

INTERNATIONAL STANDARD

**Industrial communication networks – Fieldbus specifications –
Part 3-2: Data-link layer service definition – Type 2 elements**

IECNORM.COM : Click to view the full PDF of IEC 61158-3-2:2023



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2023 IEC, Geneva, Switzerland

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

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

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

About the IEC

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

About IEC publications

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

IEC publications search - webstore.iec.ch/advsearchform

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 It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

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

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

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

IEC Products & Services Portal - products.iec.ch

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

Electropedia - www.electropedia.org

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

IECNORM.COM : Click to view the full text of IEC 61158-3-2:2023

INTERNATIONAL STANDARD

**Industrial communication networks – Fieldbus specifications –
Part 3-2: Data-link layer service definition – Type 2 elements**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 25.040.40; 35.100.20; 35.110

ISBN 978-2-8322-6558-1

Warning! Make sure that you obtained this publication from an authorized distributor.

CONTENTS

FOREWORD.....	6
INTRODUCTION.....	8
1 Scope.....	9
1.1 General.....	9
1.2 Specifications	9
1.3 Conformance	9
2 Normative references	10
3 Terms, definitions, symbols, abbreviated terms and conventions	10
3.1 Reference model terms and definitions	10
3.2 Service convention terms and definitions	12
3.3 Common data-link service terms and definitions	12
3.4 Additional Type 2 data-link specific definitions	14
3.5 Common symbols and abbreviations	16
3.6 Additional Type 2 symbols and abbreviations	17
3.7 Common conventions.....	17
4 Connection-mode and connectionless-mode data-link service	18
4.1 Overview.....	18
4.1.1 Data transfer services.....	18
4.1.2 DL-management services	22
4.1.3 Timing services	22
4.2 Facilities of the data-link service	22
4.3 Model of the data-link service	23
4.3.1 General	23
4.3.2 DLS-instance identification	23
4.3.3 Model of abstract queue concepts	23
4.3.4 QoS features	24
4.3.5 DLS-TxStatus	25
4.3.6 Receive queues	25
4.4 Sequence of primitives.....	25
4.4.1 Constraints on sequence of primitives	25
4.4.2 Relation of primitives at DLSAPs	26
4.4.3 Sequence of primitives at one DLSAP	27
4.5 Connection-mode data transfer	27
4.5.1 General	27
4.5.2 Function	27
4.5.3 Types of primitives and parameters	27
4.5.4 Sequence of primitives	29
4.6 Connectionless-mode data transfer.....	29
4.6.1 General	29
4.6.2 Function	29
4.6.3 Types of primitives and parameters	30
4.6.4 Sequence of primitives	32
4.7 Queue maintenance	32
4.7.1 Function	32
4.7.2 Types of primitives and parameters	32
4.7.3 Request primitive.....	33
4.7.4 Confirmation primitive.....	33

4.7.5	Sequence of primitives	33
4.8	Tag filter	34
4.8.1	Function	34
4.8.2	Types of primitives and parameters	34
4.8.3	Sequence of primitives	35
5	DL-management services	35
5.1	Sequence of primitives	35
5.2	Link synchronization	36
5.2.1	Function	36
5.2.2	Types of primitives and parameters	36
5.2.3	Sequence of primitives	37
5.3	Synchronized parameter change	37
5.3.1	Function	37
5.3.2	Types of primitives and parameters	37
5.3.3	Sequence of primitives	39
5.4	Event reports	40
5.4.1	Function	40
5.4.2	Types of primitives and parameters	40
5.4.3	Sequence of primitives	41
5.5	Bad FCS	41
5.5.1	Function	41
5.5.2	Types of primitives and parameters	41
5.5.3	Sequence of primitives	41
5.6	Current moderator	42
5.6.1	Function	42
5.6.2	Types of primitives and parameters	42
5.6.3	Sequence of primitives	42
5.7	Enable moderator	42
5.7.1	Function	42
5.7.2	Types of primitives and parameters	42
5.7.3	Sequence of primitives	43
5.8	Power-up and online	43
5.8.1	Function	43
5.8.2	Types of primitives and parameters	43
5.8.3	Sequence of primitives	44
5.9	Listen only	44
5.9.1	Function	44
5.9.2	Types of primitives and parameters	45
5.9.3	Sequence of primitives	45
5.10	Time distribution	45
5.10.1	Function	45
5.10.2	Types of primitives and parameters	45
	Bibliography	48
	Figure 1 – Relationships of DLSAPs, DLSAP-addresses and group DL-addresses	13
	Figure 2 – NUT structure	19
	Figure 3 – Medium access during scheduled time	20
	Figure 4 – Medium access during unscheduled time	21

Figure 5 – Queue model for the peer and multipoint DLS, DLSAPs and their DLCEPs	22
Figure 6 – Queue model of a multipoint DLS between a sending DLS-user and one or more receiving DLS-users	24
Figure 7 – DLS primitive time-sequence diagram	26
Figure 8 – State transition diagram for sequences of DLS primitives at one DLSAP	27
Figure 9 – Sequence of primitives for a successful connection-mode transfer	29
Figure 10 – Sequence of primitives for an unsuccessful connection-mode transfer	29
Figure 11 – Sequence of primitives for a successful connectionless-mode transfer	32
Figure 12 – Sequence of primitives for an unsuccessful connectionless-mode transfer	32
Figure 13 – Sequence of primitives for a queue maintenance request	34
Figure 14 – Sequence of primitives for a tag filter request	35
Figure 15 – Sequence of primitives for a local link synchronization	37
Figure 16 – Sequence of primitives for a DLM-get/set parameters request	39
Figure 17 – Sequence of primitives for a DLM-tMinus change request	39
Figure 18 – Sequence of primitives for a DLM-event indication	41
Figure 19 – Sequence of primitives for a DLM-bad-FCS indication	42
Figure 20 – Sequence of primitives for a DLM-current-moderator indication	42
Figure 21 – Sequence of primitives for a DLM-enable-moderator request	43
Figure 22 – Sequence of primitives for a DLM-power-up indication	44
Figure 23 – Sequence of primitives for a DLM-online request	44
Figure 24 – Sequence of primitives for a DLM-listen-only request	45
Table 1 – Summary of connection-mode and connectionless-mode primitives and parameters	26
Table 2 – DL-connection-mode transfer primitives and parameters	28
Table 3 – DL-connectionless-mode transfer primitives and parameters	30
Table 4 – Fixed tag services available to the DLS-user	31
Table 5 – DL-queue maintenance primitives and parameters (single flush)	33
Table 6 – DL-queue maintenance primitives and parameters (flush by QoS)	33
Table 7 – DL-connectionless-mode tag filter primitives and parameters	34
Table 8 – Summary of DL-management primitives and parameters	36
Table 9 – Link synchronization primitives and parameters	37
Table 10 – Synchronized parameter change primitives and parameters (set configuration)	38
Table 11 – Synchronized parameter change primitives and parameters (get configuration)	38
Table 12 – Synchronized parameter change primitives and parameters (start countdown)	38
Table 13 – Synchronized parameter change primitives and parameters (tMinus)	38
Table 14 – DLMS-configuration-data	39
Table 15 – Event report primitives and parameters	40
Table 16 – DLMS events being reported	40
Table 17 – Bad FCS primitives and parameters	41
Table 18 – Current moderator primitives and parameters	42
Table 19 – Enable moderator primitives and parameters	43

Table 20 – Power-up primitives and parameters.....	43
Table 21 – Online primitives and parameters	44
Table 22 – Listen-only primitives and parameters	45
Table 23 – DLMS time and time quality parameters	46
Table 24 – Time distribution source quality	46

IECNORM.COM : Click to view the full PDF of IEC 61158-3-2:2023

INTERNATIONAL ELECTROTECHNICAL COMMISSION

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 3-2: Data-link layer service definition – Type 2 elements

FOREWORD

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

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

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

IEC 61158-3-2 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation. It is an International Standard.

This third edition cancels and replaces the second edition published in 2014 and Amendment 1:2019. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

- a) update of normative and bibliographic references;
- b) use of more inclusive terminology ("master" replaced by "active" or "supervisor");
- c) miscellaneous editorial corrections.

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

Draft	Report on voting
65C/1201/FDIS	65C/1242/RVD

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

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

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

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

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under webstore.iec.ch in the data related to the specific document. At this date, the document will be

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

INTRODUCTION

This 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.

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 data-link layer service defined in this document is a conceptual architectural service, independent of administrative and implementation divisions.

IECNORM.COM : Click to view the full PDF of IEC 61158-3-2:2023

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 3-2: Data-link layer service definition – Type 2 elements

1 Scope

1.1 General

This part of IEC 61158 provides common elements for basic time-critical messaging communications between devices in an automation environment. 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 document defines in an abstract way the externally visible service provided by the Type 2 fieldbus data-link layer in terms of:

- the primitive actions and events of the service;
- the parameters associated with each primitive action and event, and the form which they take; and
- the interrelationship between these actions and events, and their valid sequences.

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

- the Type 2 fieldbus application layer at the boundary between the application and data-link layers of the fieldbus reference model;
- systems management at the boundary between the data-link layer and systems management of the fieldbus reference model.

Type 2 DL-service provides both a connected and a connectionless subset of those services specified in ISO/IEC 8886.

1.2 Specifications

The principal objective of this document is to specify the characteristics of conceptual data-link layer services suitable for time-critical communications and thus supplement the OSI Basic Reference Model in guiding the development of data-link protocols for time-critical communications. A secondary objective is to provide migration paths from previously existing industrial communications protocols.

This document can be used as the basis for formal DL-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:

- the sizes and octet ordering of various multi-octet service parameters;
- the correlation of paired request and confirm, or indication and response, primitives.

1.3 Conformance

This document does not specify individual implementations or products, nor does it constrain the implementations of data-link entities within industrial automation systems.

There is no conformance of equipment to this data-link layer service definition standard. Instead, conformance is achieved through implementation of the corresponding data-link protocol that fulfills the Type 2 data-link layer services defined in this document.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

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

IEC 61158-4-2:2023, *Industrial communication networks – Fieldbus specifications – Part 4-2: Data-link layer protocol specification – Type 2 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 8886, *Information technology – Open Systems Interconnection – Data link service definition*

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

3 Terms, definitions, symbols, abbreviated terms and conventions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

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

3.1 Reference model terms and definitions

This document is based in part on the concepts developed in ISO/IEC 7498-1 and ISO/IEC 7498-3, and makes use of the following terms defined therein:

3.1.1	DL-address	[ISO/IEC 7498-3]
3.1.2	DL-address-mapping	[ISO/IEC 7498-1]
3.1.3	called-DL-address	[ISO/IEC 7498-3]
3.1.4	calling-DL-address	[ISO/IEC 7498-3]
3.1.5	centralized multi-end-point-connection	[ISO/IEC 7498-1]
3.1.6	DL-connection	[ISO/IEC 7498-1]
3.1.7	DL-connection-end-point	[ISO/IEC 7498-1]
3.1.8	DL-connection-end-point-identifier	[ISO/IEC 7498-1]
3.1.9	DL-connection-mode transmission	[ISO/IEC 7498-1]
3.1.10	DL-connectionless-mode transmission	[ISO/IEC 7498-1]
3.1.11	correspondent (N)-entities correspondent DL-entities (N=2) correspondent Ph-entities (N=1)	[ISO/IEC 7498-1]
3.1.12	DL-duplex-transmission	[ISO/IEC 7498-1]
3.1.13	(N)-entity DL-entity (N=2) Ph-entity (N=1)	[ISO/IEC 7498-1]
3.1.14	DL-facility	[ISO/IEC 7498-1]
3.1.15	flow control	[ISO/IEC 7498-1]
3.1.16	(N)-layer DL-layer (N=2) Ph-layer (N=1)	[ISO/IEC 7498-1]
3.1.17	layer-management	[ISO/IEC 7498-1]
3.1.18	DL-local-view	[ISO/IEC 7498-3]
3.1.19	DL-name	[ISO/IEC 7498-3]
3.1.20	naming-(addressing)-domain	[ISO/IEC 7498-3]
3.1.21	peer-entities	[ISO/IEC 7498-1]
3.1.22	primitive name	[ISO/IEC 7498-3]
3.1.23	DL-protocol	[ISO/IEC 7498-1]
3.1.24	DL-protocol-connection-identifier	[ISO/IEC 7498-1]
3.1.25	DL-protocol-data-unit	[ISO/IEC 7498-1]
3.1.26	DL-relay	[ISO/IEC 7498-1]
3.1.27	reset	[ISO/IEC 7498-1]
3.1.28	responding-DL-address	[ISO/IEC 7498-3]
3.1.29	routing	[ISO/IEC 7498-1]
3.1.30	segmenting	[ISO/IEC 7498-1]
3.1.31	(N)-service DL-service (N=2) Ph-service (N=1)	[ISO/IEC 7498-1]
3.1.32	(N)-service-access-point DL-service-access-point (N=2) Ph-service-access-point (N=1)	[ISO/IEC 7498-1]
3.1.33	DL-service-access-point-address	[ISO/IEC 7498-3]
3.1.34	DL-service-connection-identifier	[ISO/IEC 7498-1]
3.1.35	DL-service-data-unit	[ISO/IEC 7498-1]
3.1.36	DL-simplex-transmission	[ISO/IEC 7498-1]

3.1.37	DL-subsystem	[ISO/IEC 7498-1]
3.1.38	systems-management	[ISO/IEC 7498-1]
3.1.39	DLS-user-data	[ISO/IEC 7498-1]

3.2 Service convention terms and definitions

This document also makes use of the following terms defined in ISO/IEC 10731 as they apply to the data-link layer:

- 3.2.1** **acceptor**
- 3.2.2** **asymmetrical service**
- 3.2.3** **confirm (primitive);
requestor.deliver (primitive)**
- 3.2.4** **deliver (primitive)**
- 3.2.5** **DL-confirmed-facility**
- 3.2.6** **DL-facility**
- 3.2.7** **DL-local-view**
- 3.2.8** **DL-mandatory-facility**
- 3.2.9** **DL-non-confirmed-facility**
- 3.2.10** **DL-provider-initiated-facility**
- 3.2.11** **DL-provider-optional-facility**
- 3.2.12** **DL-service-primitive;
primitive**
- 3.2.13** **DL-service-provider**
- 3.2.14** **DL-service-user**
- 3.2.15** **DLS-user-optional-facility**
- 3.2.16** **indication (primitive);
acceptor.deliver (primitive)**
- 3.2.17** **multi-peer**
- 3.2.18** **request (primitive);
requestor.submit (primitive)**
- 3.2.19** **requestor**
- 3.2.20** **response (primitive);
acceptor.submit (primitive)**
- 3.2.21** **submit (primitive)**
- 3.2.22** **symmetrical service**

3.3 Common data-link service terms and definitions

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

NOTE Many definitions are common to more than one protocol Type; they are not necessarily used by all protocol Types.

3.3.1

DL-segment link local link

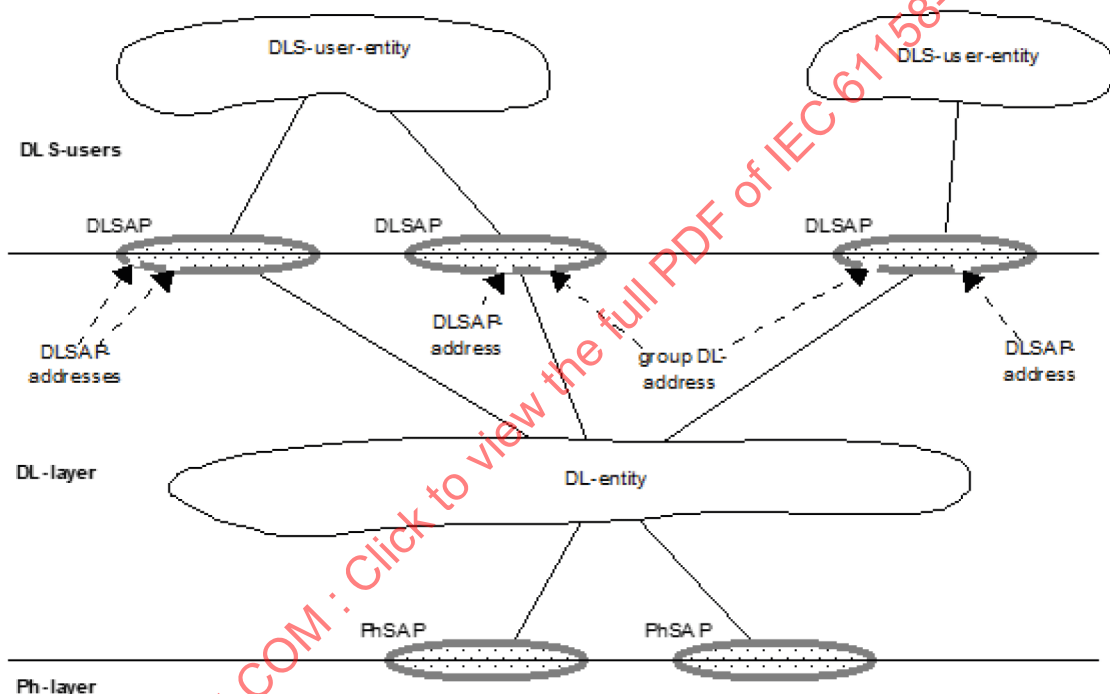
single DL-subnetwork in which any of the connected DLEs may communicate directly, without any intervening DL-relaying, whenever all of those DLEs that are participating in an instance of communication are simultaneously attentive to the DL-subnetwork during the period(s) of attempted communication

3.3.2

DLSAP

distinctive point at which DL-services are provided by a single DL-entity to a single higher-layer entity

Note 1 to entry: This definition, derived from ISO/IEC 7498-1, is repeated here to facilitate understanding of the critical distinction between DLSAPs and their DL-addresses (see Figure 1).



IEC

NOTE 1 DLSAPs and PhSAPs are depicted as ovals spanning the boundary between two adjacent layers.

NOTE 2 DL-addresses are depicted as designating small gaps (points of access) in the DLL portion of a DLSAP.

NOTE 3 A single DL-entity can have multiple DLSAP-addresses and group DL-addresses associated with a single DLSAP.

Figure 1 – Relationships of DLSAPs, DLSAP-addresses and group DL-addresses

3.3.3

DL(SAP)-address

either an individual DLSAP-address, designating a single DLSAP of a single DLS-user, or a group DL-address potentially designating multiple DLSAPs, each of a single DLS-user

Note 1 to entry: This terminology is chosen because ISO/IEC 7498-3 does not permit the use of the term DLSAP-address to designate more than a single DLSAP at a single DLS-user.

3.3.4

(individual) DLSAP-address

DL-address that designates only one DLSAP within the extended link

Note 1 to entry: A single DL-entity may have multiple DLSAP-addresses associated with a single DLSAP.

3.3.5

extended link

DL-subnetwork, consisting of the maximal set of links interconnected by DL-relays, sharing a single DL-name (DL-address) space, in which any of the connected DL-entities may communicate, one with another, either directly or with the assistance of one or more of those intervening DL-relay entities

Note 1 to entry: An extended link may be composed of just a single link.

3.3.6

frame

denigrated synonym for DLPDU

3.3.7

group DL-address

DL-address that potentially designates more than one DLSAP within the extended link

Note 1 to entry: A single DL-entity may have multiple group DL-addresses associated with a single DLSAP. A single DL-entity also may have a single group DL-address associated with more than one DLSAP.

3.3.8

node

single DL-entity as it appears on one local link

3.3.9

receiving DLS-user

DL-service user that acts as a recipient of DLS-user-data

Note 1 to entry: A DL-service user can be concurrently both a sending and receiving DLS-user.

3.3.10

sending DLS-user

DL-service user that acts as a source of DLS-user-data

3.4 Additional Type 2 data-link specific definitions

3.4.1

application

function or data structure for which data is subscribed or published

3.4.2

behavior

indication of how the object responds to particular events

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

3.4.3

bridge, DL-router

DL-relay entity which performs selective store-and-forward and routing functions to connect two or more separate DL-subnetworks (links) to form a unified DL-subnetwork (the extended link)

3.4.4

cyclic

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

3.4.5

device

physical hardware connection to the link

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

3.4.6

DL-subnetwork

series of nodes connected by PhEs and, where appropriate, DL-routers

3.4.7

DLPDU

Data-link Protocol Data unit

Note 1 to entry: A DLPDU consists of a source MAC ID, zero or more Lpackets, and an FCS, as transmitted or received by an associated PhE.

3.4.8

error

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

3.4.9

fixed tag

two octet identifier (tag) which identifies a specific service to be performed by either

- a) that receiving node on the local link which has a specified MAC ID, or
- b) all receiving nodes on the local link

Note 1 to entry: Identification of the target node(s) is included in the two octet tag.

3.4.10

generic tag

three octet identifier (tag) which identifies a specific piece of application information

3.4.11

guardband

time slot allocated for the transmission of the moderator DLPDU

3.4.12

link

collection of nodes with unique MAC IDs

Note 1 to entry: Ph-segments connected by Ph-repeaters make up a link; links connected by DL-routers make up an extended link (sometimes called a local area network).

3.4.13

Lpacket

well-defined sub-portion of a DLPDU containing (among other things)

- c) a fixed tag or a generic tag, and
- d) DLS-user data or, when the tag has DL-significance, DL-data

3.4.14

moderator

node with the lowest MAC ID that is responsible for transmitting the moderator DLPDU

3.4.15

moderator DLPDU

DLPDU transmitted by the node with the lowest MAC ID for the purpose of synchronizing the nodes and distributing the link configuration parameters

3.4.16 multipoint DLC

centralized multi-end-point DL-connection offering DL-simplex-transmission between a single distinguished DLS-user, known as the publisher or publishing DLS-user, and a set of peer but undistinguished DLS-users, known collectively as the subscribers or subscribing DLS-users, where the publishing DLS-user can send to the subscribing DLS-users as a group (but not individually)

Note 1 to entry: A multipoint DLC always provides asymmetrical service.

3.4.17 node

logical connection to a local link, requiring a single MAC ID

Note 1 to entry: A single physical device can appear as many nodes on the same local link. For the purposes of this protocol, each node is considered to be a separate DLE.

3.4.18 peer-to-peer DLC

point-to-point DL-connection offering DL-simplex-transmission between a single distinguished sending DLS-user and a single distinguished receiving DLS-user

Note 1 to entry: A peer-to-peer DLC always provides asymmetrical service.

3.4.19 rogue

node that has received a moderator DLPDU that disagrees with the link configuration currently used by this node

3.4.20 scheduled

data transfers that occur in a deterministic and repeatable manner on predefined NUTs.

3.4.21 tMinus

number of NUTs before a new set of link configuration parameters are to be used

3.4.22 tone

instant of time which marks the boundary between two NUTs

3.4.23 unscheduled

data transfers that use the remaining allocated time in the NUT after the scheduled transfers have been completed

3.5 Common symbols and abbreviations

NOTE Many symbols and abbreviations are common to more than one protocol Type; they are not necessarily used by all protocol Types.

DL-	Data-link layer (as a prefix)
DLC	DL-connection
DLCEP	DL-connection-end-point
DLE	DL-entity (the local active instance of the data-link layer)
DLL	DL-layer
DLPCI	DL-protocol-control-information
DLPDU	DL-protocol-data-unit

DLM	DL-management
DLME	DL-management Entity (the local active instance of DL-management)
DLMS	DL-management Service
DLS	DL-service
DLSAP	DL-service-access-point
DLSDU	DL-service-data-unit
FIFO	First-in first-out (queuing method)
OSI	Open systems interconnection
Ph-	Physical layer (as a prefix)
PhE	Ph-entity (the local active instance of the physical layer)
PhL	Ph-layer
QoS	Quality of service

3.6 Additional Type 2 symbols and abbreviations

MAC ID	DL-address of a node
MDS	Medium dependent sublayer
NUT	Network (actually, local link) update time

NOTE The use of the term "network" in the preceding definition is maintained for historic reasons, even though the scope involved is only a portion of a single DL-subnetwork.

RMS	root mean square
SMAx	MAC ID of the maximum scheduled node
Tx	Transmit
TUI	Table unique identifier
UCMM	Unconnected message manager
UMAx	MAC ID of maximum unscheduled node
USR	Unscheduled start register

3.7 Common conventions

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

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

Service primitives, used to represent service user/service provider interactions (see ISO/IEC 10731), convey parameters that indicate information available in the user/provider interaction.

This document uses a tabular format to describe the component parameters of the DLS primitives. The parameters that apply to each group of DLS primitives are set out in tables throughout the remainder of this document. Each table consists of up to six columns, containing the name of the service parameter, and a column each for those primitives and parameter-transfer directions used by the DLS:

- the request primitive's input parameters;
- the request primitive's output parameters;
- the indication primitive's output parameters;

- the response primitive's input parameters; and
- the confirm primitive's output parameters.

NOTE The request, indication, response and confirm primitives are also known as requestor.submit, acceptor.deliver, acceptor.submit, and requestor.deliver primitives, respectively (see ISO/IEC 10731).

One parameter (or part 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 and parameter direction specified in the column.

- M** – parameter is mandatory for the primitive.
- U** – parameter is a User option, and can be provided or not depending on the dynamic usage of the DLS-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 DLS-user.
- (blank) – parameter is never present.

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.

In any particular interface, not all parameters need be explicitly stated. Some may be implicitly associated with the DLSAP at which the primitive is issued.

In the diagrams which illustrate these interfaces, dashed lines indicate cause-and-effect or time-sequence relationships, and wavy lines indicate that events are roughly contemporaneous.

4 Connection-mode and connectionless-mode data-link service

4.1 Overview

4.1.1 Data transfer services

The primary task of a DLE is to determine, in co-operation with other DLEs on the same local link, the granting of permission to transmit on the medium. At its upper interface, the DLL provides services to receive and deliver service data units (DLSDUs) for higher level entities.

NOTE 1 The following access mechanisms are not visible to the higher level entities. They are described here as an aid to understanding the purpose and use of DLS parameters and services that are visible to higher layer entities.

This DLL protocol is based on a fixed repetitive time cycle, called the network update time (NUT). The NUT is maintained in close synchronism among all nodes on the local link. A node is not permitted access to transmit if its configured NUT does not agree with the NUT currently being used on the local link. Different local links within the extended link may have different NUT durations.

Each node contains its own timer synchronized to the local link's NUT. Medium access is determined by local sub-division of the NUT into variable-duration access slots. Access to the medium is in sequential order based on the MAC ID of the node. Specific behaviors have been incorporated into the access protocol allowing a node which temporarily assumes a MAC ID of zero to perform link maintenance. The MAC ID numbers of all nodes on a link are unique. Any

DLE detecting the presence of a MAC ID duplicating its own MAC ID immediately stops transmitting.

An implicit token passing mechanism is used to grant access to the medium. Each node monitors the source MAC ID of each DLPDU received. At the end of a DLPDU, each DLE sets an "implicit token register" to the received source MAC ID + 1. If the implicit token register is equal to the local MAC ID, then the DLE transmits one DLPDU containing zero or more Lpackets with data. In all other cases, the node watches for either a new DLPDU from the node identified by the "implicit token register" or a time-out value if the identified node fails to transmit. In each case, the "implicit token" is automatically advanced to the next MAC ID. All nodes have the same value in their "implicit token register" preventing collisions on the medium.

The time-out period (called the "slot time") is based on the amount of time required for

- a) the current node to hear the end of the transmission from the previous node, and
- b) the current node to begin transmitting, and
- c) the next node to hear the beginning of the transmission from the current node.

The slot time is adjusted to compensate for the total length of the medium since the propagation delay of the medium effects the first and last item on the previous list.

NOTE 2 The calculation of slot time is the responsibility of System Management.

Each NUT is divided into three major parts: scheduled, unscheduled, and guardband as shown in Figure 2. This sequence is repeated in every NUT. The implicit token passing mechanism is used to grant access to the medium during both the unscheduled and scheduled intervals.

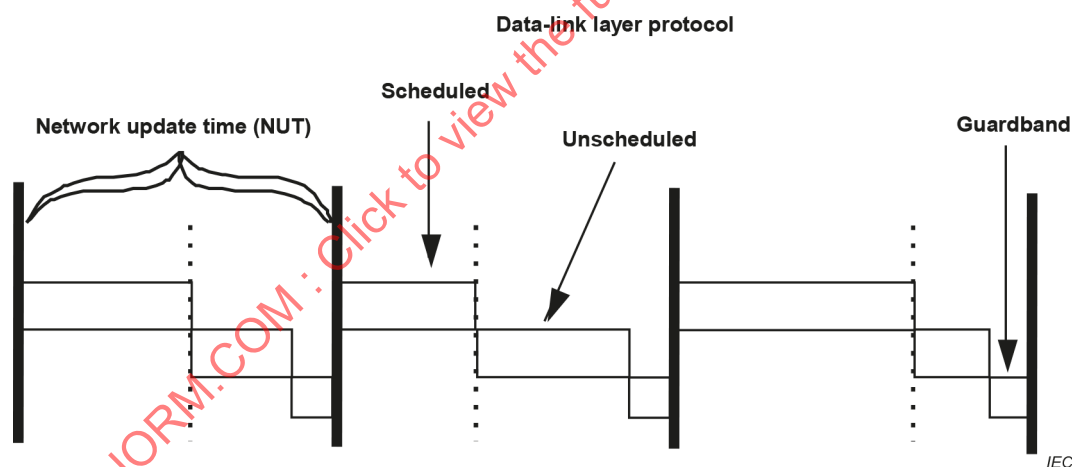
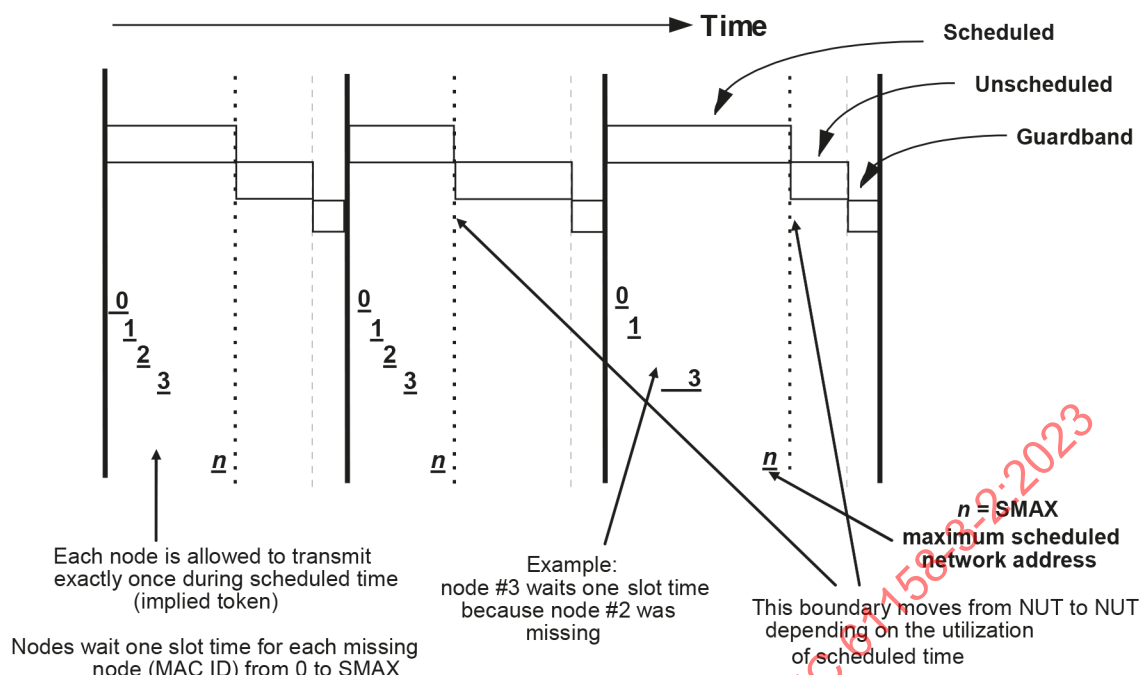


Figure 2 – NUT structure

During the scheduled part of the NUT, each node, starting with node 0 and ending with node SMAX, gets a chance to transmit time-critical (scheduled) data. SMAX is the MAC ID of the highest numbered node that has access to the medium during the scheduled part of the NUT. Every node between 0 and SMAX has only one opportunity to send one DLPDU of scheduled data in each NUT. The opportunity to access the medium during the scheduled time is the same for each node in every NUT. This allows data that is transmitted during the scheduled portion of the NUT to be sent in a predictable and deterministic manner.

Figure 3 shows how the permission to transmit is granted during the scheduled time. The DLS-user regulates the amount of data that each node may transmit during this scheduled token pass.



IEC

Figure 3 – Medium access during scheduled time

During the unscheduled part of the NUT, each node from 0 to UMAX shares the opportunity to transmit one DLPDU of non-time-critical data in a round robin fashion, until the allocated NUT duration is exhausted. UMAX is the MAC ID of the highest numbered node that has access to the medium during the unscheduled part of the NUT. The round robin method of access opportunity enables every node between 0 and UMAX to have zero, one or many opportunities to send unscheduled data depending on how much of the NUT remains after the completion of the scheduled time. Variations in scheduled traffic means the opportunity to access the medium during the unscheduled time can be different for each node in every NUT.

Figure 4 shows how the permission to transmit is granted during the unscheduled time. The MAC ID of the node that goes first in the unscheduled part of the NUT is incremented by 1 for each NUT. The unscheduled token begins at the MAC ID specified in the unscheduled start register (USR) of the previous moderator DLPDU. The USR increments by one modulo (UMAX+1) each NUT. If the USR reaches UMAX before the guardband, it returns to zero and the token pass continues.

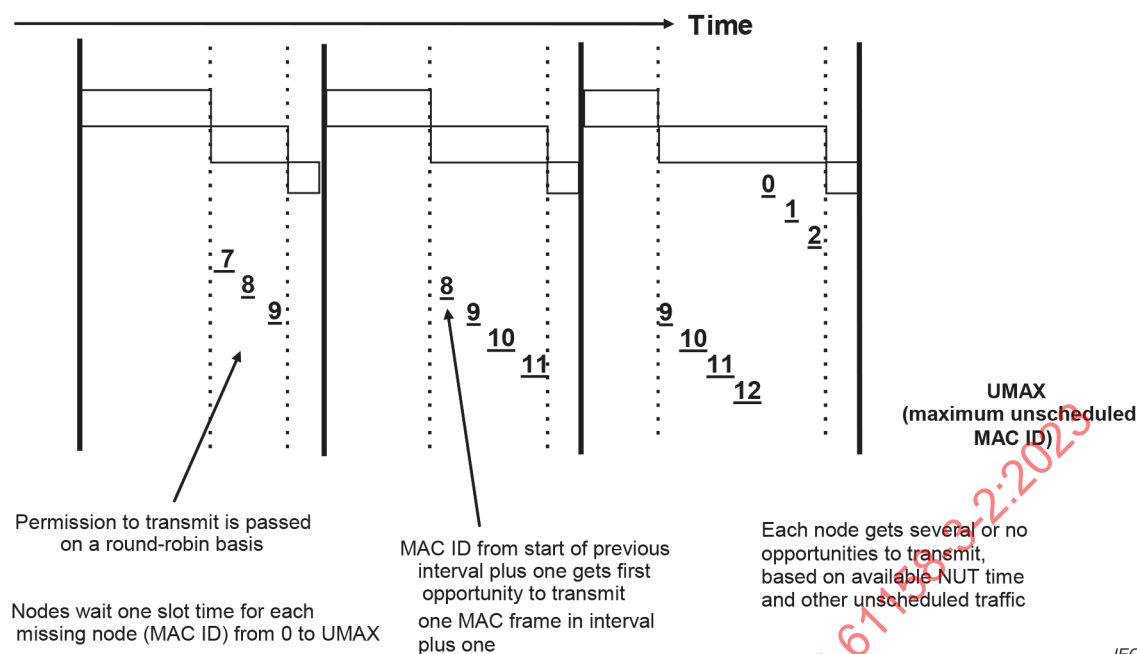


Figure 4 – Medium access during unscheduled time

When the guardband is reached, all nodes stop transmitting. A node is not allowed to start a transmission unless it can be completed before the beginning of the guardband. During the guardband, the node with the lowest MAC ID (called the "moderator") transmits a maintenance message (called the "moderator DLPDU") that accomplishes two things.

- 1) It keeps the NUT timers of all nodes synchronized.
- 2) It publishes critical link parameters enabling all DLEs on the local link to share a common version of important local link values such as NUT, slot time, SMAX, UMAX, USR, etc.

The moderator transmits the moderator DLPDU, which re-synchronizes all nodes and restarts the NUT. Following the receipt of a valid moderator DLPDU, each node compares its internal values with those transmitted in the moderator DLPDU. A node using link parameters that disagree with the moderator disables itself. If the moderator DLPDU is not heard for two consecutive NUTs, the node with the lowest MAC ID assumes the moderator role and begins transmitting the moderator DLPDU in the guardband of the third NUT. A moderator node that notices another node online and transmitting with a MAC ID lower than its own immediately cancels its moderator role.

Situations that can cause disruption of the DL-protocol arise due to problems in the underlying PhL service. Some examples of the types of PhL problems which can disrupt the DL-protocol are:

- induced noise within the distributed PhE;
- poor quality PhE components or installation practices;
- physically connecting two Ph-segments together while the link is operating.

One common consequence of such disruption is that nodes can be caused to disagree as to which node should be transmitting; this is called a "non-concurrence". Another potential problem occurs when the nodes do not agree to the same values of the link configuration parameters. A node that disagrees with the link parameters as transmitted by the moderator is called a "rogue" and immediately stops transmitting. The DL-protocol is designed to recover a rogue node and bring it back online.

4.1.2 DL-management services

DL-management services support:

- a) setting of address filters by receiving DLS users;
- b) queue maintenance support for sending DLS users;
- c) local link synchronization and online change of local link parameters;
- d) event reporting of important variables and events within the layer;
- e) non-disruptive addition of nodes to the link;
- f) tuning of link parameters;
- g) time distribution and clock synchronization between nodes.

4.1.3 Timing services

This DLL is quite flexible. It can provide deterministic and synchronized I/O transfer at cyclic intervals up to 1 ms and node separations up to 25 km. This performance is adjustable online by configuring the link parameters of the local link. These parameters, which govern the access to the link, can be tuned as required to match different applications. DL-management allows these parameters to be changed online, while the local link is operating, it also allows the local link to continue functioning while connections to new nodes are added and removed.

DLEs can maintain clock synchronization across the extended link with a precision better than 10 μ s.

4.2 Facilities of the data-link service

The DLS provides the following facilities to the DLS-user:

- a) A means of transferring DLSDUs of limited length between two or more DLS-users who have negotiated peer or multipoint connection-mode services, see Figure 5.

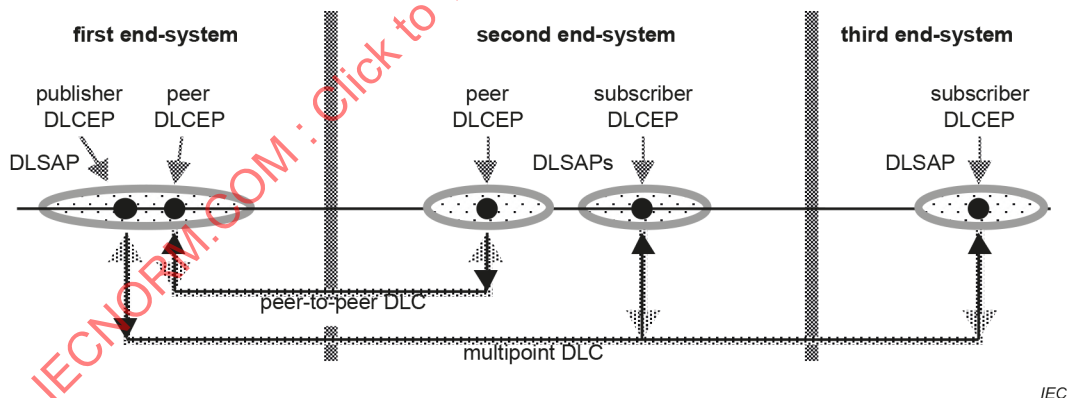


Figure 5 – Queue model for the peer and multipoint DLS, DLSAPs and their DLCEPs

- b) A means of maintaining time synchronization for service execution and cyclic transfer of DLSDUs based on selected QoS parameters.
- c) A means of transferring DLSDUs of limited length from one source DLSAP to a destination DLSAP or group of DLSAPs, without establishing or later releasing a DLC. The transfer of DLSDUs is transparent, in that the boundaries of DLSDUs and the contents of DLSDUs are preserved unchanged by the DLS, and there are no constraints on the DLSDU content (other than limited length) imposed by the DLS. QoS for this transmission can be selected by the sending DLS-user.

NOTE The length of a DLSDU is limited because of internal mechanisms employed by the DL-protocol.

- d) A means by which the status of dispatch to the destination DLSAP or group of DLSAPs can be returned to the source DLSAP.

- e) A means of cancelling either a specific outstanding DLSDU transfer service request, or all outstanding DLSDU transfer service requests of a specified QoS.

4.3 Model of the data-link service

4.3.1 General

This document uses the abstract model for a layer service defined in ISO/IEC 10731:1994, Clause 5. The model defines interactions between the DLS-user and the DLS provider that take place at a DLSAP. Information is passed between the DLS-user and the DLS provider by DLS primitives that convey parameters.

4.3.2 DLS-instance identification

A DLS-user is able to distinguish among several DLCEPs at the same DLSAP. This is done by an address structure named generic-tag and supported by address filtering services available to each receiving DLS-user.

For connectionless service, a DLS-user is able to distinguish among several DLSAPs using an address structure named fixed-tag. Address filtering services are available for each receiving DLS-user.

A local identification mechanism is provided for each use of the DLS which needs to correlate a confirmation or subsequent cancellation request with its associated request.

4.3.3 Model of abstract queue concepts

4.3.3.1 General

After establishment of the DLC using a generic-tag address, there exists a relationship between the publishing DLS-user and the subscribing DLS-user(s).

DL services using a fixed-tag address do not need establishment as they use pre-defined fixed relationships between permanent DLSAPs associated with each DLS-user.

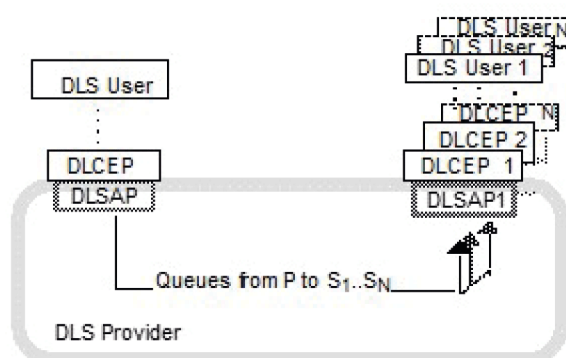
As a means of specifying these relationships, an abstract queue model of a multipoint DLC, which is described in 4.3.3.2, is used.

NOTE 1 Establishment and management of a DLC and its identifying generic-tag is provided by higher layer entities above the DLS-interface.

NOTE 2 The internal mechanisms that support the operation of the DLS are not visible to the DLS-user.

4.3.3.2 Queue model concepts

The queue model represents the operation of a multipoint DLS in the abstract by a set of abstract queues linking the sending DLSAP-user with the receiving DLSAP-user(s) – one queue per receiving DLSAP (see Figure 6).



IEC

Figure 6 – Queue model of a multipoint DLS between a sending DLS-user and one or more receiving DLS-users

Each queue represents one direction of transfer. The ability of a sending or receiving DLS-user to remove objects from a queue is determined by the behavior of the DLS provider.

DLSDU objects identified by DL-generic-tag primitives or DL-fixed-tag primitives and their parameters can be placed in the abstract queue by the sending DLS-user and will be delivered to receiving DLS-users as determined by the DLSDU object's associated address and QoS parameters.

Queue management services are available to the sending DLS-user for flushing unsent objects from a transmit queue. These can be either identified individual objects or all objects loaded at a specific QoS.

4.3.4 QoS features

4.3.4.1 Sending priority and timing

The available QoS options for the connection-mode and connectionless-mode services are sending priority and timing.

The choice of sending priority implicitly selects the timing characteristics of the DLS supplier execution of the transmission. Three alternative priorities are available: scheduled, high and low.

NOTE 1 To ensure guaranteed access, the active Keeper uses scheduled priority for regular publication of a TUI fixed tag message containing the current Table Unique Identifier (TUI). The TUI is a unique reference to the current link and node configuration parameters. All participating DLEs receive the TUI and use it to ensure their link details are current.

High and low priorities are recommended for all connectionless-mode services except those involved with TUI messages.

NOTE 2 High and low priorities are used only in a local sense to set the order of servicing locally submitted DLS-user-data; they do not have link-wide connotations.

4.3.4.2 Scheduled priority

This QoS provides accurate time-based cyclic and acyclic sending of DLSDUs. The execution timing for this scheduled service can be accurate and repeatable to better than 1 ms.

4.3.4.3 High priority

This QoS provides acyclic sending of DLSDUs with a bounded upper time for the sending delay. Data on this priority is sent only when all scheduled data has been sent and a non-scheduled sending opportunity is available.

4.3.4.4 Low priority

This QoS provides sending of DLSDUs only on a time-available basis. Data on this priority is sent only when all other priorities of data have been sent and a non-scheduled sending opportunity is available.

4.3.5 DLS-TxStatus

This parameter allows a sending DLS-user to determine the status of a corresponding requested transmission. The value conveyed in this parameter is as follows:

- a) "OK" — success — message successfully sent;
- b) "TXABORT" — failure — sending process failed;
- c) "FLUSHED" — failure — message has been removed from the pending queue before being sent.

NOTE 1 The FLUSHED status is only used in response to the Queue maintenance service of 4.7.

NOTE 2 The parameter value OK is not an indication that the message has been received.

4.3.6 Receive queues

The receiving DLS-user has an implicit queue of indeterminate capacity which is used as the receive queue, and the DLSDU is delivered as the DLS-user-data parameter of the associated indication primitive.

If it is not possible to append the received DLSDU to the receive queue, then the DLSDU is discarded and an indication primitive is not issued to the DLS-user.

4.4 Sequence of primitives

4.4.1 Constraints on sequence of primitives

Subclause 4.4.1 defines the constraints on the sequence in which the primitives defined in 4.5 and 4.6 may occur. The constraints determine the order in which primitives occur, but do not fully specify when they may occur. Other aspects of actual system operation, such as PHL problems affecting messages in transit, will affect the ability of a DLS-user or a DLS provider to issue a primitive at any particular time.

The connection-mode and connectionless-mode primitives and their parameters are summarized in Table 1.

Table 1 – Summary of connection-mode and connectionless-mode primitives and parameters

Service	Service subtype	Primitive	Parameter
Data Transfer	Connection-mode	DL-GENERIC-TAG request	(<i>in</i> request DLS-user-identifier, DLS-user-data, DLS-QoS, DLS-generic-tag)
		DL-GENERIC-TAG indication	(<i>out</i> DLS-user-data, DLS-generic-tag)
		DL-GENERIC-TAG confirm	(<i>out</i> DLS-TxStatus)
	Connectionless-mode	DL-FIXED-TAG request	(<i>in</i> request DLS-user-identifier, DLS-user-data, DLS-QoS, DLS-fixed-tag, DLS-destination-DLE-ID)
		DL-FIXED-TAG indication	(<i>out</i> DLS-user-data, DLS-fixed-tag, DLS-source-DLE-ID)
		DL-FIXED-TAG confirm	(<i>out</i> DLS-TxStatus)
Super vision	Queue maintenance	DL-FLUSH-SINGLE-REQUEST request	(<i>in</i> request DLS-user-identifier)
		DL-FLUSH-SINGLE-REQUEST confirm	(<i>out</i> DLS-TxStatus)
		DL-FLUSH-REQUESTS-BY-QoS request	(<i>in</i> DLS-QoS)
		DL-FLUSH-REQUESTS-BY-QoS confirm	<none>
	Tag filter	DL-ENABLE-TAG request	(<i>in</i> DLS-tag)
		DL-ENABLE-TAG confirm	(<i>out</i> DLS-result)
		DL-DISABLE-TAG request	(<i>in</i> DLS-tag)
		DL-DISABLE-TAG confirm	(<i>out</i> DLS-result)

NOTE 1 Request DLS-user-identifiers are locally assigned by the DLS-user and used to flush a specific request from the DLS-provider's queues.

NOTE 2 The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.

4.4.2 Relation of primitives at DLSAPs

With few exceptions, a primitive issued at one DLSAP will have consequences at one or more other DLSAPs. The relations of primitives of each type at one DLSAP to primitives at the other DLSAPs are defined in 4.5 and 4.6, and summarized in Figure 7.

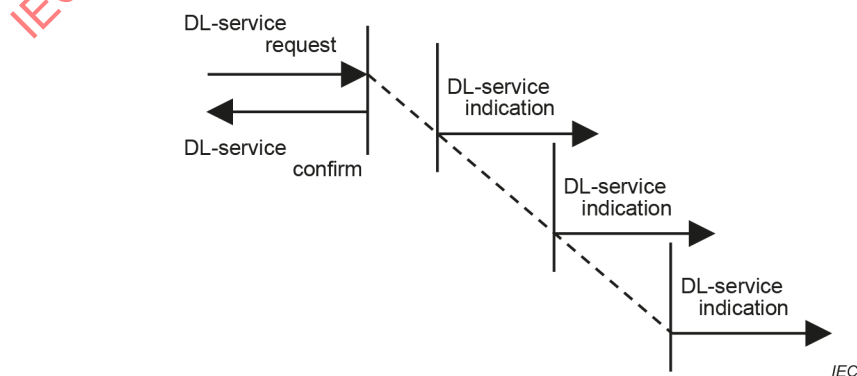


Figure 7 – DLS primitive time-sequence diagram

4.4.3 Sequence of primitives at one DLSAP

The possible overall sequences of primitives at a DLSAP are defined in the state transition diagram shown in Figure 8. In the diagram, the use of a state transition diagram to describe the allowable sequences of service primitives does not impose any requirements or constraints on the internal organization of any implementation of the service.

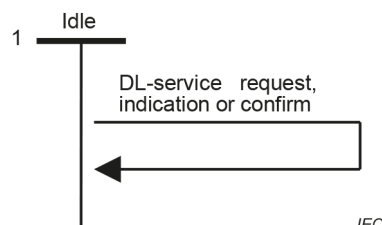


Figure 8 – State transition diagram for sequences of DLS primitives at one DLSAP

4.5 Connection-mode data transfer

4.5.1 General

DL-connection-mode service primitives can be used to transmit DLSDUs from one DLSAP to one or more peer DLSAPs using a generic-tag address to identify a connection between DLS-users. Each DLSDU is transmitted in a single DLPDU. All the information required to deliver the DLSDU is presented to the DLS provider, together with the user data to be transmitted, in a single service access.

DLS-users which are higher layer protocol entities can provide negotiation and management of connections above the DLL through additional interpretation of the DLS-generic-tag.

No means are provided by which the receiving DLS-user may control the rate at which the sending DLS-user may send DLSDUs. This is managed externally by appropriate scheduling tools which match the capability of sending and receiving DLS users and the configured service schedule of the DLS provider.

4.5.2 Function

This service provides the facilities of 4.2 a), b), c), d) and e). It can be used to transmit a DL-connection-mode DLSDU from one DLSAP to another or to a group of DLSAPs, in a single service access.

NOTE Delivery status (if required) is provided by higher-layer services provided by the DLS-user, it is not returned as part of the local DLS invocation.

In the absence of errors, the DLS provider maintains the integrity of individual DLSDUs, and delivers them to the receiving DLS-users in the order in which they are presented by the sending DLS-user.

4.5.3 Types of primitives and parameters

4.5.3.1 Primitive specifications

Table 2 indicates the types of primitives and the parameters needed for the DL-connection-mode transmission service.

Table 2 – DL-connection-mode transfer primitives and parameters

DL-GENERIC-TAG Parameter name	Request	Indication	Confirm
	input	output	output
Request DLS-user-identifier (handle)	M		
DLS-user-data (packet)	M	M(=)	
DLS-QoS (priority)	M		
DLS-generic-tag	M	M(=)	
DLS-TxStatus			M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

4.5.3.2 Request DLS-user-identifier

This parameter, which is specified by the DLS-user on cancelable DL-request primitives, provides a local means by which the DLS-user can subsequently attempt to cancel that request through a DL-FLUSH-SINGLE queue maintenance request. The naming-domain of this identifier is the DLS-user-local-view.

4.5.3.3 DLS-user data

This parameter provides the data to be transmitted between DLS-users without alteration by the DLS provider. The initiating DLS-user may transmit any integral number of octets greater than zero, up to the limit determined by the service type parameter specified in the service request.

4.5.3.4 DLS-QoS

This parameter is specified in 4.3.4.

4.5.3.5 DLS-generic-tag

This parameter conveys a connection identification or DLSAP-address identifying the remote DLSAP(s) to which the DLS is to be provided. It is a DL(SAP)-address in the request primitive, but takes the form of a local DL(SAP)-address DLS-user-identifier in the indication primitive(s). It may be a DLSAP-address or a multi-cast DL-address.

4.5.3.6 Request primitive

If the initiating DLS-user has implemented a FIFO queue of maximum depth K as a source queue for the DLSAP-address at the specified QoS priority, then a DL-GENERIC-REQUEST primitive attempts to append a DLSDU to the queue, but fails if the queue already contains K DLSDUs. If the append operation is successful, then the DLSDU will be transmitted at the first opportunity, after all preceding DLSDUs in the queue.

NOTE 1 The queue provides a means of managing multiple DLS-user requests for the efficiency advantage of combining them in a single transmission opportunity.

NOTE 2 The queue depth K is implementation specific.

4.5.3.7 Indication primitive for DLSDUs associated with generic tags

The receiving DLS-user is able to identify Generic Tag values of interest to it and pass them to the local DLS provider using the DLS-tag-filter management services. The set of local tag values are used to filter arriving associated DLSDUs. For DLSDUs with associated Generic tags that are acceptable to the filter, the following indication parameters are delivered to the local DLS-user:

- DLS-user-data;
- DLS-generic-tag, the value of the generic tag associated with the DLSDU.

4.5.3.8 DLS-TxStatus

This parameter is specified in 4.3.5.

4.5.4 Sequence of primitives

The sequence of primitives in a successful or unsuccessful generic-tag transfer is defined in the time-sequence diagrams in Figure 9 and Figure 10.

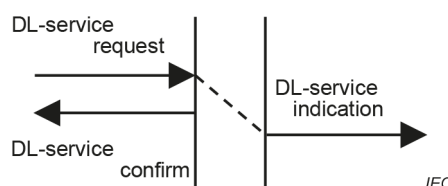


Figure 9 – Sequence of primitives for a successful connection-mode transfer

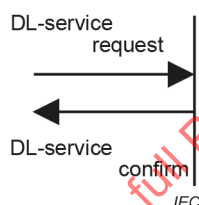


Figure 10 – Sequence of primitives for an unsuccessful connection-mode transfer

4.6 Connectionless-mode data transfer

4.6.1 General

DL-connectionless-mode service primitives can be used to transmit independent DLSDUs from one DLSAP to another DLSAP using a fixed-tag address to identify the destination DLSAP. Each DLSDU is transmitted in a single DLPDU. The DLSDU is independent in the sense that it bears no relationship to any other DLSDU transmitted through an invocation of the DLS. The DLSDU is self-contained in that all the information required to deliver the DLSDU is presented to the DLS provider, together with the user data to be transmitted, in a single service access.

No means are provided by which the receiving DLS-user may control the rate at which the sending DLS-user may send DLSDUs. This is managed externally by appropriate scheduling tools which match the capabilities of sending and receiving DLS-users with the configured service schedule of the DLS provider.

4.6.2 Function

This service provides the facilities of 4.2 b), c), d) and e). It can be used to transmit an independent, self-contained DLSDU from one DLSAP to a group of DLSAPs, all in a single service access. Delivery status is not returned as part of the local DLS invocation.

A DLSDU transmitted using DL-connectionless-mode data transfer is not considered by the DLS provider to be related in any way to any other DLSDU. In the absence of errors, it maintains the integrity of individual DLSDUs, and delivers them to the receiving DLS-users in the order in which they are presented by the sending DLS-user.

4.6.3 Types of primitives and parameters

4.6.3.1 Primitive specifications

Table 3 indicates the types of primitives and the parameters needed for the DL-connectionless-mode transmission service.

Table 3 – DL-connectionless-mode transfer primitives and parameters

DL-FIXED-TAG Parameter name	Request	Indication	Confirm
	input	output	output
Request DLS-user-identifier (handle)	M		
DLS-user-data (packet)	M	M(=)	
DLS-QoS (priority)	M		
DLS-fixed-tag	M	M(=)	
DLS destination-DLE-ID (station MAC ID)	M		
DLS-source-DLE-ID (station MAC ID)		M	
DLS-TxStatus			M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

4.6.3.2 Request DLS-user-identifier

This parameter is specified in 4.5.3.2.

4.6.3.3 DLS-user data

This parameter provides the data to be transmitted between DLS-users without alteration by the DLS provider. The initiating DLS-user may transmit any integral number of octets greater than zero, up to the limit inherent for the specified service.

4.6.3.4 DLS-QoS

This parameter is specified in 4.3.4.

NOTE DLS-scheduled-priority is generally reserved for generic-tag connection-mode services. The only normal exception is for periodic TUI fixed tag messages published by the active Keeper to ensure that all DLS providers share a common sense of link parameters.

4.6.3.5 DLS-fixed-tag

This parameter specifies the destination DLSAP in the DLE identified by the DLS-destination-DLE-ID address. The DLSAP to be used is selected from the set of Fixed Tag service types available in the destination DLE.

The set of Fixed-tag services available to the DLS-user are listed in Table 4.

Table 4 – Fixed tag services available to the DLS-user

Fixed tag service code (hexadecimal)	Meaning of service
0x01 — 0x08	Vendor specific
0x09	Ping request
0x0A — 0x14	Vendor specific
0x15	tMinus
0x16 — 0x28	Vendor specific
0x29	Ping reply
0x2A — 0x3F	Vendor specific
0x70 — 0x7F	Vendor-specific
0x83	UCMM
0x88	Keeper UCMM
0x8C	Time distribution
0xF0 — 0xFF	Vendor specific

Fixed-tag service codes in the vendor-specific range may be assigned by the DLS-user.

The UCMM fixed tag is reserved for DLS-users wishing to send messages via the Unconnected Message Manager object in the destination DLE.

The Keeper UCMM fixed tag is reserved for DLS-users wishing to send messages via the Keeper Unconnected Message Manager object in the destination DLE.

Specific uses for other fixed tags in the table are presented in Clause 5 and IEC 61158-4-2.

NOTE All other fixed tags are reserved or used internally by the DLS provider.

4.6.3.6 DLS-destination-DLE-ID

This parameter conveys the node DL-address of the destination node; it is a MAC ID address.

4.6.3.7 Request primitive

If the initiating DLS-user has implemented a FIFO queue of maximum depth K to the DLSAP-address at the specified priority as a source, then a DL-request primitive attempts to append a DLSDU to the queue, but fails if the queue already contains K DLSDUs. If the append operation is successful, then the DLSDU will be transmitted at the first opportunity, after all preceding DLSDUs in the queue. The queue serves to assemble multiple DLS-user requests for the efficiency advantage of combining them in a single transmission opportunity for the specified QoS or better.

NOTE The queue depth K is implementation specific.

4.6.3.8 Indication primitives

4.6.3.8.1 General

The receiving DLS-user has an implicit queue of indeterminate capacity which is used as the receive queue, and the DLSDU is delivered as the DLS-user-data parameter of the associated indication primitive.

If it is not possible to append the received DLSDU to the receive queue, then the DLSDU is discarded and an indication primitive is not issued to the DLS-user.

4.6.3.8.2 Indication for fixed tag DLSDUs

The receiving DLS-user is able to identify a number of Fixed Tag values of interest to it and pass them to the local DLS provider using the DLS-tag-filter management services. The set of local tag values are used to filter associated arriving DLSDUs. For DLSDUs with associated Fixed tags that are acceptable to the filter, the following indication parameters are delivered to the local DLS-user:

- DLS-user-data;
- DLS-fixed-tag, the value of the fixed tag service code associated with the DLSDU;
- DLS-source-DLE-ID, the source DLE MAC ID.

4.6.3.8.3 DLS-source-DLE-ID

This parameter conveys an address identifying the local DLE from which the fixed tag DLSDU has been sent. It is a DLE MAC ID address on the local link.

4.6.3.9 DLS-TxStatus

This parameter is specified in 4.3.5.

4.6.4 Sequence of primitives

The sequence of primitives in a successful or unsuccessful fixed-tag transfer is defined in the time-sequence diagrams in Figure 11 and Figure 12.

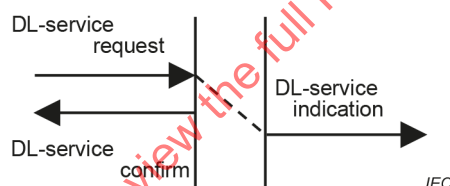


Figure 11 – Sequence of primitives for a successful connectionless-mode transfer

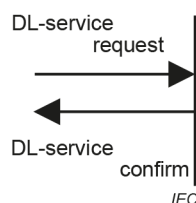


Figure 12 – Sequence of primitives for an unsuccessful connectionless-mode transfer

4.7 Queue maintenance

4.7.1 Function

DLS-send requests are held in a pending queue by the DLS provider until the requested sending opportunity is available. This queue is not visible to the DLS-user. To support efficient operation, the queue maintenance service is provided to de-queue pending requests that have not been sent.

4.7.2 Types of primitives and parameters

4.7.2.1 Primitive specifications

Table 5 and Table 6 indicate the primitives and parameters of the DL-queue maintenance service. This is a local service at each DLSAP.

Table 5 – DL-queue maintenance primitives and parameters (single flush)

DL-FLUSH SINGLE-REQUEST		Request	Confirm
Parameter name		input	output
request DLS-user-identifier (handle)		M	
DLS-TxStatus			M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

Table 6 – DL-queue maintenance primitives and parameters (flush by QoS)

DL-FLUSH REQUESTS-BY-QoS		Request	Confirm
Parameter name		input	output
DLS-QoS (priority)		M	
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

4.7.2.2 Request DLS-user-identifier and DLS-QoS

The request DLS-user-identifier and DLS-QoS parameters have the same meanings as specified in 4.5. Their purpose in these primitives is to identify the set of requests, or the single request, which is to be flushed from the request queue if they have not yet been irrevocably committed for transmission.

4.7.2.3 DLS-TxStatus

The DLS-TxStatus parameter has the same meaning and purpose as specified in 4.5.

4.7.3 Request primitive

When used with a DL-FLUSH REQUESTS-BY-QoS request, all untransmitted transfers at that QoS priority are cancelled.

When used with a DL-FLUSH SINGLE-REQUEST request, only the specified individual transfer is cancelled.

4.7.4 Confirmation primitive

4.7.4.1 DL-Flush-single-request

When the single pending transfer identified by request DLS-user-id has been cancelled, the confirmation for the original transfer request (DL-GENERIC-TAG or DL-FIXED-TAG) is returned with the DLS-TxStatus specifying the value FLUSHED.

4.7.4.2 DL-Flush-requests-by-QoS

When all pending transfers of the specified QoS have been cancelled, a confirmation is returned.

4.7.5 Sequence of primitives

The sequence of primitives for a queue maintenance request is defined in the time sequence diagrams of Figure 13.

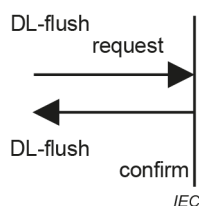


Figure 13 – Sequence of primitives for a queue maintenance request

4.8 Tag filter

4.8.1 Function

By default, the receiving DLS provider accepts and processes only the DLS-fixed-tag messages which have the fixed-tag value of 00 (moderator tag) and all other messages are discarded.

The tag filter service allows the DLS user to enable or disable reception of other messages based on the contents of their DLS parameter tag.

The DLS provider will deliver incoming messages to the DLS-user only for DLS-tags that have been enabled.

4.8.2 Types of primitives and parameters

4.8.2.1 Primitive specifications

Table 7 indicates the primitives and parameters of the DL-connectionless-mode queue maintenance service. This is a local service at each DLSAP.

Table 7 – DL-connectionless-mode tag filter primitives and parameters

Parameter name	DL-ENABLE-TAG DL-DISABLE-TAG	Request	Confirm
		input	output
DLS-tag		M	
DLS-result			M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

4.8.2.2 Request DLS-user-identifier and DLS-tag

These parameters have the same meanings and purpose as specified in 4.5. The DLS-tag can be either a DLS-generic-tag or a DLS-fixed-tag.

4.8.2.3 DLS-result

This parameter conveys the status of the corresponding request:

- a) TRUE — the service request completed successfully;
- b) FALSE — the service request failed to complete successfully.

NOTE If the DLS provider is unable to accept filtering requests for additional generic tags, the status returned will be FALSE.

4.8.3 Sequence of primitives

The sequence of primitives for a tag filter request is defined in the time sequence diagrams of Figure 14.

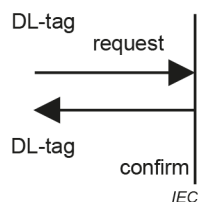


Figure 14 – Sequence of primitives for a tag filter request

5 DL-management services

5.1 Sequence of primitives

Subclause 5.1 defines the constraints on the sequence in which the primitives defined in 5.2 to 5.9 may occur. The constraints determine the order in which primitives occur, but do not fully specify when they may occur. Other aspects of actual system operation, such as PhL problems affecting messages in transit, will affect the ability of a DLS-user or a DLS provider to issue a primitive at any particular time.

The DL-management primitives and their parameters are summarized in Table 8.

Table 8 – Summary of DL-management primitives and parameters

Service	Service subtype	Primitive	Parameter
Management	Local link synchronization	DLM-TONE INDICATION	(out DLMS-cycle)
	Synchronized parameter change	DLM-SET-PENDING request	(in DLMS-configuration-data)
		DLM-SET-PENDING confirm	(out DLMS-result)
		DLM-GET-PENDING request	<none>
		DLM-GET-PENDING confirm	(out DLMS-configuration-data)
		DLM-SET-CURRENT request	(in DLMS-configuration-data)
		DLM-SET-CURRENT confirm	(out DLMS-result)
		DLM-GET-CURRENT request	<none>
		DLM-GET-CURRENT confirm	(out DLMS-configuration-data)
		DLM-TMINUS-START-COUNTDOWN request	(in DLMS-start-count)
		DLM-TMINUS-START-COUNTDOWN confirm	(out DLMS-result)
		DLM-TMINUS-ZERO indication	<none>
	Event reports	DLM-EVENT indication	(out DLMS-event, DLMS-source-DLE-ID)
	Bad FCS	DLM-BAD-FCS indication	(out DLMS-channel)
	Current moderator	DLM-CURRENT-MODERATOR indication	(out DLMS-source-DLE-ID)
	Enable moderator	DLM-ENABLE-MODERATOR request	(in DLMS-enable-moderator)
		DLM-ENABLE-MODERATOR confirm	(out DLMS-enable-moderator)
	Power-up and Online	DLM-POWER-UP indication	<none>
		DLM-ONLINE request	(in DLMS-online)
		DLM-ONLINE confirm	(out DLMS-online)
	Listen only	DLM-LISTEN-ONLY request	(in DLMS-listen only)
		DLM-LISTEN-ONLY confirm	(out DLMS-listen only)

NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.

5.2 Link synchronization

5.2.1 Function

The scheduled QoS is based on a repeating cycle of DLS transmission opportunities which are time locked to better than 1 ms. The basic time interval is the NUT or Network Update Time and an incrementing count is maintained for each NUT within the repeating cycle. This service indicates to the DLMS-user the current NUT count within the cycle.

5.2.2 Types of primitives and parameters

5.2.2.1 Primitive specifications

Table 9 indicates the primitives and parameters of the Link synchronization service. This is a local service.

Table 9 – Link synchronization primitives and parameters

Parameter name	DLM-TONE	Indication
		output
DLMS-cycle		M

5.2.2.2 DLMS-cycle

This parameter indicates the interval count for the NUT which has just been received within the overall cycle of scheduled access intervals. The DLS provider uses internal timing facilities to simulate this indication if expected moderator DLPDUs are not available.

5.2.3 Sequence of primitives

The sequence of primitives for a link synchronization is defined in the time sequence diagrams of Figure 15.

**Figure 15 – Sequence of primitives for a local link synchronization**

5.3 Synchronized parameter change

5.3.1 Function

All DLEs maintain two local copies of DLMS-configuration-data parameters: current and pending. The current copy is used for the ongoing operation of the DLS. The pending copy is maintained to allow a synchronized change of DLS configuration parameters. This service manages these DLMS-configuration-data parameters and their changeover.

At the system management level, a required set of DLMS-configuration-data parameters and the count down trigger for a change-over are distributed to all DLMS-users using data transmit services and fixed tags (link parameters tag and tMinus tag).

The synchronized parameter change service enables each DLMS-user to transfer required configuration-data values to the local DLS provider.

The moderator fixed tag DLPDU contains a parameter, called tMinus, that counts down to zero as a trigger to synchronize the change-over from current to pending sets of the DLS configuration parameters. The DLM-TMINUS-START-COUNTDOWN request from a DLMS-user causes its local DLS provider to participate in a tMinus countdown, and, if the node is the moderator, it initializes the tMinus parameter of the moderator. The moderator decrements this parameter count before transmitting each moderator DLPDU until the parameter equals zero. When tMinus transitions from 1 to 0, each local DLS provider participating in the countdown locally generates a DLM-TMINUS-ZERO indication and copies its pending DLMS-configuration-data parameters into its current copy. If the tMinus field transitions to 0 from any value except 1, the countdown is aborted and no DLM-TMINUS-ZERO indication is generated.

5.3.2 Types of primitives and parameters

5.3.2.1 Primitive specifications

Table 10, Table 11, Table 12 and Table 13 indicate the primitives and parameters of the DLM synchronized parameter change service. This is a local service.

Table 10 – Synchronized parameter change primitives and parameters (set configuration)

Parameter name	DLM-SET-PENDING DLM-SET-CURRENT	Request	Confirm
		input	output
DLMS-configuration-data	M		
DLMS-result			M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

Table 11 – Synchronized parameter change primitives and parameters (get configuration)

Parameter name	DLM-GET-PENDING DLM-GET-CURRENT	Request	Confirm
		input	output
DLMS-configuration-data			M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

Table 12 – Synchronized parameter change primitives and parameters (start countdown)

Parameter name	DLM-TMINUS-START-COUNTDOWN	Request	Confirm
		input	output
DLMS-start-count	M		
DLMS-result			M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.			

Table 13 – Synchronized parameter change primitives and parameters (tMinus)

Parameter name	DLM-TMINUS-ZERO	Indication
		output
<none>		

5.3.2.2 DLMS-result

This parameter has the same meaning and purpose as specified in 4.8 for DLS-result.

5.3.2.3 DLMS-configuration-data

This parameter conveys the set of configuration data values specified in Table 14.

Table 14 – DLMS-configuration-data

Subparameter	Meaning
my_addr	the MAC ID of this DLE
NUT_length	the length of the NUT in 10 μ s increments
SMAX	highest MAC ID allowed to transmit scheduled
UMAX	highest MAC ID allowed to transmit unscheduled
slotTime	time allowed for Ph layer line turnaround in 1 μ s increments
blanking	time to disable RX after DLPDU in 1 600 ns increments
gb_start	10 μ s intervals from start of guardband to tone
gb_center	10 μ s intervals from start of moderator to tone
modulus	modulus of the interval counter for intervals in a cycle of NUTs
gb_prestart	transmit cut-off, 10 μ s intervals before tone, shall not transmit past this limit

5.3.2.4 DLMS-start-count

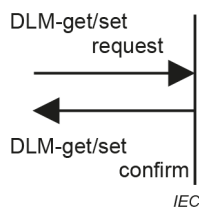
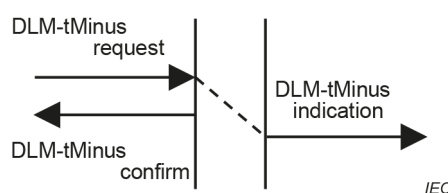
In all DLEs but the moderator, the presence of this parameter enables the local DLS provider to track the tMinus countdown contained in successive moderator messages and when the count changes from 1 to 0, to change to the pending set of DLS configuration parameters previously requested by the local DLMS user. If the final tMinus transition to 0 is from any value other than 1, the change of configuration data parameters is aborted.

If the local DLE is the moderator, this parameter initializes the tMinus parameter in the moderator messages and initiates its decrementing by 1 for each successive moderator message until it reaches 0.

If the final tMinus transition is from 1 to 0, this indication is locally generated by each participating DLS provider and passed to local DL-management, which then transforms any pending link DLS configuration parameters into current parameters.

5.3.3 Sequence of primitives

The sequence of primitives for synchronized parameter change is defined in the time sequence diagrams of Figure 16 and Figure 17.

**Figure 16 – Sequence of primitives for a DLM-get/set parameters request****Figure 17 – Sequence of primitives for a DLM-tMinus change request**