BS EN 61158-3-2:2014

BSI Standards Publication

Industrial communication networks — Fieldbus specifications

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

bsi.

...making excellence a habit.™

National foreword

This British Standard is the UK implementation of EN 61158-3-2:2014. It is identical to IEC 61158-3-2:2014. It supersedes BS EN 61158-3-2:2008 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee AMT/7, Industrial communications: process measurement and control, including fieldbus.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Published by BSI Standards Limited 2014

ISBN 978 0 580 79362 2

ICS 25.040.40; 35.100.20; 35.240.50

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 October 2014.

Amendments/corrigenda issued since publication

Date	Text affected

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 61158-3-2

October 2014

ICS 25.040.40; 35.100.20; 35.110

Supersedes EN 61158-3-2:2008

English Version

Industrial communication networks - Fieldbus specifications - Part 3-2: Data-link layer service definition - Type 2 elements (IEC 61158-3-2:2014)

Réseaux de communication industriels - Spécifications des bus de terrain - Partie 3-2: Définition des services de la couche liaison de données - Eléments de type 2 (CEI 61158-3-2:2014)

Industrielle Kommunikationsnetze - Feldbusse - Teil 3-2: Dienstfestlegungen des Data Link Layer (Sicherungsschicht) - Typ 2-Elemente (IEC 61158-3-2:2014)

This European Standard was approved by CENELEC on 2014-09-17. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Ref. No. EN 61158-3-2:2014 E

Foreword

The text of document 65C/759/FDIS, future edition 2 of IEC 61158-3-2, prepared by SC 65C "Industrial networks" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 61158-3-2:2014.

The following dates are fixed:

• latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement	(dop)	2015-06-17
• latest date by which the national standards conflicting with the document have to be withdrawn	(dow)	2017-09-17

This document supersedes EN 61158-3-2:2008.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association.

Endorsement notice

The text of the International Standard IEC 61158-3-2:2014 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

IEC 61158-1:2014	NOTE	Harmonized as EN 61158-1:2014 (not modified).
IEC 61158-2:2014	NOTE	Harmonized as EN 61158-2:2014 (not modified).
IEC 61158-5-2:2014	NOTE	Harmonized as EN 61158-5-2:2014 (not modified).
IEC 61158-6-2:2014	NOTE	Harmonized as EN 61158-6-2 [1)] (not modified).
IEC 61784-1:2014	NOTE	Harmonized as EN 61784-1 [1)] (not modified).
IEC 61784-2:2014	NOTE	Harmonized as EN 61784-2:2014 (not modified).

---

1) To be published.

Annex ZA
(normative)

Normative references to international publications with their corresponding European publications

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 1 When an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cenelec.eu.

Publication	Year	Title	EN/HD	Year
IEC 61158-4-2	2014	Industrial communication networks - Fieldbus specifications - Part 4-2: Data-link layer protocol specification - Type 2 elements	EN 61158-4-2	[2]
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	-	-

2) To be published.

CONTENTS

INTRODUCTION

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.

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 standard is a conceptual architectural service, independent of administrative and implementation divisions.

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 standard defines in an abstract way the externally visible service provided by the Type 2 fieldbus data-link layer in terms of:

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

The purpose of this standard 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 standard 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 specification may 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:

a) the sizes and octet ordering of various multi-octet service parameters;
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 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 1 data-link layer services defined in this standard.

2 Normative references

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

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

IEC 61158-4-2:2014, *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, abbreviations and conventions

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

3.1 Reference model terms and definitions

This standard 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 [ISO/IEC 7498-1]
correspondent DL-entities (N=2)
correspondent Ph-entities (N=1)

3.1.12 DL-duplex-transmission [ISO/IEC 7498-1]

3.1.13 (N)-entity [ISO/IEC 7498-1]
DL-entity (N=2)
Ph-entity (N=1)

3.1.14 DL-facility [ISO/IEC 7498-1]

3.1.15 flow control [ISO/IEC 7498-1]

3.1.16 (N)-layer [ISO/IEC 7498-1]
DL-layer (N=2)
Ph-layer (N=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 [ISO/IEC 7498-1]
DL-service (N=2)
Ph-service (N=1)

3.1.32 (N)-service-access-point [ISO/IEC 7498-1]
DL-service-access-point (N=2)
Ph-service-access-point (N=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 standard 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

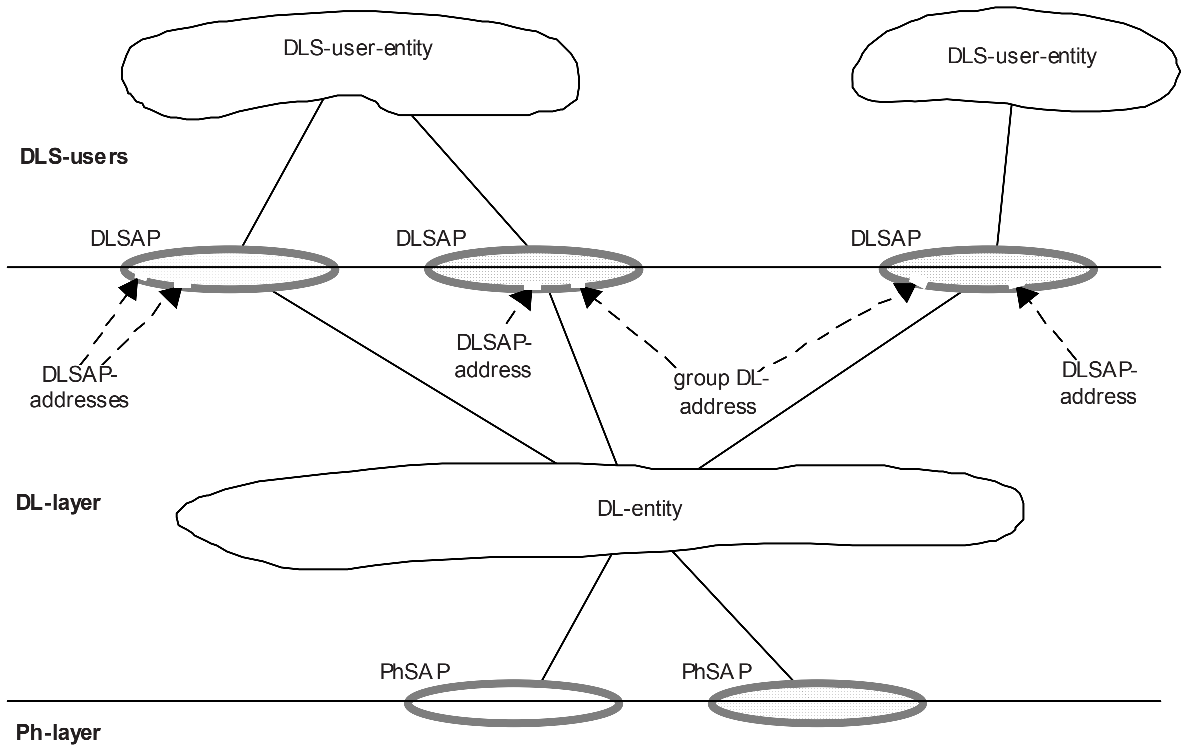
For the purposes of this standard, the following terms and definitions apply.

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.



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)

a) a fixed tag or a generic tag, and

b) 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 may 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.

r.m.s. 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 standard 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 standard 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 standard. 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 may or may not be provided 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.

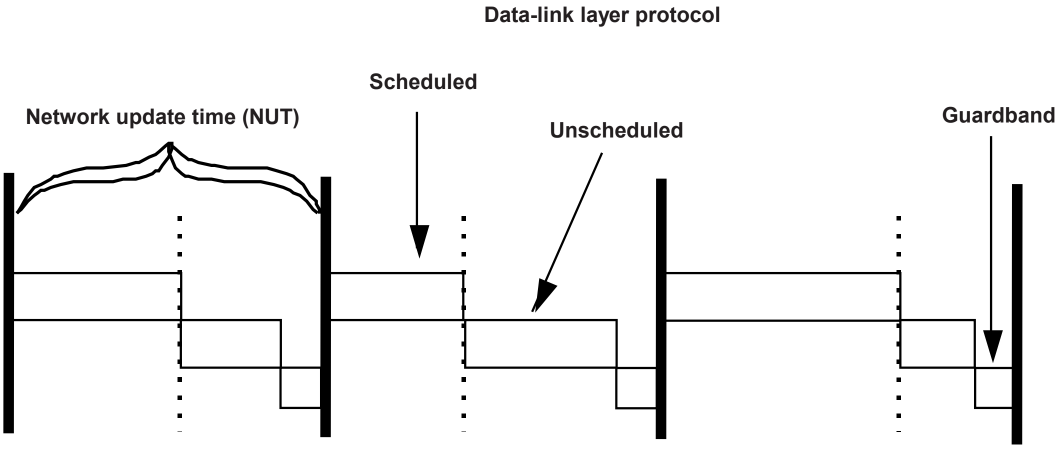


Figure 2 – NUT structure

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.

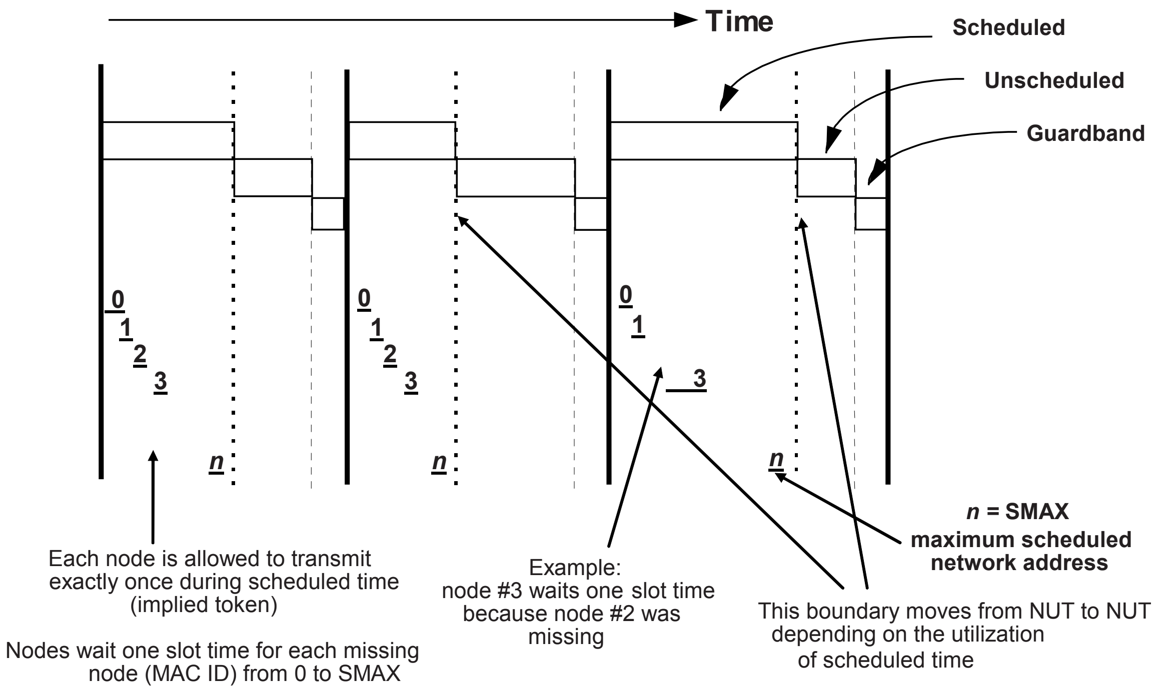


Figure 3 – Medium access during scheduled time

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 may 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.

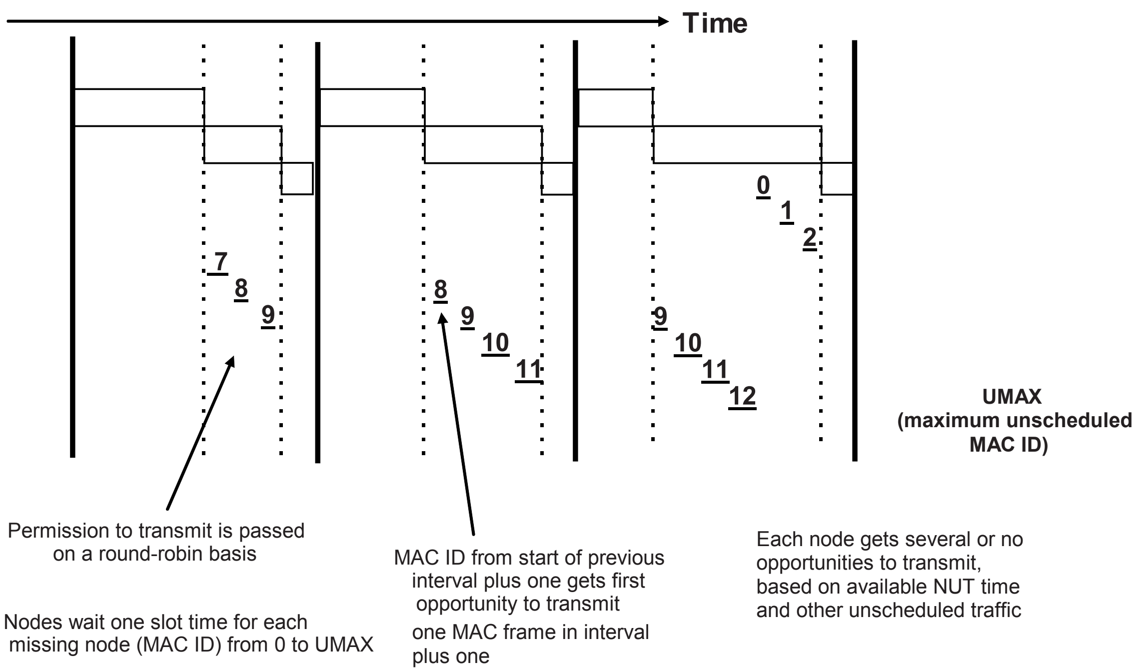


Figure 4 – Medium access during unscheduled time

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 may 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 may 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 µs.

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.

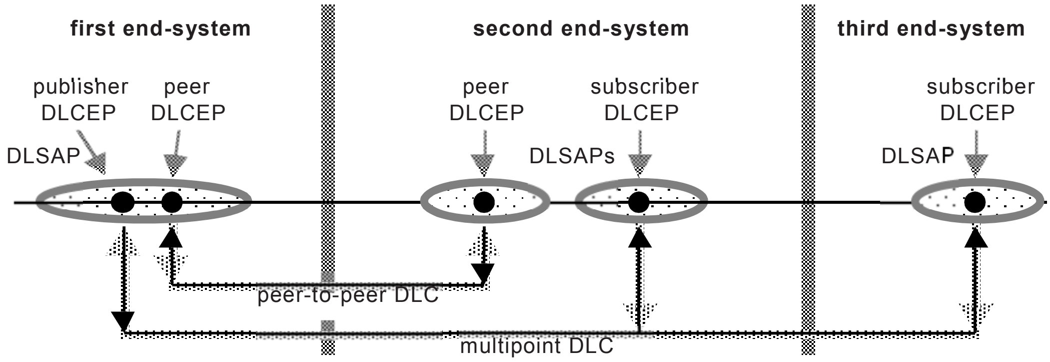


Figure 5 – Queue model for the peer and multipoint DLS, DLSAPs and their DLCEPs

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 standard 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).

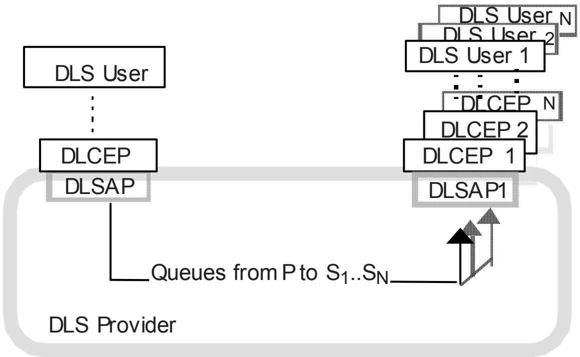


Figure 6 – Queue model of a multipoint DLS between a sending DLS-user and one or more receiving DLS-users

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 may 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 may 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 master 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	(*in* request DLS-user-identifier, DLS-user-data, DLS-QoS, DLS-generic-tag)
		DL-GENERIC-TAG indication	(*out* DLS-user-data, DLS-generic-tag)
		DL-GENERIC-TAG confirm	(*out* DLS-TxStatus)
	Connectionless-mode	DL-FIXED-TAG request	(*in* request DLS-user-identifier, DLS-user-data, DLS-QoS, DLS-fixed-tag, DLS-destination-DLE-ID)
		DL-FIXED-TAG indication	(*out* DLS-user-data, DLS-fixed-tag, DLS-source-DLE-ID)
		DL-FIXED-TAG confirm	(*out* DLS-TxStatus)
Super vision	**Queue maintenance**	DL-FLUSH-SINGLE-REQUEST request	(in request DLS-user-identifier)
		DL-FLUSH-SINGLE-REQUEST confirm	(*out* DLS-TxStatus)
		DL-FLUSH-REQUESTS-BY-QOS request	(*in* DLS-QoS)
		DL-FLUSH-REQUESTS-BY-QOS confirm	<none>
	Tag filter	DL-ENABLE-TAG request	(*in* DLS-tag)
		DL-ENABLE-TAG confirm	(*out* DLS-result)
		DL-DISABLE-TAG request	(*in* DLS-tag)
		DL-DISABLE-TAG confirm	(*out* 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.

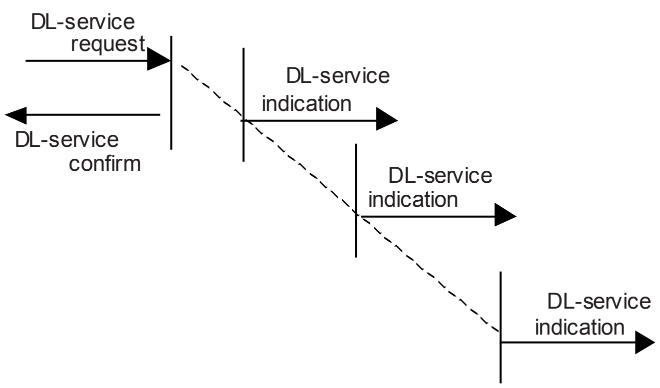


Figure 7 – DLS primitive time-sequence diagram

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.

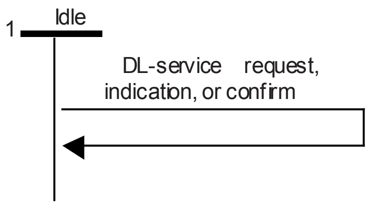


Figure 8 – State transition diagram for sequences of DLS primitives at one DLSAP

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 input	Indication output	Confirm 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.

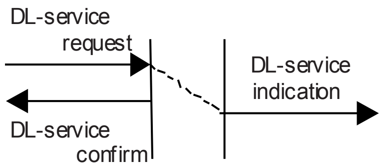


Figure 9 – Sequence of primitives for a successful connection-mode transfer

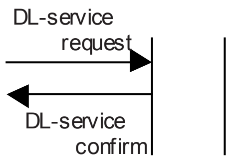


Figure 10 – Sequence of primitives for an unsuccessful connection-mode transfer

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 input	Indication output	Confirm 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 master 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
01 — 08	Vendor specific
09	Ping request
0A – 14	Vendor specific
15	tMinus
16 — 28	Vendor specific
29	Ping reply
2A — 3F	Vendor specific
70 – 7F	Vendor-specific
83	UCMM
88	Keeper UCMM
8C	Time distribution
F0 — FF	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.

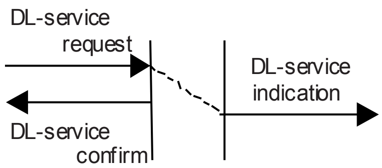


Figure 11 – Sequence of primitives for a successful connectionless-mode transfer

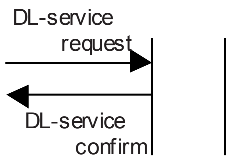


Figure 12 – Sequence of primitives for an unsuccessful connectionless-mode transfer

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 indicates 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

DL-FLUSH SINGLE-REQUEST Parameter name	Request input	Confirm 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.		

DL-FLUSH REQUESTS-BY-QOS Parameter name	Request input	Confirm 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.

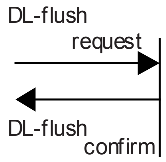


Figure 13 – Sequence of primitives for a queue maintenance request

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

Table 6 – DL-connectionless-mode tag filter primitives and parameters

DL-ENABLE-TAG DL-DISABLE-TAG **Parameter name**	**Request** input	**Confirm** 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 of 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.

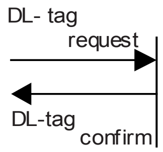


Figure 14 – Sequence of primitives for a tag filter request

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 7.

Table 7 – 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 8 indicates the primitives and parameters of the Link synchronization service. This is a local service.

Table 8 – Link synchronization primitives and parameters

DLM-TONE	Indication
Parameter name	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 9 indicates the primitives and parameters of the DLM synchronized parameter change service. This is a local service.

Table 9 – Synchronized parameter change primitives and parameters

DLM-SET-PENDING DLM-SET-CURRENT Parameter name	Request input	Confirm 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.		

DLM-GET-PENDING DLM-GET-CURRENT Parameter name	Request input	Confirm 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.		

DLM-tMINUS-START-COUNTDOWN Parameter name	Request input	Confirm 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.		

DLM-tMINUS-ZERO Parameter name	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 10.

Table 10 – DLMS-configuration-data

Subparameter	Meaning
my_addr	the MAC ID of this DLE
NUT_length	the length of the NUT in 10 µs 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 µs increments
blanking	time to disable RX after DLPDU in 1 600 ns increments
gb_start	10 µs intervals from start of guardband to tone
gb_center	10 µs 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 µs intervals before tone, may 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.

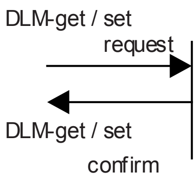


Figure 16 – Sequence of primitives for a DLM-get/set parameters request

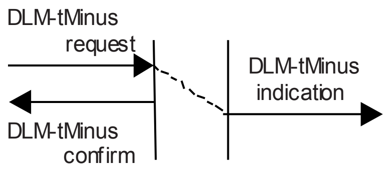


Figure 17 – Sequence of primitives for a DLM-tMinus change request

5.4 Event reports

5.4.1 Function

The event report service provides indications to DL-management about events internal to the DLS provider.

5.4.2 Types of primitives and parameters

5.4.2.1 Primitive specifications

Table 11 indicates the primitives and parameters of the event report service. This is a local service.

Table 11 – Event report primitives and parameters

DLM-EVENT **Parameter name**	**Indication** output
DLMS-event	M
DLMS-source-DLE-ID	C

5.4.2.2 DLMS-event

This parameter takes one of the values in Table 12.

Table 12 – DLMS events being reported

DLMS event	Description
DLMS_EV_rxGoodFrame	A good DLPDU was received. This includes DLPDUs that contain no data (null DLPDUs), but excludes moderator DLPDUs.
DLMS_EV_txGoodFrame	A good DLPDU was transmitted. This includes DLPDUs that contain no data (null DLPDUs), but excludes moderator DLPDUs.
DLMS_EV_badFrame	A damaged DLPDU was received. The apparent source MAC ID of the transmitting DLE is reported via the optional parameter.
DLMS_EV_errA	A bad DLPDU was received on channel A of the physical medium, or a good DLPDU was received on channel B and PH-FRAME indication from channel A stayed FALSE.
DLMS_EV_errB	A bad DLPDU was received on channel B of the physical medium, or a good DLPDU was received on channel A and PH-FRAME indication from channel B stayed FALSE.
DLMS_EV_txAbort	A transmit DLPDU was terminated with an abort sequence.
DLMS_EV_NUT_overrun	NUT is not large enough to accommodate all the scheduled traffic.
DLMS_EV_dribble	Scheduled Lpackets could not be sent during scheduled time.
DLMS_EV_nonconcurrence	An event was detected that indicates that this node is out of step with the access control protocol.
DLMS_EV_rxAbort	A DLPDU was received that was terminated with an abort sequence.
DLMS_EV_lonely	Have not heard a DLPDU from another node on the link for 8 NUTs
DLMS_EV_dupNode	Another node on the link is using this node’s MAC ID.
DLMS_EV_noisePulse	Ph-LOCK indication went TRUE then FALSE before PH-FRAME indication went TRUE, but Ph-LOCK indication was not TRUE long enough to indicate a possibly damaged DLPDU.
DLMS_EV_collision	PH-FRAME indication was TRUE when this node was about to transmit.
DLMS_EV_invalidModAddress	A moderator was received from a node that does not have the lowest MAC ID on the link.
DLMS_EV_rogue	A moderator DLPDU was received that does not match the link configuration information at this node.
DLMS_EV_deafness	Cannot hear the moderator DLPDU even though other link traffic is present.
DLMS_EV_supernode	A moderator was received from MAC ID 0.

5.4.2.3 DLMS-source-DLE-ID

This parameter is used in conjunction with the DLMS_EV_badFrame event to indicate the probable transmitting DLE.

NOTE As the DLPDU was damaged, the indicated DLMS-source-DLE-ID could be incorrect.

5.4.3 Sequence of primitives

The sequence of primitives for an event indication is defined in the time sequence diagrams of Figure 18.

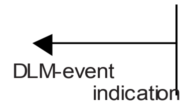


Figure 18 – Sequence of primitives for a DLM-event indication

5.5 Bad FCS

5.5.1 Function

The BAD-FCS indication service alerts the DLMS-user that a received DLPDU had an invalid frame check sequence.

5.5.2 Types of primitives and parameters

5.5.2.1 Primitive specifications

Table 13 indicates the primitives and parameters of the bad-FCS service. This is a local service.

Table 13 – Bad FCS primitives and parameters

DLM-BAD-FCS Parameter name	Indication output
DLMS-channel	M

5.5.2.2 DLMS-channel

This parameter indicates which PhE provided the DLPDU that failed the FCS check. The parameter value is either CHANNEL-A or CHANNEL-B indicating the PhL channel on which the erroneous DLPDU was received. This indication is provided not more than once per erroneous DLPDU per channel.

5.5.3 Sequence of primitives

The sequence of primitives for a bad-FCS is defined in the time sequence diagram of Figure 19.

Figure 19 – Sequence of primitives for a DLM-bad-FCS indication

5.6 Current moderator

5.6.1 Function

This service informs the DLMS-user which DLE is the current moderator.

5.6.2 Types of primitives and parameters

5.6.2.1 Primitive specifications

Table 14 indicates the primitives and parameters of the current moderator indication. This is a local service.

Table 14 – Current moderator primitives and parameters

DLM-CURRENT-MODERATOR Parameter name	Indication output
DLMS-source-DLE-ID	M

5.6.2.2 DLMS-source-DLE-ID

This parameter indicates an address identifying the source DLE MAC ID for the most recently received moderator DLPDU on the local link.

5.6.3 Sequence of primitives

The sequence of primitives for a current moderator indication is defined in the time sequence diagrams of Figure 20.

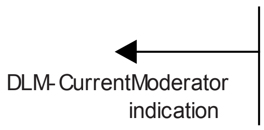


Figure 20 – Sequence of primitives for a DLM-current-moderator indication

5.7 Enable moderator

5.7.1 Function

This service enables the DLMS user to enable and disable the ability of its local DLS provider to join the group of DLEs that co-operate in assigning one of its members to the role of current moderator.

5.7.2 Types of primitives and parameters

5.7.2.1 Primitive specifications

Table 15 indicates the primitives and parameters of the enable moderator service. This is a local service.

Table 15 – Enable moderator primitives and parameters

DLM-ENABLE-MODERATOR Parameter name	Request input	Confirm output
DLMS-enable-moderator	M	M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.		

5.7.2.2 DLMS-enable-moderator

This parameter takes values TRUE and FALSE which respectively enable or disable the moderator capability in the local DLS provider.

5.7.3 Sequence of primitives

The sequence of primitives for enable moderator is defined in the time sequence diagrams of Figure 21.

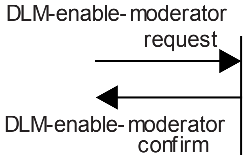


Figure 21 – Sequence of primitives for a DLM-enable-moderator request

5.8 Power-up and online

5.8.1 Function

This service enables the DLMS-user to request its local DLS provider to enter an online state or an offline state.

5.8.2 Types of primitives and parameters

5.8.2.1 Primitive specifications

Table 16 indicates the primitives and parameters of the power-up and online services. This is a local service.

Table 16 – Power-up and online primitives and parameters

DLM-POWER-UP	**Indication**
Parameter name	output
<none>	

DLM-ONLINE	**Request**	**Confirm**
Parameter name	input	output
DLMS-online	M	M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter.		

5.8.2.2 DLM-power-up

The DLM-power-up indication notifies that the Data-link layer has completed its initialization.

NOTE Following this initialization indication the DL user can continue the process of going on line by sending an “I’m alive” fixed tag messages to inform all other DL users.

5.8.2.3 DLM-online

At power-up, the local DLS provider waits until the request DLMS-online parameter equals TRUE. The DLS provider then begins the process of going online and reports TRUE or FALSE, representing success or failure transition to online, respectively, via the confirmation parameter.

When the request parameter DLMS-online is FALSE, the local DLS provider goes offline at the end of the current NUT, and reports back the new status via the confirmation parameter. When offline, the local DLS provider does not transmit, and ignores any link activity.

5.8.3 Sequence of primitives

The sequence of primitives for power-up and online is defined in the time sequence diagrams of Figure 22 and Figure 23.

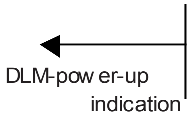


Figure 22 – Sequence of primitives for a DLM-power-up indication

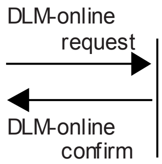


Figure 23 – Sequence of primitives for a DLM-online request

5.9 Listen only

5.9.1 Function

This service enables the DLMS user to enable and disable the ability of its local DLS provider to transmit.

5.9.2 Types of primitives and parameters

5.9.2.1 Primitive specifications

Table 17 indicates the primitives and parameters of the listen only service. This is a local service.

Table 17 – Listen-only primitives and parameters

DLM-LISTEN-ONLY Parameter name	Request input	Confirm output
DLMS-listen-only	M	M

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

5.9.2.2 DLMS-listen-only

This parameter takes values TRUE and FALSE which respectively enable or disable the DLPDU transmission capability in the local DLS provider. When the enable parameter is FALSE, transmission of DLPDUs is disabled, however the ability of the DLE to receive DLPDUs is not affected.

5.9.3 Sequence of primitives

The sequence of primitives for listen only is defined in the time sequence diagrams of Figure 24.

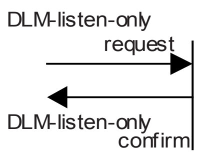


Figure 24 – Sequence of primitives for a DLM-listen-only request

5.10 Time distribution

5.10.1 Function

The moderator DLPDU provides a common reference marker that is synchronized between all nodes on the local link. By distributing and processing time stamps relative to the reference instead of to the time distribution message, implementations are simplified whilst accuracy is improved by several orders of magnitude. Phase and frequency synchronization is inherent in this DL-protocol to a very high level of accuracy. The accuracy of time synchronization using the time distribution format defined in 5.10 is implementation dependent, however it can be better than 10 µs.

5.10.2 Types of primitives and parameters

5.10.2.1 General

Table 18 is a summary of the DLMS time and time quality parameters sent as DLMS-user data by the time distribution service.

Table 18 – DLMS time and time quality parameters

Subparameter	Meaning
revision	revision of time distribution format
leap	leap second offset
goodness	time relay control field
gse	global squared error relative to ultimate master
dctz	distribution channel time zero
ts_ref	time stamp of previous reference pulse
ts_tx	time stamp of this message's transmission

5.10.2.2 revision

This parameter has the value zero; it represents the revision of the time distribution format.

5.10.2.3 leap

This parameter specifies the Universal Coordinated Time (UTC) leap seconds. This number, when added to the system time, gives actual UTC time. This number takes unpredictable jumps as dictated by the US Naval Observatory. If zero, then the number of leap seconds is unknown.

This parameter should not be used in any control situations, but may be needed in some time relays to distribution channels that are based on UTC rather than Global Positioning Satellite System (GPSS) time.

5.10.2.4 goodness

5.10.2.4.1 Definition

This compound parameter specifies the source quality of the distributed time and the number of hops in the time distribution path.

5.10.2.4.2 source quality

This subparameter indicates the quality of the source of the distributed time as shown in Table 19.

Table 19 – Time distribution source quality

Value	Meaning
7	Absolute system time — acting as master
6	Absolute system time — acting as dependent
5	Human set system time — acting as master
4	Human set system time — acting as dependent
3	Lock established to node on distribution channel other than this one
2	Lock established to node on this channel; system time unknown
1	(invalid)
0	Not synchronized with any other node

5.10.2.4.3 stratum

This parameter specifies the number of time relays between this message and a source of absolute time. A value of 0 signifies an exact reference, and the value is incremented for every intermediate time relay. If the priority field is set to zero (lock not achieved), or the number of intermediate time relays exceeds 15, the ctrl parameter is set to 15. Bits 3 through 11 are reserved and have the value zero.

NOTE A time relay is a DL-router which distributes time synchronization messages on its connected links based on the time synchronization messages received on its other links.

5.10.2.5 gse

This parameter indicates the cumulated r.m.s stability squared. This parameter should approximate the node's worst-case stability relative to the rest of the system. The units of this parameter are $(100\ \text{ns})^2$. When the r.m.s stability is unknown or not yet determined, the value for this parameter is $FFFFFFFF_{16}$.

5.10.2.6 dctz

This parameter indicates the system time offset from the distribution channel's arbitrary time zero, established when the distribution channel and local link are synchronized. The unit of measurement is 100 ns.

5.10.2.7 ts_ref

This parameter indicates the time stamp of the last tone following a moderator DLPDU which had its interval_count equal to zero. The value of zero indicates that this value is not known. System time zero is defined as that originally used for the Global Positioning Satellite System: 12:00 midnight, Jan 6, 1980 GMT. The unit of measurement is 100 ns.

NOTE This DLL does not use GPS time; it uses only the original time zero point for GPS. Thus the 1999 roll-over of GPS time has no relevance to system time.

5.10.2.8 ts_tx

This parameter indicates the time stamp at the transmission of this message. The value of zero indicates that this value is not known. System time zero is defined in 5.10.2.7.

Bibliography

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 bibliographic references.

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

IEC 61158-2:2014, *Industrial communication networks – Fieldbus specifications – Part 2: Physical layer specification and service definition*

IEC 61158-5-2:2014, *Industrial communication networks – Fieldbus specifications – Part 5-2: Application layer service definition – Type 2 elements*

IEC 61158-6-2:2014, *Industrial communication networks – Fieldbus specifications – Part 6-2: Application layer protocol specification – Type 2 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*

ODVA: THE CIP NETWORKS LIBRARY – *Volume 1: Common Industrial Protocol (CIP™) – Edition 3.13, November 2012, available at <http://www.odva.org>*

ODVA: THE CIP NETWORKS LIBRARY – *Volume 4: ControlNet™ Adaptation of CIP – Edition 1.7, April 2011, available at <http://www.odva.org>*

____

This page deliberately left blank

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Useful Contacts:

Customer Services
Tel: +44 845 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 845 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

bsi.

...making excellence a habit.™