Cet attribut facultatif définit le nom de l'objet Action.

Type d'attribut: Visible String(32)

Local Detail
Cet attribut facultatif contient des informations locales complémentaires concernant l'objet Action.

Type d'attribut: Octet String

6.2.8.2.2.3 Invocation de l'objet action

Pour l'invocation de l'objet Action, les règles suivantes s'appliquent:

— Au moins une AR doit avoir accès à l'objet Action.
— Un seul objet Load Region doit être connexe à un objet Action.
— Les droits d'accès doivent être établis en conséquence au(x) service(s) autorisé(s).

6.2.8.3 Spécification de services de Function invocation

6.2.8.3.1 Service Start

Ce service confirmé est utilisé pour demander qu'une Function Invocation soit lancée. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 175 montre les paramètres du service.

Tableau 175 – Start

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Execution Argument	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Function Invocation souhaité dans une baie spécifique (typiquement un module).

FI Index
Ce paramètre spécifie l'indice de l'objet Function Invocation à lancer.

Execution Argument
Ce paramètre facultatif contient des informations complémentaires pour accomplir le démarrage de la Function Invocation. Il doit être présent si l'attribut Execution Argument Data Description est défini pour l'objet Function Invocation adressé.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: state conflict (conflit d'états), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide), resource busy (ressource occupée).

6.2.8.3.2 Service Stop

Ce service confirmé est utilisé pour arrêter une Function Invocation en retenant son contexte afin qu'elle puisse être reprise. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 176 montre les paramètres du service.

Tableau 176 – Stop

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Function Invocation souhaité dans une baie spécifique (typiquement un module).

FI Index
Ce paramètre spécifie l'indice de l'objet Function Invocation à arrêter.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: state conflict (conflit d'états), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide), feature not supported (caractéristique non prise en charge).

6.2.8.3.3 Service Resume

Ce service confirmé est utilisé pour reprendre l'exécution d'une Function Invocation qui a été arrêtée. L'exécution est reprise en utilisant le contexte sauvegardé lorsque la Function Invocation avait été arrêtée. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 177 montre les paramètres du service.

Tableau 177 – Resume

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Execution Argument	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Function Invocation souhaité dans une baie spécifique (typiquement un module).

FI Index
Ce paramètre spécifie l'indice de l'objet Function Invocation à reprendre.

Execution Argument
Ce paramètre facultatif contient des informations complémentaires pour reprendre l'exécution de la Function Invocation. Il doit être présent si l'attribut Execution Argument Data Description est défini pour l'objet Function Invocation adressé.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: state conflict (conflit d'états), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide), feature not supported (caractéristique non prise en charge), resource busy (ressource occupée).

6.2.8.3.4 Service Reset

Ce service confirmé est utilisé pour réinitialiser une Function Invocation avec son contexte initial. Son contexte initial est défini comme étant le contexte de la Function Invocation établi par ses procédures d'initialisation. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 178 montre les paramètres du service.

Tableau 178 – Reset

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Function Invocation souhaité dans une baie spécifique (typiquement un module).

FI Index
Ce paramètre spécifie l'indice de l'objet Function Invocation à réinitialiser.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: state conflict (conflit d'états), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide).

6.2.8.3.5 Service Get FI state

Ce service confirmé est utilisé pour lire l'état réel d'un objet Function Invocation. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 179 montre les paramètres du service.

Tableau 179 – Get FI state

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
FI Index	M	M(=)		
Result(+)			S	S(=)
AREP			M	M(=)
FI State			M	M(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Function Invocation souhaité dans une baie spécifique (typiquement un module).

FI Index
Ce paramètre spécifie l'indice de l'objet Function Invocation à arrêter.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

FI State
Ce paramètre contient l'état réel de l'objet Function Invocation.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide).

6.2.8.3.6 Service Call

Ce service confirmé est utilisé pour invoquer l'exécution d'un objet Action. Ce service doit seulement être utilisé conjointement à l'AR MS1 ou à l'AR MS2.

Le Tableau 180 montre les paramètres du service.

Tableau 180 – Call

Nom de paramètre	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
AREP	M	M(=)		
Slot Number	M	M(=)		
Entity Number	U	U(=)		
FI Index	M	M(=)		
Execution Argument	U	U(=)		
Result(+)			S	S(=)
AREP			M	M(=)
Result Argument			U	U(=)
Result(-)			S	S(=)
AREP			M	M(=)
Error Code			M	M(=)

Argument
L'argument doit acheminer les paramètres spécifiques au service relatifs à la demande de service.

AREP
Ce paramètre est l'identificateur local pour l'AR souhaitée.

Slot Number
Le paramètre Slot Number est utilisé dans l'appareil de destination pour adresser l'objet Action souhaité dans une baie spécifique (typiquement un module).

Entity Number
Le paramètre Entity Number (numéro d'entité) est spécifique à un profil.

FI Index
Ce paramètre spécifie l'indice de l'objet Action à invoquer. La place d'indices disponible est structurée comme suit:

– Indices réservés pour une utilisation spécifique au constructeur (0 - 32 767).
– Indices réservés aux services du système PROFIBUS (32 768 - 64 999).
– Indices réservés aux services Identification et Maintenance (I&M) (65 000 - 65 199).

Execution Argument
Ce paramètre facultatif contient des informations complémentaires pour accomplir l'invocation de l'objet Action. Il doit être présent si l'attribut Execution Argument Data Description est défini pour l'objet Action adressé.

Result(+)
Ce paramètre de type sélection indique que la demande de service a réussi.

Result Argument
Ce paramètre facultatif contient le résultat de l'invocation de l'objet Action. Il doit être présent si l'attribut Result Argument Data Description est défini pour l'objet Action adressé.

Result(-)
Ce paramètre de type sélection indique que la demande de service a échoué.

Error Code
Ce paramètre indique la cause de la défaillance.

Valeurs admissibles: type conflict (conflit de types), access denied (accès refusé), invalid index (indice non valide), invalid slot (baie non valide), module failure (défaillance de module), invalid parameter (paramètre non valide), invalid type (type non valide).

6.2.8.4 Comportement de l'objet function invocation

6.2.8.4.1 Description du diagramme d'états des Function invocation

6.2.8.4.1.1 Généralités

Le comportement d'un objet Function Invocation est défini par le diagramme d'états des Function Invocation. Chaque objet Function Invocation est représenté par une propre instance de ce diagramme d'états.

Le diagramme d'états des Function Invocation représente une partie de la fonctionnalité d'utilisateur et est inclus pour assurer l'interopérabilité. Il est supposé que les autres parties de la fonctionnalité d'utilisateur accomplissent les vérifications suivantes:

— validation de l'existence et de l'adressage de l'objet Function Invocation,

— validation des droits d'accès,

— validation des restrictions temporaires du système relatives à l'utilisation de l'objet Function Invocation

— validation de la classe correcte de l'objet

et génèrent les Error Code appropriés, s'il y a lieu.

L'objet Function Invocation utilise un objet Load Region. Par conséquent, le diagramme d'états des Function Invocation fournit une interface aux diagrammes d'états des Load Region pour des besoins de coordination.

Les principales fonctionnalités du diagramme d'états des Function Invocation sont:

— vérifier la validité des services par rapport à l'état de l'objet Function Invocation,

— assurer qu'aucun service parallèle n'est exécuté sur l'objet Function Invocation,

— assurer qu'un objet Function Invocation est utilisé seulement si l'objet Load Region connexe est disponible,

— surveiller l'activité Client.

6.2.8.4.1.2 Définitions des primitives

####### 6.2.8.4.1.2.1 Primitives échangées entre l'utilisateur et le diagramme d'états des invocations de fonction

Le Tableau 181 et le Tableau 182 montrent la primitive échangée entre l'utilisateur et le diagramme d'états des Function Invocation.

Tableau 181 – Primitives émises par l'utilisateur vers le diagramme d'états des Function Invocation

Nom de primitive	Source	Paramètres associés	Fonctions
Abort.ind	User("Utilisateur")	AREP, Locally Generated, Subnet, Instance, Reason Code, Additional Detail	—
Call.ind	User ("Utilisateur")	AREP, Slot Number, FI Index, Execution Argument	—
DP-slave Stopped.ind	User ("Utilisateur")	AREP	—
Reset.ind	User ("Utilisateur")	AREP, Slot Number, FI Index	—
Resume.ind	User ("Utilisateur")	AREP, Slot Number, FI Index, Execution Argument	—
Start.ind	User ("Utilisateur")	AREP, Slot Number, FI Index, Execution Argument	—
Stop.ind	User ("Utilisateur")	AREP, Slot Number, FI Index	—
Get FI State.ind	User ("Utilisateur")	AREP, Slot Number, FI Index	—

Tableau 182 – Primitives émises par le diagramme d'états des Function Invocation vers l'utilisateur

Nom de primitive	Source	Paramètres associés	Fonctions
Reset.rsp(+)	Function Invocation ("Invocation de fonction")	AREP	—
Reset.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Resume.rsp(+)	Function Invocation	AREP	—
Resume.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Start.rsp(+)	Function Invocation	AREP	—
Start.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Stop.rsp(+)	Function Invocation	AREP	—
Stop.rsp(-)	Function Invocation	AREP, Error Code, Function Invocation State	—
Get FI State.rsp(+)	Function Invocation	AREP, FI State	—

Les paramètres utilisés avec les primitives sont décrits dans la Spécification de service de l'objet Function Invocation (voir 6.2.8.3).

6.2.8.4.1.2.2 Primitives échangées entre les diagrammes d'états des LR et des invocations de fonction

Le Tableau 183 et le Tableau 184 montrent la primitive échangée entre le diagramme d'états des Load Region et le diagramme d'états des Function Invocation.

Tableau 183 – Primitives émises par le diagramme d'états des Load Region vers le diagramme d'états des Function Invocation

Nom de primitive	Source	Paramètres associés	Fonctions
DeleteInUse.cnf(+)	Load Region ("Région de charge")	FI Identifier, LR Identifier	L'objet Load Region confirme positivement, avec la primitive, une DeleteInUse.req émise précédemment.
SetInUse.cnf(+)	Load Region	FI Identifier, LR Identifier	L'objet Load Region confirme positivement, avec la primitive, une SetInUse.req émise précédemment.
SetInUse.cnf(-)	Load Region	FI Identifier, LR Identifier	L'objet Load Region confirme négativement, avec la primitive, une SetInUse.req émise précédemment.