

Digital data communications for measurement and control — Fieldbus for use in industrial control systems —

Part 5: Application layer service definition

The European Standard EN 61158-5:2004 has the status of a
British Standard

ICS 25.040; 35.100.70; 35.240.50

National foreword

This British Standard is the official English language version of EN 61158-5:2004. It is identical with IEC 61158-5:2003. It, together with BS EN 61158-2:2004, BS EN 61158-3:2004, BS EN 61158-4:2004, BS EN 61158-6:2004 and BS EN 61784-1:2004, supersedes BS EN 50170:1997 and BS EN 50254:1999, which are withdrawn.

The UK participation in its preparation was entrusted to Technical Committee AMT/7, Industrial communications: Process measurement and control, including Fieldbus, which has the responsibility to:

- aid enquirers to understand the text;
- present to the responsible international/European committee any enquiries on the interpretation, or proposals for change, and keep the UK interests informed;
- monitor related international and European developments and promulgate them in the UK.

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

Cross-references

The British Standards which implement international or European publications referred to in this document may be found in the *BSI Catalogue* under the section entitled “International Standards Correspondence Index”, or by using the “Search” facility of the *BSI Electronic Catalogue* or of British Standards Online.

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

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, the EN title page, pages 2 to 1295 and a back cover.

The BSI copyright notice displayed in this document indicates when the document was last issued.

Amendments issued since publication

Amd. No.	Date	Comments

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 August 2004

© BSI 31 August 2004

ISBN 0 580 44290 X

EUROPEAN STANDARD

EN 61158-5

NORME EUROPÉENNE

EUROPÄISCHE NORM

July 2004

ICS 25.040; 35.100.70; 35.240.50

Partially supersedes EN 50170:1996 + A1:2002 + A2:1999 + A3:2002 and EN 50254:1998

English version

**Digital data communication for measurement and control -
Fieldbus for use in industrial control systems
Part 5: Application layer service definition
(IEC 61158-5:2003 + corrigendum 2004)**

Communications numériques pour les
systèmes de mesure et de commande -
Bus de terrain utilisés dans les systèmes
de commande industriels
Partie 5: Définition des services
de la couche d'application
(CEI 61158-5:2003 + corrigendum 2004)

Digitale Datenkommunikation
in der Leittechnik -
Feldbus für industrielle Leitsysteme
Teil 5: Definition der Dienste der
Anwendungsschicht (Application Layer)
(IEC 61158-5:2003 + Corrigendum 2004)

This European Standard was approved by CENELEC on 2004-03-16. 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 Central Secretariat or to any CENELEC member.

This European Standard exists in one official version (English). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the Central Secretariat has the same status as the official versions.

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

CENELEC

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

Central Secretariat: rue de Stassart 35, B - 1050 Brussels

© 2004 CENELEC - All rights of exploitation in any form and by any means reserved worldwide for CENELEC members.

Ref. No. EN 61158-5:2004 E



Documents
Document Sharing Hub

Foreword

The text of the International Standard IEC 61158-5:2003, prepared by SC 65C, Digital communications, of IEC TC 65, Industrial-process measurement and control, was submitted to the Unique Acceptance Procedure and was approved by CENELEC as EN 61158-5 on 2004-03-16 with inclusion of the accepted future IEC corrigendum.

This European Standard, together with EN 61158-2:2004 to EN 61158-4:2004, EN 61158-6:2004 and EN 61784-1:2004, supersedes EN 50170:1996 + A1:2002 (+ corr. Aug. 2002) + A2:1999 + A3:2002 (+ corr. Aug. 2002) and EN 50254:1998.

The following dates were fixed:

- latest date by which the EN has to be implemented
at national level by publication of an identical
national standard or by endorsement (dop) 2005-04-01
- latest date by which the national standards conflicting
with the EN have to be withdrawn (dow) 2007-04-01

Annex ZA has been added by CENELEC.

Endorsement notice

The text of the International Standard IEC 61158-5:2003 and its corrigendum July 2004 was approved by CENELEC as a European Standard without any modification.

CONTENTS

0	Introduction	25
0.1	General	25
0.2	Nomenclature for references within this standard	25
1	Scope	26
2	Normative references	26
3	Terms and definitions	27
3.1	ISO/IEC 7498-1 terms	27
3.2	ISO/IEC 8822 terms	28
3.3	ISO/IEC 9545 terms	28
3.4	ISO/IEC 8824 terms	28
3.5	Fieldbus Data Link Layer terms	28
3.6	Fieldbus Application Layer specific definitions	29
3.7	Abbreviations and symbols	40
3.8	Conventions	41
4	Concepts	44
4.1	Overview	44
4.2	Architectural relationships	45
4.3	Fieldbus Application Layer structure	47
4.4	Fieldbus Application Layer naming and addressing	59
4.5	Architecture summary	59
4.6	FAL service procedures	60
4.7	Common FAL attributes	61
4.8	Common FAL service parameters	61
4.9	APDU size	62
5	Data type ASE	62
5.1	General	62
5.2	Formal definition of data type objects	65
5.3	FAL defined data types	66
5.4	Data type ASE service specification	101
6	Type 1 communication model specification	101
6.1	Concepts	101
6.2	ASEs	101
6.3	ARs	245
6.4	Summary of FAL classes	269
6.5	Permitted FAL services by AREP role	270
7	Type 2 communication model specification	272
7.1	Concepts	272
7.2	ASEs	280
7.3	ARs	345
7.4	Summary of FAL classes	381
7.5	Permitted FAL services by AR type	382
8	Type 3 communication model specification	382
8.1	DP concepts	382

8.2	ASEs	400
8.3	Summary of AL classes	733
8.4	Permitted AL services by AREP role	734
8.5	Conformance classes	738
8.6	Application characteristics	738
9	Type 4 communication model specification	739
9.1	Concepts	739
9.2	Variable ASE	746
9.3	Application relationship ASE	765
10	Type 5 communication model specification	771
10.1	Concepts	771
10.2	ASEs	792
10.3	FDA sessions	827
10.4	Summary of FAL Type 9 and Type 5 classes	836
10.5	Permitted FAL Type 9 and Type 5 services by AREP role	837
11	Type 6 communication model specification	839
11.1	Application relationship (AR) and Application relationship endpoint (AREP) characteristics	839
11.2	Relationship of Type 6 to Type 1	841
11.3	Permitted services by AREP role	845
12	Type 7 communication model specification	847
12.1	Concepts	847
12.2	ASEs	863
12.3	ARs	1032
13	Type 8 communication model specification	1052
13.1	Concepts	1052
13.2	ASEs	1054
13.3	Application relationships	1069
13.4	Permitted FAL services by AREP role	1070
14	Type 9 communication model specification	1071
14.1	Concepts	1071
14.2	Common parameters	1071
14.3	ASEs	1072
14.4	ARs	1162
14.5	Summary of classes	1165
14.6	Permitted services by AREP role	1165
15	Type 10 communication model specification	1167
15.1	Type 10 Concepts	1167
15.2	ASE Data Types	1173
15.3	ASEs	1174
15.4	ARs	1288
15.5	Summary of FAL classes	1291
15.6	Summary of FAL services	1291
Annex ZA	(normative) Normative references to international publications with their corresponding European publications	1294

Figure 1 – Relationship of IEC 61158-3 to other fieldbus layers and to users of the fieldbus Application Service	25
Figure 2 – Relationship to the OSI basic reference model	45
Figure 3 – Architectural positioning of the fieldbus Application Layer	46
Figure 4 – Client/server interactions	48
Figure 5 – Pull model interactions	49
Figure 6 – Push model interactions	50
Figure 7 – APOs services conveyed by the FAL	51
Figure 8 – Application entity structure	53
Figure 9 – Example FAL ASEs	54
Figure 10 – FAL management of objects	55
Figure 11 – ASE service conveyance	56
Figure 12 – Defined and established AREPs	58
Figure 13 – FAL architectural components	60
Figure 14 – Data type class hierarchy example	63
Figure 15 – The AR ASE conveys APDUs between APs	132
Figure 16 – 1-to-1 AR establishment	144
Figure 17 – 1-to-many AR establishment	144
Figure 18 – Event model overview	184
Figure 19 – Residence timeliness	260
Figure 20 – Synchronized timeliness	261
Figure 21 – Residence timeliness	266
Figure 22 – Synchronized timeliness	267
Figure 23 – Overview of ASE's and object classes	274
Figure 24 – Addressing format using MAC, class, instance and attribute IDs	274
Figure 25 – Static Assembly state transition diagram	292
Figure 26 – Dynamic Assembly state transition diagram	293
Figure 27 – Example of Find_Next_Object_Instance service	322
Figure 28 – Context of transport services within the connection model	348
Figure 29 – Application-to-application view of data transfer	348
Figure 30 – Data flow diagram for a link producer	349
Figure 31 – Data flow diagram for a link consumer	350
Figure 32 – Triggers	351
Figure 33 – Binding transport instances to the producer and consumer of a transport connection that does not have a reverse data path	352
Figure 34 – Binding transport instances to the producers and consumers of a transport connection that does have a reverse data path	352
Figure 35 – Binding transport instances to the producer and consumers of a multipoint connection when the transport connection does not have a reverse data path	353
Figure 36 – Binding transport instances to the producers and consumers of a multipoint connection when the transport connection does have reverse data paths	353
Figure 37 – Example of DP communication with a single controlling device	385
Figure 38 – Example of DP communication with several controlling devices	385
Figure 39 – Example of DP communication between field devices	385
Figure 40 – DP-slave model (modular DP-slave)	388
Figure 41 – DP-slave model (compact DP-slave)	388
Figure 42 – Overview of application processes	389

Figure 43 – DP-slave model (modular DP-slave).....	390
Figure 44 – Application Service Elements (ASEs)	392
Figure 45 – Application Process with application Objects (APOs).....	393
Figure 46 – Access to a remote APO	394
Figure 47 – Access to a remote APO for publisher/subscriber association	395
Figure 48 – Example of one AR with two AREPs	396
Figure 49 – Relation of a simple process data object to the real object	401
Figure 50 – Relation of a combined process data object to the real objects.....	404
Figure 51 – Sequence of an isochronous DP cycle with one DP-master (class 1).....	432
Figure 52 – Additional time relationships in a DP system operating in isochronous mode....	433
Figure 53 – DP system with optimised isochronous DP cycle	435
Figure 54 – Buffered synchronised isochronous mode at the DP-master (class 1).....	436
Figure 55 – Enhanced synchronised isochronous mode at the DP-master (class 1)	437
Figure 56 – Input, output and PLL state machine interaction	437
Figure 57 – PLL state diagram	443
Figure 58 – OUTPUT state diagram	447
Figure 59 – INPUT state diagram	451
Figure 60 – Treatment of an alarm in the DP system.....	480
Figure 61 – Load Region state diagram for erasable memory.....	567
Figure 62 – Load region state diagram for non erasable memory	567
Figure 63 – Function invocation state diagram	594
Figure 64 – System architecture	604
Figure 65 – Assignment of communication relationship to application relationship	611
Figure 66 – MS0 application relationship	617
Figure 67 – Output buffer model of a DP-slave without sync functionality.....	618
Figure 68 – Output buffer model of a DP-slave with sync functionality.....	618
Figure 69 – Input buffer model of a DP-slave without freeze functionality.....	619
Figure 70 – Input buffer model of a DP-slave with freeze functionality.....	619
Figure 71 – MS1 application relationship	620
Figure 72 – MS2 application relationship	621
Figure 73 – Example of inter-network communication	622
Figure 74 – Example without inter-network addressing.....	622
Figure 75 – First example with inter-network addressing	623
Figure 76 – Second example with inter-network addressing	623
Figure 77 – MS3 application relationship	624
Figure 78 – MM1 application relationship	625
Figure 79 – MM2 application relationship	626
Figure 80 – Cycle time of the DP system	739
Figure 81 – FAL AE	740
Figure 82 – Summary of the FAL architecture	743
Figure 83 – FAL service procedure overview.....	744
Figure 84 – Time sequence diagram for the confirmed services	745
Figure 85 – Time sequence diagram for unconfirmed services	746
Figure 86 – VCR initiation	777
Figure 87 – Misordered message handling.....	783
Figure 88 – FF SM port message processing order	784

Figure 89 – FF FDA port message processing order	784
Figure 90 – FF TCP connection message processing order	785
Figure 91 – Session endpoint message processing order.....	785
Figure 92 – FDA LAN redundancy port message processing order	785
Figure 93 – Message processing by receiving entity	786
Figure 94 – Organisation of the ASEs and ARs	847
Figure 95 – Object model of the MPS ASE	867
Figure 96 – Time-out evaluation net.....	880
Figure 97 – Asynchronous promptness status evaluation net	884
Figure 98 – Synchronous promptness status evaluation net	885
Figure 99 – Punctual promptness status evaluation net	887
Figure 100 – Asynchronous refreshment status evaluation net.....	890
Figure 101 – Synchronous refreshment status evaluation net.....	891
Figure 102 – Punctual refreshment status evaluation net	893
Figure 103 – A_Readloc service procedure.....	896
Figure 104 – A_Writeloc service procedure.....	897
Figure 105 – A_Update service procedure	899
Figure 106 – A_Readfar service procedure	901
Figure 107 – A_Writefar service procedure	903
Figure 108 – A_Sent service procedure	904
Figure 109 – A_Received service procedure	905
Figure 110 – A_Read service procedure	907
Figure 111 – A_Read service state machine	908
Figure 112 – A_Write service procedure	909
Figure 113 – A_Write service state machine	910
Figure 114 – Model of a resynchronised variable	913
Figure 115 – Principles for resynchronisation of a produced variable	914
Figure 116 – Resynchronisation mechanism state machine for a produced variable.....	916
Figure 117 – Asynchronous refreshment private mechanism evaluation net.....	917
Figure 118 – Asynchronous refreshment public mechanism evaluation net	918
Figure 119 – Synchronous refreshment private mechanism evaluation net.....	919
Figure 120 – Synchronous refreshment public mechanism evaluation net	920
Figure 121 – Punctual refreshment private mechanism evaluation net	921
Figure 122 – Punctual refreshment public mechanism evaluation net.....	922
Figure 123 – Principles for the resynchronisation of a consumed variable.....	923
Figure 124 – Resynchronisation mechanism state machine for consumed variable	925
Figure 125 – Asynchronous promptness public mechanism evaluation net.....	926
Figure 126 – Asynchronous promptness private mechanism evaluation net	927
Figure 127 – Synchronous promptness public mechanism evaluation net.....	928
Figure 128 – Synchronous promptness private mechanism evaluation net	930
Figure 129 – Punctual promptness public mechanism evaluation net	931
Figure 130 – Punctual promptness private mechanism evaluation net.....	932
Figure 131 – Spatial consistency list variables interchange mechanism	935
Figure 132 – Spatial consistency – consistency variable interchange mechanism	935
Figure 133 – Spatial consistency – list recovery mechanism	936
Figure 134 – Spatial consistency – validity of the spatial consistency status	

Figure 135 – Object model of a variable list	937
Figure 136 – A_Readlist service procedure	942
Figure 137 – Consistency variable value evaluation net	948
Figure 138 – Consistency interchange timing diagram	949
Figure 139 – Recovery mechanism evaluation net	950
Figure 140 – Recovery interchange timing diagram	950
Figure 141 – Flowchart of the sub-MMS environment management state	957
Figure 142 – Domain management state chart	987
Figure 143 – Domain upload flowchart	990
Figure 144 – Domain download sequence diagram	991
Figure 145 – Domain upload sequence diagram	991
Figure 146 – Program invocation state chart	1003
Figure 147 – A_Associate service procedure	1041
Figure 148 – A_Release service procedure	1044
Figure 149 – A_Abort service procedure	1045
Figure 150 – A_Data service procedure	1047
Figure 151 – A_Unidata service procedure	1050
Figure 152 – Associated mode service state chart	1051
Figure 153 – Non-associated mode service state chart	1052
Figure 154 – Architectural positioning of the fieldbus Application Layer	1052
Figure 155 – VFD model	1072
Figure 156 – Abstract model of an automation system (VFD)	1072
Figure 157 – Source OD/remote OD	1079
Figure 158 – Put OD state machine	1092
Figure 159 – Transaction object state machine	1099
Figure 160 – Context test of two features-supported with different bitstring length	1106
Figure 161 – Overview of event	1126
Figure 162 – Event state machine	1132
Figure 163 – Domain GenericDownload/Download state machine (server)	1146
Figure 164 – Domain Upload state machine (server)	1148
Figure 165 – State diagram	1160
Figure 166 – FAL ASEs Communication Architecture	1169
Figure 167 – Runtime object model	1170
Figure 168 – Relationship between Engineering and Runtime	1171
Figure 169 – Navigation in the runtime object model	1172
Figure 170 – Operating state block diagram	1197
Figure 171 – Device status model for the common diagnosis	1198
Figure 172 – ACCO ASE structure	1222
Figure 173 – Productive operation of data connections	1223
Figure 174 – Productive operation of event connections	1224
Figure 175 – Quality Code transfer – standard behavior	1232
Figure 176 – Quality Code with communication fault	1233
Figure 177 – Quality Code when an connection is cleared	1234
Figure 178 – Quality Code when an connection is deactivated	1234
Figure 179 – Quality Code during the transfer of "incorrect" connection data	1235
Figure 180 – Quality Code for provider in "Ready" state	

Figure 181 — Quality Code when clearing an object from the provider.....	1236
Figure 182 — Quality Code when a connection is forced.....	1236
Figure 183 — Quality Code at QoS violation	1237
Figure 184 — Quality Code for initial value	1237
Figure 185 — Productive operation of data connections.....	1241
Figure 186 — Productive operation of event connections	1242
Figure 187 — Failure of the provider in productive operation	1245
Figure 188 — Failure of the consumer	1246
Figure 189 — Failure of the provider when changing the connection status	1247
Figure 190 — Failure of the provider when changing the interconnection status	1248
Figure 191 — Failure of the provider when clearing connections.....	1249
Figure 192 — Information levels	1249
Figure 193 — ACCO ASE status model for the common diagnosis.....	1250
Figure 194 — ACCO ASE status model for the detailed diagnosis	1250
Figure 195 — Structure of the transmitted connection data	1273
Table 1 — PERSISTDEF	71
Table 2 — VARTYPE	72
Table 3 — ITEMQUALITYDEF.....	73
Table 4 — STATEDEF	76
Table 5 — GROUPEXCEPTIONDEF.....	76
Table 6 — ACCESSRIGHTSDEF.....	77
Table 7 — HRESULT	77
Table 8 — UUID.....	84
Table 9 — Data type names for value.....	99
Table 10 — UUID	101
Table 11 — Create service parameters	102
Table 12 — Delete service parameters.....	103
Table 13 — Get Attributes service parameters	105
Table 14 — Set Attributes service parameters.....	107
Table 15 — Begin Set Attributes	109
Table 16 — End Set Attributes	110
Table 17 — Subscribe service parameters	119
Table 18 — Identify	122
Table 19 — Get status	123
Table 20 — Status notification.....	124
Table 21 — Initiate.....	125
Table 22 — Terminate.....	128
Table 23 — Conclude.....	130
Table 24 — Reject	130
Table 25 — Conveyance of service primitives by AREP role.....	133
Table 26 — Valid combinations of AREP roles involved in an AR	133
Table 27 — AR-Unconfirmed Send.....	139
Table 28 — AR-Confirmed Send.....	141
Table 29 — AR-Establish service	143
Table 30 — Valid combinations of AREP classes to be related	

Table 31 – AR-DeEstablish service	146
Table 32 – AR-Abort	147
Table 33 – AR-Compel service.....	148
Table 34 – AR-Get Buffered Message service.....	149
Table 35 – AR-Schedule Communication service	150
Table 36 – AR-Cancel Scheduled Sequence service	151
Table 37 – AR-Status.....	152
Table 38 – AR-XON-OFF	153
Table 39 – AR-Remote Read service	154
Table 40 – AR-Remote Write service	155
Table 41 – Read service parameters.....	164
Table 42 – Read List service parameters	167
Table 43 – Write service parameters.....	169
Table 44 – Write List service parameters	171
Table 45 – Information Report service	173
Table 46 – Information Report List service	175
Table 47 – Exchange service parameters	177
Table 48 – Exchange List service parameters	181
Table 49 – Acknowledge Event	193
Table 50 – Acknowledge Event List service parameters	194
Table 51 – Enable event	196
Table 52 – Event Notification service parameters.....	197
Table 53 – Enable Event List	199
Table 54 – Notification recovery service parameters	200
Table 55 – Get Event Summary service parameters.....	201
Table 56 – Get Event Summary List service parameters	203
Table 57 – Query Event Summary List service parameters	207
Table 58 – Initiate Load service parameters.....	215
Table 59 – Terminate Load service parameters.....	217
Table 60 – Push Segment service parameters	218
Table 61 – Pull Segment service parameters	219
Table 62 – Discard service parameters	220
Table 63 – Pull Upload Sequencing of service primitives	222
Table 64 – Pull Upload service parameter constraints.....	222
Table 65 – Pull Upload state table	223
Table 66 – Pull Download Sequencing of service primitives	224
Table 67 – Pull Download service parameter constraints	224
Table 68 – Pull Download state table	225
Table 69 – Push Download sequencing of service primitives.....	226
Table 70 – Push Download service parameter constraints	227
Table 71 – Push Download state table	227
Table 72 – Start service parameters	234
Table 73 – Stop service parameters.....	235
Table 74 – Resume service parameters	236
Table 75 – Reset service parameters.....	237
Table 76 – Kill service parameters	

Table 77 – Action Invoke service parameters	239
Table 78 – Action Return service parameters	240
Table 79 – State transitions for a function invocation object	242
Table 80 – FAL class summary	270
Table 81 – Services by AREP role	270
Table 82 – Common elements	277
Table 83 – ST language elements	278
Table 84 – Type conversion operations	278
Table 85 – Values of implementation-dependent parameters	279
Table 86 – Extensions to IEC 61131-3	280
Table 87 – Static Assembly state event matrix	292
Table 88 – Dynamic Assembly state event matrix	293
Table 89 – Message Router object Forward_Open parameters	296
Table 90 – Status codes	298
Table 91 – Format of extended status	300
Table 92 – Get_Attribute_All service parameters	301
Table 93 – Set_Attribute_All service parameters	302
Table 94 – Get_Attribute_List service parameters	304
Table 95 – Set_Attribute_List service parameters	306
Table 96 – Reset service parameters	308
Table 97 – Start service parameters	310
Table 98 – Stop service parameters	312
Table 99 – Create service parameters	314
Table 100 – Delete service parameters	316
Table 101 – Get_Attribute_Single service parameters	317
Table 102 – Set_Attribute_Single service parameters	319
Table 103 – Find_Next_Object_Instance service parameters	321
Table 104 – NOP service parameters	323
Table 105 – Apply_Attributes service parameters	324
Table 106 – Save service parameters	326
Table 107 – Restore service parameters	328
Table 108 – CM_Open service parameters	336
Table 109 – CM_Close service parameters	338
Table 110 – CM_Unconnected_Send service parameters	339
Table 111 – CM_Get_Connection_Data service parameters	341
Table 112 – CM_Search_Connection_Data service parameters	343
Table 113 – CM_Get_Connection_Data service parameters	344
Table 114 – How production trigger, transport class, and CM_RPI determine when data is produced	347
Table 115 – Transport classes	357
Table 116 – UCMM_Create service parameters	374
Table 117 – UCMM_Delete service parameters	375
Table 118 – UCMM_Write service parameters	376
Table 119 – UCMM_Abort service parameters	378
Table 120 – TR_Write service parameters	378
Table 121 – TR_Trigger service parameters	379

Table 122 – TR_Packet_arrived service parameters	379
Table 123 – TR_Ack_received service parameters	380
Table 124 – TR_Verify service parameters	380
Table 125 – TR_Status_updated service parameters	381
Table 126 – FAL class summary	381
Table 127 – FAL services by AR type	382
Table 128 – Requirements and features of fieldbus DP	384
Table 129 – Status values of the service primitives	400
Table 130 – Access Rights MS1	402
Table 131 – Access Rights MS2	403
Table 132 – Access Rights MS1	405
Table 133 – Access Rights MS2	405
Table 134 – SCL matching rules	406
Table 135 – Read	407
Table 136 – Write	408
Table 137 – Data transport	409
Table 138 – Format (simple input data description)	413
Table 139 – Consistency (simple input data description)	413
Table 140 – Format (simple output data)	415
Table 141 – Consistency (simple output data)	415
Table 142 – Format (extended input data)	416
Table 143 – Consistency (extended input data)	416
Table 144 – Format (extended output data)	418
Table 145 – Consistency (extended output data)	418
Table 146 – Set Input	419
Table 147 – Read Input	420
Table 148 – Get Input	421
Table 149 – New Input	421
Table 150 – Set Output	423
Table 151 – Final	423
Table 152 – Read Output	424
Table 153 – Get Output	425
Table 154 – Clear Flag	425
Table 155 – New Flag	425
Table 156 – New Output	425
Table 157 – Clear Flag	426
Table 158 – Global Control	426
Table 159 – Clear Command	427
Table 160 – Sync Command	427
Table 161 – Freeze Command	427
Table 162 – New publisher data	427
Table 163 – Get publisher data	428
Table 164 – New Flag	428
Table 165 – SYNCH	429
Table 166 – SYNCH Delayed	429
Table 167 – DX Finished	

Table 168 – SYNCH Event.....	430
Table 169 – Status.....	430
Table 170 – Primitives issued by the AL to the PLL state machine	439
Table 171 – Primitives issued by the user to the PLL state machine	439
Table 172 – Allowed values of status	439
Table 173 – Primitives issued by the user to the input state machine	439
Table 174 – Primitives issued by the user to the output state machine	440
Table 175 – Primitives issued by the PLL to the output state machine	440
Table 176 – Primitives issued by the output to the PLL state machine	440
Table 177 – Primitives issued by the PLL to the input state machine.....	440
Table 178 – Primitives issued by the output to the input state machine	440
Table 179 – Primitives issued by the output state machine to the AL	441
Table 180 – Primitives issued by the AL to the output state machine	441
Table 181 – Primitives issued by the input state machine to the AL	441
Table 182 – Primitives issued by the AL to the input state machine	441
Table 183 – PLL state table	444
Table 184 – OUTPUT state table	448
Table 185 – INPUT state table	452
Table 186 – Identifier status.....	454
Table 187 – Channel type	455
Table 188 – IO type	455
Table 189 – Error type	456
Table 190 – Status type	456
Table 191 – Status specifier	457
Table 192 – Status specifier	458
Table 193 – Module status	458
Table 194 – Status specifier	459
Table 195 – Link status	459
Table 196 – Link error.....	460
Table 197 – Set Slave Diag	461
Table 198 – Ext Diag Flag.....	461
Table 199 – Get Slave Diag	463
Table 200 – Read Slave Diag	472
Table 201 – New Slave Diag.....	480
Table 202 – Alarm type	482
Table 203 – Add Ack.....	482
Table 204 – Alarm specifier	482
Table 205 – Alarm notification.....	483
Table 206 – Alarm Ack.....	484
Table 207 – Prm data type.....	489
Table 208 – Supported feature.....	498
Table 209 – Supported profile feature	498
Table 210 – Role	499
Table 211 – Check user Prm.....	501
Table 212 – Prm structure.....	502
Table 213 – MS1 Command.....	

Table 214 – Check user Prm result	506
Table 215 – Status values.....	506
Table 216 – Check Ext user Prm.....	507
Table 217 – Check Ext user Prm result	510
Table 218 – Status values.....	510
Table 219 – Check Cfg	511
Table 220 – Check Cfg result.....	512
Table 221 – Status values.....	512
Table 222 – Set Cfg	513
Table 223 – Get Cfg	513
Table 224 – Set Slave Add	514
Table 225 – Initiate	515
Table 226 – Abort	518
Table 227 – Instance	518
Table 228 – MS0 init DP-slave.....	519
Table 229 – MS1 init DP-slave.....	519
Table 230 – MS2 init DP-slave.....	519
Table 231 – DP-slave started.....	520
Table 232 – Alarm limit	520
Table 233 – DP-slave stopped	521
Table 234 – Reset DP-slave	521
Table 235 – DP-slave fault.....	521
Table 236 – Application ready DP-slave.....	521
Table 237 – Start subscriber	522
Table 238 – Stop subscriber	522
Table 239 – Publisher active.....	523
Table 240 – Status.....	523
Table 241 – Init DP-master CI1	524
Table 242 – DP-master CI1 started	525
Table 243 – Alarm limit	525
Table 244 – DP-master CI1 stopped	525
Table 245 – Reset DP-master CI1	526
Table 246 – DP-master CI1 fault	526
Table 247 – DP-master CI1 reject	526
Table 248 – Set mode DP-master CI1	527
Table 249 – DP-master CI1 mode changed	528
Table 250 – Load bus Par DP-master CI1	528
Table 251 – Mark DP-master CI1	529
Table 252 – Abort DP-master CI1	530
Table 253 – Read value DP-master CI1	530
Table 254 – Delete SC DP-master CI1	530
Table 255 – DP-master CI1 event	531
Table 256 – Init DP-master CI2.....	531
Table 257 – Reset DP-master CI2.....	532
Table 258 – DP-master CI2 fault	533
Table 259 – DP-master CI2 reject	

Table 260 – DP-master CI2 closed	533
Table 261 – DP-master CI2 event	534
Table 262 – USIF state	535
Table 263 – Data rate	538
Table 264 – USIF state	539
Table 265 – Isochronous mode	539
Table 266 – Slave type	542
Table 267 – Alarm mode	542
Table 268 – Get Master Diag	545
Table 269 – MDiag identifier	545
Table 270 – Start Seq	546
Table 271 – Area code (start seq)	547
Table 272 – Download	548
Table 273 – Upload	549
Table 274 – End Seq	550
Table 275 – Act Para Brct	550
Table 276 – Area code (Act Para Brct)	551
Table 277 – Act param	551
Table 278 – Area code (Act param)	552
Table 279 – Activate	552
Table 280 – Access rights MS1	554
Table 281 – Access rights MS2	555
Table 282 – Load region state	555
Table 283 – Initiate load	557
Table 284 – Default values for the parameter Intersegment Request Timeout	557
Table 285 – Push segment	558
Table 286 – Pull segment	560
Table 287 – Terminate load	561
Table 288 – Primitives issued by the user to the Load Region state machine	563
Table 289 – Primitives issued by the Load Region state machine to the user	563
Table 290 – Primitives issued by the Function Invocation to the Load Region state machine	564
Table 291 – Primitives issued by the Load Region to the Function Invocation state machine	564
Table 292 – Load Region state definitions	564
Table 293 – Load Region function table	565
Table 294 – Load Region state table for erasable memory	568
Table 295 – Load Region state table for non erasable memory	577
Table 296 – Access rights MS1	581
Table 297 – Access rights MS2	581
Table 298 – Function Invocation state	582
Table 299 – Load Region object in use	582
Table 300 – Access rights MS1	583
Table 301 – Access rights MS2	583
Table 302 – Load Region object in use	584
Table 303 – Start	585

Table 304 – Stop	586
Table 305 – Resume	586
Table 306 – Reset	587
Table 307 – Get FI state	588
Table 308 – Call	589
Table 309 – Primitives issued by the user to the Function Invocation state machine	591
Table 310 – Primitives issued by the Function Invocation state machine to the user	591
Table 311 – Primitives issued by the Load Region to the Function Invocation state machine	592
Table 312 – Primitives issued by the Function Invocation to the Load Region state machine	592
Table 313 – Function Invocation state definitions	592
Table 314 – Function definitions	593
Table 315 – Function Invocation state table	594
Table 316 – CS status	606
Table 317 – Summertime	606
Table 318 – Synchronisation active	607
Table 319 – Announcement hour	608
Table 320 – Summertime	608
Table 321 – Accuracy	608
Table 322 – Set time	609
Table 323 – Sync interval violation	610
Table 324 – Parameter of Initiate service without inter-network addressing	623
Table 325 – Parameter of Initiate service with inter-network addressing (first example)	623
Table 326 – Parameter of Initiate service with inter-network addressing (second example)	624
Table 327 – AR type	629
Table 328 – Sync supported	631
Table 329 – Freeze supported	631
Table 330 – Group identifier	633
Table 331 – DPV1 enabled	633
Table 332 – Fail safe	634
Table 333 – WD base	634
Table 334 – No Add change	636
Table 335 – Alarm mode supported	638
Table 336 – Isochronous mode supp	642
Table 337 – Isochronous mode	642
Table 338 – Alarm mode	643
Table 339 – Time device type	644
Table 340 – S_SAP_index	647
Table 341 – D_addr	647
Table 342 – Service_activate	648
Table 343 – Role_in_service	648
Table 344 – Indication_mode	649
Table 345 – Max_DLSDU_length_req_low	649
Table 346 – Max_DLSDU_length_req_high	650
Table 347 – Max_DLSDU_length_ind_low	650

Table 348 – Max_DLSDU_length_ind_high	651
Table 349 – S_SAP_index	655
Table 350 – D_SAP_index	656
Table 351 – D_addr	656
Table 352 – Service_activate	657
Table 353 – Role_in_service	657
Table 354 – Indication_mode	658
Table 355 – Max_DLSDU_length_req_low	658
Table 356 – Max_DLSDU_length_req_high	658
Table 357 – Max_DLSDU_length_ind_low	659
Table 358 – Max_DLSDU_length_ind_high	659
Table 359 – Sync	660
Table 360 – Freeze	660
Table 361 – DPV1 enabled	662
Table 362 – Fail safe	662
Table 363 – Enable publisher	662
Table 364 – WD base	662
Table 365 – Alarm mode	664
Table 366 – Fail safe	672
Table 367 – S_SAP_index	679
Table 368 – D_SAP_index	679
Table 369 – D_addr	680
Table 370 – Service_activate	680
Table 371 – Role_in_service	680
Table 372 – Max_DLSDU_length_req_low	680
Table 373 – Max_DLSDU_length_req_high	681
Table 374 – Max_DLSDU_length_ind_low	681
Table 375 – Max_DLSDU_length_ind_high	681
Table 376 – DLL init DP-slave	682
Table 377 – Load ARL DP-slave	682
Table 378 – Get ARL DP-slave	688
Table 379 – Set ARL isochronous mode	693
Table 380 – Load ARL DP-master CI1	694
Table 381 – Get ARL DP-master CI1	696
Table 382 – ARL Slave update DP-master CI1	698
Table 383 – Load ARL DP-master CI2	699
Table 384 – Get ARL DP-master CI2	700
Table 385 – Load CRL DP-slave	701
Table 386 – Load CRL DXB link entries	702
Table 387 – Get CRL DP-slave	703
Table 388 – Load CRL DP-master CI1	705
Table 389 – Get CRL DP-master CI1	716
Table 390 – CRL Slave activate	727
Table 391 – CRL Slave new Prm	728
Table 392 – CRL Slave new Prm data	729
Table 393 – Load CRL DP-master CI2	729

Table 394 – Get CRL DP-master CI2	732
Table 395 – Fieldbus AL class summary	734
Table 396 – Assignment of the services to DP-masters and DP-slaves	735
Table 397 – Support of AR types in the different DP-device types.....	736
Table 398 – Support of services at the different AREPs respectively CREPs	736
Table 399 – Conformance classes DP-master (class 1)	738
Table 400 – Conformance classes DP-master (class 2)	738
Table 401 – REQUEST service parameters	761
Table 402 – RESPONSE service parameters	762
Table 403 – Error codes by source	763
Table 404 – Reserve REP service parameters	763
Table 405 – Free AREP service parameters	764
Table 406 – Get REP attribute service parameters.....	764
Table 407 – Set REP attribute service parameters	765
Table 408 – AR Send service parameters	768
Table 409 – AR Acknowledge service parameters.....	769
Table 410 – AR Get Attributes service parameters.....	769
Table 411 – AR Set Attributes service parameters	770
Table 412 – Scope of Invoke Id.....	781
Table 413 – Types of misordering detectable by message numbers.....	782
Table 414 – Delivery of misordered message types on publisher/subscriber VCRs	782
Table 415 – Statistics gathered per VCR	782
Table 416 – Determination of misordering type at a subscriber VCR	783
Table 417 – Mapping of received messages to primitives.....	783
Table 418 – Mapping of received primitives to messages.....	784
Table 419 – Defined network addresses	787
Table 420 – Use of network addresses	787
Table 421 – Use of endpoint selectors in server VCRs.....	788
Table 422 – Use of endpoint selectors in publisher VCRs	788
Table 423 – Use of endpoint selectors in source VCRs	788
Table 424 – Network address and port numbers for device annunciation	790
Table 425 – Network address and port numbers for set/clear assignment info and clear address	790
Table 426 – Network address and port numbers for SM identify	790
Table 427 – Network address and port numbers for SM find tag	791
Table 428 – Network address and port numbers for clients and servers (part 1).....	791
Table 429 – Network address and port numbers for clients and servers (part 2).....	791
Table 430 – Network address and port numbers for publishers and subscribers.....	791
Table 431 – Network address and port numbers for report distribution.....	791
Table 432 – Network address and port numbers for LAN redundancy get and put information.....	791
Table 433 – Network address and port numbers for LAN redundancy diagnostics	791
Table 434 – VCR types	793
Table 435 – Use of VCR user id.....	794
Table 436 – Use of FDA address	794
Table 437 – Initiate	796

Table 438 – Find tag query service parameters.....	801
Table 439 – SMK IDs	802
Table 440 – Find tag reply service parameters.....	803
Table 441 – Identify service parameters	806
Table 442 – Annunciate service parameters.....	809
Table 443 – Set assignment info service parameters	811
Table 444 – Clear assignment info service parameters	814
Table 445 – Clear address service parameters	816
Table 446 – Diagnostic message service	821
Table 447 – Get redundancy info service	822
Table 448 – Put redundancy info service	824
Table 449 – Get redundancy statistics service	826
Table 450 – Open session service	833
Table 451 – Idle session service	836
Table 452 – FAL class summary	837
Table 453 – Services by AREP role	838
Table 454 – Services by AREP role	846
Table 455 – Access protection.....	862
Table 456 – Binary time coding.....	878
Table 457 – Asynchronous promptness events and actions	884
Table 458 – Synchronous promptness events and actions	885
Table 459 – Punctual promptness events and actions	887
Table 460 – Asynchronous refreshment events and actions	890
Table 461 – Synchronous refreshment events and actions.....	891
Table 462 – Punctual refreshment events and actions	894
Table 463 – A_Readloc service parameters	895
Table 464 – A_Writeloc service parameters	896
Table 465 – A_Update service parameters.....	898
Table 466 – A_Readfar service parameters	900
Table 467 – A_Writefar service parameters	902
Table 468 – A_Sent service parameters.....	904
Table 469 – A_Received service parameters	905
Table 470 – A_Read service parameters	906
Table 471 – A_Write service parameters.....	908
Table 472 – Asynchronous refreshment private mechanism events and actions	917
Table 473 – Asynchronous refreshment public mechanism events and actions	918
Table 474 – Synchronous refreshment private mechanism events and actions.....	919
Table 475 – Synchronous refreshment public mechanism events and actions	920
Table 476 – Punctual refreshment private mechanism events and actions	922
Table 477 – Punctual refreshment public mechanism events and actions.....	923
Table 478 – Asynchronous promptness public mechanism events and actions.....	926
Table 479 – Asynchronous promptness private mechanism events and actions	927
Table 480 – Synchronous promptness public mechanism events and actions.....	929
Table 481 – Synchronous promptness privatemechanism events and actions	930
Table 482 – Punctual promptness public mechanism events and actions	932
Table 483 – Punctual promptness privatemechanism events and actions.....	

Table 484 – A_Readlist service parameters	941
Table 485 – Confirmed initiate service parameters.....	961
Table 486 – Detailed structure of the extension calling parameter.....	962
Table 487 – Detailed structure of the init request detail parameter	963
Table 488 – Detailed structure of the extension called parameter	964
Table 489 – Detailed structure of the init request detail parameter	965
Table 490 – Conclude service parameter	966
Table 491 – Unconfirmed abort service parameters	968
Table 492 – Unconfirmed reject service parameters.....	969
Table 493 – Confirmed status service parameters.....	971
Table 494 – Unconfirmed unsolicited status service parameter	971
Table 495 – Confirmed identify service parameters.....	972
Table 496 – Confirmed get name list service paramaters	973
Table 497 – Access group attribute description for domain object	975
Table 498 – Access rights attribute description for domain object	976
Table 499 – Confirmed delete domain service parameters	976
Table 500 – Confirmed initiate download sequence service parameters.....	977
Table 501 – Confirmed download segment service parameters	979
Table 502 – Confirmed terminate download sequence service parameters.....	980
Table 503 – Confirmed initiate upload sequence service parameters	981
Table 504 – Confirmed upload segment service parameters	982
Table 505 – Confirmed terminate upload sequence service parameters	983
Table 506 – Confirmed get domain attributes service parameters	984
Table 507 – Access group attribute details for program invocation object	993
Table 508 – Access rights attribute details for program invocation object.....	993
Table 509 – Confirmed create program invocation service parameters.....	994
Table 510 – Confirmed delete program invocation service parameters.....	996
Table 511 – Confirmed start service parameters	997
Table 512 – Confirmed stop service parameters	998
Table 513 – Confirmed resume service parameters	999
Table 514 – Confirmed reset service parameters	1000
Table 515 – Confirmed kill service parameters.....	1001
Table 516 – Access group attribute details for variable object.....	1005
Table 517 – Access rights attribute details for variable object.....	1006
Table 518 – Access group attribute details for variable list object	1007
Table 519 – Access right attribute details for variable list objects.....	1007
Table 520 – Confirmed read service parameters	1008
Table 521 – Confirmed write service parameters	1010
Table 522 – Unconfirmed information report service parameters.....	1011
Table 523 – Confirmed define variable-list service parameters.....	1012
Table 524 – Confirmed delete variable-list service parameters.....	1013
Table 525 – Confirmed get variable access attributes service parameters.....	1014
Table 526 – Confirmed get variable-list attributes service parameters.....	1016
Table 527 – Data type specification	1017
Table 528 – Variable access specification.....	1018
Table 529 – Variable access description attribute details	

Table 530 – Path selection parameters	1020
Table 531 – Access group attribute detail for event object	1023
Table 532 – Access rights attribute details for event object	1023
Table 533 – Unconfirmed event notification service parameters	1024
Table 534 – Event type parameter details	1024
Table 535 – Confirmed acknowledged event notification service parameter	1026
Table 536 – Confirmed alter event condition monitoring service parameters	1027
Table 537 – Confirmed get alarm summary service parameters	1029
Table 538 – Confirmed get event condition attributes service parameters	1031
Table 539 – Classification of service quality parameters	1034
Table 540 – Identification parameters	1038
Table 541 – List of MCS AR ASE services	1039
Table 542 – A_Associate service parameters	1039
Table 543 – A_Release service parameters	1044
Table 544 – A_Abort service parameters	1045
Table 545 – A_Data service parameters	1046
Table 546 – A_Unidata service parameters	1047
Table 547 – Get Attributes service parameters	1055
Table 548 – Initiate service parameters	1058
Table 549 – Conveyance of service primitives by AREP role	1059
Table 550 – Valid combinations of AREP roles involved in an AR	1059
Table 551 – AR-Unconfirmed Send service parameters	1061
Table 552 – AR-Establish service parameters	1062
Table 553 – AR-Data-Send-Acknowledge service parameters	1062
Table 554 – Read service parameters	1065
Table 555 – Write service parameters	1065
Table 556 – Information Report service parameters	1066
Table 557 – Start service parameters	1067
Table 558 – Stop service parameters	1067
Table 559 – Resume service parameters	1068
Table 560 – Reset service parameters	1068
Table 561 – Services by AREP role	1071
Table 562 – Logical status	1074
Table 563 – Status	1075
Table 564 – Unsolicited status	1076
Table 565 – Identify	1077
Table 566 – Structure of the object dictionary	1080
Table 567 – Structure of the static list of types	1080
Table 568 – Structure of the static object dictionary	1080
Table 569 – Structure of the dynamic list of variable lists	1081
Table 570 – Structure of the dynamic list of program invocations	1081
Table 571 – Empty object dictionary	1085
Table 572 – Get OD service parameters	1088
Table 573 – Initiate Put OD service parameters	1090
Table 574 – Put OD service parameters	1091
Table 575 – Terminate Put OD service parameters	

Table 576 – Put OD state transitions.....	1094
Table 577 – Attribute FMS features supported	1096
Table 578 – Transaction object state transitions	1099
Table 579 – Initiate service parameters	1100
Table 580 – Failure reasons	1101
Table 581 – Abort service parameters.....	1102
Table 582 – User abort reasons	1103
Table 583 – APO ASE abort reasons	1103
Table 584 – Reject service parameters	1104
Table 585 – Reject APDU reasons	1104
Table 586 – Compatibility of the local context to the remote context	1105
Table 587 – Unconfirmed Send service parameters	1108
Table 588 – Confirmed Send service parameters	1109
Table 589 – AR-Abort service parameters.....	1109
Table 590 – Compel service parameters	1110
Table 591 – Get Buffered Message service parameters	1111
Table 592 – AR-Status service parameters	1112
Table 593 – Simple variable access group membership	1114
Table 594 – Simple variable access rights membership	1114
Table 595 – Array variable access group membership	1116
Table 596 – Array variable access rights membership.....	1116
Table 597 – Variable list access group membership.....	1118
Table 598 – Variable list access rights membership.....	1118
Table 599 – Read service parameters.....	1121
Table 600 – Write service parameters.....	1122
Table 601 – Information Report service parameters	1123
Table 602 – Define Variable List service parameters	1124
Table 603 – Delete Variable List service parameters	1125
Table 604 – Event access group membership	1127
Table 605 – Event access rights membership	1128
Table 606 – Event Notification service parameters.....	1129
Table 607 – Acknowledge Event Notification service parameters	1130
Table 608 – Alter Event Condition Monitoring service parameters.....	1131
Table 609 – Event state transitions	1132
Table 610 – Domain access group membership	1133
Table 611 – Domain access rights membership	1133
Table 612 – GenericInitiateDownloadSequence	1135
Table 613 – GenericDownloadSegment	1136
Table 614 – GenericTerminateDownloadSequence	1137
Table 615 – InitiateDownloadSequence	1138
Table 616 – DownloadSegment	1139
Table 617 – TerminateDownloadSequence	1140
Table 618 – RequestDomainDownload.....	1141
Table 619 – InitiateUploadSequence.....	1142
Table 620 – UploadSegment.....	1143
Table 621 – TerminateUploadSequence	

Table 622 – RequestDomainUpload	1145
Table 623 – Domain GenericDownload/Download state machine (server)	1146
Table 624 – Domain Upload state machine (server)	1148
Table 625 – Program invocation access group membership	1150
Table 626 – Program invocation access group membership	1150
Table 627 – Create Program Invocation service parameters	1152
Table 628 – Delete Program Invocation service parameters	1153
Table 629 – Start service parameters	1154
Table 630 – Stop service parameters	1155
Table 631 – Resume service parameters	1156
Table 632 – Reset service parameters	1157
Table 633 – Kill service parameters	1158
Table 634 – Program invocation state machine	1160
Table 635 – Class summary	1165
Table 636 – Services by AREP role	1166
Table 637 – QueryInterface	1175
Table 638 – AddRef	1176
Table 639 – Release	1177
Table 640 – GetTypeInfoCount	1178
Table 641 – GetTypeInfo	1179
Table 642 – GetIDsOfNames	1180
Table 643 – Invoke	1181
Table 644 – get_Producer	1185
Table 645 – get_Product	1186
Table 646 – get_SerialNo	1187
Table 647 – get_ProductionDate	1188
Table 648 – Revision	1189
Table 649 – get_LogicalDevice	1190
Table 650 – get_Count	1191
Table 651 – BrowseItems	1192
Table 652 – Save	1193
Table 653 – get_Name	1198
Table 654 – get_Producer	1199
Table 655 – get_Product	1200
Table 656 – get_SerialNo	1201
Table 657 – get_ProductionDate	1202
Table 658 – Revision	1203
Table 659 – get_ACCO	1204
Table 660 – get_RTAuto	1205
Table 661 – get_State	1206
Table 662 – Activate	1207
Table 663 – Deactivate	1208
Table 664 – Reset	1209
Table 665 – AdviseState	1210
Table 666 – UnadviseState	1211
Table 667 – get_Time	

Table 668 – put_Time	1213
Table 669 – get_Count	1214
Table 670 – BrowseItems	1215
Table 671 – GroupError	1216
Table 672 – AdviseGroupError.....	1217
Table 673 – UnadviseGroupError.....	1218
Table 674 – PingFactor values.....	1220
Table 675 – QoS Types and Values	1225
Table 676 – Quality Codes.....	1232
Table 677 – Quality Code priority table	1238
Table 678 – Error codes for the ACCO ASE detailed diagnosis	1251
Table 679 – AddConnections	1254
Table 680 – RemoveConnections.....	1256
Table 681 – ClearConnections	1257
Table 682 – SetActivationState	1258
Table 683 – GetInfo	1259
Table 684 – GetIDs.....	1260
Table 685 – GetConnections.....	1261
Table 686 – ReviseQoS	1262
Table 687 – get_PingFactor	1264
Table 688 – put_PingFactor	1265
Table 689 – get_CDBCookie.....	1266
Table 690 – Connect.....	1267
Table 691 – Disconnect	1268
Table 692 – DisconnectMe.....	1270
Table 693 – SetActivation	1271
Table 694 – Ping.....	1272
Table 695 – OnDataChanged.....	1273
Table 696 – Data Type Names for Values	1275
Table 697 – ReadItems	1276
Table 698 – WriteItems.....	1277
Table 699 – WriteItemsQCD	1278
Table 700 – GroupError	1279
Table 701 – AdviseGroupError.....	1280
Table 702 – UnadviseGroupError.....	1281
Table 703 – Allowed Data Types.....	1283
Table 704 – get_Name.....	1284
Table 705 – Revision	1285
Table 706 – get_Count	1286
Table 707 – BrowseItems	1287
Table 708 – CoCreateInstance.....	1289
Table 709 – Call	1290
Table 710 – Fieldbus AL class summary	1291
Table 711 – Assignment of the services to client and server	1291

0 Introduction

0.1 General

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, which is based in part on the Basic Reference Model for Open Systems Interconnection. Both Reference Models subdivide the area of standardization for interconnection into a series of layers of specification, each of manageable size.

The Application Service is provided by the Application Protocol making use of the services available from the Data Link or other immediately lower layer. This part of the IEC 61158 series defines the Application Service characteristics that any immediately higher-level protocols may exploit. The relationship between the International Standards for fieldbus Application Service, fieldbus Application Protocol and Systems Management is illustrated in Figure 1.

NOTE Systems Management, as used in this standard, is a local mechanism for managing the layer protocols

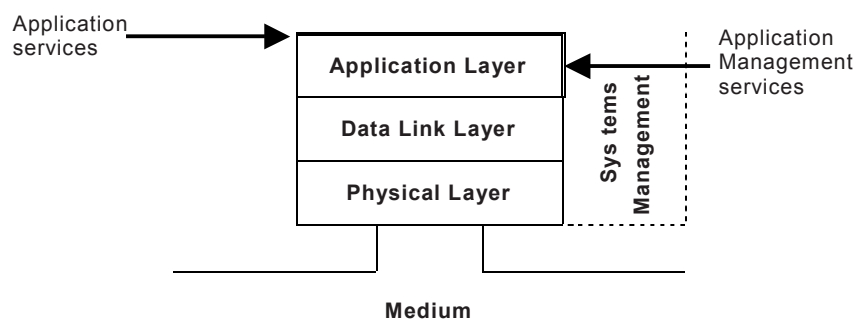


Figure 1 – Relationship of IEC 61158-3 to other fieldbus layers and to users of the fieldbus Application Service

This application layer standard does not specify individual implementations or products, nor does it constrain the implementations of application entities and interfaces within the industrial automation system.

This application layer standard does not contain test specifications used to demonstrate compliance with IEC 61158-5 and IEC 61158-6.

0.2 Nomenclature for references within this standard

Clauses, including annexes, can be referenced in their entirety, including any subordinate subclauses, as “clause N” or “Annex N”, where N is the number of the clause or letter of the annex.

Subclauses can be referenced in their entirety, including any subordinate subclauses, as “N.M” or “N.M.P” and so forth, depending on the level of the subclause, where N is the number of the subclause or letter of the annex, and M, P and so forth represent the successive levels of subclause up to and including the subclause of interest.

When a clause or subclause contains one or more subordinate subclauses, the text between the clause or subclause heading and its first subordinate subclause can be referenced in its entirety as “N.0” or “N.M.0” or “N.M.P.0” and so forth, where N, M and P are as above. Stated differently, a reference ending with “.0” designates the text and figures between a clause or subclause header and its first subordinate subclause.

DIGITAL DATA COMMUNICATIONS FOR MEASUREMENT AND CONTROL – FIELDBUS FOR USE IN INDUSTRIAL CONTROL SYSTEMS –

Part 5: Application Layer Service definition

1 Scope

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

The FAL is an Application Layer Communication Standard designed to support the conveyance of time-critical and non-time-critical application requests and responses among devices in an automation environment. The term “time-critical” is used to represent the presence of an application time-window, within which one or more specified actions are required to be completed with some defined level of certainty.

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

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

This part of IEC 61158 specifies interactions between remote applications in terms of

- an abstract model for defining application resources (objects) capable of being manipulated by users via the use of FAL Services,
- the primitives (interactions between the FAL and the FAL user) associated with each FAL Service;
- the parameters associated with each primitive;
- the interrelationship between and the valid sequences of the primitives for each service.

Several models of communications are specified in this document. Each model is specified as a communication “type”. Each type has its own separate clause in the document.

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

2 Normative references

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

IEC 61131, *Programmable controllers*

IEC 61131-1:1992, *Part 1: Programmable controllers — General information*

IEC 61131-3:1993, *Part 3: Programmable controllers — Programming languages*

IEC 61158, *Digital data communications for measurement and control – Fieldbus for use in industrial control systems*

IEC 61158-3:2003, *Digital data communications for measurement and control – Fieldbus for use in industrial control systems — Part 3: Data Link Layer service definition*

IEC 61158-4:2003, *Digital data communications for measurement and control – Fieldbus for use in industrial control systems — Part 4: Data Link Layer Protocol Specification*

IEC 61158-6:2003, *Digital data communications for measurement and control – Fieldbus for use in industrial control systems — Part 6: Application Layer Protocol Specification*

ISO/IEC 646:1991, *Information technology – ISO 7-bit coded character set for information interchange*

ISO/IEC 7498, *Information technology – Open Systems Interconnection – Basic Reference Model*

ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model — Part 1: The Basic Model*

ISO/IEC 7498-3:1997, *Information technology – Open Systems Interconnection – Basic Reference Model — Part 3: Naming and addressing*

ISO/IEC 8822:1994, *Information technology – Open Systems Interconnection – Presentation service definition*

ISO/IEC 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*

ISO/IEC 8859-1:1998, *Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*

ISO/IEC 9545:1994, *Information technology – Open Systems Interconnection – Application Layer structure*

ISO/IEC 10646-1:1993, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Architecture and Basic Multilingual Plane*

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

IEEE Std 754:1985 (R1990), *IEEE Standard for Binary Floating Point Arithmetic*

3 Terms and definitions

For the purposes of this document, the following terms as defined in these publications apply:

3.1 ISO/IEC 7498-1 terms

- a) application entity
- b) application process
- c) application protocol data unit
- d) application service element
- e) application entity invocation

- f) application process invocation
- g) application transaction
- h) real open system
- i) transfer syntax

3.2 ISO/IEC 8822 terms

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

- a) abstract syntax
- b) presentation context

3.3 ISO/IEC 9545 terms

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

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

3.4 ISO/IEC 8824 terms

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

- a) object identifier
- b) type

3.5 Fieldbus Data Link Layer terms

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

- a) DL-Time
- b) DL-Scheduling-policy
- c) DLCEP
- d) DLC
- e) DL-connection-oriented mode
- f) DLPDU
- g) DLSDU
- h) DLSAP
- i) fixed tag
- j) generic tag
- k) link
- l) MAC ID
- m) network address

- n) node address
- o) node
- p) tag
- q) scheduled
- r) unscheduled

3.6 Fieldbus Application Layer specific definitions

For the purposes of this part of IEC 61158, the following terms and definitions apply.

3.6.1 access protection

limitation of the usage of an application object to one client

3.6.2 active connection control object

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

3.6.3 address assignment table

mapping of the client's internal I/O-Data object storage to the decentralised input and output data objects

3.6.4 allocate

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

3.6.5 application

function or data structure for which data is consumed or produced

3.6.6 application layer interoperability

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

3.6.7 application objects

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

3.6.8 application process

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

3.6.9 application process identifier

distinguishes multiple application processes used in a device

3.6.10 application process object

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

NOTE Application process object definitions are composed of a set of values for the attributes of their class (see the definition for Application Process Object Class Definition). Application process object definitions may be accessed remotely using the services of the FAL Object Management ASE. FAL Object Management services can be used to load or update object definitions, to read object definitions, and to dynamically create and delete application objects and their corresponding definitions.

3.6.11 application process object class

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

3.6.12 application relationship

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

3.6.13 application relationship application service element

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

3.6.14 application relationship endpoint

context and behavior of an application relationship as seen and maintained by one of the application processes involved in the application relationship

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

3.6.15 attribute

description of an externally visible characteristic or feature of an object

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

3.6.16 behaviour

indication of how an object responds to particular events

3.6.17 bit-no

designates the number of a bit in a bitstring or an octet

3.6.18 channel

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

3.6.19 channel related diagnosis

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

EXAMPLE: validity of data

3.6.20 class

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

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

3.6.21 class attributes

attribute that is shared by all objects within the same class

3.6.22 class code

unique identifier assigned to each object class

3.6.23 class specific service

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

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

3.6.24 client

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

3.6.25 communication objects

components that manage and provide a run time exchange of messages across the network

EXAMPLES: Connection Manager object, Unconnected Message Manager (UCMM) object, and Message Router object

3.6.26 configuration check

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

3.6.27 configuration data base

interconnection information maintained by the ACCO ASE

3.6.28 configuration fault

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

3.6.29 configuration identifier

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

3.6.30 connection

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

NOTE 1 Connections may be either point-to-point or multipoint.

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

3.6.31 connection channel

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

3.6.32 connection ID (CID)

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

3.6.33 connection path

an octet stream that defines the application object to which a connection instance applies

3.6.34 connection point

buffer which is represented as a subinstance of an Assembly object

3.6.35 consume

act of receiving data from a producer

3.6.36 consumer

node or sink that is receiving data from a producer

3.6.37 consuming application

application that consumes data

3.6.38 consumerID

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

3.6.39 control commands

action invocations transferred from client to server to clear outputs, freeze inputs and/or synchronise outputs

3.6.40 conveyance path

unidirectional flow of APDUs across an application relationship

3.6.41 cyclic

repetitive in a regular manner

3.6.42 data consistency

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

3.6.43 data marshaling

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

<Type 10> NOTE This is part of the abstract ORPC model.

3.6.44 dedicated AR

AR used directly by the FAL User

NOTE On Dedicated ARs, only the FAL Header and the user data are transferred.

3.6.45 default DL-address

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

3.6.46 device

physical hardware connected to the link

NOTE A device may contain more than one node.

3.6.47 device profile

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

3.6.48 diagnosis information

all data available at the server for maintenance purposes

3.6.49 diagnosis information collection

system diagnosis information that is assembled at the client side

3.6.50 DP-master (class 1)

a controlling device which controls several DP-slaves (field devices)

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

3.6.51 DP-master (class 2)

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

3.6.52 DP-slave

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

3.6.53 dynamic AR

AR that requires the use of the AR establishment procedures to place it into an established state

3.6.54 end node

producing or consuming node

3.6.55 endpoint

one of the communicating entities involved in a connection

3.6.56 engineering

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

3.6.57 error

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

3.6.58 error class

general grouping for related error definitions and corresponding error codes

3.6.59 error code

identification of a specific type of error within an error class

3.6.60 event

an instance of a change of conditions

3.6.61 FAL subnet

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

NOTE FAL subnets are permitted to contain bridges but not routers.

3.6.62 FIFO variable

a Variable Object class, composed of a set of homogeneously typed elements, where the first written element is the first element that can be read

NOTE On the fieldbus only one, complete element can be transferred as a result of one service invocation.

3.6.63 frame

denigrated synonym for DLPDU

3.6.64 freeze

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

3.6.65 group

- a) <general> a general term for a collection of objects. Specific uses:
- b) <addressing> when describing an address, an address that identifies more than one entity
- c) <Type 3> set of DP-slaves which perform a Freeze or Sync function

3.6.66 interface

- (a) shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics as appropriate
- (b) collection of FAL class attributes and services that represents a specific view on the FAL class

3.6.67 interface definition language

syntax and semantics of describing service parameters in a formal way

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

3.6.68 interface pointer

key attribute that unambiguously addresses an object interface instance

3.6.69 invocation

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

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

3.6.70 I/O data

object designated to be transferred cyclically for the purpose of processing

3.6.71 ident number

DP-master (Class 1) or DP-slave device type

3.6.72 identifier related diagnosis

information dedicated to modules for maintenance purpose

3.6.73 index

address of an object within an application process

3.6.74 instance

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

EXAMPLE Clifornia is an instance of the object class state.

NOTE The terms object, instance, and object instance are used to refer to a specific instance.

3.6.75 instance attributes

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

3.6.76 instantiated

object that has been created in a device

3.6.77 logical device

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

3.6.78 Lpacket (or Link packet)

a piece of application information that contains a size, control byte, tag, and link data

NOTE Peer Data Link Layers use Lpackets to send and receive service data units from higher layers in the OSI stack.

3.6.79 manufacturer ID

identification of each product manufacturer by a unique number

3.6.80 management information

network-accessible information that supports managing the operation of the fieldbus system, including the application layer

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

3.6.81 master parameter set

the configuration and parameterisation data of all DP-slaves that are assigned to the corresponding DP-master and the bus parameters

3.6.82 member

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

3.6.83 message router

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

3.6.84 method

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

3.6.85 module

- a) <general> hardware or logical component of a physical device
- b) <Type 3> addressable unit inside the DP-slave

3.6.86 multipoint connection

connection from one node to many

NOTE Multipoint connections allow messages from a single producer to be received by many consumer nodes.

3.6.87 network

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

3.6.88 object

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

3.6.89 object remote procedure call

model for object oriented or component based remote method invocation

3.6.90 object specific service

service unique to the object class which defines it

3.6.91 originator

client responsible for establishing a connection path to the target

3.6.92 peer

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

3.6.93 physical device

<general> an automation or other network device

<Type10> a certain FAL class that abstracts the hardware facilities of an automation device

3.6.94 point-to-point connection

connection that exists between exactly two application objects

3.6.95 pre-defined AR endpoint

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

NOTE Pre-defined ARs that are not pre-established are established before being used

3.6.96 pre-established AR endpoint

AR endpoint that is placed in an established state during configuration of the AEs that control its endpoints

3.6.97 process data

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

3.6.98 produce

act of sending data to be received by a consumer

3.6.99 producer

node that is responsible for sending data

3.6.100 property

- a) <general> a general term for descriptive information about an object
- b) <Type 10> a synonym for ASE attributes which are readable or writeable via operational ASE services

NOTE These services are generally named "get_<Attribute Name>" or "set_<Attribute Name>" and correspond with the IDL keywords "propget" and "propput" used in Type 10 clauses of IEC 61158-6.

3.6.101 provider

source of a data connection

3.6.102 providerID

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

3.6.103 publisher

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

NOTE A publisher may not be aware of the identity or the number of subscribers and it may publish its APDUs using a dedicated AR.

3.6.104 publishing manager

role of an AR endpoint in which it issues one or more confirmed service request APDUs to a publisher to request the publisher to publish a specified object. Two types of publishing managers are defined by this standard, pull publishing managers and push publishing managers, each of which is defined separately

3.6.105 pull publisher

type of publisher that publishes an object in response to a request received from its pull publishing manager

3.6.106 pull publishing manager

type of publishing manager that requests that a specified object be published in a corresponding response APDU

3.6.107 push publisher

type of publisher that publishes an object in an unconfirmed service request APDU

3.6.108 push publishing manager

type of publishing manager that requests that a specified object be published using an unconfirmed service

3.6.109 pull subscriber

type of subscriber that recognizes received confirmed service response APDUs as published object data

3.6.110 push subscriber

type of subscriber that recognizes received unconfirmed service request APDUs as published object data

3.6.111 quality code aware

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

3.6.112 quality code

additional status information of a data item

3.6.113 quality code unaware

opposite of quality code aware

3.6.114 real configuration

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

3.6.115 resource

a processing or information capability of a subsystem

3.6.116 route endpoint

object container containing Variable Objects of a variable class

3.6.117 RT-auto

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

3.6.118 runtime object model

objects that exist in a device together with their interfaces and methods that are accessible

3.6.119 serial number

<Type 2> a unique 32-bit integer value assigned by each manufacturer to every device having Type 2 communication capabilities

NOTE The Manufacturer ID and serial number jointly form a unique identifier for each device.

3.6.120 server

- a) role of an AREP in which it returns a confirmed service response APDU to the client that initiated the request
- b) object which provides services to another (client) object

3.6.121 service

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

3.6.122 slot

address of a module within a DP-slave

3.6.123 subscriber

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

3.6.124 sync

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

3.6.125 target

end-node to which a connection is established

3.6.126 unconnected message manager (UCMM)

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

3.6.127 unconnected service

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

3.7 Abbreviations and symbols

ACCO	Active Connection Control Object
AE	Application Entity
AL	Application Layer
ALME	Application Layer Management Entity
ALP	Application Layer Protocol
APO	Application Object
AP	Application Process
APDU	Application Protocol Data Unit
API	Application Process Identifier
AR	Application Relationship
AREP	Application Relationship End Point
ASCII	American Standard Code for Information Interchange
ASE	Application Service Element
CID	Connection ID
CIM	Computer Integrated Manufacturing
CIP	Control and Information Protocol
CM_API	Actual Packet Interval
CM_RPI	Requested Packet Interval
Cnf	Confirmation
COR	Connection originator
CR	Communication Relationship
CREP	Communication Relationship End Point
DL-	(as a prefix) Data Link-
DLC	Data Link Connection
DLCEP	Data Link Connection End Point
DLL	Data Link Layer
DLM	Data Link-management
DLSAP	Data Link Service Access Point
DLSDU	DL-service-data-unit
DNS	Domain Name Service
DP	Decentralised Peripherals
FAL	Fieldbus Application Layer
FIFO	First In First Out
HMI	Human-Machine Interface
ID	Identifier
IDL	Interface Definition Language
IEC	International Electrotechnical Commission
Ind	Indication
IP	Internet Protocol
ISO	International Organization for Standardization
LDev	Logical Device

LME	Layer Management Entity
O2T	Originator to target (connection characteristics)
O⇒T	Originator to target (connection characteristics)
ORPC	Object Remote Procedure Call
OSI	Open Systems Interconnect
PDev	Physical Device
PDU	Protocol Data Unit
PL	Physical Layer
QoS	Quality of Service
QC	Quality Code
REP	Route Endpoint
Req	Request
Rsp	Response
RT	Runtime
SAP	Service Access Point
SCL	Security Level
SDU	Service Data Unit
SEM	State event matrix
SMIB	System Management Information Base
SMK	System Management Kernel
STD	State transition diagram, used to describe object behaviour
S-VFD	Simple Virtual Field Device
T2O	Target to originator (connection characteristics)
T⇒O	Target to originator (connection characteristics)
VAO	Variable Object

3.8 Conventions

3.8.1 Overview

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

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

3.8.2 General conventions

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

3.8.3 Conventions for class definitions

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

FAL ASE:	ASE Name
CLASS:	Class Name
CLASS ID:	#

PARENT CLASS:		Parent Class Name
ATTRIBUTES:		
1	(o)	Key Attribute: numeric identifier
2	(o)	Key Attribute: name
3	(m)	Attribute: attribute name(values)
4	(m)	Attribute: attribute name(values)
4.1	(s)	Attribute: attribute name(values)
4.2	(s)	Attribute: attribute name(values)
4.3	(s)	Attribute: attribute name(values)
5.	(c)	Constraint: constraint expression
5.1	(m)	Attribute: attribute name(values)
5.2	(o)	Attribute: attribute name(values)
6	(m)	Attribute: attribute name(values)
6.1	(s)	Attribute: attribute name(values)
6.2	(s)	Attribute: attribute name(values)

SERVICES:

1	(o)	OpsService: service name
2.	(c)	Constraint: constraint expression
2.1	(o)	OpsService: service name
3	(m)	MgtService: service name

- (1) The "FAL ASE:" entry is the name of the FAL ASE that provides the services for the class being specified.
- (2) The "CLASS:" entry is the name of the class being specified. All objects defined using this template will be an instance of this class. The class may be specified by this standard, or by a user of this standard.
- (3) The "CLASS ID:" entry is a number that identifies the class being specified. This number is unique within the FAL ASE that will provide the services for this class. When qualified by the identity of its FAL ASE, it unambiguously identifies the class within the scope of the FAL. The value "NULL" indicates that the class cannot be instantiated. Class IDs between 1 and 255 are reserved by this standard to identify standardized classes. They have been assigned to maintain compatibility with existing national standards. CLASS IDs between 256 and 2048 are allocated for identifying user defined classes.
- (4) The "PARENT CLASS:" entry is the name of the parent class for the class being specified. All attributes defined for the parent class and inherited by it are inherited for the class being defined, and therefore do not have to be redefined in the template for this class.

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

- (5) The "ATTRIBUTES" label indicate that the following entries are attributes defined for the class.
- Each of the attribute entries contains a line number in column 1, a mandatory (m) / optional (o) / conditional (c) / selector (s) indicator in column 2, an attribute type label in column 3, a name or a conditional expression in column 4, and optionally a list of enumerated values in column 5. In the column following the list of values, the default value for the attribute may be specified.
 - Objects are normally identified by a numeric identifier or by an object name, or by both. In the class templates, these key attributes are defined under the key attribute.

- c) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting is used to specify
 - i) fields of a structured attribute (4.1, 4.2, 4.3),
 - ii) attributes conditional on a constraint statement (5). Attributes may be mandatory (5.1) or optional (5.2) if the constraint is true. Not all optional attributes require constraint statements as does the attribute defined in (5.2).
 - iii) the selection fields of a choice type attribute (6.1 and 6.2).
- (6) The "SERVICES" label indicates that the following entries are services defined for the class.
 - a) An (m) in column 2 indicates that the service is mandatory for the class, while an (o) indicates that it is optional. A (c) in this column indicates that the service is conditional. When all services defined for a class are defined as optional, at least one has to be selected when an instance of the class is defined.
 - b) The label "OpsService" designates an operational service (1).
 - c) The label "MgtService" designates an management service (2).
 - d) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting within the list of services is used to specify services conditional on a constraint statement.

3.8.4 Conventions for service definitions

3.8.4.1 General

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

3.8.4.2 Service parameters

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

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

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

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

- M parameter is mandatory for the primitive
- U parameter is a User option, and may or may not be provided depending on dynamic usage of the service user. When not provided, a default value for the parameter is assumed.
- C parameter is conditional upon other parameters or upon the environment of the service user.

— (blank) parameter is never present.

S parameter is a selected item.

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

a) a parameter-specific constraint:

“(=)” indicates that the parameter is semantically equivalent to the parameter in the service primitive to its immediate left in the table.

b) an indication that some note applies to the entry:

“(n)” indicates that the following note "n" contains additional information pertaining to the parameter and its use.

3.8.4.3 Service procedures

The procedures are defined in terms of

- the interactions between application entities through the exchange of fieldbus Application Protocol Data Units, and
- the interactions between an application layer service provider and an application layer service user in the same system through the invocation of application layer service primitives.

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

4 Concepts

4.1 Overview

The fieldbus is intended to be used in factories and process plants to interconnect primary automation devices (e.g. sensors, actuators, local display devices, annunciators, programmable logic controllers, small single loop controllers, and stand-alone field controls) with control and monitoring equipment located in control rooms.

Primary automation devices are associated with the lowest levels of the industrial automation hierarchy and perform a limited set of functions within a definite time window. Some of these functions include diagnostics, data validation, and handling of multiple inputs and outputs.

These primary automation devices, also termed field devices, are located close to the process fluids, the fabricated part, the machine, the operator and the environment. This use positions the fieldbus at the lowest levels of the Computer Integrated Manufacturing (CIM) architecture.

Some of the expected benefits in using fieldbus are reduction in wiring, increase in amount of data exchanged, wider distribution of control between the primary automation devices and the control room equipment, and the satisfaction of time critical constraints.

This subclause describes fundamentals of the FAL. Detailed descriptive information about each of the FAL ASEs can be found in the “Overview” subclause of each of the Communication Model specifications.

4.2 Architectural relationships

4.2.1 Relationship to the Application Layer of the OSI basic reference model

The functions of the FAL have been described according to OSI layering principles. However, its architectural relationship to the lower layers is different, as shown in Figure 2.

- The FAL includes OSI functions together with extensions to cover time-critical requirements. The OSI Application Layer Structure standard (ISO/IEC 9545) was used as a basis for specifying the FAL.
- The FAL directly uses the services of the underlying layer. The underlying layer may be the data link layer or any layer in between. When using the underlying layer, the FAL may provide functions normally associated with the OSI Middle Layers for proper mapping onto the underlying layer.

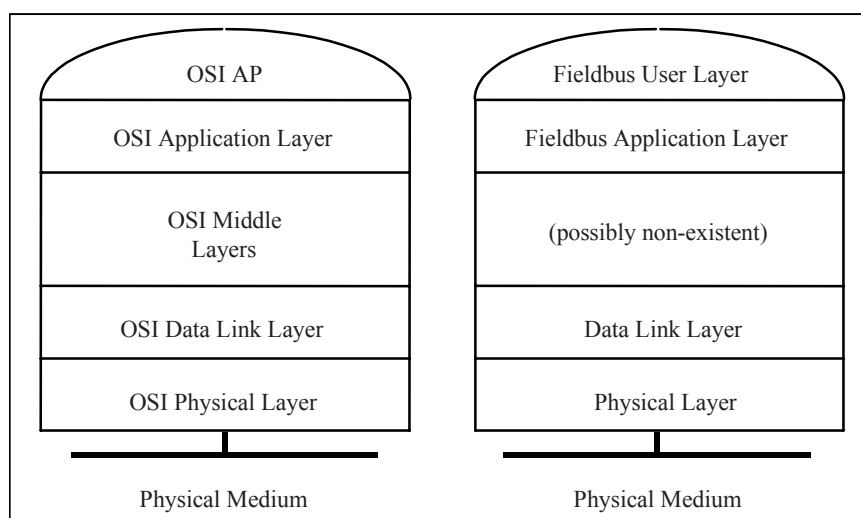


Figure 2 – Relationship to the OSI basic reference model

4.2.2 Relationships to other fieldbus entities

4.2.2.1 General

The fieldbus Application Layer (FAL) architectural relationships, as illustrated in Figure 3, have been designed to support the interoperability needs of time-critical systems distributed within the fieldbus environment.

Within this environment, the FAL provides communications services to time-critical and non-time-critical applications located in fieldbus devices.

In addition, the FAL directly uses the Data Link Layer to transfer its application layer protocol data units. It does this using a set of data transfer services and a set of supporting services used to control the operational aspects of the Data Link Layer.

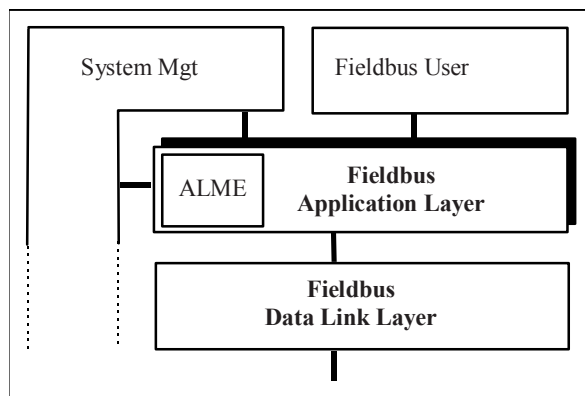


Figure 3 – Architectural positioning of the fieldbus Application Layer

The fieldbus Application Layer (FAL) provides network access to fieldbus APs. It interfaces directly to the fieldbus Data Link Layer for transfer of its APDUs.

The Data Link Layer provides various types of services to the FAL for the transfer of data between Data Link endpoints (e.g. DLSAPs, DLCEPs).

Fieldbus applications are represented to the network as application processes (APs). APs are the components of a distributed system that may be individually identified and addressed.

Each AP contains an FAL application entity (AE) that provides network access for the AP. That is, each AP communicates with other APs through its AE. In this sense, the AE provides a window of visibility into the AP.

APs contain identifiable components that are also visible across the network. These components are represented to the network as Application Process Objects (APO). They may be identified by one or more key attributes. They are located at the address of the application process that contains them.

The services used to access them are provided by APO-specific application service elements (ASEs) contained within the FAL. These ASEs are designed to support user, function block, and management applications.

The FAL services can be used to support various management operations, including management of fieldbus systems, applications, and the fieldbus network.

One layer management entity (LME) may be present in each FAL entity on the network. FALMEs provide access to the FAL for system management purposes.

The set of data accessible by the System Manager is referred to as the System Management Information Base (SMIB). Each fieldbus Application Layer Management Entity (FALME) provides the FAL portion of the SMIB. How the SMIB is implemented is beyond the scope of this standard.

4.3 Fieldbus Application Layer structure

4.3.1 Overview

The structure of the FAL is a refinement of the OSI Application Layer Structure (ISO/IEC 9545). As a result, the organization of this subclause is similar to that of ISO/IEC 9545. Certain concepts presented here have been refined from ISO/IEC 9545 for the fieldbus environment.

The FAL differs from the other layers of OSI in two principal aspects:

- OSI defines a single type of application layer communications channel, the association, to connect APs to each other. The FAL defines the Application Relationship (AR), of which there are several types, to permit application processes (APs) to communicate with each other.
- The FAL uses the DLL to transfer its PDUs and not the presentation layer. Therefore, there is no explicit presentation context available to the FAL. Between the same pair (or set) of data link service access points the FAL protocol may not be used concurrently with other application layer protocols.

4.3.2 Fundamental concepts

The operation of time critical real open systems is modeled in terms of interactions between time-critical APs. The FAL permits these APs to pass commands and data between them.

Cooperation between APs requires that they share sufficient information to interact and carry out processing activities in a coordinated manner. Their activities may be restricted to a single fieldbus segment, or they may span multiple segments. The FAL has been designed using a modular architecture to support the messaging requirements of these applications.

Cooperation between APs also sometimes requires that they share a common sense of time. The FAL or the Data Link Layer (IEC 61158-3 and IEC 61158-4) may provide for the distribution of time to all devices. They also may define local device services that can be used by APs to access the distributed time.

The remainder of this subclause describes each of the modular components of the architecture and their relationships with each other. The components of the FAL are modeled as objects, each of which provides a set of FAL communication services for use by applications. The FAL objects and their relationships are described below. The detailed specifications of FAL objects and their services are provided in the following clauses of this standard. IEC 61158-6 specifies the protocols necessary to convey these object services between applications.

4.3.3 Fieldbus application processes

4.3.3.1 Definition of the fieldbus AP

In the fieldbus environment, an application may be partitioned into a set of components and distributed across a number of devices on the network. Each of these components is referred to as a fieldbus Application Process (AP). A fieldbus AP is a variation of an Application Process as defined in ISO OSI Reference Model (ISO/IEC 7498). Fieldbus APs may be unambiguously addressed by at least one individual Data Link Layer service access point address. Unambiguously addressed, in this context, means that no other AP may simultaneously be located by the same address. This definition does not prohibit an AP from being located by more than one individual or group data link service access point address.

4.3.3.2 Communication services

Fieldbus APs communicate with each other using confirmed and unconfirmed services (ISO/IEC 10731). The services defined in this standard for the FAL specify the semantics of

the services as seen by the requesting and responding APs. The syntax of the messages used to convey the service requests and responses is defined in IEC 61158-6. The AP behavior associated with the services is specified by the AP.

Confirmed services are used to define request/response exchanges between APs.

Unconfirmed services, in contrast, are used to define the unidirectional transfer of messages from one AP to one or more remote APs. From a communications perspective, there is no relationship between separate invocations of unconfirmed services as there is between the request and response of a confirmed service.

4.3.3.3 AP interactions

4.3.3.3.1 General

Within the fieldbus environment, APs may interact with other APs as necessary to achieve their functional objectives. No constraints are imposed by this standard on the organization of these interactions or the possible relationships that may exist between them.

For example, in the fieldbus environment, interactions may be based on request/response messages sent directly between APs, or on data/events sent by one AP for use by others. These two models of interactions between APs are referred to as client/server and publisher/subscriber interactions.

The services supported by an interaction model are conveyed by application relationship endpoints (AREPs) associated with the communicating APs. The role that the AREP plays in the interaction (e.g. client, server, peer, publisher, subscriber) is defined as an attribute of the AREP.

4.3.3.3.2 Client/server interactions

Client/server interactions are characterized by a bi-directional data flow between a client AP and one or more server APs. Figure 4 illustrates the interaction between a single client and a single server. In this type of interaction, the client may issue a confirmed or unconfirmed request to the server to perform some task. If the service is confirmed then the server will always return a response. If the service is unconfirmed, the server may return a response using an unconfirmed service defined for this purpose.

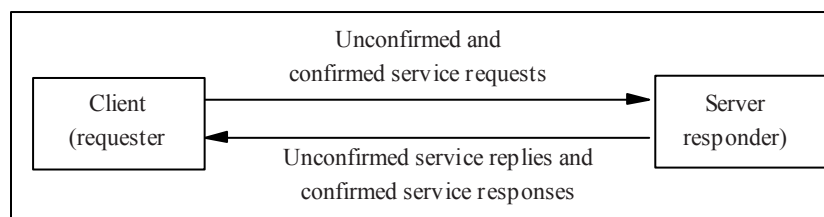


Figure 4 – Client/server interactions

4.3.3.3.3 Publisher/subscriber interactions

4.3.3.3.3.1 General

Publisher/subscriber interactions, on the other hand, involve a single publisher AP, and a group of one or more subscriber APs. This type of interaction has been defined to support variations of two models of interaction between APs, the "pull" model and the "push" model. In both models, the setup of the publishing AP is performed by management and is outside the scope of this standard.