



BSI Standards Publication

Industrial communication networks — Fieldbus specifications

Part 4-2: Data-link layer protocol specification — Type 2 elements

National foreword

This British Standard is the UK implementation of EN IEC 61158-4-2:2023. It is identical to IEC 61158-4-2:2023. It supersedes BS EN IEC 61158-4-2:2019, which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee GEL/65, Measurement and control.

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

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2023
Published by BSI Standards Limited 2023

ISBN 978 0 539 13667 8

ICS 25.040.40; 35.100.20; 35.110

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 30 April 2023.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

English Version

Industrial communication networks - Fieldbus specifications -
Part 4-2: Data-link layer protocol specification - Type 2 elements
(IEC 61158-4-2:2023)

Réseaux de communication industriels - Spécifications des
bus de terrain - Partie 4-2: Spécification du protocole de la
couche liaison de données - Éléments de type 2
(IEC 61158-4-2:2023)

Industrielle Kommunikationsnetze - Feldbusse – Teil 4-2:
Protokollspezifikation des Data Link Layer
(Sicherheitsschicht) - Typ 2-Elemente
(IEC 61158-4-2:2023)

This European Standard was approved by CENELEC on 2023-04-13. 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, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Türkiye and the United Kingdom.



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

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

European foreword

The text of document 65C/1202/FDIS, future edition 5 of IEC 61158-4-2, 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 IEC 61158-4-2:2023.

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) 2024-01-13
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2026-04-13

This document supersedes EN IEC 61158-4-2:2019 and all of its amendments and corrigenda (if any).

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

Any feedback and questions on this document should be directed to the users' national committee. A complete listing of these bodies can be found on the CENELEC website.

Endorsement notice

The text of the International Standard IEC 61158-4-2:2023 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 standard indicated:

- | | |
|----------------------|---|
| IEC 61131-9 | NOTE Approved as EN IEC 61131-9 |
| IEC 61158-1 | NOTE Approved as EN IEC 61158-1 |
| IEC 61158-2 | NOTE Approved as EN 61158-2 |
| IEC 61784-1 (series) | NOTE Approved as EN IEC 61784-1 (series) ¹ |
| IEC 61784-2 (series) | NOTE Approved as EN IEC 61784-2 (series) ² |

¹ To be published. Stage at the time of publication: FprEN IEC 61784-1-X:2023.

² To be published. Stage at the time of publication: FprEN IEC 61784-2-X:2023.

Annex ZA

(normative)

Normative references to international publications with their corresponding European publications

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 1 Where 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.

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IEC 61131-3	-	Programmable controllers - Part 3: Programming languages	EN 61131-3	-
IEC 61158-3-2	2023	Industrial communication networks - Fieldbus specifications - Part 3-2: Data-link layer service definition - Type 2 elements	EN IEC 61158-3-2	2023
IEC 61158-5-2	2023	Industrial communication networks - Fieldbus specifications - Part 5-2: Application layer service definition - Type 2 elements	EN IEC 61158-5-2	— ³
IEC 61158-6-2	2023	Industrial communication networks - Fieldbus specifications - Part 6-2: Application layer protocol specification - Type 2 elements	EN IEC 61158-6-2	— ⁴
IEC 61588	-	Precision clock synchronization protocol for-networked measurement and control systems		-
IEC 61784-3-2	-	Industrial communication networks - Profiles - Part 3-2: Functional safety fieldbuses - Additional specifications for CPF 2	EN IEC 61784-3-2	-
IEC 62026-3	2014	Low-voltage switchgear and controlgear - Controller-device interfaces (CDIs) - Part 3: DeviceNet	EN 62026-3	2015
IEC 62439-3	2016	Industrial communication networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)	EN IEC 62439-3	2018

³ Under preparation. Stage at time of publication: prEN IEC 61158-5-2:2023.

⁴ Under preparation. Stage at time of publication: prEN IEC 61158-6-2:2023.

EN IEC 61158-4-2:2023 (E)

ISO/IEC 13239	-	Information technology - Telecommunications and information exchange between systems - High-level data link control (HDLC) procedures	-	-
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/IEEE 8802-3	-	Telecommunications and exchange between information technology systems - Requirements for local and metropolitan area networks - Part 3: Standard for Ethernet	-	-
ISO 11898-1	2015	Road vehicles - Controller area network (CAN) - Part 1: Data link layer and physical signalling	-	-
IEEE Std 802.1AB	2016	IEEE Standard for Local and metropolitan area networks: Station and Media Access Control Connectivity Discovery	-	-
IEEE Std 802.1ABcu	2021	Standard for Local and metropolitan area networks - Station and Media Access Control Connectivity Discovery Amendment: YANG Data Model	-	-
IEEE Std 802.1Q	2018	IEEE Standard for local and metropolitan area networks - Bridges and bridged networks	-	-
IEEE Std 802.3	2018	IEEE Standard for Ethernet	-	-
IETF RFC 951	1985	Bootstrap Protocol (BOOTP)	-	-
IETF RFC 1213	1991	Management Information Base for Network-Management of TCP/IP-based Internets: MIB-II	-	-
IETF RFC 1542	1993	Clarifications and Extensions for the Bootstrap Protocol	-	-
IETF RFC 1643	1994	Definitions of Managed Objects for the Ethernet-like interface types	-	-
IETF RFC 2131	1997	Dynamic Host Configuration Protocol	-	-
IETF RFC 2132	1997	DHCP Options and BOOTP Vendor Extensions	-	-
IETF RFC 2863	2000	The Interfaces Group MIB	-	-
IETF RFC 3418	2002	Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)	-	-
IETF RFC 3635	2003	Definitions of Managed Objects for the Ethernet-like Interface Types	-	-
IETF RFC 4541	2006	Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches	-	-

IETF RFC 5227	2008	IPv4 Address Conflict Detection	-	-
---------------	------	---------------------------------	---	---

CONTENTS

FOREWORD.....	16
INTRODUCTION.....	18
1 Scope.....	19
1.1 General.....	19
1.2 Specifications	19
1.3 Procedures	19
1.4 Applicability	20
1.5 Conformance	20
2 Normative references	20
3 Terms, definitions, symbols, abbreviated terms and conventions	22
3.1 Reference model terms and definitions	22
3.2 Service convention terms and definitions	24
3.3 Common terms and definitions	24
3.4 Additional Type 2 definitions	25
3.5 Type 2 symbols and abbreviated terms	33
3.6 Conventions for station management objects	34
4 Overview of the data-link protocol.....	35
4.1 General.....	35
4.1.1 DLL architecture	35
4.1.2 Access control machine (ACM) and scheduling support functions	37
4.1.3 Connection-mode, connectionless-mode data transfer and DL service.....	37
4.2 Services provided by the DL	37
4.2.1 Overview	37
4.2.2 QoS.....	38
4.3 Structure and definition of DL-addresses	38
4.3.1 General	38
4.3.2 MAC ID address	39
4.3.3 Generic tag address	40
4.3.4 Fixed tag address	40
4.4 Services assumed from the PhL.....	41
4.4.1 General requirements	41
4.4.2 Data encoding rules.....	41
4.4.3 DLL to PhL interface	42
4.5 Functional classes	44
5 General structure and encoding of PhIDUs and DLPDUs and related elements of procedure	44
5.1 Overview.....	44
5.2 Media access procedure	44
5.3 DLPDU structure and encoding	48
5.3.1 General	48
5.3.2 DLPDU components	48
5.3.3 Preamble	48
5.3.4 Start and end delimiters.....	48
5.3.5 DLPDU octets and ordering	48
5.3.6 Source MAC ID.....	49
5.3.7 Lpackets field	49
5.3.8 Frame check sequence (FCS).....	49

5.3.9	Null DLPDU	52
5.3.10	Abort DLPDU	52
5.4	Lpacket components	52
5.4.1	General Lpacket structure	52
5.4.2	Size	53
5.4.3	Control	53
5.4.4	Generic tag Lpackets	53
5.4.5	Fixed tag Lpackets	54
5.5	DLPDU procedures	54
5.5.1	General	54
5.5.2	Sending scheduled DLPDUs	55
5.5.3	Sending unscheduled DLPDUs	55
5.5.4	Receiving DLPDUs	55
5.6	Summary of DLL support services and objects	56
6	Specific DLPDU structure, encoding and procedures	57
6.1	Modeling language	57
6.1.1	State machine description	57
6.1.2	Use of DLL- prefix	58
6.1.3	Data types	58
6.2	DLS user services	59
6.2.1	General	59
6.2.2	Connected mode and connectionless mode transfer service	60
6.2.3	Queue maintenance service	61
6.2.4	Tag filter service	61
6.2.5	Link synchronization service	62
6.2.6	Synchronized parameter change service	62
6.2.7	Event reports service	63
6.2.8	Bad FCS service	64
6.2.9	Current moderator service	64
6.2.10	Power up and online services	64
6.2.11	Enable moderator service	65
6.2.12	Listen only service	65
6.3	Generic tag Lpacket	65
6.3.1	General	65
6.3.2	Structure of the generic-tag Lpacket	65
6.3.3	Sending and receiving the generic-tag Lpacket	66
6.4	Moderator Lpacket	66
6.4.1	General	66
6.4.2	Structure of the moderator Lpacket	66
6.4.3	Sending and receiving the moderator Lpacket	66
6.5	Time distribution Lpacket	67
6.5.1	General	67
6.5.2	Structure of the time distribution Lpacket	67
6.5.3	Sending and receiving the time distribution Lpacket	69
6.6	UCMM Lpacket	70
6.6.1	General	70
6.6.2	Structure of the UCMM Lpacket	70
6.6.3	Sending and receiving the UCMM Lpacket	70
6.7	Keeper UCMM Lpacket	70

6.7.1	General	70
6.7.2	Structure of the Keeper UCMM Lpacket	70
6.7.3	Sending and receiving the Keeper UCMM Lpacket.....	70
6.8	TUI Lpacket	71
6.8.1	General	71
6.8.2	Structure of the TUI Lpacket	71
6.8.3	Sending and receiving the TUI Lpacket.....	72
6.9	Link parameters Lpacket and tMinus Lpacket.....	72
6.9.1	General	72
6.9.2	Structure of link parameters and tMinus Lpackets	72
6.9.3	Sending and receiving the tMinus and Link parameters Lpackets.....	73
6.10	I'm-alive Lpacket.....	73
6.10.1	General	73
6.10.2	Structure or the I'm-alive Lpacket	73
6.10.3	Sending and receiving I'm Alive	74
6.10.4	I'm alive state processing	74
6.11	Ping Lpackets	75
6.11.1	General	75
6.11.2	Structure of the ping Lpackets	76
6.11.3	Sending and receiving the ping Lpackets	76
6.12	WAMI Lpacket.....	76
6.12.1	General	76
6.12.2	Structure of the WAMI Lpacket	77
6.12.3	Sending and receiving the WAMI Lpacket	77
6.13	Debug Lpacket.....	77
6.14	IP Lpacket	78
6.15	Ethernet Lpacket.....	78
7	Objects for station management	78
7.1	General.....	78
7.2	ControlNet™ object.....	80
7.2.1	Overview	80
7.2.2	Class attributes	80
7.2.3	Instance attributes	80
7.2.4	Common services	89
7.2.5	Class specific services	90
7.2.6	Behavior	91
7.2.7	Module status indicator.....	91
7.3	Keeper object	92
7.3.1	Overview	92
7.3.2	Revision history	92
7.3.3	Class attributes	92
7.3.4	Instance attributes	92
7.3.5	Common services	101
7.3.6	Class specific services	102
7.3.7	Service error codes	108
7.3.8	Behavior	108
7.3.9	Miscellaneous notes	109
7.3.10	Keeper power up sequence	110
7.4	Scheduling object	111

7.4.1	Overview	115
7.4.2	Class attributes	116
7.4.3	Instance attributes	116
7.4.4	Common services	117
7.4.5	Class specific services	119
7.4.6	Typical scheduling session	125
7.5	TCP/IP Interface object	126
7.5.1	Overview	126
7.5.2	Revision history	126
7.5.3	Class attributes	126
7.5.4	Instance attributes	127
7.5.5	Diagnostic connection points	142
7.5.6	Common services	142
7.5.7	Class specific services	145
7.5.8	Behavior	147
7.5.9	Address Conflict Detection (ACD)	148
7.6	Ethernet Link object	154
7.6.1	Overview	154
7.6.2	Revision history	154
7.6.3	Class attributes	154
7.6.4	Instance attributes	155
7.6.5	Diagnostic connection points	165
7.6.6	Common services	166
7.6.7	Class specific services	167
7.6.8	Behavior	168
7.7	DeviceNet™ object	169
7.7.1	Overview	169
7.7.2	Revision history	170
7.7.3	Class attributes	170
7.7.4	Instance attributes	170
7.7.5	Common services	177
7.7.6	Class specific services	178
7.8	Connection Configuration object (CCO)	179
7.8.1	Overview	179
7.8.2	Revision history	179
7.8.3	Class attributes	179
7.8.4	Instance attributes	181
7.8.5	Connection Configuration object change control	190
7.8.6	Common services	190
7.8.7	Class specific services	196
7.8.8	Behavior	200
7.9	DLR object	200
7.9.1	Overview	200
7.9.2	Revision history	201
7.9.3	Class attributes	201
7.9.4	Instance attributes	201
7.9.5	Diagnostic connection points	213
7.9.6	Common services	213
7.9.7	Class specific services	

7.10	QoS object.....	218
7.10.1	Overview	218
7.10.2	Revision History	218
7.10.3	Class attributes	218
7.10.4	Instance Attributes.....	219
7.10.5	Common services	220
7.11	Port object	221
7.11.1	Overview	221
7.11.2	Revision History	221
7.11.3	Class attributes	222
7.11.4	Instance attributes	222
7.11.5	Common services	229
7.12	PRP/HSR Protocol object.....	231
7.12.1	Overview	231
7.12.2	Revision history	231
7.12.3	Class attributes	231
7.12.4	Instance attributes	231
7.12.5	Diagnostic connection points	239
7.12.6	Common Services.....	239
7.13	PRP/HSR Nodes Table object.....	241
7.13.1	Overview	241
7.13.2	Revision history	241
7.13.3	Class attributes	242
7.13.4	Instance attributes	242
7.13.5	Common services	244
7.14	LLDP Management object	245
7.14.1	Overview	245
7.14.2	Revision history	245
7.14.3	Class attributes	246
7.14.4	Instance attributes	246
7.14.5	Common services	247
7.15	LLDP Data Table object	248
7.15.1	Overview	248
7.15.2	Revision history	249
7.15.3	Class attributes	249
7.15.4	Instance attributes	249
7.15.5	Common services	253
8	Other DLE elements of procedure.....	254
8.1	Network attachment monitor (NAM).....	254
8.1.1	General	254
8.1.2	Default parameters	256
8.1.3	Auto-addressing	256
8.1.4	Valid MAC IDs	257
8.1.5	State machine description.....	257
8.2	Calculating link parameters.....	263
8.2.1	Link parameters.....	263
8.2.2	Conditions affecting link parameters	263
8.2.3	Moderator change.....	263
8.2.4	NUT timing	264

8.2.5	Slot timing	265
8.2.6	Blanking	266
8.2.7	Example implementation	266
9	Detailed specification of DL components	271
9.1	General	271
9.2	Access control machine (ACM)	271
9.3	TxLLC	290
9.4	RxLLC	295
9.5	Transmit machine (TxM)	298
9.6	Receive machine (RxM)	302
9.7	Serializer	308
9.8	Deserializer	310
9.8.1	Octet construction	310
9.8.2	FCS checking	311
9.8.3	End of DLPDU processing	311
9.9	DLL management	311
10	Device Level Ring (DLR) protocol	313
10.1	General	313
10.2	Support for Multiple DLR Ring Pairs	314
10.3	Supported topologies	315
10.4	Overview of DLR operation	316
10.4.1	Normal operation	316
10.4.2	Link failures	317
10.5	Classes of DLR implementation	318
10.6	DLR behavior	319
10.6.1	DLR variables	319
10.6.2	Ring supervisor	319
10.6.3	Ring node	322
10.6.4	Sign on process	323
10.6.5	Neighbor check process	324
10.7	Implementation requirements	324
10.7.1	Embedded switch requirements and recommendations	324
10.7.2	DLR implementation requirements	325
10.7.3	IEC 61588 and Type 2 Ethernet considerations	326
10.7.4	IEEE Std 802.1Q-2018 STP/RSTP/MSTP considerations	326
10.8	Using non-DLR nodes in the ring network	326
10.8.1	General considerations	326
10.8.2	Non-DLR end devices	327
10.8.3	Non-DLR switches	327
10.9	Redundant gateway devices on DLR network	329
10.9.1	General	329
10.9.2	Supported topologies	330
10.9.3	Redundant gateway capable device	330
10.9.4	Redundant gateway device behavior	331
10.10	DLR messages	334
10.10.1	General	334
10.10.2	Common frame header	335
10.10.3	Beacon frame	336
10.10.4	Neighbor_Check request	336

10.10.5	Neighbor_Check_response	337
10.10.6	Link_Status/Neighbor_Status	337
10.10.7	Locate_Fault	338
10.10.8	Announce	338
10.10.9	Sign_On	338
10.10.10	Advertise	339
10.10.11	Flush_Tables	339
10.10.12	Learning_Update	340
10.11	State diagrams and state-event-action matrices	340
10.11.1	Beacon-based ring node	340
10.11.2	Announce-based ring node	347
10.11.3	Ring supervisor	351
10.11.4	Redundant gateway	366
10.12	Performance analysis	369
10.12.1	General	369
10.12.2	Redundant gateway switchover performance	373
11	PRP and HSR redundancy protocols	375
11.1	General	375
11.2	PRP overview	375
11.2.1	General	375
11.2.2	Address Conflict Detection (ACD)	376
11.3	HSR overview	377
12	LLDP protocol	378
12.1	General	378
12.2	LLDP overview	379
12.3	Type 2 LLDP Transmission Requirements	379
12.3.1	General	379
12.3.2	Chassis ID TLV (TLV Type = 1)	380
12.3.3	Port ID TLV (TLV Type = 2)	380
12.3.4	System Capabilities TLV (TLV Type = 7)	381
12.3.5	Management Address (TLV Type = 8)	381
12.3.6	Type 2 Identification TLV (TLV Type = 127)	381
12.3.7	Type 2 MAC Address TLV (TLV Type = 127)	381
12.3.8	Type 2 Interface Label TLV (TLV Type = 127)	381
12.3.9	Additional Ethernet Capabilities TLV (TLV Type = 127)	382
12.4	Type 2 LLDP Reception Requirements	382
12.5	Type 2 LLDP Reporting Requirements	382
12.6	EtherNet/IP LLDP Link State Transition Requirements	382
Annex A (normative)	Indicators and switches	383
A.1	Purpose	383
A.2	Indicators	383
A.2.1	General indicator requirements	383
A.2.2	Common indicator requirements	383
A.2.3	Fieldbus specific indicator requirements – option 1	385
A.2.4	Fieldbus specific indicator requirements – option 2	389
A.2.5	Fieldbus specific indicator requirements – option 3	393
A.3	Switches	398
A.3.1	Common switch requirements	398
A.3.2	Fieldbus specific switch requirements – option 1	398

A.3.3	Fieldbus specific switch requirements – option 2	398
A.3.4	Fieldbus specific switch requirements – option 3	399
Bibliography		400
Figure 1 – Data-link layer internal architecture		36
Figure 2 – Relationships of DLSAPs, DLSAP-addresses, and group DL-addresses		39
Figure 3 – Basic structure of a MAC ID address		39
Figure 4 – Basic structure of a generic tag address		40
Figure 5 – Basic structure of a fixed tag address		40
Figure 6 – M_symbols and Manchester encoding at 5 MHz		42
Figure 7 – NUT structure		45
Figure 8 – Media access during scheduled time		46
Figure 9 – Media access during unscheduled time		47
Figure 10 – DLPDU format		48
Figure 11 – Aborting a DLPDU during transmission		52
Figure 12 – Lpacket format		53
Figure 13 – Generic tag Lpacket format		54
Figure 14 – Fixed tag Lpacket format		54
Figure 15 – Goodness parameter of TimeDist_Lpacket		68
Figure 16 – Example I'm alive processing algorithm		75
Figure 17 – Keeper CRC algorithm		99
Figure 18 – Keeper object power-up state diagram		111
Figure 19 – Keeper object operating state diagram		112
Figure 20 – Synchronized network change processing		115
Figure 21 – State transition diagram for TCP/IP Interface object		147
Figure 22 – State transition diagram for TCP/IP Interface object		148
Figure 23 – ACD Behavior		150
Figure 24 – State transition diagram for Ethernet Link object		169
Figure 25 – Connection Configuration object edit flowchart		200
Figure 26 – Communication objects diagram for example device		229
Figure 27 – NAM state machine		256
Figure 28 – Devices with Multiple DLR Ring Pairs		314
Figure 29 – DLR rings connected to switches		315
Figure 30 – Normal operation of a DLR network		316
Figure 31 – Beacon and Announce frames		316
Figure 32 – Link failure		317
Figure 33 – Network reconfiguration after link failure		318
Figure 34 – Neighbor Check process		324
Figure 35 – Unsupported topology – example 1		328
Figure 36 – Unsupported topology – example 2		328
Figure 37 – DLR ring connected to switches through redundant gateways		330
Figure 38 – DLR redundant gateway capable device		331
Figure 39 – Advertise frame		333

Figure 40 – State transition diagram for Beacon frame based non-supervisor ring node.....	341
Figure 41 – State transition diagram for Announce frame based non-supervisor ring node	347
Figure 42 – State transition diagram for ring supervisor	351
Figure 43 – State transition diagram for redundant gateway.....	366
Figure 44 – PRP network	376
Figure 45 – Directly Attached SANs	377
Figure 46 – Virtual DANs	377
Figure 47 – HSR network	378
Figure 48 – IEEE LLDP PDU Format (source IEEE Std 802.1AB-2016).....	379
Figure 49 – Type 2 LLDP PDU Format	380
Figure 50 – Type 2 Identification TLV Format.....	381
Figure 51 – Type 2 MAC Address TLV Format	381
Figure 52 – Type 2 Interface Label TLV Format	382
Figure A.1 – Non redundant network status indicator labeling	389
Figure A.2 – Redundant network status indicator labeling	389
Figure A.3 – Network status indicator state diagram	392
Figure A.4 – Examples of multiple network status indicators	392
Table 1 – Format of attribute tables	34
Table 2 – Data-link layer components	36
Table 3 – MAC ID addresses allocation	40
Table 4 – Fixed tag service definitions	40
Table 5 – Data encoding rules	42
Table 6 – M Data symbols.....	43
Table 7 – Truth table for ph_status_indication.....	43
Table 8 – FCS length, polynomials and constants	49
Table 9 – DLL support services and objects.....	56
Table 10 – Elementary data types.....	59
Table 11 – DLL events	63
Table 12 – Time distribution priority	69
Table 13 – Format of the TUI Lpacket	71
Table 14 – ControlNet object class attributes	80
Table 15 – ControlNet object instance attributes	81
Table 16 – TUI status flag bits	86
Table 17 – Mac_ver bits.....	87
Table 18 – Channel state bits	87
Table 19 – ControlNet object common services.....	89
Table 20 – ControlNet object class specific services	90
Table 21 – Keeper object revision history	92
Table 22 – Keeper object class attributes	92
Table 23 – Keeper object instance attributes	93
Table 24 – Keeper operating state definitions	97

Table 25 – Port status flag bit definitions	97
Table 26 – TUI status flag bits	98
Table 27 – Keeper attributes	101
Table 28 – Memory requirements (in octets) for the Keeper attributes.....	101
Table 29 – Keeper object common services	102
Table 30 – Keeper object class specific services	102
Table 31 – Service error codes	103
Table 32 – Wire order format of the TUI Lpacket.....	107
Table 33 – Service error codes	108
Table 34 – Keeper object operating states	109
Table 35 – Keeper object state event matrix	113
Table 36 – Scheduling object class attributes	116
Table 37 – Scheduling object instance attributes	117
Table 38 – Scheduling object common services	117
Table 39 – Status error descriptions for Create	118
Table 40 – Status error descriptions for Delete and Kick_Timer	119
Table 41 – Scheduling object class specific services	119
Table 42 – Status error descriptions for Read	121
Table 43 – Status error descriptions for Conditional_Write	122
Table 44 – Status error descriptions for Forced_Write.....	122
Table 45 – Status error descriptions for Change_Start.....	123
Table 46 – Status error descriptions for Break_Connections	124
Table 47 – Status error descriptions for Change_Complete.....	124
Table 48 – Status error descriptions for Restart_Connections	125
Table 49 – Revision history	126
Table 50 – TCP/IP Interface object class attributes	127
Table 51 – TCP/IP Interface object instance attributes	127
Table 52 – Status bits	132
Table 53 – Configuration capability bits	133
Table 54 – Configuration control bits.....	134
Table 55 – Example path	135
Table 56 – Interface configuration components	136
Table 57 – Alloc control values	138
Table 58 – AcdActivity values	139
Table 59 – ArpPdu – ARP Response PDU in binary format	139
Table 60 – Admin Capability member bit definitions	140
Table 61 – Admin Capability member bit definitions	141
Table 62 – TCP/IP Interface connection point 1, Standard Network Diagnostics.....	142
Table 63 – TCP/IP Interface object common services	143
Table 64 – Get_Attributes_All response format	144
Table 65 – Set_Attributes_All request format	145
Table 66 – TCP/IP Interface object class specific services.....	145
Table 67 – Set_Port_Admin_State service request parameters	146

Table 68 – Set_Protocol_Admin_State service request parameters.....	146
Table 69 – Class specific error codes	147
Table 70 – Ethernet link object revision history	154
Table 71 – Ethernet link object class attributes	155
Table 72 – Ethernet link object instance attributes	155
Table 73 – Interface flags bits	160
Table 74 – Control bits	162
Table 75 – Interface type	163
Table 76 – Interface state	163
Table 77 – Admin state	163
Table 78 – Capability Bits	164
Table 79 – Ethernet Link connection point 1, Standard Network Diagnostics.....	165
Table 80 – Ethernet Link object common services.....	166
Table 81 – Get_Attributes_All response format	167
Table 82 – Ethernet Link object class specific services	168
Table 83 – DeviceNet object revision history.....	170
Table 84 – DeviceNet object class attributes.....	170
Table 85 – DeviceNet object instance attributes.....	170
Table 86 – Bit rate attribute values	173
Table 87 – BOI attribute values.....	173
Table 88 – Diagnostic counters bit description	176
Table 89 – DeviceNet object common services	177
Table 90 – Reset service parameter.....	177
Table 91 – Reset service parameter values	177
Table 92 – DeviceNet object class specific services.....	178
Table 93 – Connection Configuration object revision history	179
Table 94 – Connection Configuration object class attributes	179
Table 95 – Format number values.....	181
Table 96 – Connection Configuration object instance attributes	181
Table 97 – Originator connection status values.....	185
Table 98 – Target connection status values	185
Table 99 – Connection flags	186
Table 100 – I/O mapping formats	188
Table 101 – Services valid during a change operation	190
Table 102 – Connection Configuration object common services	190
Table 103 – Get_Attributes_All Response – class level.....	191
Table 104 – Get_Attributes_All response – instance level.....	191
Table 105 – Set_Attributes_All error codes	193
Table 106 – Set_Attributes_All request	193
Table 107 – Create request parameters	195
Table 108 – Create error codes	195
Table 109 – Delete error codes.....	195
Table 110 – Restore error codes.....	196

Table 111 – Connection Configuration object class specific services	196
Table 112 – Change_Start error codes.....	197
Table 113 – Get_Status service parameter	198
Table 114 – Get_Status service response	198
Table 115 – Get_Status service error codes	198
Table 116 – Change_Complete service parameter	199
Table 117 – Change_Complete service error codes	199
Table 118 – Audit_Changes service parameter	199
Table 119 – Audit_Changes service error codes	199
Table 120 – Revision history	201
Table 121 – DLR object class attributes	201
Table 122 – DLR object instance attributes	202
Table 123 – Network Status values	206
Table 124 – Ring Supervisor Status values	206
Table 125 – Capability flags	210
Table 126 – Redundant Gateway Status values	212
Table 127 – DLR connection point 1, Standard Network Diagnostics.....	213
Table 128 – DLR connection point 2, Standard Network Diagnostics.....	213
Table 129 – DLR object common services	214
Table 130 – Get_Attributes_All Response – Object Revision 1, non supervisor device.....	214
Table 131 – Get_Attributes_All Response – Object Revision 1, supervisor-capable device	215
Table 132 – Get_Attributes_All Response – Object Revision 2, non supervisor device.....	215
Table 133 – Get_Attributes_All Response – All other cases	216
Table 134 – DLR object class specific services	217
Table 135 – QoS object revision history	218
Table 136 – QoS object class attributes	219
Table 137 – QoS object instance attributes	219
Table 138 – Default DCSP values and usages	220
Table 139 – QoS object common services	221
Table 140 – Port object revision history	222
Table 141 – Port object class attributes	222
Table 142 – Port object instance attributes	223
Table 143 – Port Type and associated Logical Link Object classes and Port Type Name values.....	225
Table 144 – Port Routing Capabilities attribute bit definitions.....	228
Table 145 – Contents of Associated Communication objects attribute 11 for the two Port object instances of the example device	229
Table 146 – Port object common services	230
Table 147 – Get_Attributes_All response– class level	230
Table 148 – Get_Attributes_All response– instance level	230
Table 149 – Revision history	231
Table 150 – Class attributes	231
Table 151 – Instance attributes	231

Table 152 – Node Type	235
Table 153 – Switching Node	236
Table 154 – HSR Mode	237
Table 155 – RedBox ID	238
Table 156 – PRP/HSR Protocol connection point 1, Standard Network Diagnostics	239
Table 157 – PRP/HSR Protocol object common services	240
Table 158 – Get_Attributes_All response	240
Table 159 – Revision history	242
Table 160 – Class attributes	242
Table 161 – Instance attributes	242
Table 162 – Remote Node Type	244
Table 163 – PRP/HSR Nodes Tables object common services	245
Table 164 – Get_Attributes_All response	245
Table 165 – Revision history	245
Table 166 – Class attributes	246
Table 167 – Instance attributes	246
Table 168 – Bit Definitions of the LLDP Enable Array	247
Table 169 – LLDP Management object common services	248
Table 170 – Get_Attributes_All response	248
Table 171 – Revision history	249
Table 172 – Class attributes	249
Table 173 – Instance attributes	249
Table 174 – Bitmaps of supported capabilities & enabled capabilities	253
Table 175 – LLDP Management object common services	253
Table 176 – Get_Attributes_All response	254
Table 177 – NAM states	255
Table 178 – Default link parameters	256
Table 179 – PhL timing characteristics	264
Table 180 – DLR variables	319
Table 181 – DLR Link speed and duplex requirements	325
Table 182 – Redundant gateway variables	332
Table 183 – MAC addresses for DLR messages	335
Table 184 – IEEE Std 802.1Q-2018 common frame header format	335
Table 185 – DLR message payload fields	335
Table 186 – DLR frame types	336
Table 187 – Format of the Beacon frame	336
Table 188 – Ring State values	336
Table 189 – Format of the Neighbor_Check request	337
Table 190 – Format of the Neighbor_Check response	337
Table 191 – Format of the Link_Status/Neighbor_Status frame	337
Table 192 – Link/Neighbor status values	338
Table 193 – Format of the Locate_Fault frame	338
Table 194 – Format of the Announce frame	338

Table 195 – Format of the Sign_On frame	339
Table 196 – Format of the Advertise frame	339
Table 197 – Gateway state values	339
Table 198 – Format of the Flush_Tables frame	340
Table 199 – Format of the Learning_Update frame	340
Table 200 – Parameter values for Beacon frame based non-supervisor ring node	341
Table 201 – LastBcnRcvPort bit definitions	342
Table 202 – State-event-action matrix for Beacon frame based non-supervisor ring node	342
Table 203 – Parameter values for Announce frame based non-supervisor ring node	348
Table 204 – State-event-action matrix for Announce frame based non-supervisor ring node	348
Table 205 – Parameter values for ring supervisor node	352
Table 206 – LastBcnRcvPort bit definitions	353
Table 207 – State-event-action matrix for ring supervisor node	353
Table 208 – Parameter values for redundant gateway node	366
Table 209 – State-event-action matrix for redundant gateway node	367
Table 210 – Parameters/assumptions for example performance calculations	370
Table 211 – Example ring configuration parameters and performance	373
Table 212 – Variables for performance analysis	374
Table 213 – LLDP support requirements	379
Table 214 – LLDP TLV Type Values	380
Table A.1 – Module status indicator	384
Table A.2 – Time Sync status indication	385
Table A.3 – Network status indicators	387
Table A.4 – Network status indicator	391
Table A.5 – Combined Module/Network status indicator	393
Table A.6 – Network status indicator	395
Table A.7 – Combined module/network status indicator	396
Table A.8 – I/O status indicator	397
Table A.9 – Bit rate switch encoding	399

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**INDUSTRIAL COMMUNICATION NETWORKS –
FIELDBUS SPECIFICATIONS –****Part 4-2: Data-link layer protocol specification –
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 61158-4-2 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 fifth edition cancels and replaces the fourth edition published in 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" and "slave" replaced, mainly in 7.3 and 7.7;
- c) new STIME, UTIME, NTIME, STRINGI and EPATH data types in 6.1.3;
- d) updates, addition of diagnostics connection points and new service for TCP/IP interface object in 7.5;
- e) addition of diagnostics connection points and new service for Ethernet Link object in 7.6;
- f) update of Get/Set_Attributes_All parameters for the Connection Configuration object in 7.8;
- g) addition of diagnostics connection points and new service for DLR object in 7.9;
- h) extensions and clarifications of Port object in 7.11;
- i) addition of diagnostics connection points and new service for PRP/HSR Protocol object in 7.12;
- j) addition of LLDP Management and LLDP Data Table objects in 7.1, 7.14 and 7.15;
- k) addition of LLDP protocol support in Clause 12;
- l) addition of a combined module/network indicator in A.2.4.5;
- m) removal of all references to CPF and CPs (material moved to profile documents);
- n) miscellaneous editorial corrections.

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

Draft	Report on voting
65C/1202/FDIS	65C/1243/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 parts of the IEC 61158 series, published under the general title *Industrial communication networks – Fieldbus specifications*, can be found on the IEC web site.

The committee has decided that the contents of this 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.

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

INTRODUCTION

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

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

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

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

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents. IEC takes no position concerning the evidence, validity, and scope of these patent rights.

The holders of these patent rights have assured IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holders of these patent rights is registered with IEC. Information may be obtained from the patent database available at <http://patents.iec.ch>.

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

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 4-2: Data-link layer protocol specification – Type 2 elements

1 Scope

1.1 General

The data-link layer provides basic time-critical messaging communications between devices in an automation environment.

This part of IEC 61158 specifies a main protocol with the following characteristics.

- This protocol provides communication opportunities to all participating data-link entities, sequentially and in a cyclic synchronous manner. Foreground scheduled access is available for time-critical activities together with background unscheduled access for less critical activities.
- Deterministic and synchronized transfers can be provided at cyclic intervals up to 1 ms and device separations of 25 km. This performance is adjustable dynamically and on-line by re-configuring the parameters of the local link whilst normal operation continues. By similar means, DL connections and new devices can be added or removed during normal operation.
- This protocol provides means to maintain clock synchronization across an extended link with a precision better than 10 μ s.
- This protocol optimizes each access opportunity by concatenating multiple DLSDUs and associated DLPCI into a single DLPDU, thereby improving data transfer efficiency for data-link entities that actively source multiple streams of data.
- The maximum system size is an unlimited number of links of 99 nodes, each with 255 DLSAP-addresses. Each link has a maximum of 2^{24} related peer and publisher DLCEPs.

This document specifies additional lower layers protocols or implementations of additional lower layers protocols for use in combination with ISO/IEC/IEEE 8802-3.

This document specifies a set of corresponding objects providing a consistent management interface to the lower layers.

1.2 Specifications

This document specifies

- a) procedures for the timely transfer of data and control information from one data-link user entity to a peer user entity, and among the data-link entities forming the distributed data-link service provider;
- b) the structure of the fieldbus DLPDUs used for the transfer of data and control information by the protocol of this document, and their representation as physical interface data units.

1.3 Procedures

The procedures are defined in terms of

- a) the interactions between peer DL-entities (DLEs) through the exchange of fieldbus DLPDUs;
- b) the interactions between a DL-service (DLS) provider and a DLS-user in the same system through the exchange of DLS primitives;

- c) the interactions between a DLS-provider and a Ph-service provider in the same system through the exchange of Ph-service primitives.

1.4 Applicability

These procedures are applicable to instances of communication between systems which support time-critical communications services within the data-link layer of the OSI or fieldbus reference models, and which require the ability to interconnect in an open systems interconnection environment.

Profiles provide a simple multi-attribute means of summarizing capabilities of an implementation, and thus its applicability to various time-critical communications needs.

1.5 Conformance

This document also specifies conformance requirements for systems implementing these procedures. This document does not contain tests to demonstrate compliance with such requirements.

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 61131-3, *Programmable controllers – Part 3: Programming languages*

IEC 61158-3-2:2023, *Industrial communication networks – Fieldbus specifications – Part 3-2: Data-link layer service definition – Type 2 elements*

IEC 61158-5-2:2023, *Industrial communication networks – Fieldbus specifications – Part 5-2: Application layer service definition – Type 2 elements*

IEC 61158-6-2:2023, *Industrial communication networks – Fieldbus specifications – Part 6-2: Application layer protocol specification – Type 2 elements*

IEC 61588, *Precision clock synchronization protocol for networked measurement and control systems*

IEC 61784-3-2, *Industrial communication networks – Profiles – Part 3-2: Functional safety fieldbuses – Additional specifications for CPF 2*

IEC 62026-3:2014, *Low-voltage switchgear and controlgear – Controller-device interfaces (CDIs) – Part 3: DeviceNet*

IEC 62439-3:2016¹, *Industrial communication networks – High availability automation networks – Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)*

¹ A newer edition of this standard has been published, but only the cited edition applies.

ISO/IEC 13239, *Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures*

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/IEEE 8802-3, *Telecommunications and exchange between information technology systems – Requirements for local and metropolitan area networks – Part 3: Standard for Ethernet*

ISO 11898-1:2015, *Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signalling*

IEEE Std 802.1AB-2016, *IEEE Standard for local and metropolitan area networks – Station and Media Access Control Connectivity Discovery*

IEEE Std 802.1ABcu-2021, *Standard for local and metropolitan area networks – Station and Media Access Control Connectivity Discovery Amendment: YANG Data Model*

IEEE Std 802.1Q-2018, *IEEE standard for local and metropolitan area networks – Bridges and bridged networks*

IEEE Std 802.3-2018, *IEEE Standard for Ethernet*

IETF RFC 951, W.J. Croft, J. Gilmore, *Bootstrap Protocol*, September 1985, available at <https://www.rfc-editor.org/info/rfc951> [viewed 2022-02-18]

IETF RFC 1213, K. McCloghrie, M. Rose, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II*, March 1991, available at <https://www.rfc-editor.org/info/rfc1213> [viewed 2022-02-18]

IETF RFC 1542, W. Wimer, *Clarifications and Extensions for the Bootstrap Protocol*, October 1993, available at <https://www.rfc-editor.org/info/rfc1542> [viewed 2022-02-18]

IETF RFC 1643, F. Kastenholz, *Definitions of Managed Objects for the Ethernet-like Interface Types*, July 1994, available at <https://www.rfc-editor.org/info/rfc1643> [viewed 2022-02-18]

IETF RFC 2131, R. Droms, *Dynamic Host Configuration Protocol*, March 1997, available at <https://www.rfc-editor.org/info/rfc2131> [viewed 2022-02-18]

IETF RFC 2132, S. Alexander, R. Droms, *DHCP Options and BOOTP Vendor Extensions*, March 1997, available at <https://www.rfc-editor.org/info/rfc2132> [viewed 2022-02-18]

IETF RFC 2863, K. McCloghrie, F. Kastenholz, *The Interfaces Group MIB*, June 2000, available at <https://www.rfc-editor.org/info/rfc2863> [viewed 2022-02-18]

IETF RFC 3418, R. Presuhn, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*, December 2002, available at <https://www.rfc-editor.org/info/rfc3418> [viewed 2022-02-18]

IETF RFC 3635, J. Flick, *Definitions of Managed Objects for the Ethernet-like Interface Types*, September 2003, available at <https://www.rfc-editor.org/info/rfc3635> [viewed 2022-02-18]

IETF RFC 4541, M. Christensen, K. Kimball, F. Solensky, *Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches*, May 2006, available at <https://www.rfc-editor.org/info/rfc4541> [viewed 2022-02-18]

IETF RFC 5227:2008, S. Cheshire, *IPv4 Address Conflict Detection*, July 2008, available at <https://www.rfc-editor.org/info/rfc5227> [viewed 2022-02-18]

3 Terms, definitions, symbols, abbreviated terms and conventions

For the purposes of this document, the following terms, definitions, symbols and abbreviated terms 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	called-DL-address	[ISO/IEC 7498-3]
3.1.2	calling-DL-address	[ISO/IEC 7498-3]
3.1.3	centralized multi-end-point-connection	[ISO/IEC 7498-1]
3.1.4	correspondent (N)-entities correspondent DL-entities (N=2) correspondent Ph-entities (N=1)	[ISO/IEC 7498-1]
3.1.5	demultiplexing	[ISO/IEC 7498-1]
3.1.6	DL-address	[ISO/IEC 7498-3]
3.1.7	DL-address-mapping	[ISO/IEC 7498-1]
3.1.8	DL-connection	[ISO/IEC 7498-1]
3.1.9	DL-connection-end-point	[ISO/IEC 7498-1]
3.1.10	DL-connection-end-point-identifier	[ISO/IEC 7498-1]
3.1.11	DL-connection-mode transmission	[ISO/IEC 7498-1]
3.1.12	DL-connectionless-mode transmission	[ISO/IEC 7498-1]
3.1.13	DL-data-sink	[ISO/IEC 7498-1]
3.1.14	DL-data-source	[ISO/IEC 7498-1]
3.1.15	DL-duplex-transmission	[ISO/IEC 7498-1]
3.1.16	DL-facility	[ISO/IEC 7498-1]
3.1.17	DL-local-view	[ISO/IEC 7498-3]
3.1.18	DL-name	[ISO/IEC 7498-3]
3.1.19	DL-protocol	[ISO/IEC 7498-1]
3.1.20	DL-protocol-connection-identifier	[ISO/IEC 7498-1]
3.1.21	DL-protocol-control-information	[ISO/IEC 7498-1]
3.1.22	DL-protocol-data-unit	[ISO/IEC 7498-1]
3.1.23	DL-protocol-version-identifier	[ISO/IEC 7498-1]
3.1.24	DL-relay	[ISO/IEC 7498-1]

3.1.25	DL-service-connection-identifier	[ISO/IEC 7498-1]
3.1.26	DL-service-data-unit	[ISO/IEC 7498-1]
3.1.27	DL-simplex-transmission	[ISO/IEC 7498-1]
3.1.28	DL-subsystem	[ISO/IEC 7498-1]
3.1.29	DL-user-data	[ISO/IEC 7498-1]
3.1.30	flow control	[ISO/IEC 7498-1]
3.1.31	layer-management	[ISO/IEC 7498-1]
3.1.32	multiplexing	[ISO/IEC 7498-3]
3.1.33	naming-(addressing)-authority	[ISO/IEC 7498-3]
3.1.34	naming-(addressing)-domain	[ISO/IEC 7498-3]
3.1.35	naming-(addressing)-subdomain	[ISO/IEC 7498-3]
3.1.36	(N)-entity DL-entity Ph-entity	[ISO/IEC 7498-1]
3.1.37	(N)-interface-data-unit DL-service-data-unit (N=2) Ph-interface-data-unit (N=1)	[ISO/IEC 7498-1]
3.1.38	(N)-layer DL-layer (N=2) Ph-layer (N=1)	[ISO/IEC 7498-1]
3.1.39	(N)-service DL-service (N=2) Ph-service (N=1)	[ISO/IEC 7498-1]
3.1.40	(N)-service-access-point DL-service-access-point (N=2) Ph-service-access-point (N=1)	[ISO/IEC 7498-1]
3.1.41	(N)-service-access-point-address DL-service-access-point-address (N=2) Ph-service-access-point-address (N=1)	[ISO/IEC 7498-1]
3.1.42	peer-entities	[ISO/IEC 7498-1]
3.1.43	Ph-interface-control-information	[ISO/IEC 7498-1]
3.1.44	Ph-interface-data	[ISO/IEC 7498-1]
3.1.45	primitive name	[ISO/IEC 7498-3]
3.1.46	reassembling	[ISO/IEC 7498-1]
3.1.47	recombining	[ISO/IEC 7498-1]
3.1.48	reset	[ISO/IEC 7498-1]
3.1.49	responding-DL-address	[ISO/IEC 7498-3]
3.1.50	routing	[ISO/IEC 7498-1]
3.1.51	segmenting	[ISO/IEC 7498-1]
3.1.52	sequencing	[ISO/IEC 7498-1]
3.1.53	splitting	[ISO/IEC 7498-1]
3.1.54	synonymous name	[ISO/IEC 7498-3]
3.1.55	systems-management	[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 DL-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 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: Definition is provided in a generic manner "(N)-SAP" in ISO/IEC 7498-1.

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 DL-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 DL-user-data

3.4 Additional Type 2 definitions**3.4.1****actual packet interval****API**

measure of how frequently a specific connection produces its Lpacket data

3.4.2

allocate

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

3.4.3

application

function or data structure for which data is consumed or produced

3.4.4

application objects

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

3.4.5

attribute

description of an externally visible characteristic or feature of an object

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

3.4.6

behavior

indication of how the object responds to particular events

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

3.4.7

bit

unit of information consisting of a 1 or a 0

Note 1 to entry: This is the smallest data unit that can be transmitted.

3.4.8

blanking or blanking time

length of time required after transmitting before the node is allowed to receive

3.4.9

client

- 1) object which uses the services of another (server) object to perform a task
- 2) initiator of a message to which a server reacts

3.4.10

communication objects

components that manage and provide run time exchange of messages across the network such as the ControlNet object, the Unconnected Message Manager (UCMM)

3.4.11

connection

logical binding between two application objects within the same or different devices

3.4.12

connection ID

CID

identifier assigned to a transmission, associated with a particular connection between producers and consumers, that identifies a specific piece of application information

Note 1 to entry: Within the data link this is a generic tag.

3.4.13**cyclic redundancy check****CRC**

residual value computed from an array of data and used as a representative signature for the array

3.4.14**cyclic**

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

3.4.15**deafness**

situation where the node cannot hear the moderator DLPDU but can hear other link traffic

3.4.16**device**

physical hardware connection to the link

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

3.4.17**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.

3.4.18**end delimiter**

unique set of M_symbols that identifies the end of a DLPDU

3.4.19**error**

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

3.4.20**extended link**

links connected by DL-routers (sometimes called a local area network)

3.4.21**FCS error**

error that occurs when the computed frame check sequence value after reception of all the octets in a DLPDU does not match the expected residual

3.4.22**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.

3.4.23**frame**

denigrated synonym for DLPDU

3.4.24

Frame Check Sequence

FCS

redundant data derived from a block of data within a DLPDU (frame), using a hash function, and stored or transmitted together with the block of data, in order to detect data corruption

3.4.25

generic tag

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

3.4.26

guardband

time slot allocated for the transmission of the moderator DLPDU

3.4.27

implicit token

mechanism that governs the right to transmit

Note 1 to entry: No actual token message is transmitted on the medium. Each node keeps track of the MAC ID of the node that it believes currently holds the right to transmit. The right to transmit is passed from node to node by keeping a record of the node that last transmitted. A slot time is used to allow a missing node to be skipped in the rotation.

3.4.28

implicit token register

register that contains the MAC ID of the node that holds the right to transmit

3.4.29

instance

actual physical occurrence of an object within a class, identifying one of many objects within the same object class

EXAMPLE California is an instance of the object class state.

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

3.4.30

instance attribute

attribute whose value is unique to an object instance and whose definition is shared by all instances of an object

3.4.31

instantiated

object that has been created in a device

3.4.32

Keeper

object responsible for distributing link configuration data to all nodes on the link

3.4.33

link

collection of nodes with unique MAC IDs, attached to Ph-segments connected by Ph-repeaters

3.4.34

little endian

model of memory organization which stores the least significant octet at the lowest address, and of transmission organization which transfers the least significant octet first on the medium

3.4.35**Lpacket**

well-defined sub-portion of a DLPDU consisting of

- a) size and control information,
- b) a fixed tag or a generic tag, and
- c) DLS-user data or, when the tag has DL-significance, DL-data

3.4.36**M_symbol(s)**

symbols that represent the data bits and related information encoded and transmitted by the PhL

3.4.37**maximum scheduled node**

node with highest MAC ID that can use scheduled time on a link

3.4.38**maximum unscheduled node**

node with highest MAC ID that can use unscheduled time on a link

3.4.39**Message Router**

object within a node that distributes messaging requests to the appropriate application objects

3.4.40**moderator**

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

3.4.41**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

3.4.42**multipoint connection**

connection from one node to many

Note 1 to entry: Multipoint connections allow messages from a single producer (publisher) to be received by many consumer (subscriber) nodes.

3.4.43**network**

series of nodes connected by some type of communication medium

Note 1 to entry: The connection paths between any pair of nodes can include repeaters, routers and gateways.

3.4.44**network access monitor****NAM**

component within a node that manages communication with a temporary node connected to the nodes local NAP

3.4.45**network access port****NAP**

PhL variant that allows a temporary node to be connected to the link by connection to the NAP of a permanent node

3.4.46

network address or node address

node's address on the link

Note 1 to entry: This is also called MAC ID.

3.4.47

network status indicators

indicators on a node indicating the status of the Physical and Data Link Layers

3.4.48

network update time

NUT

repetitive time interval in which data can be sent on the link

3.4.49

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.

3.4.50

non-concurrence

situation where a transmission is received from an unexpected MAC ID, which appears to violate the time based access protocol

Note 1 to entry: This can occur when a connection is made between two working links that are not synchronized with each other but who have the same configuration information.

3.4.51

non-data symbol

PhL symbol which violates the requirements of Manchester biphase L encoding

3.4.52

object

abstract representation of a particular component within a device

Note 1 to entry: An object can be

- 1) an abstract representation of a computer's capabilities. Objects can be composed of any or all of the following components:
 - a) data (information which changes with time);
 - b) configuration (parameters for behavior);
 - c) methods (things that can be done using data and configuration);
- 2) a collection of related data (in the form of variables) and methods (procedures) for operating on that data that have clearly defined interface and behavior.

3.4.53

object specific service

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

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

3.4.54

originator

client responsible for establishing a connection path to the target

3.4.55**permanent node**

node whose connection to the network does not utilize the network access port (NAP) PhL variant

Note 1 to entry: This node may optionally support a NAP PhL variant to allow temporary nodes to connect to the network.

3.4.56**produce**

act of sending data to be received by a consumer

3.4.57**producer**

node that is responsible for sending data

3.4.58**redundant media**

system using more than one medium to help prevent communication failures

3.4.59**requested packet interval****RPI**

measure of how frequently the originating application requires the transmission of Lpackets or data packets from the target application

3.4.60**rogue**

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

3.4.61**scheduled**

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

3.4.62**server**

object which provides services to another (client) object

3.4.63**service**

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

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

3.4.64**slot time**

maximum time required for detecting an expected transmission

Note 1 to entry: Each node waits a slot time for each missing node during the implied token pass.

3.4.65**start delimiter**

unique set of M_symbols that identifies the beginning of a DLPDU

3.4.66

supernode

DLE with a MAC ID of zero

Note 1 to entry: This DLE DL-address is reserved for special DLL functions

3.4.67

table unique identifier

TUI

unique reference code used by the ControlNet object and the Keeper object to reference a consistent set of link configuration attribute

3.4.68

tag

shorthand name for a specific piece of application information, two or three octets in length

3.4.69

target

end-node to which a connection is established

3.4.70

temporary node

same as transient node

3.4.71

tMinus

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

3.4.72

tone

instant of time which marks the boundary between two NUTs

3.4.73

tool

executable software program which interacts with the user to perform some function

EXAMPLE Link scheduling software (LSS).

3.4.74

transaction id

field within a UCMM header that matches a response with the associated request, which a server echoes in the response message

3.4.75

transient node

node that is only intended to be connected to the network on a temporary basis using the NAP PhL medium connected to the NAP of a permanent node

3.4.76

Unconnected Message Manager

UCMM

component within a node that transmits and receives unconnected explicit messages and sends them directly to the Message Router object

3.4.77

unconnected service

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

3.4.78**unscheduled**

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

3.4.79**vendor ID**

identification of each product manufacturer/vendor by a unique number

Note 1 to entry: Vendor IDs are assigned by the ODVA, Inc. organization (see <www.odva.org>).

3.5 Type 2 symbols and abbreviated terms

ACD	Address Conflict Detection	
ACM	Access control machine	
BOOTP	Bootstrap Protocol	[IETF RFC 951]
CA	Clock accuracy	
CAN	Controller Area Network	[ISO 11898-1]
CCITT	Consultative Committee for International Telephony and Telegraphy	
COP	Connection originator password	
DANP	Doubly Attached Node operating PRP	
DHCP	Dynamic Host Configuration Protocol	[IETF RFC 2131]
DLR	Device level ring	
DSCP	DiffServ code point	
FCS	Frame Check Sequence	
HSR	High-availability Seamless Redundancy	[IEC 62439-3]
IP	Internet Protocol	[IETF RFC 791]
LAN	Local Area Network	
LED	Light emitting diode	
LRE	Link Redundancy Entity (associated with PRP)	
LSS	Link scheduling software	
MAC ID	MAC address of a node	
MSTP	Multiple spanning tree protocol	[IEEE Std 802.1Q-2018]
NAM	Network attachment monitor	
NAP	Network access port	
ND	Non-data symbol	
PRP	Parallel Redundancy Protocol	[IEC 62439-3]
NUT	Network update time	
PT	Programming terminal (a temporary network connection)	
PTP	Precision Time Protocol	[IEC 61588]
QoS	Quality of service	
Rcv	Receive	
RedBox	PRP Redundancy Box (for attaching non-PRP nodes)	
RPI	Requested Lpacket interval	
RSTP	Rapid spanning tree protocol	[IEEE Std 802.1Q-2018]
Rx	Receive	

RxLLC	Receive logical link control	
RxM	Receive machine	
SAN	Singly Attached Node (in a PRP topology)	
SEM	State event matrix	
SMAX	MAC ID of the maximum scheduled node	
STD	State transition diagram, used to describe object behavior	
STP	Spanning tree protocol	[IEEE Std 802.1Q-2018]
TCP	Terminal Control Protocol	[IETF RFC 793]
TFTP	Trivial File Transfer Protocol	[IETF RFC 1350]
TUI	Table unique identifier	
Tx	Transmit	
TxLLC	Transmit logical link control	
TxM	Transmit machine	
UCCM	Unconnected Message Manager	
UMAX	MAC ID of maximum unscheduled node	
USR	Unscheduled start register	
VLAN	Virtual LAN	

3.6 Conventions for station management objects

Objects for station management are specified in Clause 7, using attributes, services and behavior.

Object attributes are specified in tables formatted according to Table 1.

Table 1 – Format of attribute tables

Attribute ID	Need in implementation	Access rule	NV	Name	Data type	Description of attribute	Semantics of values
See a)	See b)	See c)	See d)	See e)	See 6.1.3	See f)	See g)

a) "Attribute ID" is the identification value assigned to an attribute.

b) "Need in implementation" specifies whether or not the attribute is necessary in the implementation. An attribute may be optional, required or conditional.

A conditional attribute is required if certain object behaviors and/or attributes are implemented as defined by the class.

If an Attribute is optional, then a default value shall be defined in the semantics. An optional attribute's default value shall provide the same behavior as if the attribute was not implemented. If the default value of an optional attribute does not match the behavior of the object when the object does not implement the attribute, the object definition shall declare the behavior.

In all cases, the term "default" indicates a "factory default" as shipped from the vendor.

c) "Access rule" specifies how a requestor can access an attribute. The definitions for access rules are:

- **Settable (Set)** – The attribute shall be accessed by at least one of the set services. For enumerated attributes, if the device supports only one of the values and the object definition does not require more than one value to be supported, then the attribute may be implemented as Get only.

Settable attributes, unless otherwise specified by the object definition, shall be also accessed by get services.

- Gettable (Get) – The attribute shall be accessed by at least one of the get services.
- d) "NV" indicates whether an attribute value is maintained through power cycles. This column is used in object definitions where non-volatile storage of attribute values is required. An entry of "NV" indicates value shall be saved, "V" means not saved.
- e) Name refers to the attribute.
- f) Description of Attribute provides general information about the attribute.
- g) Semantics of Values specifies the meaning of the value of the attribute.

4 Overview of the data-link protocol

4.1 General

4.1.1 DLL architecture

The Type 2 DLL is modeled as an integrated Access Control Machine (ACM), with scheduling support functions, designed for reliable and efficient support of higher-level connection-mode and connectionless data transfer services. Interoperating with these higher level functions are DLL management functions.

PhL framing and delimiters are managed by DLL functions for serializing and deserializing M_symbol requests and indications.

Within the Data Link Layer, the Access Control Machine (ACM) has the primary responsibility for detecting and recovering from network disruptions. The major objectives of the ACM are

- to make sure that the local node detects and fully utilizes its assigned access slots in the schedule;
- to make sure that the local node does not interfere with the transmissions of other nodes, especially the moderator node;
- to start the next NUT (see 4.1.2) on schedule, whether the moderator DLPDU is heard or not;
- if the local node is the moderator, to transmit each moderator DLPDU strictly on schedule.

The Data Link Layer is comprised of the components listed in Table 2:

Table 2 – Data-link layer components

Components	Description
Access Control Machine (ACM)	receives and transmits control DLPDUs and header information, and determines the timing and duration of transmissions.
Transmit LLC (TxLLC)	buffers DLSDUs received from Station Management and the DLS-user, and determines which should be transmitted next.
Receive LLC (RxLLC)	performs the task of quarantining received link units until they are validated by a good FCS.
Transmit Machine (TxM)	receives requests to send DLPDU headers and trailers, and Lpackets from the ACM, and breaks them down into octet symbol requests that are sent to the Serializer.
Receive Machine (RxM)	assembles received Lpackets from octet symbols received from the Deserializer, and submits them to the RxLLC.
Serializer	receives octet symbols, encodes and serializes them, and sends them as M_symbols to the PhL. It is also responsible for generating the FCS.
Deserializer	receives M_symbols from the PhL, converts M_symbols into octets and sends them to the receive machine. It is also responsible for checking the FCS.
DLL Management Interface	holds the Station Management variables that belong to the DLL, and helps manage synchronized changes of the link parameters.

The internal arrangement of these components, and their interfaces, are shown in Figure 1. The arrowheads illustrate the primary direction of flow of data and control.

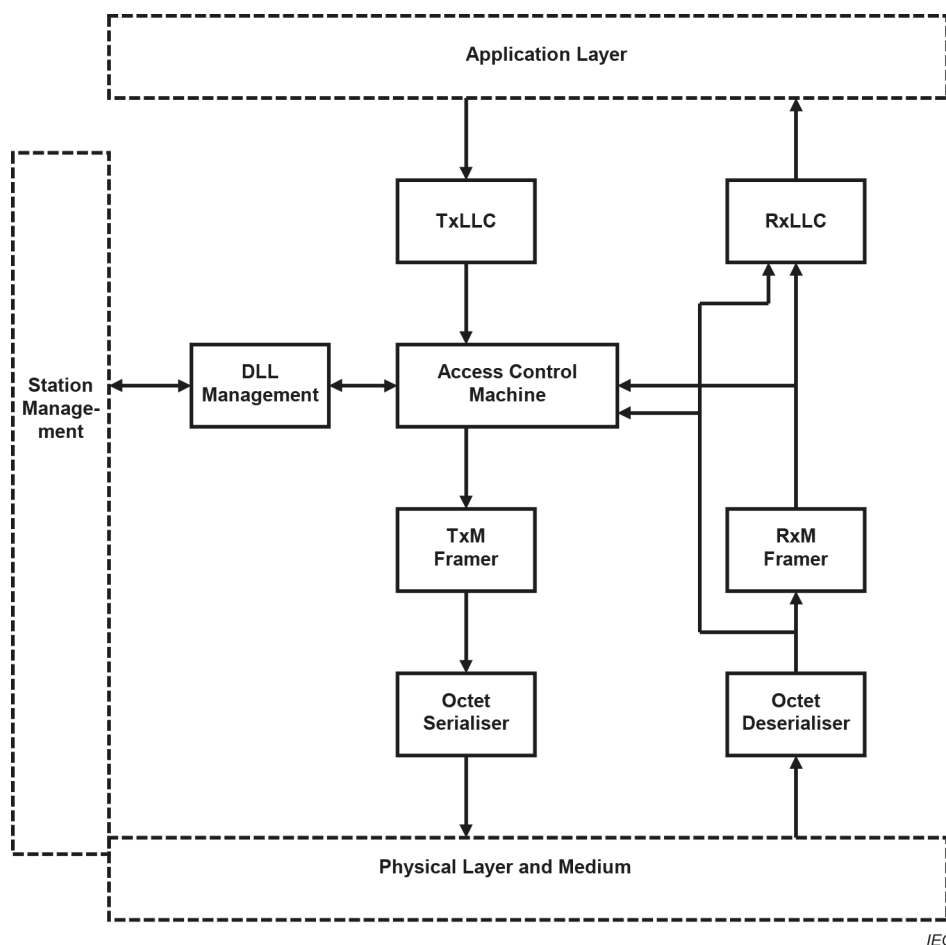


Figure 1 – Data-link layer internal architecture

4.1.2 Access control machine (ACM) and scheduling support functions

The ACM functions schedule all communication from one DLE to another. The timing of this communication is regulated at two levels,

- a) to provide an accurate time based distribution of scheduled opportunities for designated communications in accord with, and with a pre-established schedule for, the local link and any extended network of links,
- b) to provide a democratic distribution of unscheduled opportunities for arbitrary communications, generally in a cyclic but asynchronous manner.

Accurate schedule timing as provided by a) is very important to support many control and data collection tasks in the applications domain of this protocol. The schedule is based on a fixed, repetitive time cycle called network update time (NUT). The NUT is maintained in close synchronism among all nodes participating in the DLL and used to provide two types of access opportunity:

- 1) One opportunity for each active station to transfer one scheduled DLPDU containing multiple scheduled DLSDUs.

Only DLSDUs with scheduled as their QoS parameter may be included in a scheduled DLPDU.

NOTE DLSDUs with scheduled as their QoS parameter are usually connection mode transfers with generic as their tag parameter.

- 2) One opportunity for at least one active station to send one background (unscheduled) DLPDU containing multiple DLSDUs of any QoS type. If time is available within a NUT, one background MAC opportunity is offered to each other active station in the order of their MAC ID address. The station MAC ID for the first background opportunity in each NUT is incremented (+1 modulo the set of background addresses) for each successive NUT to democratically share the available time among active stations.

Support procedures are included for automatic transfer to backup scheduling services and to manage the use of redundant Ph media.

4.1.3 Connection-mode, connectionless-mode data transfer and DL service

The connection mode, connectionless mode and DL services provide the necessary functionality for

- managing DL provider interactions with the DL user by converting request and response primitives into necessary DLE operations and generating indications and confirm primitives where appropriate,
- determination of the DL address and control information to be added to each DLSDU,
- local confirmation of status for each outgoing DLSDU,
- DLPDU formation by concatenation of multiple DL user DLSDUs for efficient transfer as single PhPDUs, subject to DLPDU size limitations, QoS parameters and the type of access opportunity.

4.2 Services provided by the DL

4.2.1 Overview

The DLL provides connectionless data transfer services and connection-mode data transfer services for limited-size DLSDUs, an internally synchronized time service, scheduling services to control the time allocation of the underlying shared PhL service, a DL(SAP)-address and queue management service, and a packing/unpacking service that combines multiple DLSDUs into a single data transfer DLPDU for efficient use of each access opportunity.

4.2.2 QoS

4.2.2.1 Definition

The QoS parameter specifies priority options for DLSDUs and access opportunities. The available values are as follows:

4.2.2.2 DLL scheduled priority

The scheduled priority QoS provides accurate time based cyclic and acyclic sending of DLSDUs. The execution timing for this fieldbus scheduled service can be accurate and repeatable to better than 1 ms. Scheduled priority is normally used with connected mode services.

NOTE The active Keeper is responsible for regular publication of a unique table identifier (TUI) which ensures all nodes have a reference for the current set of link operating parameters. This TUI publication uses a fixed tag and is sent with QoS priority set to scheduled.

4.2.2.3 DLL high priority (unscheduled)

The high priority 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.

4.2.2.4 DLL low priority (unscheduled)

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

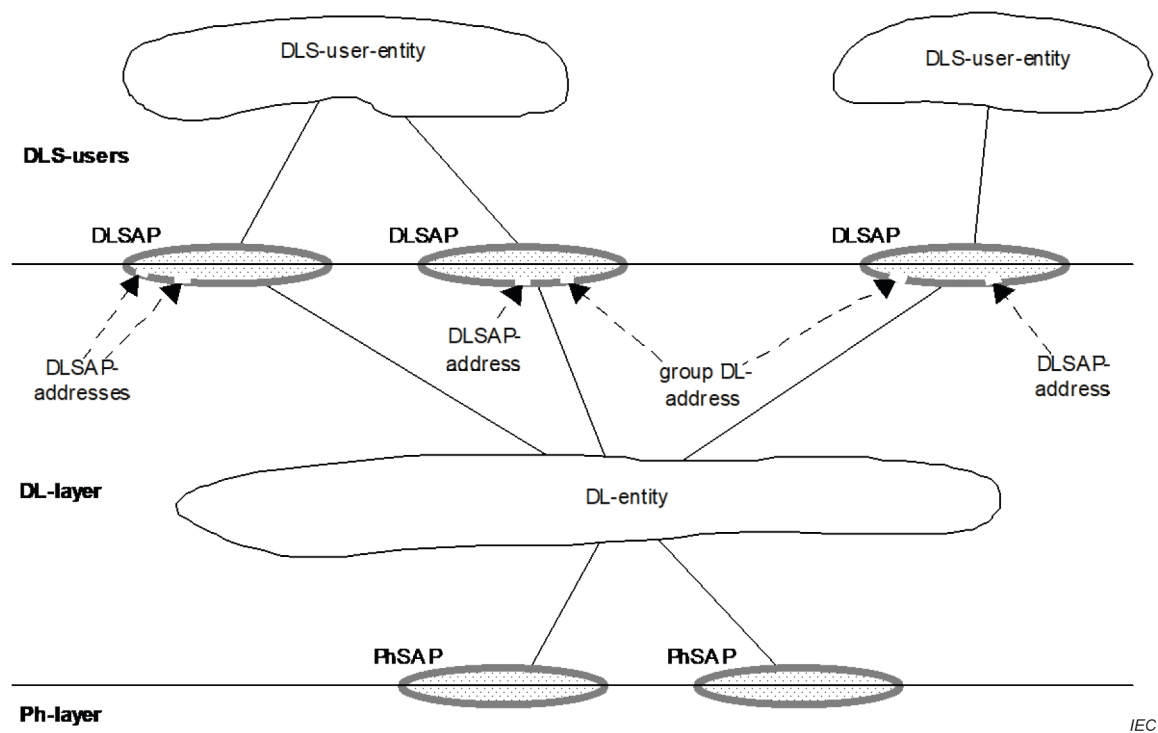
NOTE High and Low priority are used only in a local sense to set the order of servicing among locally submitted, unscheduled DLS-user-data.

4.3 Structure and definition of DL-addresses

4.3.1 General

DL-addresses are used as MAC ID addresses (Link node designators), Fixed tag addresses (DLSAP-addresses and group DL-addresses), and Generic tag addresses (DLCEP-addresses).

NOTE The definition of DLSAP, derived from ISO/IEC 7498-1, is explained here to facilitate understanding of the critical distinction between DLSAPs and their DL-addresses (see Figure 2).



NOTE 1 DLSAPs and physical layer service access points (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.

Figure 2 – Relationships of DLSAPs, DLSAP-addresses, and group DL-addresses

A single DLE may have multiple DLSAP-addresses and group DL-addresses associated with a single DLSAP.

Subclause 4.3 defines the form and structure of DL-addresses and includes specific definitions for some standard DL-addresses.

All addresses are formed of octets. Each octet shall be transferred to the Ph layer low-bit first and multiple octets shall be transferred low-order octet first (little endian format).

Three types of DL addresses are used (see 4.3.2, 4.3.3 and 4.3.4).

4.3.2 MAC ID address

MAC ID addresses identify physical nodes on the local link as shown in Figure 3.

Permitted values are within the range 0 to 99, values from 100 to 254 are reserved.



Figure 3 – Basic structure of a MAC ID address

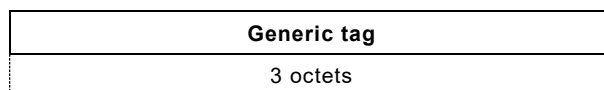
Table 3 specifies the use of MAC ID addresses.

Table 3 – MAC ID addresses allocation

MAC ID	Meaning and purpose
0x0	temporary address used for maintenance
SMAX	current maximum address value for Scheduled access opportunities. SMAX ≤ UMAX
UMAX	current maximum address value for Unscheduled access opportunities SMAX. ≤ UMAX ≤ 0x63
0x64 to 0xFE	reserved
0xFF	broadcast address to all MAC ID nodes on this link

4.3.3 Generic tag address

Generic tag addresses types identify all data transfers for connected mode services. The available capacity in the DLL is 3 octets, see Figure 4.

**Figure 4 – Basic structure of a generic tag address**

Generic tags are used to send connected DLSDUs and may be associated with any of the available QoS priorities (Scheduled, High, Low).

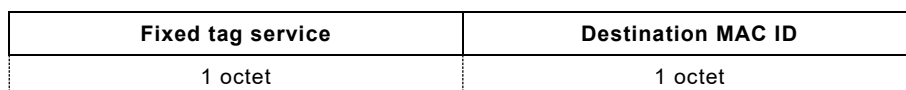
NOTE Negotiation and management of generic tags and connected mode services are the responsibility of the DL user.

4.3.4 Fixed tag address

Fixed tag addresses types identify DLSAPs within a designated station MAC ID. They are used with all data transfers for connectionless mode services.

NOTE Fixed tags are used to send connectionless DLSDUs and are usually associated with unscheduled QoS priorities (High, Low). Typical uses of fixed tags are for station management and to establish connections.

Each fixed tag address shall consist of two octets as shown in Figure 5.

**Figure 5 – Basic structure of a fixed tag address**

The first octet shall specify the service access point in the destination node as specified in Table 4. The second octet shall be the address of the destination node. The destination address shall be either a MAC ID or 0xFF, which indicates the broadcast address to all MAC IDs on the local link.

Table 4 – Fixed tag service definitions

Fixed tag service	Meaning
0x00	moderator
0x01 – 0x08	vendor specific
0x09	ping request
0x0A – 0x14	vendor specific
0x15	tMinus
0x16 – 0x28	vendor specific

Fixed tag service	Meaning
0x29	ping reply
0x2A – 0x3F	vendor specific
0x40 – 0x6F	reserved
0x70 – 0x7F	vendor-specific
0x80	I'm alive
0x81	link parameters
0x82	reserved
0x83	UCMM
0x84	TUI
0x85	IP (reserved for Internet Protocol)
0x86	WAMI
0x87	reserved
0x88	Keeper UCMM
0x89	Ethernet (used for address resolution protocol)
0x8A – 0x8B	reserved
0x8C	Time distribution
0x8D – 0x8F	reserved for time distribution
0x90	debug
0x91 – 0xCF	reserved
0xD0 – 0xEF	group addresses
0xF0 – 0xFF	vendor specific

Fixed tags in the range 0xD0 to 0xEF shall be reserved to permit group screening of connection identifiers.

4.4 Services assumed from the PhL

4.4.1 General requirements

Subclauses 4.4.2 to 4.4.3 include requirements for interface to a Type 2 PhL that are not required for other Types:

- Transmit Machine;
- Receive Machine;
- Serializer;
- Deserializer.

This is necessary because the Ph-Interface Data Units (PhIDUs) that Type 2 passes across the DLL–PhL interface are individual serial data and non-data symbols, designated on the DLE side as M_Symbols (see Table 5), whereas PhIDUs for other Types are not conceptualized in this way. As a consequence, a Type 2 Data Link Layer requires a Type 2 PhL.

4.4.2 Data encoding rules

The M_Symbols transferred at the DLL to PhL interface are of four types designated 0, 1, +, -. They will be encoded inside the PhL into appropriate Ph_Symbols as shown in Table 5. The M_0 and M_1 will be encoded into Ph_Symbols that represent Manchester biphasic L data encoding rules. The M_ND symbols are non-data symbols to allow for the creation of unique data patterns used for start and end delimiters. The example of signal voltage waveform as il

might be applied to an electrically-conductive medium is shown in an idealized form in Figure 6 to provide an example of the data encoding rules shown in Table 5.

Table 5 – Data encoding rules

Data bits (common name)	M_Symbol representation	Ph_Symbol encoding	Manchester encoded
data "zero"	M_0 or {0}	{L,H}	0
data "one"	M_1 or {1}	{H,L}	1
"non_data+"	M_ND+ or {+}	{H,H}	no data
"non_data–"	M_ND– or {–}	{L,L}	no data

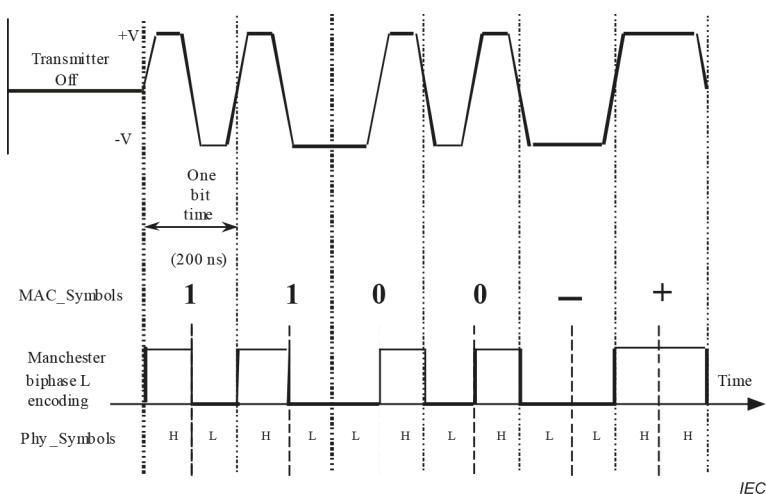


Figure 6 – M_symbols and Manchester encoding at 5 MHz

4.4.3 DLL to PhL interface

4.4.3.1 General

The DLL to PhL interface need not be exposed in the implementation of any PhL variant. This interface may be internal to the node. If conformance to the DLL to PhL interface is claimed, it shall conform to the requirements of 4.4.3.

4.4.3.2 Ph_lock_indication

ph_lock_indication shall provide an indication of either data lock or Ph_Symbol synchronization by the PhL. Valid states for ph_lock_indication shall be true and false. ph_lock_indication shall be true whenever valid Ph_Symbols are present at the PhL interface and the PhL to DLL interface timing of M_Symbols conforms to the PhL requirements. It shall be false between frames (when no Ph_Symbols are present on the medium) or whenever data synchronization is lost or the timing fails to conform to the PhL requirements. ph_lock_indication shall be true prior to the beginning of the start delimiter.

4.4.3.3 Ph_frame_indication

ph_frame_indication shall provide an indication of a valid data frame from the PhL. Valid states for ph_frame_indication shall be true and false. ph_frame_indication shall be true upon ph_lock_indication = true and reception of the first valid start delimiter. ph_frame_indication shall be false at reception of next M_ND symbol (following the start delimiter) or ph_lock_indication = false.

NOTE This signal provides octet synchronization to the DLL.

4.4.3.4 Ph_carrier_indication

ph_carrier_indication shall represent the presence of a signal carrier on the medium. The ph_carrier_indication shall be true if internal PhL identification of signal carrier has been true during any of the last 4 M_symbol times and it shall be false otherwise.

4.4.3.5 Ph_data_indication

ph_data_indication shall represent the M_Symbols that are decoded from the PhL internal representations such as those shown in Table 5. Valid M_symbols shall be M_0 {0}, M_1 {1}, M_ND+ {+}, and M_ND– {-} as shown in Table 6.

Table 6 – M Data symbols

Data bits (common name)	M_Symbol representation
data "zero"	M_0 or {0}
data "one"	M_1 or {1}
"non_data+"	M_ND+ or {+}
"non_data–"	M_ND– or {-}

The ph_data_indication shall represent the M_Symbols as decoded within the PhL whenever ph_lock_indication is true.

4.4.3.6 Ph_status_indication

ph_status_indication shall represent the delimiter status of the frame which was received from the PhL as shown in Table 7. Valid symbols shall be Normal, Abort, and Invalid. ph_status_indication shall indicate Normal after reception of a frame (ph_frame_indication = true) composed of a start delimiter, valid data (no M_ND symbols) and an end delimiter. ph_status_indication shall indicate Abort after reception of a frame (ph_frame_indication = true) composed of a start delimiter, valid data, and a second start delimiter. ph_status_indication shall indicate Invalid after reception of a frame (ph_frame_indication = true) composed of a start delimiter and the detection of any M_ND symbol which was not part of a start or end delimiter.

Table 7 – Truth table for ph_status_indication

Ph_status_indication	Ph_frame_indication	Start delimiters in a single frame	End delimiter detection	Any non-delimiter Manchester violations
Normal	true	1	true	false
Abort	true	2	don't care	false
Invalid	true	1	don't care	true

4.4.3.7 Ph_data_request

ph_data_request shall present the M_Symbols to be transmitted. Valid symbols shall be M_0, M_1, M_ND+ or M_ND– as shown in Table 6. ph_data_request shall indicate M_0 when no data is to be transmitted (and ph_frame_request = false).

4.4.3.8 Ph_frame_request

ph_frame_request shall be true when ph_data_request presents M_Symbols to be encoded into the appropriate Ph_Symbols by the PhL, and shall be false when no valid M_Symbols are to be transferred to the PhL.